CLI Reference

August 01, 2025

HPE aruba networking

Copyright and Trademarks

Support

For product and technical support, contact support at either of the following:

1.800.943.4526 (toll-free in USA and Canada)
+1.408.941.4300
www.silver-peak.com/support

We are dedicated to continually improving our products and documentation. If you have suggestions or feedback for our documentation, send an e-mail to sp-techpubs@hpe.com.

Table of Contents

CLI Reference **12**

Using the Command Line Interface . . . 13
CLI Modes . . . 14
User Privilege Levels . . . 17
Authorization Credentials . . . 18
Naming Objects . . . 19
Understanding the Command Syntax . . . 20
Conventions Used in this Manual . . . 22
Using the Command Line-Editing Keys . . . 23
Configuring DB-9 Console Access to the Appliance . . . 24
Configuration Commands . . . 25
aaa authentication login . . . 26
aaa authorization map . . . 27
access-list . . . 28
active-flows . . . 32
alarms . . . 33
application . . . 34
application-group . . . 36
arp . . . 38
banner login . . . 39
banner motd . . . 40
bgp . . . 41
bgp neighbor soft-reconfiguration . . . 43
boot system . . . 44
bridge . . . 45
cc enable / disable . . . 46
cdp holdtime . . . 48
cdp timer . . . 49
clear . . . 50
clear cdp counters . . . 51
clear cdp table . . . 52
clear lldp counters . . . 53
clear lldp table . . . 54
cli . . . 55
clock set . . . 57
clock timezone . . . 58
cluster . . . 60
configuration . . . 61
configure terminal . . . 64
debug generate dump . . . 65

disable . . . 66
discoveryd enable / disable . . . 67
dns cache . . . 68
enable . . . 69
enable password . . . 70
excess-flow . . . 71
exit . . . 72
fips enable / disable . . . 73
fips secure erase . . . 74
fips show . . . 75
flow-debug . . . 76
flow-export . . . 78
flow-redirection . . . 80
help . . . 81
hostname . . . 82
iflabel . . . 83
igmp interface enable . . . 84
image boot . . . 85
image install . . . 86
image upgrade . . . 87
interface cdp enable / disable . . . 88
interface dhcp . . . 89
interface inbound-max-bw . . . 90
interface ip address . . . 91
interface label . . . 92
interface lldp enable / disable . . . 93
interface mac address . . . 94
interface mtu . . . 95
interface outbound-max-bw . . . 96
interface pass-through . . . 97
interface security-mode . . . 98
interface shutdown . . . 99
interface speed-duplex . . . 100
interface tunnel admin . . . 101
interface tunnel alias . . . 102
interface tunnel bind-tunnel . . . 103
interface tunnel control-packet . . . 104
interface tunnel create . . . 105
interface tunnel gre-protocol . . . 107
interface tunnel ipsec . . . 108
interface tunnel max-bandwidth . . . 110
interface tunnel min-bandwidth . . . 111
interface tunnel mode . . . 112
interface tunnel mtu . . . 113
interface tunnel nat-mode . . . 114
interface tunnel packet . . . 115
interface tunnel peer-name . . . 117
interface tunnel revert . . . 118

interface tunnel tag-name . . . 119
interface tunnel threshold . . . 120
interface tunnel traceroute . . . 122
interface tunnel udp-flow . . . 123
interface tunnel udp-port . . . 124
interface virtual . . . 125
interface vrrp (no) . . . 126
interface vrrp admin . . . 127
interface vrrp authentication . . . 129
interface vrrp debug action . . . 131
interface vrrp debug packet-trace . . . 132
interface vrrp description . . . 133
interface vrrp ip . . . 134
interface vrrp preempt . . . 136
interface vrrp priority . . . 138
interface vrrp timers advertise . . . 140
interface vrrp timers holddown . . . 142
interface vrrp version . . . 144
ip default-gateway . . . 145
ip domain-list . . . 146
ip host . . . 147
ip mgmt-ip . . . 148
ip multicast route group . . . 149
ip name-server . . . 150
ip route . . . 151
ip-tracking . . . 152
license . . . 153
lldp holdtime . . . 154
lldp timer . . . 155
logging . . . 156
logging facility . . . 157
logging files . . . 158
logging local . . . 160
logging trap . . . 161
monitor . . . 162
mtr . . . 163
multicast enable / disable . . . 165
multicast filtername . . . 166
nat-map . . . 167
nat-map (no) . . . 169
nat-map activate . . . 170
nat-map comment . . . 171
nat-map match . . . 172
nat-map modify-priority . . . 174
nat-map set . . . 175
ntp . . . 177
ntpdate . . . 179
opt-map . . . 180

opt-map (no) . . . 182
opt-map activate . . . 183
opt-map comment . . . 184
opt-map match . . . 185
opt-map modify-priority . . . 188
opt-map set . . . 189
overlay . . . 193
pim interface dr-priority . . . 195
pim interface enable . . . 196
pim interface hello-interval . . . 197
pim interface join-prune-interval . . . 198
pim rp ip . . . 199
ping . . . 200
proxy-arp . . . 203
qos-map . . . 204
qos-map (no) . . . 205
qos-map activate . . . 206
qos-map comment . . . 207
qos-map match . . . 208
qos-map modify-priority . . . 211
qos-map set . . . 212
radius-server . . . 214
reboot . . . 216
reload . . . 217
route-map . . . 218
route-map (no) . . . 219
route-map activate . . . 220
route-map comment . . . 221
route-map match . . . 222
route-map modify-priority . . . 225
route-map set . . . 226
saas . . . 229
selftest . . . 230
shaper inbound . . . 231
shaper outbound . . . 233
slogin . . . 235
snmp-server . . . 239
snmp-server user v3 . . . 242
ssh client global host-key-check . . . 244
ssh client global known-host . . . 246
ssh client global known-hosts-file . . . 247
ssh client user authorized-key . . . 248
ssh client user identity . . . 250
ssh client user known-host remove . . . 251
ssh server enable . . . 252
ssh server encryption-algos . . . 253
ssh server host-key . . . 255
ssh server key-exchange-algos . . . 257

ssh server mac-algos . . . 259
ssh server permit-scp-sftp . . . 260
ssh server ports . . . 261
ssl auth-certificate . . . 262
ssl builtin-signing . . . 263
ssl cert-substitution . . . 264
ssl host-certificate . . . 265
ssl signing-certificate . . . 266
ssl subs-certificate . . . 268
subnet . . . 269
system arp-table-size . . . 271
system auto-ipid . . . 272
system auto-mac-configure . . . 273
system auto-policy-lookup . . . 274
system auto-subnet . . . 275
system auto-syn . . . 276
system bandwidth . . . 277
system bonding . . . 278
system bypass . . . 279
system cc enable / disable . . . 280
system contact . . . 282
system disk . . . 283
system disk encryption . . . 284
system dpc . . . 285
system eclicense . . . 286
system fips enable / disable . . . 287
system fips secure erase . . . 288
system firmware . . . 289
system hostname . . . 290
system int-hairpin . . . 291
system ip-broadcast enable . . . 292
system location . . . 293
system mode . . . 294
system nat-all-inbound . . . 296
system nat-all-outbound . . . 297
system network-memory . . . 298
system passthru-to-sender . . . 299
system peer-list . . . 300
system registration . . . 301
system router . . . 302
system routing . . . 304
system smb-signing . . . 305
system ssl-ipsec-override . . . 306
tacacs-server . . . 307
tca . . . 309
tcpdump . . . 312
tcptraceroute . . . 316
telnet . . . 319

terminal . . . 321
traceroute . . . 322
traffic-class . . . 324
username (no) . . . 325
username capability . . . 326
username disable . . . 327
username password . . . 328
vrrp vmac enable / disable . . . 330
wccp . . . 331
web . . . 334
write . . . 336
Display Commands . . . 337
show aaa . . . 338
show access-list . . . 339
show alarms . . . 340
show application . . . 342
show application-builtin . . . 344
show application-group . . . 345
show arp . . . 347
show banner . . . 348
show bgp . . . 349
show bootvar . . . 350
show bridge . . . 351
show cc . . . 352
show cdp . . . 353
show cdp neighbors . . . 354
show cdp traffic . . . 355
show cli . . . 356
show clock . . . 357
show cluster . . . 358
show configuration . . . 359
show edgeha hasync . . . 361
show excess-flow . . . 362
show files . . . 363
show flow-debug . . . 365
show flow-export . . . 366
show flow-redirection . . . 367
show hosts . . . 368
show iflabels . . . 369
show igmp interfaces . . . 370
show image . . . 371
show interfaces . . . 372
show interfaces cdp . . . 374
show interfaces cdp neighbors . . . 375
show interfaces lldp . . . 376
show interfaces lldp neighbors . . . 377
show interfaces pass-through . . . 378
show interfaces security . . . 381

show interfaces tunnel . . . 382
show interfaces virtual . . . 385
show interfaces vrrp . . . 386
show ip . . . 387
show ip multicast static routes . . . 388
show ip-tracking . . . 389
show licenses . . . 390
show lldp . . . 391
show lldp neighbors . . . 392
show lldp traffic . . . 393
show log . . . 394
show log-files . . . 397
show log-list matching . . . 399
show logging . . . 400
show memory . . . 402
show nat-map . . . 403
show nat statistics . . . 405
show ntp . . . 406
show opt-map . . . 407
show overlay . . . 411
show overlay-common . . . 412
show pass-through . . . 413
show pim debug . . . 415
show pim interfaces . . . 416
show pim interfaces stats . . . 417
show pim internal stats . . . 418
show pim mroute . . . 419
show pim neighbors . . . 420
show pim neighbors stats . . . 421
show pim rp . . . 422
show pim rtm . . . 423
show pimlite adjacencies . . . 424
show pimlite mroutes . . . 425
show pimlite oifs . . . 426
show pimlite stats . . . 427
show proxy-arp . . . 429
show qos-map . . . 430
show radius . . . 433
show route-map . . . 434
show running-config . . . 437
show selftest disk . . . 438
show shaper . . . 439
show snmp . . . 440
show ssh client . . . 442
show ssh server . . . 443
show ssl . . . 445
show stats . . . 446
show stats tunnel . . . 448

show subif 450
show subnet 451
show system 452
show system cc 455
show system fips 456
show tacacs 457
show tca 458
show terminal 460
show transceiver 461
show tunnel 463
show usernames 466
show users 467
show users history 468
show version 469
show vlan 470
show vrrp 471
show wccp 473
show web 476
show whoami 477

CLI Reference

This document provides information about the command line interface (CLI) for Aruba EdgeConnect appliance software.

This content does not provide feature descriptions or explanations of the technologies. For information about the various features and technologies supported by EdgeConnect physical and virtual appliances, see the Silver Peak Appliance Manager Operator's Guide.

Using the Command Line Interface

This section provides details of the command line syntax for Aruba EdgeConnect appliance software.

This content does not provide feature descriptions or explanations of the technologies. For information about the various features and technologies supported by EdgeConnect physical and virtual appliances, see the Silver Peak Appliance Manager Operator's Guide.

CLI Modes

This section describes the three command modes defined for the EdgeConnect appliance CLI:

- User EXEC Mode
- Privileged EXEC Mode
- Global Configuration Mode

Being in a particular command mode determines which commands you may execute. To display a list of the command that are available to you, enter that command mode and type **?** (a question mark).

User EXEC Mode

When you first log in to an EdgeConnect appliance, you are in the User EXEC mode. The User EXEC mode provides access to commands for non-configuration tasks, such as checking the appliance status. When you are in this mode, the following prompt displays:

<appliance> >

where *appliance* is the name of the appliance on which you logged in.

In the User EXEC mode, you have access to the following commands:

Command	Result
cli	Configure CLI shell options
enable	Enter enable mode
exit	Log out of the CLI
no	Negate or clear certain configuration options
ping	Send ICMP echo requests to a specified host
show	Display system configuration or statistics
slogin	Log into another system securely using ssh
telnet	Log into another system using telnet
terminal	Set terminal parameters
traceroute	Trace the route packets take to a destination
wccp	Configure WCCP

Privileged EXEC Mode

The Privileged EXEC mode provides access to all the commands you could execute in User EXEC mode, as well as several additional commands. Also, from this mode, you can enter Global Configuration mode. Most of the commands that the Privileged EXEC mode makes available are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces.

To enter Privileged EXEC mode, type **enable** from the User EXEC prompt, which displays the following prompt:

<appliance> #

where *appliance* is the name of the appliance on which you logged in.

In Privileged EXEC mode, you access to the following commands:

clear	Reset certain statistics or clear caches
cli	Configure CLI shell options
configure	Enter configuration mode
debug	Debugging commands
disable	Leave enable mode
email	Configure e-mail and event notification via e-mail
exit	Log out of the CLI
file	Manipulate files on disk
image	Manipulate system software images
job	Configure scheduled jobs
logging	Configure event logging
no	Negate or clear certain configuration options
ntpdate	Set system clock once from a remote server using NTP
ping	Send ICMP echo requests to a specified host
reboot	Reboot or shut down the system
show	Display system configuration or statistics
slogin	Log into another system securely using ssh
system	Configure system level information
tcpdump	Display packets on a network
telnet	Log into another system using telnet
terminal	Set terminal parameters
traceroute	Trace the route packets take to a destination
write	Save the running configuration to persistent storage

Global Configuration Mode

The Global Configuration mode allows you to make changes to the running configuration. If you later save the configuration, these commands are stored across appliance reboots. To enter the Global Configuration mode, you must first enter the Privileged EXEC mode and then type **configure terminal** at the prompt. When you press *Enter*, the following prompt displays:

<appliance> (config) #

where *appliance* is the name of the appliance on which you logged in.

Global Configuration mode provides access to all CLI commands, including those available in User EXEC and Privileged EXEC modes.

You must have an Administrator user privilege level to access Global Configuration mode.

To leave Global Configuration mode, you can use the “no configure” or “exit” commands:

<appliance> (config) # no configure

User Privilege Levels

The CLI has two user privilege levels, which determine the CLI modes you may enter and the commands you can execute. You can log in to one of the following user privilege levels:

- Administrator
- Monitor

To execute a CLI command at the prompt, you must be logged in at the required user privilege level for that command. For example, most configuration commands require you to have the Administrator privilege level.

You cannot delete user IDs in the CLI; you can only change the password for a user.

Monitor

The Monitor user privilege level is the default privilege level for the CLI. This privilege level provides access to the both the User EXEC and Privileged EXEC modes. The Monitor user privilege level does not have access to most configuration commands.

Administrator

The Administrator user privilege level has full access to all modes and commands in the CLI.

Authorization Credentials

Accessing the CLI requires a username and password.

Usernames

When you create a username, ensure that the first character of the name is alphabetical (a-z or A-Z). The remaining characters must include one of the following:

- alphabetical (upper or lower case)
- numerical
- dash (-)
- underscore (_)
- dot (.)

No spaces are allowed.

Passwords

- You can establish passwords for a user to enter the Privilege EXEC or Global Configuration modes.
- The CLI provides no restrictions on the password you create for a user.
- You may enter a clear-text password or use a utility to create an encrypted password for a user.
- There are also no restrictions on the use of, or requirement for, special characters in the password.

Naming Objects

When you create a name for an object, such as a tunnel, access control list, or a route map, you can use one of the following characters:

- alphabetical (upper or lower case)
- numerical
- dash (-)
- underscore (_)
- dot (.)

The Silver Peak command line interface (CLI) supports only the US character set.

Understanding the Command Syntax

The following symbols are used in the CLI documentation to describe the command syntax. When you execute commands in the CLI, do not type these characters:

Symbol Name	Symbol	Syntax
Angled brackets	**< >**	Enclose a variable or a value that you must specify in the command. For example, in the syntax: **configure vlan ip address** , you must supply a VLAN name for the variable and an IP address for the variable when you enter the command.
Vertical bars	**|**	Separate mutually exclusive items in a list, one of which must be entered. For example, in the syntax **file upload | cancel**, you must specify either the file name variable or the word, **cancel**, when you enter the command.
Curly brackets	**{ }**	Enclose a required value or list of required arguments. One or more values or arguments can be specified in square brackets. For example, in the syntax **configure snmp community {read-only | read-write} <***string__>,__ *you must include either the* read-only* or *read-write* argument in the command.
Square brackets	**[ ]**	Enclose an optional value or a list of optional arguments. You can specify in curly brackets one or more values or arguments that are not required to execute the command. For example, in the syntax **reboot [| cancel]**, you can choose to use the reboot command without any arguments. Alternately, you can specify either a particular date and time combination or the keyword **cancel** to cancel a previously scheduled reboot.

Syntax Helper

The CLI has a built-in Syntax Helper. If you are not sure of the complete syntax for a particular command, enter the first three letters of the command and press the **Tab** key. The Syntax Helper provides a list of options for the remainder of the command, and places the cursor at the end of the command you have entered so far, ready for the next option.

The Syntax Helper also provides assistance by informing you if you have entered an incorrect command.

Command History

The Silver Peak operating system keeps the last commands you entered in its memory. You can “walk” through these commands one at a time by using the **Up** and **Down** arrows on your keyboard.

Conventions Used in this Manual

The following topics are discussed in this section:

Typographical Conventions

- In examples, terminal sessions and system displays are shown in `Courier` font.
- The commands that you need to type exactly as shown are displayed in ***`Courier bold`***.

Syntax Notation

- Commands and keywords are in **bold** text.
- Angled brackets (< >) indicate nonprinting characters, such as passwords, and variables that you need to replace with a value.
- Arguments for which you supply values are in *italics*.
- Curly brackets ({ }) contain required choices.
- Square brackets ([ ]) contain optional elements.
- Vertical bars (|) separate the alternative elements.
- Curly brackets and vertical bars within square brackets ([{ | }]) mean a required choice within an optional element.

Using the Command Line-Editing Keys

These line-editing keys are available when you are using the CLI:

Key	Description
Backspace	This key deletes character to left of cursor and shifts remainder of line to left.
Delete or [*Ctrl*] + *D*	Deletes character under cursor and shifts remainder of line to left.
[*Ctrl*] + *K*	Deletes characters from under cursor to end of line.
Insert	Toggles between on and off. When on, inserts text and shifts previous text to right.
Left Arrow	Moves cursor to left.
Right Arrow	Moves cursor to right.
Home or [*Ctrl*] + *A*	Moves cursor to first character in line.
End or [*Ctrl*] + *E*	Moves cursor to last character in line.
[*Ctrl*] + *L*	Clears screen and movers cursor to beginning of line.
[*Ctrl*] + *P* or *Up Arrow*	Displays previous command in command history buffer and places cursor at end of command.
[*Ctrl*] + *N* or *Down Arrow*	Displays next command in command history buffer and places cursor at end of command.
[*Ctrl*] + *U*	Clears all characters typed from cursor to beginning of line.
[*Ctrl*] + *W*	Deletes previous word.

When you choose to display output in multiple pages, the CLI has additional “editor” keys available:

Key	Description
1 + [Shift] + g	Moves to the top of the screen display.
1 + g	Moves to the bottom of the screen display.
/*textstring*	Searches forward for the textstring you enter.
?textstring	Searches backward for the textstring you enter.
Spacebar	Moves forward a page.
[Enter]	Moves forward one line.
q	Quits out of what it was doing and returns you to the command prompt.

Configuring DB-9 Console Access to the Appliance

For console port access, the appropriate settings are as follows:

Parameter	Setting
Bits per second	9600
Data bits	8
Parity	None
Stop bits	1
Flow control	None

Configuration Commands

Configuration commands allow you to configure Silver Peak gateways:

aaa authentication login

Use the **aaa authentication login** default command to configure the order in which authentication methods are tried. **Authentication** is the process of validating that the end user, or device, is who they claim to be. Generally, authentication precedes authorization.

Use the **no** form of this command to clear all authentication login settings.

Command Mode: Global configuration mode

Syntax

aaa authentication login default { *method-1* | *method-1 method-2* | *method-1 method-2 method-3* }
no aaa authentication login

Arguments

Parameter	Description
method-x	Specifies the methods for authenticating the default login in the order that they will be used. The method options are: - **local** - **radius** - **tacacs+**

Defaults

No default behavior or values.

Usage Guidelines

You can use up to three methods (or databases) for authentication, place the methods in any order, and/or use any method more than once.

However, one of the methods that you include must be **local**.

Examples

To set the authentication login methods to be local and TACACS+, in that order:

```
ECV (config) # aaa authentication login default local tacacs+
```

aaa authorization map

Use the **aaa authorization map default-user** command to configure authorization mapping settings. *Authorization* is the action of determining what a user is allowed to do. Generally, authentication precedes authorization.

Command Mode: Global configuration map

Syntax

aaa authorization map default-user *user*
no aaa authorization map default-user

aaa authorization map order *policy*
no aaa authorization map order

Parameters

user: Specifies the user ID of a valid local user. Generally, this is **admin** or **monitor**.

map default-user *user*: Sets the local user default mapping. Use the **no** form of this command to clear the local user default mapping.

policy: Specifies the order for handling remote-to-local user mapping. Available policies:

- **remote-only** Only honor user mapping from remote authentication server.
- **remote-first** Honor user mapping from remote auth server, if provided; otherwise use local mapping.
- **local-first** Ignore user mapping from remote auth server; use local mapping only.

  The **no** form of the command clears the authorization user mapping order policy. |

Usage Guidelines

When you enter a user name, the system verifies in the database that the user ID is valid.

Examples

To set authorization mapping to check the remote database first:

```
ECV (config)# aaa authorization map order remote-first
```

access-list

Use the **access-list** command to configure Access Lists and their rules.

Use the **no access-list** command to delete a specific ACL rule or an entire ACL.

Command Mode: Global configuration mode

Syntax

access-list *acl-name priority-value* { **permit** | **deny** } **protocol** { *IP-protocol-number* | *protocol-name* } { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [**dscp** { *dscp-value* | **any** }]

access-list *acl-name priority-value* { **permit** | **deny** } **protocol** { *IP-protocol-number* | *protocol-name* } { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [**vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }]

access-list *acl-name priority-value* { **permit** | **deny** } **protocol-ip** { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [**app** { *app-name* | **any** }] [**dscp** { *dscp-value* | **any** }][**vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }]

access-list *acl-name priority-value* { **permit** | **deny** } **protocol-ip** { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [**app** { *app-name* | **any** }] [**dscp** { *dscp-value* | **any** }]

access-list *acl-name priority-value* { **permit** | **deny** } **protocol-ip** { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [**vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }]

access-list *acl-name priority-value* { **permit** | **deny** } **protocol** { **tcp** | **udp** } { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [{ *source-port-number* | **any** } { *dest-port-number* | **any** }] [**dscp** { *dscp-value* | **any** }]

access-list *acl-name priority-value* { **permit** | **deny** } **protocol** { **tcp** | **udp** } { *source-IP-addr/netmask* | **any** } { *dest-IP-addr/netmask* | **any** } [{ *source-port-number* | **any** } { *dest-port-number* | **any** }] [**vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }]

access-list *acl-name priority-value* { **permit** | **deny** } **app** { *app-name* | **any** }

access-list *acl-name priority-value* { **permit** | **deny** } **dscp** { *dscp-value* | **any** } [**vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }]

access-list *acl-name priority-value* { **permit** | **deny** } **matchstr** { *match-string* | **any** }

access-list *acl-name priority-value* { **permit** | **deny** } **vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }

access-list *acl-name priority-value* **comment** *comment-text*

no access-list *acl-name* [*priority-value*]

Arguments

Parameter	Description
access-list *acl-name priority-value*	Specifies the name of the ACL and the priority value for the (ACL) rule that you want to add or modify. You can set any priority value between 1 and 65535.
permit	Permits access to this ACL rule.
deny	For traffic that matches this ACL rule, discontinue further processing **by this ACL**, and continue to look for a match in the subsequent policy entries.
comment	Add a comment for specified access list entry.
protocol { *IP-protocol-number* | *IP-protocol-name* | **ip** | **tcp** | **udp** }	Specifies the protocol to match:The available IP protocol numbers include 1 through 254.When you specify **protocol ip**, the assumption is that you are allowing *any* IP protocol. In that case, you also need to specify an application. If you don't, the CLI defaults to specifying **any** application.
{ *source-IP-addr/netmask* | **any** }	Matches against traffic that has a specific source IP address and netmask (in slash notation). For example, enter *10.2.0.0 0.0.255.255* as *10.2.0.0/16.*If you want to include traffic to all destinations, use **any**.
{ *dest-IP-addr/netmask* | **any** }	Matches against traffic that has a specific destination IP address and netmask (in slash notation). For example, *10.2.0.0/16.*If you want to include traffic to all destinations, use **any**.
{ *source-port-number* | **any** } { *dest-port-number* | **any** }	When you specify **protocol tcp** or **protocol udp**, you can limit the traffic to specific source and/or destination ports. **any** is a wildcard.
app { *app-name* | **any** }	Specifies a default or user-defined application name, or the name of a user-defined application group. **any** is a wildcard.
dscp { *dscp-value* | **any** }	Specifies a DSCP value. The available values include:af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, or ef.__any__ is a wildcard.
matchstr *match-string*	Adds a match string for specified access list entry.
vlan { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }	Matches an interface and 802.1q VLAN tag. The available values include: - *1..4094* number assigned to a VLAN - *interface.tag* as in **lan0.10** - *any.tag* as in **any.10** - *interface.any* as in **lan0.any** - *interface.native* as in **lan0.native** - **any** is a wildcard

Parameter	Description
any	Is a wildcard.

Usage Guidelines

You name a rule with a *priority*, which not only identifies the rule, but also specifies its sequence in that ACL. Within an ACL, every priority value must be unique. In other words, no two rules in a given ACL can have the same priority value. We recommend that you don't make the priority values contiguous, so that you can later insert a new rule between two existing rules, without needing to change the priority values you've already set. For example, you might create an ACL with rules (priorities) 10, 20, 30, and 40. If you need to add several rules at a later time, you can easily place them between any of the existing rules.

If you need to replace an existing rule, just name the new rule with the same priority as the one you want to replace. The CLI overwrites the existing rule with your new one.

If you specify a priority to create a rule for an ACL that doesn't already exist, the CLI creates the new ACL and populates it with the new rule.

Use the **no** form of this command to delete a rule within an ACL. If you delete the last rule of an ACL, that ACL is removed. If you don't specify a priority value in the **no** command, the entire ACL is deleted.

IP Address and Netmasks

Source and destination IP addresses are immediately followed by a netmask "/n" where *n* is the number of contiguous non-wildcard bits counting from the left-most bit. For example, 10.10.10.0 /24 refers to the 10.10.10 class C subnet. Use the keyword **any** to specify that all bits are wildcards.

Using Deny

Since access lists define the matching criteria and not the action, you should remember that **deny** in this context does not actually "drop" traffic. Rather, the **deny** keyword is effectively a sort of break statement, preventing further processing by that particular ACL, and sending the traffic to look for matches against subsequent **policy** entries.

For example, if you wanted to accelerate all IP traffic except for ICMP traffic, you could enter the following commands:

```
access-list a1 100 deny protocol icmp any any  \newline
access-list a1 200 permit protocol ip any any  \newline
.  \newline
.  \newline
.  \newline
route-map map1 10 match acl a1  \newline
route-map map1 10 set tunnel tun1.  \newline
.  \newline
.  \newline
```

In this example, any ICMP traffic that attempts to match the ACL, *a1*, would immediately stop processing at the **deny** statement and would pass through.

Examples

To create a rule for an ACL named *acl2*, that matches against all IGP traffic that has a DSCP value of *be* (best effort):

```
ECV (config) # access-list acl2 10 permit protocol igp any any dscp be
```

To accelerate all IP traffic except for ICMP traffic:

```
ECV (config) # access-list a1 100 deny protocol icmp any any   \newline
ECV (config) # access-list a1 200 permit protocol ip any any
```

To create a rule to match all IP traffic coming from the source 10.2.0.0 0.0.255.255:

```
ECV (config) # access-list a2 40 permit protocol ip 10.2.0.0/16 any
```

To create a rule to match all UDP traffic going to port 53:

```
ECV (config) # access-list a1 500 protocol udp any any any 53
```

To delete the priority 100 rule from the ACL named *ac18*:

```
ECV (config) # no access-list acl8 100
```

active-flows

Use the **active-flows** command to configure all active flows.

Command Mode: Privileged EXEC mode

Syntax

active-flows { **reset-all** }

Arguments

Parameter	Description
reset-all	Resets all non-TCP accelerated active flows.

Examples

None

alarms

Use the **alarms** command to manage the alarms in the system.

Command Mode: Global configuration mode

Syntax

alarms { **acknowledge** | **unacknowledge** } *alarm-seq-number*

alarms clear *alarm-seq-number*

Arguments

Parameter	Description
acknowledge	Acknowledges an alarm in the system.
clear	Clears an alarm in the system.
unacknowledge	Unacknowledges an alarm in the system.
alarm-seq-number	Specifies the sequence number of the alarm.

Usage Guidelines

For a list of current alarms, use the following command:

```
show alarms outstanding
```

```
ECV (config) # show alarms outstanding

###   Seq  Date                Type  Sev A Source       Description
--- ---- ------------------- ----- --- - ------------ -----------------
  1    5 2007/06/19 19:23:54   EQU MAJ N system       Datapath Gateway Connectivity
    Test Failed
  2    4 2007/06/19 19:21:58   TUN CRI N HQ-to-Branch Tunnel state is Down
  3    2 2007/06/19 19:20:44   EQU MAJ N wan0         Network Interface Link Down
```

The *alarm sequence number* is **not** the same as the *alarm ID* number.

Examples

None

application

Use the **application** command to configure applications on the appliance.

Use the **no application** command to delete an application.

Command Mode: Global configuration mode

Syntax

application *app-priority app-name* **dscp** *dscp-value*

application *app-priority app-name* **protocol** *IP-protocol-number-or-name*

application *app-priority app-name* **protocol** *IP-protocol-number-or-name* **src-ip** { *source-IP-addr-range* | **any** } [**src-port** { *source-port-range* | **any** }]

application *app-priority app-name* **protocol** *IP-protocol-number-or-name* **src-ip** { *source-IP-addr-range* | **any** } **src-port** { *source-port-range* | **any** } **dst-ip** {*dest-IP-addr-range* | **any** } [**dst-port** { *dest-port-range* | **any**}]

application *app-priority app-name* **protocol** *IP-protocol-number-or-name* **src-ip** { *source-IP-addr-range* | **any** } **src-port** { *source-port-range* | **any** } **dst-ip** {*dest-IP-addr-range* | **any** } **dst-port** { *dest-port-range* | **any** } [**dscp** *dscp-value*]

application *app-priority app-name* **protocol** *IP-protocol-number-or-name* **src-ip** { *source-IP-addr-range* | **any** } **src-port** { *source-port-range* | **any** } **dst-ip** {*dest-IP-addr-range* | **any** } **dst-port** { *dest-port-range* | **any** } **dscp** *dscp-value* [**vlan** { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }]

no application *app-priority*

Arguments

Parameter	Description
app-priority	Specifies the priority value of the application.
app-name	Specifies the name of the application.
protocol *IP-protocol-number-or-name*	Specifies the application protocol.
src-ip { *source-IP-addr-range* | **any** }	You can specify a comma-delimited list. For example: 192.1.2.0/24,192.10.10.100-200If you want to include all addresses, use **any**.
src-port { *source-port-range* | **any** }	Comma-separated port ranges. If you want to include all ports, use **any**.

Parameter	Description
dst-ip { *dest-IP-addr-range* | **any** }	You can specify a comma-delimited list. For example: 192.1.2.0/24,192.10.10.100-200If you want to include all addresses, use **any**.
dst-port { *dest-port-range* | **any** }	Comma separated port ranges. If you want to include all ports, use **any**.
dscp { *dscp-value* | **any** }	Specifies a DSCP value. The available values include:af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, or ef. **any** is a wildcard.
vlan { **any** | *1..4094* | *interface.tag* | *any.tag* | *interface.any* | *interface.native* }	Matches an interface and 802.1q VLAN tag. The available values include: *1..4094* number assigned to a VLAN *interface.tag* as in **lan0.10** *any.tag* as in **any.10** *interface.any* as in **lan0.any** *interface.native* as in **lan0.native** **any** is a wildcard
any	Is a wildcard

Examples

To create an application, *surf*, for traffic that comes from the IP address, 192.4.4.11:

```
ECV > application 10 surf protocol any src-ip 192.4.4.11
```

application-group

Use the **application-group** command to specify a group of (one or more) applications.

Use **no application-group** to remove one or more applications from an application group or to delete the group, itself.

Command Mode: Global configuration mode

Syntax

application-group *app-group-name app-1* [, *app-2*, *app-3*... ,*app-n*]
no application-group *app-group-name* [, *app1*, *app2*... ,*app-n*]

Arguments

Parameter	Description
app-group-name	Defines a unique group name. The name is checked against existing application groups and, if the name does not exist, the CLI creates it. If the name does exist, then the application(s) you specify are added to the existing group.
app-x	Specifies an existing application name, whether it's built-in or user-defined.

Usage Guidelines

If your ACLs or policy maps contain match conditions that involve multiple applications, you can simplify the match conditions with *application groups*. Application groups are identifiers that you can create to represent a list of applications.

You create an application group by naming the group and specifying at least one application that belongs in it. After creating it, you can modify the application group by adding or removing applications.

To add applications to an application group that already exists, enter the name of the application group, followed by the applications you are adding. For example, to add two applications to the application group, *omega*, you might use the following command:

```
ECV (config) # application-group omega http, tftp
```

If *omega* did not exist, the CLI would create it and it would contain these two applications.

If you then wanted to remove *http* from *omega*, you would issue the following command:

```
ECV (config) # no application-group omega http
```

The **application-group** command has the following restrictions:

- If you specify more than one application at a time for an application group, you must separate the applications with commas. If you just use spaces, the CLI will respond with an error message.
- If you attempt to delete an application that is not in the application group that you specify, then the CLI displays an error message.

Examples

To create an application group, *encrypted*, that contains the applications SSH, HTTPS, and SFTP:

```
ECV (config) # application-group encrypted ssh, https, sftp
```

To add two applications to the existing application group, *omega*:

```
ECV (config) # application-group omega http, tftp
```

arp

Use the **arp** command to add static entries to the Address Resolution Protocol (ARP) cache.

Use the **no** form of this command to remove a static entry from the ARP cache.

Command Mode: Global configuration mode

Syntax

arp *ip-addr mac-addr*
no arp *ip-addr*

Arguments

Parameter	Description
ip-addr	Specifies an IP address.
mac-addr	Defines the 48-bit MAC address that the IP address to which the IP address will be mapped.

Examples

To create an entry in the ARP table for a machine with the IP address 10.10.1.1 and MAC address 00107654bd33:

```
ECV (config) # arp 10.10.1.1 00107654bd33
```

banner login

Use the **banner login** command to create a message for the system login banner, such as legal or welcome text.

Use the **no** form of this command to reset the system login banner.

Command Mode: Global configuration mode

Syntax

banner login *message-string*
no banner login

Arguments

Parameter	Description
message-string	Specifies the message to display before a user logs into the appliance. A message that includes spaces requires quotes at the beginning and end of the message string.

Examples

To configure the banner message, *Gotcha!*, to display at login:

```
ECV (config) # banner login Gotcha!
```

To configure the banner message, *“How about some coffee?”*, to display at login:

```
ECV (config) # banner login “How about some coffee?”
```

banner motd

Use the **banner motd** command to create a “Message of the Day” banner.

Use the **no** form of this command to reset the system Message of the Day banner.

Command Mode: Global configuration mode

Syntax

banner motd *message-string*
no banner motd

Arguments

Parameter	Description
message-string	Specifies the message to display for the Message of the Day. A message that includes spaces requires quotes at the beginning and end of the message string. The Message of the Day appears after successful login.

Examples

To configure the Message of the Day, *Greetings*, to display at login:

```
ECV (config) # banner motd Greetings
```

To configure the banner message, *“Time for a margarita”*, to display at login:

```
ECV (config) # banner motd “Time for a margarita”
```

bgp

Use the **bgp** command to configure BGP (Border Gateway Protocol) on the appliance.

Command Mode: Global Configuration mode

Syntax

bgp asn *1-65535*
no bgp asn *1-65535*

bgp { **disable** | **enable** }

bgp neighbor *Neighbor-IP-addr* **export-map** *Custom-BGP-bit-map-of-permitted-route-types-to-export-(decimal)*
no bgp neighbor *Neighbor-IP-addr* **export-map**

bgp neighbor *Neighbor-IP-addr* **import-disable**
no bgp neighbor *Neighbor-IP-addr* **import-disable**

bgp neighbor *Neighbor-IP-addr* **metric** *Neighbor-additional-route-cost*
no bgp neighbor *Neighbor-IP-addr* **metric**

bgp neighbor *Neighbor-IP-addr* **password** *Neighbor-MD5-pwd*
no bgp neighbor *Neighbor-IP-addr* **password**

bgp neighbor *Neighbor-IP-addr* **remote-as** *Neighbor-ASN* { **Branch** | **Branch-transit** | **PE-router** }

bgp router-id *IPv4-addr-recognizable-to-remote-peer*
no bgp router-id *IPv4-addr-recognizable-to-remote-peer*

no bgp neighbor *Neighbor-IP-addr*

Arguments

Parameter	Description
asn *1-65535*	Autonomous System Number
disable	Disables BGP globally.
enable	Enables BGP globally.

Parameter	Description
export-map *Custom-BGP-bit-map-of-permitted-route-types-to-export-(decimal)*	Creates a BGP neighbor with customized export rules. Use the numbers listed for the following options: **1 Local** Locally configured **2 Shared** Learned via subnet sharing (from a non-BGP source) **4 BGPBr** Learned from a local BGP branch peer **8 BGPTr** Learned from a local BGP branch-transit peer **16 BGPPe** Learned from a local BGP Provider Edge peer **32 RemBGP** Remote BGP (learned via subnet sharing, but originally from a BGP peer) **64 RemBGPTr** Remote BGP branch-transit (learned via subnet sharing, but originally from a BGP branch-transit peer)
neighbor *Neighbor-IP-addr*	Specifies a BGP neighbor.
import-disable	Disables the learning of routes from the neighbor.
metric *Neighbor-additional-route-cost*	Configures additional metric for BGP neighbor.
password *Neighbor-MD5-pwd*	Creates an MD5 password for the BGP neighbor.
remote-as *Neighbor-ASN* { **Branch** | **Branch-transit** | **PE-router** }	Creates a BGP neighbor with a remote ASN (Autonomous System Number): **Branch** Configures Neighbor as branch type **Branch-transit** Configures Neighbor as branch transit type **PE-router** Configures Neighbor as Provider Edge type
router-id *IPv4-addr-recognizable-to-remote-peer*	Configures router IP ID. The router identifier is the IPv4 address which the remote peer identifies the appliance for BGP purposes.

Examples

None

bgp neighbor soft-reconfiguration

The **bgp neighbor soft-reconfiguration** command enables the BGP soft reconfigure function. By default, the VRF segment (router) does not send a route-refresh message to the specified BGP peer when a policy is changed. When soft reconfiguration is enabled, the segment applies policy changes against BGP peer learned routes stored in memory. Commands that do not include the **segment** parameter modify the BGP configuration for the default segment.

The **no bgp neighbor soft-reconfiguration** command disables the soft-reconfiguration function. By default, soft reconfiguration is disabled

Border Gateway Protocol (BGP) is a dynamic routing protocol for exchanging routing and reachability information among routing domains, also referred to as autonomous systems (AS). BGP makes routing decisions based on paths, network policies, and rules defined by a network administrator.

Command Mode: Global Configuration mode

Syntax

bgp neighbor *neighbor-addr* **soft-reconfiguration**
bgp segment *segment-id* **neighbor** *neighbor-addr* **soft-reconfiguration**

no bgp neighbor *neighbor-addr* **soft-reconfiguration**
no bgp segment *segment-id* **neighbor** *neighbor-addr* **soft-reconfiguration**

Parameters

neighbor-addr: The IP address of the BGP neighbor for which soft-reconfiguration is enabled (Dotted decimal notation).

segment-id: VRF segment modified by the command. The default segment is modified when this parameter is omitted.

Defaults

BGP neighbor soft-reconfiguration is disabled by default.

Examples

This command enables BGP soft reconfiguration for the peer at 10.3.2.3 on VRF segment 1.

```
ECV-A (config) # bgp segment 1 neighbor 10.3.2.3 soft-reconfiguration
ECV-A (config) #
```

boot system

Use the **boot system** command to specify which partition to boot from next time.

Command Mode: Global configuration mode

Syntax

boot system *partition-number*

Arguments

Parameter	Description
partition-number	Specifies the next boot partition. The partition options are: - **1** Partition 1 - **2** Partition 1 - **next** The partition that is not currently running.

Examples

To set the appliance to start using partition 2, by default, beginning at the next system boot:

```
ECV (config) # boot system 2
```

To boot from the other partition at the next system boot:

```
ECV (config) # boot system next
```

bridge

Use the **bridge** command to configure bridge mode.

Command Mode: Global Configuration mode

Syntax

bridge propagate-linkdown { **enable** | **disable** }
bridge transition-fdb-age *1-300*
bridge transition-time *1-300*

Arguments

Parameter	Description
propagate-linkdown { **enable** | **disable** }	When enabled, forces the WAN interface link to go down when the corresponding LAN interface goes down, and vice versa.
transition-fdb-age *1-300*	Specifies the maximum age of a MAC entry, in seconds, during the time that a link is going down.
transition-time *1-300*	Specifies, in seconds, the time to wait after the first link goes down before propagating the second link down.

Examples

To configure 30 seconds as the time to wait before propagating the WAN interface's link down to the LAN:

```
ECV (config) # bridge transition-time 30
```

cc enable / disable

The **cc enable** command enables Common Criteria mode on the appliance. This command also enables FIPS mode and reboots the appliance. By default, Common Criteria mode is disabled.

The **cc disable** command disables Common Criteria mode, disables FIPS mode, and reboots the appliance.

The **noconfirm** parameter prompts the CLI to provide command execution status up through the reboot of the appliance.

Common Criteria is an international standard for computer security certification. When Common Criteria mode is enabled, the appliance is Common Criteria compliant to a set of guidelines and certifications that ensure the appliance meets the security standard that includes PKI certificates, online certificate status protocol, and enhanced logging.

Command Mode: Global Configuration mode

Syntax

cc enable
cc enable noconfirm

cc disable
cc disable noconfirm

Usage Guidelines

The **cc enable** and **cc disable** commands are not available in ECOS version 9.4.3 and all later versions. Equivalent commands available in these versions are **system cc enable** and **system cc disable**. The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command enables Common Criteria on the appliance.

```
ECV (config) # cc enable noconfirm

Enabling Common Criteria mode will automatically enable FIPS mode

This operation will cause a system reboot.

Additional security configurations will be applied and
any unsaved configuration changes will get saved.

Configuration changes saved, and cc mode enabled

The appliance is going to reboot...
```

```
ECV (config) #


System shutdown initiated -- logging off.

This will take a few minutes...

Connection to 10.80.171.181 closed.
[root@abcde ~]#
```

cdp holdtime

The **cdp holdtime** command configures the CDP hold time. Hold time is the period that the receiver retains CDP packet information.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Global Configuration mode

Syntax

cdp holdtime *hold-period*

Parameters

hold-period: CDP packet information retention period (seconds). Value range is 10 through 255. Default is 120 seconds.

Examples

This command sets the CDP hold time to 210 seconds.

```
ECV-A (config) # cdp holdtime 210
ECV-A (config) # show cdp
Global CDP information:
        Sending CDP packets every 75 seconds
        Sending a holdtime value of 210 seconds
        Sending CDPv2 advertisements is enabled
ECV-A (config) #
```

cdp timer

The **cdp timer** command configures the CDP timer. The CDP timer is the interval between the transmission of CDP packets.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Global Configuration mode

Syntax

cdp timer *cdp-rate*

Parameters

cdp-rate: CDP packet transmission interval (seconds per packet). Range is 5 through 254. Default is 60 seconds per packet.

Examples

This command sets the CDP timer to 75 seconds.

```
ECV-A (config) # cdp timer 75
ECV-A (config) # show cdp
Global CDP information:
        Sending CDP packets every 75 seconds
        Sending a holdtime value of 180 seconds
        Sending CDPv2 advertisements is enabled
ECV-A (config) #
```

clear

Use the **clear** command to clear entries and/or counters.

Command Mode: EXEC mode (clear cluster, clear flow-redirection, clear proxy-ip-address)
Command Mode: Global configuration mode (all other clear commamds)

Syntax

clear arp-cache
clear bridge counters
clear bridge mac-address-table
clear cluster spcp
clear flow-redirection

Arguments

Parameter	Description
arp-cache	Clears dynamic entries from the ARP cache.
bridge counters	Clears the bridge counters.
bridge mac-address-table	Flushes the bridge MAC address table.
cluster spcp	Clears the cluster's Silver Peak Communication Protocol counters. These are used when doing flow redirection.
flow-redirection	Clears the flow redirection counters.

Examples

None

clear cdp counters

The **clear cdp counters** command resets CDP data counters to zero for all ports.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC modde

Syntax

clear cdp counters

Examples

These commands reset CDP data counters, then display the CDP counters.

```
ECV-A (config) # clear cdp counters
ECV-A (config) # show cdp traffic
CDP counters:
        Total packets output: 0, Input: 0
        Hdr syntax: 0, No memory: 0
ECV-A (config) #
```

clear cdp table

The **clear cdp table** command removes all entries from the CDP neighbor table.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC modde

Syntax

clear cdp table

Examples

None

clear lldp counters

The **clear lldp counters** command clears the LLDP table for all ports.

The Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can include device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

clear lldp counters

Examples

These commands reset LLDP data counters, then display the counters content.

```
ECV-A (config) # clear lldp counters
ECV-A (config) # show lldp traffic
LLDP counters:
        Total packets output: 0, Input: 0
        Hdr syntax: 0, No memory: 0
ECV-A (config) #
```

clear lldp table

The **clear lldp table** command removes all entries from the LLDP neighbor table.

The Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can include device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

clear lldp table

Examples

None

cli

Description

Use the **cli** command to configure CLI shell options.

Command Mode: Global configuration mode (cli session)
Command Mode: EXEC mode (all other cli commands)

Syntax

cli clear-history

cli default allow-all-show { **enable** | **disable** }

cli default auto-logout *number-minutes*
no cli default auto-logout

cli session auto-logout *number-minutes*
no cli session auto-logout

cli session paging enable
no cli session paging enable

cli session terminal length *number-lines*

cli session terminal type { **xterm** | **ansi** | **vt100** }
no cli session terminal type

cli session terminal width *number-char*

Arguments

Parameter	Description
clear-history	Clears the current user's command history.
default allow-all-show { **enable** | **disable** }	When enabled, allows the user to view all possible show commands. When disabled, the commands a user can see are based on privilege level.
default auto-logout *number-minutes*	Configures — for all **future** sessions — the amount of time for keyboard inactivity before automatically logging out a user. The default auto-logout setting is 15 minutes. Use the **no** form of this command to prevent users from being automatically logged out because of keyboard inactivity.

Parameter	Description
session auto-logout *number-minutes*	Configures — **for this session only** — how long the keyboard can be inactive before automatically logging out a user. Use the **no** form of this command to prevent users from being automatically logged out because of keyboard inactivity.
session paging enable	Configures — **for this session only** — the ability to view text one screen at a time. Paging is enabled, by default. Use the **no** form of this command to prevent parsing of text into individual, sequential screens.
session terminal length *number-lines*	Sets — **for this session only** — the number of lines of text for this terminal. The default terminal length is 24 rows.
session terminal type { **xterm** | **ansi** | **vt100** }	Sets — **for this session only** — the terminal type:**xterm** – Sets terminal type to xterm.__ansi__ – Sets terminal type to ANSI.__vt100__ – Sets terminal type to VT100.The default type is **xterm**. Use the **no** form of the command to clear the terminal type.
session terminal width *number-char*	Sets — **for this session only** — the maximum number of characters in a line.

Defaults

- The default auto-logout setting is 15 minutes.
- Paging is enabled, by default.
- The default terminal length is 24 rows.
- The default terminal type is **xterm**.
- The default number of characters per line is 80.

Examples

To set 1.5 hours as the maximum time a session will last without keyboard activity, for this session only:

```
ECV (config) # cli session auto-logout 75
```

To set the number of lines of text per page to 30 rows:

```
ECV (config) # cli session terminal length 30
```

clock set

Use the **clock set** command to set the system time and/or date.

Command Mode: Global Configuration mode

Syntax

clock set *<hh>:<mm>:<ss>* [*<yyyy>/<mm>/<dd>*]

Arguments

Parameter	Description
<hh>:<mm>:<ss>	Sets the hour, minute, and second of the current time, but leaves the date unchanged. Time is based on a 24-hour clock.
<yyyy>/<mm>/<d	Sets the system's date by year/month/date.

Examples

To set the time and date to exactly one minute after midnight on the morning of August 11, 2007:

```
ECV (config) # clock set 00:01:00 2007/08/11
```

clock timezone

Use the **clock timezone** command to set the time zone for the system.

Use the **no** form of the command to reset the time to its default of Greenwich Mean Time, GMT (also known as UTC).

Command Mode: Global Configuration mode

Syntax

clock timezone *region . . .*
no clock timezone

Arguments

Parameter	Description
region	Specify the region, country, locality, or timezone for the system.

Usage Guidelines

You set the timezone by selecting from a series of menus. To see the list of possible values for timezone, enter the following command:

```
ECV (config) # clock timezone ?
```

The CLI displays a list of world regions, followed by the command prompt:

```
Africa
America
Antarctica
Arctic
Asia
Atlantic_Ocean
Australia
Europe
GMT-offset
Indian_Ocean
Pacific_Ocean
UTC
```

Choose a region from the list and append the region to the command, along with a question mark (?). For example, to specify America, you would enter the following command:

```
ECV (config) # clock timezone America ?
```

The CLI displays the regions in America, such as in the following example:

```
Caribbean
Central
North
South
```

Continue specifying the appropriate menu selections, ending each command with a question mark to display the next menu. When the CLI displays <*cr*>, press **Enter** to complete the command.

The CLI is case-sensitive.

Examples

None

cluster

Use the **cluster** command to configure a cluster of appliances for flow redirection.

Use the **no** form of this command to delete a peer appliance from a cluster.

Command Mode: Global Configuration mode

Syntax

cluster interface *intf-name*
cluster peer *IP-addr-1, IP-addr-2, …., IP-addr-N*
no cluster peer *IP-addr-X*

Arguments

Parameter	Description
interface *intf-name*	Specifies an interface for intra-cluster communication. Generally, Silver Peak recommends using **mgmt1**.
peer *ip-addr-X*	Specifies a comma-delimited list of peer IP addresses. Use the **no** form of the command to delete a peer from a cluster.

Usage Guidelines

If you specify **mgmt1** as the cluster interface, then when created a list of peers, use the **mgmt1** IP addresses in the comma-delimited list.

Examples

To configure **mgmt1** as the cluster interface:

```
ECV (config) # cluster interface mgmt1
```

To create a cluster from appliances with the cluster interfaces, 10.10.10.3, 10.10.20.2, and 10.10.30.5:

```
ECV (config) # cluster peer 10.10.10.3, 10.10.20.2, 10.10.30.5
```

configuration

Use the **configuration** command to manipulate configuration files.

Command Mode: Global configuration mode

Syntax

configuration copy *source-file dest-file*
configuration delete *filename*
configuration download *URL or scp://user:pwd@host/path/filename* [*filename*]
configuration download cancel
configuration factory *filename*
configuration merge *filename*
configuration move *source-file dest-file*
configuration new *filename*
configuration reboot-next *filename*
configuration revert saved
configuration upload { **active** | *filename* } *URL or scp://user:pwd@host/path/filename*
configuration upload cancel
configuration write
configuration write to *filename*

Arguments

Parameter	Description
copy *source-file dest-file*	Makes a copy of a configuration file. Specify, in order, the names of the existing source file and the new destination (configuration) file.
delete *filename*	Deletes the named configuration file. The filename you specify must be one of the configuration files listed on the appliance.
download { *URL* or *scp://user:pwd@host/path/filename* } [*new filename*]	Downloads a configuration file from a remote host. Optionally, you can rename the downloading file.
download cancel	Cancels a configuration file download.
factory *filename*	Creates a new configuration file.
merge *filename*	Merges settings from the specified configuration file to the currently active configuration file.
move *source-file dest-file*	Renames a configuration file. First enter the current file name, followed by the new file name.

Parameter	Description
new *filename*	Creates a new configuration file with all defaults plus active licenses.
reboot-next *filename*	Loads the named configuration file at the next reboot.
revert saved	Reverts to the last saved configuration.
upload *filename* URL* or *scp://user:pwd@host/path/filename*	Uploads an existing, inactive configuration file to a remote host, as specified by a URL or an SCP path.
upload active *URL* or *scp://user:pwd@host/path/filename*	Uploads the currently active configuration file to a remote host, as specified by a URL or an SCP path.
upload cancel	Cancels the configuration file upload.
write	Saves the running configuration to the active configuration file (same as the *write memory*).
write to *filename*	Saves the running configuration to an inactive file and makes that copy the active file.

Usage Guidelines

To display a list of available files, enter the command that displays the required information:

```
ECV (config) # configuration copy ?  \newline
ECV (config) # configuration delete ?  \newline
ECV (config) # configuration merge ?  \newline
ECV (config) # configuration move ?  \newline
ECV (config) # configuration reboot-next ?  \newline
ECV (config) # configuration upload ?
```

Examples

To make a copy of the configuration file, “Texas”, and rename it “Texarkana” (three methods):

```
ECV (config) # configuration copy Texas Texarkana   \newline
ECV (config) # config copy Texas Texarkana   \newline
ECV (config) # copy Texas Texarkana
```

To create a new, clean configuration file named, “wholesale”:

```
ECV (config) # config new wholesale
```

To merge the inactive configuration file, “lanes”, with the currently active configuration file:

```
ECV (config) # config merge lanes
```

To download the configuration file, “horsemen” from the URL, www.apocalypse.com/four/, and keep the original file name:

```
ECV (config) # configuration download www.apocalyse.com/four/horseme
```

To upload the configuration file, “initial.bak” to an account at the remote SCP host, “abcd”, and rename the file to “coyote.bak”:

```
ECV (config) # configuration upload initial.bak scp://root:semi@abcd/tmp/coyote.bk
```

To upload the configuration file, “initial.bak”, to an account at the remote SCP host, 10.0.55.28, and rename the file to “coyotes.bak” at the destination:

```
ECV (config) # configuration upload initial.bak scp://root:semi@10.0.55.28/tmp/coyote.
    bk
```

To rename the local configuration file, “laurel” to “andhardy”:

```
ECV (config) # configuration move laurel andhardy
```

To load the configuration file, “wolves”, at the next reboot:

```
ECV (config) # configuration reboot-next wolves
```

To save the running configuration as a new file named, “newDeployment”, and make it the active configuration:

```
ECV (config) # configuration write to newDeployment
```

configure terminal

Use the **configure terminal** command to enter configuration mode. Use the **no** form of this command to leave the configuration mode.

Command Mode: Privileged EXEC mode (not available in Global configuration mode)

Syntax

configure terminal

Usage Guidelines

To exit the configuration mode, you may also use the **exit** command.

The CLI also accepts these two shortened versions of **configure terminal**:

```
ECV # config t
```

```
ECV # co t
```

As a result, the prompt changes to:

```
ECV (config) #
```

Examples

None

debug generate dump

Use the **debug generate dump** command to generate files that are useful for debugging the system. These are also commonly known as “sysdump” files.

Command Mode: Global configuration mode

Syntax

debug generate dump

Examples

None

disable

Use the **disable** command to exit Privileged EXEC mode.

Command Mode: Privileged EXEC mode (not available in Global configuration mode)

Syntax

disable

Usage Guidelines

When you use the **disable** command, you enter the User EXEC mode.

Examples

To go from Privileged EXEC Mode to User EXEC mode (command followed by result):

```
ECV # disable
ECV  >
```

discoveryd enable / disable

The **discoveryd enable** command enables CDP and LLDP globally.

The **discoveryd disable** command disables CDP and LLDP globally. By default, both protocols are disabled.

Link Layer Discovery Protocol (LLDP) and Cisco Discovery Protocol (CDP) are layer two protocols that allow Ethernet network devices to advertise details about themselves to directly connected devices on the network that use the same protocols. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices, whereas Link Layer Discovery Protocol (LLDP) is an open standard.

Command Mode: Global Configuration mode

Syntax

discoveryd enable
discoveryd disable

Examples

These commands 1) displays the LLDP and CDP enabled status when the protocols are disabled; 2) enables CDP and LLDP through the **discoveryd** command; and 3) displays LLDP and CDP status when they are enabled.

```
ECV-A (config) # show lldp
LLDP is not enabled
ECV-A (config) # show cdp
CDP is not enabled
ECV-A (config) #
ECV-A (config) # discoveryd enable
ECV-A (config) #
ECV-A (config) # show lldp
Global LLDP information:
        Sending LLDP packets every 55 seconds
        Sending a holdtime value of 33 seconds
        Sending LLDPv1 advertisements is enabled
ECV-A (config) # show cdp
Global CDP information:
        Sending CDP packets every 60 seconds
        Sending a holdtime value of 180 seconds
        Sending CDPv2 advertisements is enabled
ECV-A (config) #
```

dns cache

Use the **dns cache** command to configure the DNS cache.

Command Mode: Privileged EXEC mode (dns cache flush)
Command Mode: Global Configuration mode (dns cache http)

Syntax

dns cache flush
dns cache http { **disable** | **enable** }

Arguments

Parameter	Description
flush	Flushes the DNS cache.
http disable	Tells the DNS cache to ignore the HTTP request Host header.
http enable	Tells the DNS cache to use the HTTP request Host header.

Examples

None

enable

Use the **enable** command to enter Privileged EXEC mode.

Command Mode: EXEC mode

Syntax

enable

Usage Guidelines

The CLI also accepts this shortened version of **enable**:

```
ECV > en
```

Examples

To go from User EXEC Mode to Privileged EXEC mode (command followed by result):

```
ECV > enable <br/>
ECV #
```

enable password

Use the **enable password** command to set the password required to enter Privileged EXEC mode.

Use the **no** form of the command to remove the requirement of a password to enter Privileged EXEC mode.

Command Mode: Global Configuration mode

Syntax

enable password *pwd-clear*
no enable password
enable password 0 *pwd-clear*
enable password 7 *pwd-encrypt*

Arguments

Parameter	Description
password *pwd-clear*	Sets the password required to enter enable mode. By default, it will be in cleartext. Use the **no** form of this command to remove the requirement of a password to enter Privileged EXEC mode.
password 0 *pwd-clear*	Sets the enable password with a clear text string.
password 7 *pwd-encrypt*	Sets the enable password with an encrypted string. Encrypted password entries aren't visible when viewing a history of commands.

Usage Guidelines

To require the cleartext password, ratchet, for entering *enable* mode:

```
ECV (config) # enable password 0 ratchet
```

To remove the need for a password for entering *enable* mode:

```
ECV (config) # no enable password
```

Examples

None

excess-flow

Use the **excess-flow** command to manage flows that exceed the number of flows that an appliance supports.

Command Mode: Global configuration mode

Syntax

excess-flow bypass
excess-flow bypass dscp-marking { **enable** | **disable** }
excess-flow drop

Arguments

Parameter	Description
bypass	Bypasses excess flow traffic
dscp-marking enable	Enables excess flow DSCP markings
dscp-marking disable	Disables excess flow DSCP markings
drop	Drops excess flow traffic

Examples

None

exit

Use the **exit** command to log out of the CLI from the User EXEC or Privileged EXEC modes. If you use the exit command from the Global Configuration mode, you enter the Privileged EXEC mode.

Command Mode: All modes

Syntax

exit

Examples

None

fips enable / disable

The **fips enable** command enables FIPS mode and reboots the appliance. By default, FIPs mode is disabled.

The **fips disable** command disables FIPS mode.

Federal Information Processing Standards (FIPS) is a set of publicly announced standards that the National Institute of Standards and Technology (NIST) developed for use in non-military United States government agencies and contractor applications.

Command Mode: Privileged EXEC mode

Syntax

fips enable
fips disable

Usage Guidelines

The **fips enable** and **fips disable** commands commands are not available in ECOS version 9.4.3 and all later versions. Equivalent commands available in these versions are **system fips enable** and **system fips disable**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command enables FIPS mode on the appliance.

```
ECV (config) # fips enable
This operation will cause a system reboot.
Do you want to proceed? [y/n] y
```

fips secure erase

The **fips secure erase** command renders the appliance non-functional by overwriting all data with either zeros or ones. Secure erase prevents unauthorized access to sensitive information when disposing of or selling an appliance. This command provides a zeroization function as required by ISO 24759 and FIPS 140-2 implementation guidance.

Federal Information Processing Standards (FIPS) is a set of publicly announced standards that the National Institute of Standards and Technology (NIST) developed for use in non-military United States government agencies and contractor applications.

Command Mode: Privileged EXEC mode

Syntax

fips secure erase

Usage Guidelines

The **fips secure erase** command is not available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **system fips secure erase**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command renders the appliance non-functional.

```
ECV (config) # fips secure erase
Note: This command zeroizes the drive, rendering the appliance non-functional; ECOS
    will no longer run.
The entire appliance must be sent back to Silver Peak (RMA).
```

fips show

The **fips show** command displays the FIPS enable mode status for the appliance.

Federal Information Processing Standards (FIPS) is a set of publicly announced standards that the National Institute of Standards and Technology (NIST) developed for use in non-military United States government agencies and contractor applications.

Command Mode: Privileged EXEC mode

Syntax

fips show

Usage Guidelines

The **fips show** command is not available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **show system fips**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command displays the FIPS status on a appliance where FIPS is disabled.

```
ECV # fips show

FIPS mode: Disabled
ECV #
```

flow-debug

Use the **flow-debug** command to configure the flow debugging feature to isolate a single flow.

Use the **no** form of this command to remove the previous criteria for isolating a specific flow.

Command Mode: Privileged EXEC mode

Syntax

flow-debug { **disable** | **enable** }

flow-debug flow-id *flow-id*
no flow-debug flow-id *flow-id*

flow-debug ip1 { *ip-addr* | **any** } **ip2** { *ip-addr* | **any** } **protocol** { *1..255* | **any** }
no flow-debug ip1 *ip-addr* **ip2** *ip-addr* **protocol** *1..255*

flow-debug ip1 { *ip-addr* | **any** } **ip2** { *ip-addr* | **any** } **protocol** { *1..255* | **any** } **port1** { *port-no* | **any** } **port2** { *port-no* | **any** }

no flow-debug ip1 *ip-addr* **ip2** *ip-addr* **protocol** *1..255* **port1** *port-no* **port2** *port-no*

flow-debug reset

Arguments

Parameter	Description
disable	Disables flow debugging feature.
enable	Enables flow debugging feature.
flow-id *flow-id*	Specifies a flow ID for the flow specifier.
ip1 *ip-addr*	Specifies IP1 for the flow specifier.
ip2 *ip-addr*	Specifies IP2 for the flow specifier.
protocol *1..255*	Specifies the protocol for the flow specifier.
port1 *port-no*	Specifies the port number of the first endpoint.
port2 *port-no*	Specifies the port number of the second endpoint.
any	**any** is a wildcard.
reset	Resets flow debugging data.

Usage Guidelines

The **flow-debug** commands let you narrow down to a single flow and then generate output about that flow. You can isolate a flow by using the flow's ID number or by entering specifics about the endpoints, protocol, and/or ports. When more than one flow fit the criteria you specify, then the first match is what displays.

Generally, you first specify the flow, then **enable** it, and finally, use the **show flow-debug** command to generate the informational output.

You can enable and disable at will. Once you've specified a flow, it remains the target flow until you specify another flow.

Examples

None

flow-export

Use the **flow-export** command to configure the export of data to NetFlow collectors.

Command Mode: Global Configuration mode

Syntax

flow-export active-flow-timeout *<1-30 minutes>*

flow-export destination { **1** | **2** } *Collector-IP-addr Collector-port*
no flow-export destination { **1** | **2** }

flow-export { **disable** | **enable** }

flow-export engine-id < *0-255* >

flow-export engine-type < *0-255* >

flow-export traffic-type { *lan-rx* | *lan-tx* | *wan-rx* | *wan-tx* }
no flow-export traffic-type { *lan-rx* | *lan-tx* | *wan-rx* | *wan-tx* }

Arguments

Parameter	Description
active-flow-timeout < *1-30 minutes*>	Specifies the flow-export active flow timeout. The range is 1 to 30 minutes.
destination { 1 | 2 } *Collector-IP-addr Collector-port*	Specifies the IP address and port for the NetFlow collector. You can configure up to two collectors. Use the **no** form of this command to disable the export of NetFlow records to either Collector 1 or Collector 2.
disable	Disables the export of NetFlow records.
enable	Enables the export of NetFlow records.
engine-id < *0-255* >	Specifies the VIP or LC slot number of the flow switching engine.
engine-type < *0-255* >	Specifies the flow-export engine type. They are: - **0** for RP - **1** for VIP/LC.
traffic-type { *lan-rx* | *lan-tx* | *wan-rx* | *wan-tx* }	Specifies which interface to turn on for flow exporting. Use the **no** form of this command to turn off a specific interface's flow exporting.

Defaults

When you enable flow exporting, it defaults to the WAN Tx interface.

Usage Guidelines

The appliance lets you turn on up to four interfaces for flow exporting. However, you must specify each interface by using a separate command.

Examples

To configure NetFlow Collector #2, located at 10.10.10.4, using port 146:

```
ECV (config) # flow-export destination 2 10.10.10.4 146
```

To disable the export of NetFlow records to Collector #1:

```
ECV (config) # flow-export destination 1
```

To turn on the **WAN Tx** and **LAN Rx** interfaces for flow exporting:

```
ECV (config) # flow-export traffic-type wan-tx   \newline
ECV (config) # flow-export traffic-type lan-rx
```

flow-redirection

Use the **flow-redirection** command to configure flow redirection.

Command Mode: Global Configuration mode

Syntax

flow-redirection { **enable** | **disable** }
flow-redirection wait-time < *0 - 500* >

Arguments

Parameter	Description
enable	Enables flow redirection.
disable	Disables flow redirection.
wait-time < *1-500* >	Specifies flow redirection wait time in milliseconds.

Usage Guidelines

Redirection enabled simply enables and disables redirection on the selected appliance.

Examples

None

help

Use the **help** command to view a description of the interactive help system.

Command Mode: EXEC mode

Syntax

help

Examples

```
ECV > help
You may request context-sensitive help at any time by pressing '?'
on the command line.  This will show a list of choices for the
word you are on, or a list of top-level commands if you have not
typed anything yet.

If "<cr>" is shown, that means that what you have entered so far
is a complete command, and you may press Enter (carriage return)
to execute it.

Try the following to get started:
  ?
  show ?
  show c?
  show clock?
  show clock ?
  show interfaces ?     (from enable mode)
ECV >
```

hostname

Use the **hostname** command to set host name for the appliance.

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

Command Mode: Global Configuration mode

Syntax

hostname *name-text*
no hostname

Arguments

Parameter	Description
name-text	Designates the host name for the appliance, not including the domain name.

Usage Guidelines

Hostnames may contain letters, numbers, periods ('.'), and hyphens ('-'), but may not begin with a hyphen. Hostnames may **not** contain spaces.

The hostname is limited to 60 characters.

When you remove the hostname, the system reverts to the identifier assigned before shipping. For example, *silverpeak-2f8598*.

Examples

To rename the appliance to *Chicago*:

```
ECV (config) # hostname Chicago
```

iflabel

Use the **iflabel** command to assign labels to interfaces.

Command Mode: Global Configuration mode

Syntax

iflabel add { **lan-label** | **wan-label** } *label-string-with-no-spaces*
iflabel delete { **lan-label** | **wan-label** } *label-string-with-no-spaces*

Arguments

Parameter	Description
add	Add interface label.
delete	Delete interface label.
lan-label	Add LAN interface label.
wan-label	Add WAN interface label.
label-string-with-no-spaces	Specifies the name of this interface. For example: **video** or **data**.

Usage Guidelines

No spaces allowed in the label string.

Examples

To add a WAN label, *Internet*:

```
ECV (config) # iflabel wan-label internet
```

igmp interface enable

The **igmp interface enable** command enables a specified interface to send IGMP membership requests for a multicast group.

The **no igmp interface enable** command disables the interface from sending IGMP membership requests. By default, interfaces are disabled from sending IGMP membership requests.

Internet Group Management Protocol (IGMP) is a layer 3 protocol that manages multicast group memberships in IPv4 networks for the purpose of directing multicast transmissions to hosts that request them.

Command Mode: Global configuration mode

Syntax

igmp interfaces *intf-name* **enable**
no igmp interfaces *intf-name* **enable**
no igmp interfaces *intf-name*

Parameters

intf-name: The interface that is enabled to send IGMP membership requests.

Examples

These commands enable IGMP on the WAN0 interface and then displays the IGMP status of the interface.

```
ECV (config) # igmp interface wan0 enable
ECV (config) # show igmp interfaces
IfName               Interface-IP Address
wan0                 10.19.156.10
ECV (config) #
```

image boot

Use the **image boot** command to specify which system image to boot by default.

Command Mode: Global configuration mode

Syntax

image boot *partition-number*

Arguments

Parameter	Description
partition-number	Specifies the next boot partition. The partition options are: - **1** Partition 1 - **2** Partition 1 - **next** The partition that is not currently running.

Examples

None

image install

Use the **image install** command to download and install an image file onto the inactive system partition.

Command Mode: Privileged EXEC mode

Syntax

image install *URL* or *scp://username:password@hostname/path/filename*
image install cancel

Arguments

Parameter	Description
URL or *scp://username:password@hostname/path/filename*	Enter the path for the remote host from which to download and install the image file. You can specify the SCP server by IP address or hostname.
install cancel	Cancel the system upgrade.

Usage Guidelines

Software image files are .zip files.

Examples

To download the image file, “image-2.4.0.0_15984.zip”, from the remote SCP host, 10.0.55.28, to the inactive system partition:

```
ECV (config) # image install scp://root:seminole@10.0.55.28/tmp/image-2.4.0.0_15984.
    zip
```

image upgrade

Use the **image upgrade** command to download, install, and reboot using a new image file.

Command Mode: Privileged EXEC mode

Syntax

image upgrade *URL* or *scp://username:password@hostname/path/filename*

Arguments

Parameter	Description
URL or *scp://username:password@hostname/path/filename*	Enter the path for the remote host from which to download and install the image file. You can specify the SCP server by IP address or hostname.

Usage Guidelines

Software image files are .zip files.

Examples

To download the image file, “image-2.4.0.0_45678.zip”, from the remote SCP host, 10.0.55.44, to the inactive system partition, install it, and reboot to using it:

```
ECV (config) # image upgrade scp://root:seminole@10.0.55.44/tmp/image-2.4.0.0_45678.
    zip
```

interface cdp enable / disable

The **interface cdp enable** command enables CDP on a specified interface. CDP must be enabled on the appliance before it can be enabled on interfaces. The **discoveryd** command enables CDP on the appliance.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

The **interface cdp disable** command disables CDP on the specified interface. By default, CDP is disabled on all interfaces.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **cdp enable**
interface *intf-name* **cdp disable**

Parameters

intf-name: Name of the interface where CDP is enabled or disabled.

Examples

This command enables CDP on the LAN0 interface.

```
ECV-A (config) # interface lan0 cdp enable
ECV-A (config) # show interface lan0 cdp
CDP is enabled on interface lan0
ECV-A (config) #
```

interface dhcp

Use the **interface dhcp** command to enable Dynamic Host Configuration Protocol (DHCP) for this interface.

Use the **no** form of this command to disable DHCP for this interface.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **dhcp**
interface *intf-name* **dhcp renew**
no interface *intf-name* **dhcp**

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
renew	Renews DHCP for this interface.

Usage Guidelines

To see a list of the available interface names you may use, enter the following command:

```
ECV (config) # interface ?
```

Examples

None

interface inbound-max-bw

Use the **interface inbound-max-bw** command to configure the maximum bandwidth for inbound traffic.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **inbound-max-bw** *BW-in-kbps*

Arguments

Parameter	Description
BW-in-kbps	Specifies the bandwidth in kilobits per second.

Examples

None

interface ip address

The **interface ip address** command configures IP address and netmask for a specified interface.

The **no interface ip address** command erases the IP address and netmask for a specified interface.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **ip-address** *ip-addr-netmask*
interface *intf-name* **ip address** *ip-addr-netmask* **nexthop** *ip-addr*
interface *intf-name* **ip address** *ip-addr-netmask* **nexthop** *ip-addr* **portlist** *port-list-num*
no interface *intf-name* **ip address**

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
ip-addr-netmask	Specifies the source IPv4 address and netmask in standard or slash notation. For example, *10.2.0.0 0.0.255.255* could be entered as *10.2.0.0 /16.*
nexthop *ip-addr*	Next-hop address for this interface. It continues the IP format (IPv4 or IPv6) of the address for which it is the next hop.
portlist *port-list-num*	Configures the ports for this bridge interface. For example: *lan0,wan0* or *tlan0,tlan1,twan0,twan1*.

Usage Guidelines

To see a list of the available interface names you may use, enter the following command:

```
ECV (config) # interface ?
```

interface label

Use the **interface label** command to configure a label for the interface.

Use the **no** form of this command to remove the label from this interface.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **label** *label-string*
no interface *intf-name* **label**

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
label *label-string*	Specifies the label given to the interface. For example, **internet** or **voice**.

Examples

None

interface lldp enable / disable

The **interface lldp enable** command enables Link Layer Discovery Protocol (LLDP) on the specified interface. LLDP must be enabled on the appliance before it can be enabled on interfaces. The **discoveryd** command enables LLDP on the appliance.

The Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can include device configuration, capabilities, and identification.

The **interface lldp disable** command disables LLDP on the specified interface. By default, LLDP is disabled on all interfaces.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **lldp enable**
interface *intf-name* **lldp disable**

Parameters

intf-name: Name of the interface where LLDP is enabled or disabled.

Examples

This command enables LLDP on the LAN0 interface.

```
ECV-A (config) # interface lan0 lldp enable
ECV-A (config) # show interface lan0 lldp
LLDP is enabled on interface lan0
ECV-A (config) #
```

interface mac address

Use the **interface mac address** command to configure the MAC (Media Access Control) address for a selected interface.

Use the **no** form of this command to erase the MAC address for this interface.

NOTE This command is not supported on any Silver Peak hardware appliance.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **mac address** *MAC-addr-of-interface-to-use*
no interface *intf-name* **mac address**

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
mac address *MAC-addr-of-interface-to-use*	Specifies the MAC address.

Examples

None

interface mtu

Use the **interface mtu** command to configure MTU (Maximum Transmission Unit) for this interface.

Use the **no** form of this command to reset the MTU for this interface to its default.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **mtu** *MTU-bytes*
no interface *intf-name* **mtu**

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
mtu *MTU-bytes*	In bytes, the largest size packet that can be sent. The range is 700 to 2400.

Defaults

The default MTU is **1500**.

Usage Guidelines

To see a list of the available interface names you may use, enter the following command:

```
ECV (config) # interface ?
```

Examples

None

interface outbound-max-bw

Use the **interface outbound-max-bw** command to configure maximum bandwidth for outbound traffic.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **outbound-max-bw** *BW-kbps*

Arguments

Parameter	Description
BW-kbps	Specifies the bandwidth in kilobits per second.

Examples

None

interface pass-through

Use the **interface pass-through** command to configure the pass-through parameters for the WAN interface.

Command Mode: Global Configuration mode

Syntax

interface pass-through { **max-bandwidth** *bw-kbps* | **min-bandwidth** *bw-kbps* }

Arguments

Parameter	Description
max-bandwidth *bw-kbps*	Configures maximum bandwidth in kilobits per second.
min-bandwidth *bw-kbps*	Configures minimum bandwidth in kilobits per second.

Usage Guidelines

If you try to configure too high a maximum bandwidth, the CLI returns a message telling you what the maximum allowable value is, given the configured System Bandwidth.

Examples

To set the maximum bandwidth for pass-through traffic at the wan0 interface to 9000 kilobits per second:

```
ECV (config) # interface pass-through max-bandwidth 9000
```

interface security-mode

Use the **interface security-mode** command to configure the firewall mode.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **security-mode** { **0** | **1** | **2** | **3** }

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
security-mode { **0** | **1** | **2** | **3** }	The following firewall modes are expressed as integers: **0** - Open **1** - Hardened **2** - Stateful firewall **3** - Stateful firewall with Source NAT

Examples

None

interface shutdown

Use the **interface shutdown** command to disable an interface.

Use the **no** form of this command to enable this interface.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **shutdown**
no interface *intf-name* **shutdown**

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.

Usage Guidelines

To see a list of the available interface names you may use, enter the following command:

```
ECV (config) # interface ?
```

Examples

None

interface speed-duplex

Use the **interface speed-duplex** command to configure the speed and duplex of this interface.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **speed-duplex** *speed-duplex*

Arguments

Parameter	Description
intf-name	Specifies the name of this interface.
speed-duplex	Specifies the speed and duplex of this interface. Use one of the following settings, depending on your appliance model: **auto/auto** **10/full** **100/full** **1000/full** **10000/full**

Usage Guidelines

To see a list of the available interface names you may use, enter the following command:

```
ECV (config) # interface ?
```

Examples

None

interface tunnel admin

Use the **interface tunnel admin** command to configure the tunnel administrative mode.

Use the **no** form of this command to reset the tunnel administrative mode to default.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **admin** { **up** | **down** }
no interface tunnel *tunnel-name* **admin**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
up	Enables the tunnel.
down	Disables the tunnel.

Defaults

The default for Admin is **down**.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

To enable the tunnel, *Rosenkrantz*, for diagnostics only:

```
ECV (config) # interface tunnel Rosenkrantz admin diag
```

interface tunnel alias

Use the **interface tunnel alias** command to configure an alias for the tunnel for display purposes.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **alias** *tunnel-alias*

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
tunnel-alias	Specifies the alias to display for this tunnel.

Examples

None

interface tunnel bind-tunnel

Use the **interface tunnel bind-tunnel** command to bind a tunnel to a bonded tunnel.

Use the **no** form of this command to unbind a tunnel from a bonded tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **bind-tunnel** *tunnel-name*
no interface tunnel *tunnel-name* **bind-tunnel** *tunnel-name*

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.

Examples

None

interface tunnel control-packet

Use the **interface tunnel control-packet** command to configure the appliance's tunnel health and control packets.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **control-packet dscp** *DSCP-mark-for-tunnel*

The default (and recommended) tunnel health DSCP setting is **be**.

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
dscp *DSCP-mark-for-tunnel*	Specifies the DSCP option for the tunnel's control packets: **af11** AF11 dscp(001010) **af12** AF12 dscp(001100) **af13** AF13 dscp(001110) **af21** AF21 dscp(010010) **af22** AF22 dscp(010100) **af23** AF23 dscp(010110) **af31** AF31 dscp(011010) **af32** AF32 dscp(011100) **af33** AF33 dscp(011110) **af41** AF41 dscp(100010) **af42** AF42 dscp(100100) **af43** AF43 dscp(100110) **be** BE dscp(000000) **cs1** CS1 dscp(001000) **cs2** CS2 dscp(010000) **cs3** CS3 dscp(011000) **cs4** CS4 dscp(100000) **cs5** CS5 dscp(101000) **cs6** CS6 dscp(110000) **cs7** CS7 dscp(111000) **ef** EF dscp(101110)

Examples

None

interface tunnel create

Use the **interface tunnel create** command to create a tunnel interface.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **create** *ip-addr-local ip-addr-remote*

interface tunnel *tunnel-name* **create** *ip-addr-local ip-addr-remote MinBW-kbps* { *MaxBW-kbps* | **auto** } [**gre** | **gre_sp** | **gre_ip** | **udp** | **udp_sp** | **no_encap**]

interface tunnel *tunnel-name* **create** *ip-addr-local ip-addr-remote MinBW-kbps* **unshaped**

interface tunnel *tunnel-name* **create** *ip-addr-appliance ip-addr-remote*

interface tunnel *tunnel-name* **create** *ip-addr-appliance ip-addr-remote MinBW-kbps* { *MaxBW-kbps* | **auto** }

interface tunnel *tunnel-name* **create** *bonded-tunnel tag-name overlay-name* [*bonded-id overlay-ID*]

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
ip-addr-local	Specifies the IP address of the local appliance.
ip-addr-remote	Specifies the IP address of the remote appliance.
MinBW-kbps	Specifies the tunnel's minimum bandwidth in kilobits per second.
MaxBW-kbps	Specifies the tunnel's maximum bandwidth in kilobits per second.
ip-addr-appliance	Specifies the remote IP address for this tunnel.
auto	Auto-negotiates maximum bandwidth in kilobits per second.
bonded-tunnel tag-name *overlay-name*	Specifies a tag name for a bonded tunnel.
bonded-id overlay-ID	Specifies the overlay ID for a bonded tunnel.
unshaped	No traffic shaping on this tunnel

Parameter	Description
[**gre** | **gre_sp** | **gre_ip** | **udp** | **udp_sp** | **no_encap**]	Choose from one of the following tunnel types: **gre** Specifies the Generic Routing Encapsulation (GRE) mode. (legacy term) **gre_sp** Specifies the Generic Routing Encapsulation (GRE) mode. (current term) **gre_ip** Specifies a standard GRE pass-through tunnel to a third-party device. **udp** Specifies the User Datagram Protocol (UDP) mode. (legacy term) **udp_sp** Specifies the User Datagram Protocol (UDP) mode. (current term) **no_encap** Specifies no encapsulation. Use if the service doesn't support GRE.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

To remove a tunnel interface, enter the following command:

```
ECV (config) # no interface tunnel tunnel-name
```

To remove a tunnel, enter the following command:

```
ECV (config) # no interface tunnel tunnel-name
```

Examples

None

interface tunnel gre-protocol

Use the **interface tunnel gre-protocol** command to configure the GRE protocol ID for a tunnel.

Use the **no** form of this command to reset the GRE protocol ID for this tunnel to its default.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **gre-protocol** *Layer-2-protocol-ID*
no interface tunnel *tunnel-name* **gre-protocol**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
Layer-2-protocol-ID	Specifies the Layer-2 protocol ID in the GRE header (decimal). For example, **2048** for **IP**.

Defaults

The default Layer-2 protocol ID in the GRE header (decimal) is **2048**.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

None

interface tunnel ipsec

Use the **interface tunnel ipsec** command to create IPSec (Internet Protocol Security) options for this tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **ipsec auth-algorithm** { **default** | **sha1** | **sha256** | **sha384** | **sha512** }

interface tunnel *tunnel-name* **ipsec crypto-algorithm** { **default** | **aes128** | **aes256** }

interface tunnel *tunnel-name* **ipsec** { **disable** | **enable** }

interface tunnel *tunnel-name* **ipsec enable preshared-key** *key-text*

interface tunnel *tunnel-name* **ipsec enable preshared-key** *key-text* **crypto-algorithm** { **default** | **aes128** | **aes256** } [**auth-algorithm** { **default** | **sha1** | **sha256** | **sha384** | **sha512** }]

interface tunnel *tunnel-name* **ipsec preshared-key** *key-text*

interface tunnel *tunnel-name* **ipsec enable replay-check-window** { **64** | **1024** | **disable** | **auto** }

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
auth-algorithm { **default** | **sha1** | **sha256** | **sha384** | **sha512** }	Configures auth algorithm for IPSec for this tunnel.
crypto-algorithm { **default** | **aes128** | **aes256** }	Configures crypto algorithm for IPSec for this tunnel.
disable	Disables IPSec for this tunnel.
enable	Enables IPSec for this tunnel.
preshared-key *key-text*	Configures preshared key for IPSec for this tunnel.

Parameter	Description
replay-check-window { **64** | **1024** | **disable** | **auto** }	Configures the IPSec anti-replay-check window for this tunnel. The IPSec Anti-replay window provides protection against an attacker duplicating encrypted packets by assigning a unique sequence number to each encrypted packet. The decryptor keeps track of which packets it has seen on the basis of these numbers.The default window size is **64** packets.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Configurable IPSEC anti-replay Window

In environments with significant out-of-order packet delivery, IPSec may drop packets that are outside of the anti-replay window.

- To determine whether packets are falling outside of the antireplay window, execute the following CLI command:

  ```
  ECV (config) # show interfaces tunnel <tunnel name> stats ipsec
  ```

  and look for increases in “Total bytes dropped in replay check”.

- To change the IPSec anti-replay window, use the following CLI command:

  ```
  ECV (config) # interface tunnel <tunnel-name> ipsec replay-check-window < 64 |
      1024 | disable | auto >
  ```

Examples

None

interface tunnel max-bandwidth

Use the **interface tunnel max-bandwidth** command to configure maximum bandwidth for this tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **max-bandwidth** { *kbps* | **auto** }

Arguments

Parameter	Description
tunnel *tunnel-name*	Specifies the name for this tunnel.
max-bandwidth *kbps*	Specifies the maximum bandwidth in kilobits per second for this interface tunnel. The value must be a number between *0* and *4294967295*.
max-bandwidth auto	Auto-negotiates the maximum bandwidth in kilobits per second for this interface tunnel.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

None

interface tunnel min-bandwidth

Use the **interface tunnel min-bandwidth** command to configure minimum bandwidth for this tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **min-bandwidth** *kbps*

Arguments

Parameter	Description
tunnel *tunnel-name*	Specifies the name for this tunnel.
min-bandwidth *kbps*	Specifies the minimum bandwidth in kilobits per second for this interface tunnel. The value must be a number between *0* and *4294967295*.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

None

interface tunnel mode

The **interface tunnel mode** command configures the encapsulation mode for a specified tunnel as either GRE or UDP.

Use the **no** form of this command to reset the mode for this tunnel to its default.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **mode** { **gre** | **udp** }
no interface tunnel *tunnel-name* **mode**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
gre	Specifies the Generic Routing Encapsulation (GRE) mode. (legacy term)
gre_sp	Specifies the Generic Routing Encapsulation (GRE) mode. (current term)
gre_ip	Specifies a standard GRE pass-through tunnel to a third-party device.
udp	Specifies the User Datagram Protocol (UDP) mode. (legacy term)
udp_sp	Specifies the User Datagram Protocol (UDP) mode. (current term)
no_encap	Specifies no encapsulation. Use if the service doesn't support GRE.

Defaults

The default mode is **gre**.

Examples

To configure the tunnel, *Paris_London*, for UDP mode:

```
ECV (config) # interface tunnel Paris_London mode udp
```

To reset the tunnel, *Paris_London*, to the default mode, GRE:

```
ECV (config) # no interface tunnel Paris_London mode
```

interface tunnel mtu

Use the **interface tunnel mtu** command to configure Maximum Transmission Unit (MTU) for this tunnel.

Use the **no** form of this command to reset the MTU for this tunnel to its default.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **mtu** { *MTU-bytes* | **auto** }
no interface tunnel *tunnel-name* **mtu**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel. The range is 700 to 2400.
MTU-bytes	Specifies the Maximum Transmission Unit (MTU) in bytes.
auto	Sets MTU automatically.

Defaults

The default MTU is **1500**.

Examples

None

interface tunnel nat-mode

Use the **interface tunnel nat-mode** command to configure a NAT (Network Address Translation) mode for the tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel nat-mode { **none** | **snat** }

Arguments

Parameter	Description
none	Configures with no NAT.
snat	Applies Source-NAT to all outbound traffic.

Examples

None

interface tunnel packet

Use the **interface tunnel packet** command to configure packet options for this tunnel.

Use the **no** form of this command to negate or reset the packet options for this tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **packet coalesce** { **disable** | **enable** }

interface tunnel *tunnel-name* **packet coalesce wait** *TIME-msecs*
no interface tunnel *tunnel-name* **packet coalesce wait**

interface tunnel *tunnel-name* **packet fec** { **disable** | **enable** | **auto** }

interface tunnel *tunnel-name* **packet fec ratio** { **1:1** | **1:10** | **1:2** | **1:20** | **1:5** }
no interface tunnel *tunnel-name* **packet fec ratio**

interface tunnel *tunnel-name* **packet reorder wait** *TIME-msecs*
no interface tunnel *tunnel-name* **packet reorder wait**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
coalesce { **disable** | **enable** }	Disables or enables packet coalescing for this tunnel.
coalesce wait *TIME-msecs*	Specifies the coalesce wait time in milliseconds. The value must be a number between 0 and 65535. Use the **no** form of this command to reset the coalesce wait time to its default.
fec { **disable** | **enable** }	Disables or enables the packet forwarding error correction (FEC) options.
fec auto	Configures the packet forwarding error correction (FEC) options to adjust automatically. When set, it auto-tunes up to the value specified by **fec ratio**.
fec ratio { **1:1** | **1:10** | **1:20** | **1:5** | **1:2** }	Sets the packet forwarding error correction (FEC) ratios to one of the available options: 1:1, 1:10, 1:20, 1:5, or 1:2. Use the **no** form of this command to reset the FEC ratio value to its default.
reorder wait *TIME-msec*	Configures the packet reorder wait time. Use the **no** form of this command to reset the packet reorder wait time to its default.

Defaults

The default packet coalesce wait time is 0 milliseconds. The default packet reorder wait time is 0 milliseconds.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

To reset the packet coalesce wait time for the tunnel, *big-pipe*, to the default value of 0 (zero):

```
ECV (config) # no interface tunnel big-pipe packet coalesce wait
```

interface tunnel peer-name

Use the **interface tunnel peer-name** command to configure the tunnel peer name.

Use the **no** command to reset the passthrough peer name.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **peer-name** *peer-name-text*
no interface tunnel *tunnel name* **peer-name**

Arguments

Parameter	Description
peer-name *peer-name-text*	Names the destination of a tunnel that has no destination IP. That is, a passthrough tunnel.

Examples

None

interface tunnel revert

Use the **interface tunnel revert** command to configure the default values to the factory settings.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **revert**

Arguments

Parameter	Description
tunnel-name	Specifies the name of this tunnel.

Defaults

Factory defaults

Examples

None

interface tunnel tag-name

Use the **interface tunnel tag-name** command to apply a tag name to a tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **tag-name** *tag-name*

Arguments

Parameter	Description
tunnel-name	Specifies the name of this tunnel.
tag-name *tag-name*	Specifies the tunnel by calling out the WAN port names at each end of the tunnel.

Defaults

Factory defaults

Examples

None

interface tunnel threshold

The **interface tunnel threshold** command configures threshold options for this tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **threshold fastfail** { **disable** | **enable** }
interface tunnel *tunnel-name* **threshold fastfail-wait** { **base-ms** *wait-time-ms* | **rtt-x** *multiple-RTT* }
interface tunnel *tunnel-name* **threshold jitter** *jitter-ms*
interface tunnel *tunnel-name* **threshold latency** *latency-ms*
interface tunnel *tunnel-name* **threshold loss** *loss-percentage*
interface tunnel *tunnel-name* **threshold retry-count** *retry-count-number*

Arguments

Parameter	Description
tunnel-name	Specifies the name of this tunnel.
fastfail { **disable** | **enable** }	Disables or enables fast failover for this tunnel.
fastfail-wait base-ms *wait-time-ms*	Configures fast failover wait-times in milliseconds for this tunnel.
fastfail-wait rtt-x *multiple-RTT*	Configures fast failover wait-times in Return Trip Time (RTT) multiples for this tunnel.
jitter *jitter-ms*	Specifies the jitter threshold for this tunnel in milliseconds.
latency *latency-ms*	Specifies the latency threshold for this tunnel in milliseconds.
loss *loss-percentage*	Specifies the loss threshold for this tunnel in percentage.
retry-count *retry-count-number*	Specifies the number of retries.

Defaults

The default number of retries is 10.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

None

interface tunnel traceroute

Use the **interface tunnel traceroute** command to initiate traceroute for this tunnel.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **traceroute**

Arguments

None

Examples

None

interface tunnel udp-flow

Use the **interface tunnel udp-flow** command to configure the number of UDP flows for this tunnel.

Use the **no** form of this command to reset the number of UDP flows for this tunnel to its default.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **udp-flow** *flows*
no interface tunnel *tunnel-name* **udp-flow**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
flows	Sets the number of UDP flows, between 1 and 1024.

Defaults

The default number of flows is 256.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

To set the maximum number of UDP flows for the tunnel, *HastaLaVista*:

```
ECV (config) # interface tunnel HastaLaVista udp-flow 1024
```

To reset the number of UDP flows to the default of 256 for the tunnel, *HastaLaVista*:

```
ECV (config) # no interface tunnel HastaLaVista udp-flow
```

interface tunnel udp-port

Use the **interface tunnel udp-port** command to configure the UDP destination port for this tunnel.

Use the **no** form of this command to reset the UDP destination port for this tunnel to its default.

Command Mode: Global Configuration mode

Syntax

interface tunnel *tunnel-name* **udp-port** *UDP-dest-port*
no interface tunnel *tunnel-name* **udp-port**

Arguments

Parameter	Description
tunnel-name	Specifies the name for this tunnel.
UDP-dest-port	Specifies the UDP destination port for this tunnel.

Defaults

The default UDP destination port is 4163.

Usage Guidelines

To see a list of the available tunnel names you may use, enter the following command:

```
ECV (config) # interface tunnel ?
```

Examples

To make UDP port 407 the destination for the tunnel, *MataHari*:

```
ECV (config) # interface tunnel MataHari udp-port 407
```

interface virtual

Use the **interface virtual** command to create or modify a virtual network interface.

Use the **no** command to remove a virtual network interface.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **virtual** *virtual-intf-type* **username** *PPPoE-username* **password** *PPPoE-pwd* **etherdev** *phy-ether-intf*
no interface *intf-name* **virtual** *virtual-intf-type*

Arguments

Parameter	Description
intf-name	Specifies the name of the interface.
virtual *virtual-intf-type*	The type of virtual interface. Currently, the options are limited to **pppoe** (Point-to-Point over Ethernet).
username *PPPoE-username*	Specifies the PPPoE username. This is required.
password *PPPoE-pwd*	Specifies the PPPoE password. This is required.
etherdev *phy-ether-intf*	Specifies the physical ethernet interface to use for PPPoE. For example, **wan0**, **wan1**, **twan0**, or **twan1**.

Examples

None

interface vrrp (no)

The **no interface vrrp** command deletes a specified VRRP group.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

The **interface vrrp ip** command creates VRRP groups.

Command Mode: Global Configuration mode

Syntax

no interface *intf-name* **vrrp** *vrrp-id*

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

Examples

This command deletes VRRP group 51 on LAN0.

```
ECV-A (config) # no interface lan0 vrrp 51
ECV-A (config) # show vrrp
% There are no configured VRRPs.
ECV-A (config) #
```

interface vrrp admin

The **interface vrrp admin up** command enables a specified VRRP group. A VRRP group is enabled when it is created.

The **interface vrrp admin down** command disables the VRRP group.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

The **interface vrrp ip** command creates VRRP groups.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **admin up**
interface *intf-name* **vrrp** *vrrp-id* **admin down**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

Examples

This command disables VRRP group 51 on the LAN0 interface.

```
ECV-A (config) # interface lan0 vrrp 65 admin down
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 65
  Virtual IP address        : 10.19.157.65
  VRRP Version              : 2
  Admin                     : down
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 1 secs
  Holddown Timer            : 60 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : init
  State Uptime              : 0 days 0 hrs 0 mins 4 secs 603 msecs
  Master State Transitions  : 0
  Master IP address         : 0.0.0.0
  Virtual Mac Address       : 00:00:5e:00:01:41
ECV-A (config) #
```

This command enables VRRP group 51 on the LAN0 interface.

```
ECV-A (config) # interface lan0 vrrp 65 admin up
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 65
  Virtual IP address        : 10.19.157.65
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 1 secs
  Holddown Timer            : 60 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : backup
  State Uptime              : 0 days 0 hrs 0 mins 4 secs 429 msecs
  Master State Transitions  : 0
  Master IP address         : 0.0.0.0
  Virtual Mac Address       : 00:00:5e:00:01:41
ECV-A (config) #
```

interface vrrp authentication

The **interface vrrp authentication** command configures an authentication string for a specified VRRP group. All routers in a VRRP group must use the same authentication string.

When a VRRP packet arrives from another router in the VRRP group, its authentication string is compared to the string configured on the local router. The packet is accepted if the strings match; otherwise the packet is discarded.

The **no interface vrrp authentication** command deletes the authentication string from the specified group. By default, an authentication string is not assigned to a VRRP group.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **authentication** *auth-text*
no interface *intf-name* **vrrp** *vrrp-id* **authentication**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

auth-text: The authentication string. Limited to a maximum of eight characters.

Examples

This assigns the authentication string of “Baseball” to VRRP group 65.

```
ECV-A (config) # interface lan0 vrrp 65 authentication Baseball
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 65
  Virtual IP address        : 10.19.157.65
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 1 secs
  Holddown Timer            : 60 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
```

```
  Current State             : master
  State Uptime              : 0 days 0 hrs 18 mins 49 secs 429 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:41
ECV-A (config) #
```

interface vrrp debug action

The **interface vrrp debug action dump_info** command dumps all data into a log file for a specified VRRP group.

The **interface vrrp debug action clear_stats** command clears debug statistics for a specified VRRP group.

The **interface vrrp debug action mem_stats** command creates a log file of memory usage statistics for a specified VRRP group.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **debug action dump_info**
interface *intf-name* **vrrp** *vrrp-id* **debug action clear_stats**
interface *intf-name* **vrrp** *vrrp-id* **debug action mem_stats**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

Examples

None.

interface vrrp debug packet-trace

The **interface vrrp debug packet-trace** command enables a packet trace for a specified VRRP group to a log file.

The **no interface vrrp debug packet-trace** command disables the packet trace for a specified VRRP group.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **debug packet_trace**
no interface *intf-name* **vrrp** *vrrp-id* **debug packet_trace**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

Examples

None

interface vrrp description

The **interface vrrp description** command associates a text string to a specified VRRP group. The string has no functional impact. The maximum length of the string is 80 characters.

The **no interface vrrp description** command removes the text string association from the specified VRRP group. By default, no description text is associated to a VRRP group.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **description** *desc-text*
no interface *intf-name* **vrrp** *vrrp-id* **description**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

desc-text: Description string text.

Examples

This command associates the text string “abcde” to VRRP group 100 on LAN0.

```
ECV-A (config) # interface lan0 vrrp 100 description abcde
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 200
  Advertisement interval    : 2 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        : abcde
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 200
  Current State             : master
  State Uptime              : 0 days 2 hrs 41 mins 37 secs 320 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:64
ECV-A (config) #
```

interface vrrp ip

The **interface vrrp ip** command modifies the virtual IP address for a specified VRRP group. The command creates the VRRP group when the group does not exist on the appliance. The IP address of the group must be in the same subnet as the IP address of the interface.

Interfaces have a maximum capacity of four VRRP groups.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **ip** *ip-addr*

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

ip-addr: IP address assigned to group (dotted decimal notation). Value range is 0.0.0.0 through 255.255.255.255.

Examples

The first command displays information about the LAN0 interface. The next command creates VRRP group 65 on LAN0, assigning an IP address to the group that is on the same subnet as the LAN0 address.

```
ECV-A (config) # show interface lan0 brief
Interface lan0 state
   Admin up:           yes
   Link up:            yes
   IPv4 address:       10.19.157.10
   Netmask:            255.255.255.0
   IPv6 address:       fe80::20c:29ff:fe96:f667/64
   Secondary address:  10.19.157.66/24 (alias: 'lan0:v33')
   Secondary address:  10.19.157.33/24 (alias: 'lan0:v40')
   Secondary address:  10.19.157.50/24 (alias: 'lan0:v51')
   Secondary address:  10.19.157.111/24 (alias: 'lan0:v60')
   Speed:              10000Mb/s
   Duplex:             full
   Interface type:     ethernet
   MTU:                1500
   HW address:         00:0C:29:96:F6:67
```

```
ECV-A (config) # interface lan0 vrrp 65 ip 10.19.157.94
ECV-A (config) # show vrrp brief
Intf   Grp   Pre   Adv  Group Addr     Version State  Master Addr    Pri Own
lan0   65    yes   1    10.19.157.94   2       backup  0.0.0.0        128 no
ECV-A (config) #
```

interface vrrp preempt

The **interface vrrp preempt** command sets the virtual router preempt mode setting to *enabled* for a specified VRRP group. By default, preempt mode is enabled.

- When preempt mode is enabled, the appliance becomes the master if it has a higher priority than the current master.
- When preempt mode is disabled, the appliance can become the master virtual router only when a master router is not present on the subnet, regardless of VRRP priority settings.

The **no interface vrrp preempt** command sets the VRRP group preempt mode to disabled.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **preempt**
no interface *intf-name* **vrrp** *vrrp-id* **preempt**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

Examples

This command places the preempt mode to “enabled” for VRRP ID 100 on LAN0.

```
ECV-A (config) # interface lan0 vrrp 100 preempt
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 2 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : master
```

```
  State Uptime              : 0 days 1 hrs 43 mins 36 secs 531 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:64
ECV-A (config) #
```

interface vrrp priority

The **interface vrrp priority** command sets the VRRP priority value for a specified VRRP group. Priority values are used to determine the master router for the group.

The router (gateway) with the highest priority setting for a group becomes the master router. The master router controls the group IP address and is responsible for forwarding traffic sent to the address. The **vrrp preempt** command controls the periods when an appliance can become the master router.

The **no interface vrrp priority** command resets the priority to its default value (128).

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **priority** *priority-value*
no interface *intf-name* **vrrp** *vrrp-id* **priority**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

priority-value: The priority value (integer). Valid range is 1 (lowest priority) through 255.

Examples

This command sets the priority value for VRRP group 100 on LAN0 to 200.

```
ECV-A (config) # interface lan0 vrrp 100 priority 200
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 200
  Advertisement interval    : 2 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        :
```

```
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 200
  Current State             : master
  State Uptime              : 0 days 2 hrs 31 mins 22 secs 187 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:64
ECV-A (config) #
```

interface vrrp timers advertise

The **interface vrrp timers advertise** command specifies the Virtual Router Redundancy Protocol (VRRP) advertisement interval for the specified VRRP group. The master router sends advertisement packets to inform other routers in the group of its operational status.

- VRRPv2 – group routers must all be set to the same advertisement interval, measured in seconds.
- VRRPv3 – group routers may be set to different advertisement intervals, measured in centiseconds.

The **no interface vrrp timers advertise** command resets the advertisement interval to one second (VRRPv2) or 100 centiseconds (VRRPv3).

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **timers advertise** *timer-ad*
no interface *intf-name* **vrrp** *vrrp-id* **timers advertise**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

timer-ad: Advertisement interval. Range depends on active VRRP Version:

- Version 2 — Valid range is 1 to 255 (seconds)
- Version 3 — Valid range is 1 to 25500 (centiseconds)

Examples

This command sets the advertisment timer for the VRRP group 100 to 2 seconds. This group uses VRRPv2.

```
ECV-A (config) # interface lan0 vrrp 100 timers advertise 2
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
```

```
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 2 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : master
  State Uptime              : 0 days 0 hrs 42 mins 9 secs 696 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:64
ECV-A (config) #
```

This command sets the advertisment timer for the VRRP group 65 to 1.5 seconds. This group uses VRRPv3.

```
ECV-A (config) # interface lan0 vrrp 65 timers advertise 150
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 65
  Virtual IP address        : 10.19.157.65
  VRRP Version              : 3
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 150 centi-secs
  Holddown Timer            : 60 secs
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : master
  State Uptime              : 1 days 18 hrs 50 mins 24 secs 429 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:41
ECV-A (config) #
```

interface vrrp timers holddown

The **interface vrrp timers holddown** command sets the hold down timer for the specified VRRP group.

The hold down timer is the period a backup router with a higher priority waits before preempting the current master router. This prevents rapid switchovers and allows time for network convergence after a potential failure. All routers within a group must be configured with the same hold down timer.

The **no interface vrrp timers holddown** command resets the hold down timer to its default value (60 seconds).

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **timers holddown** *timer-hd*
no interface *intf-name* **vrrp** *vrrp-id* **timers holddown**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

timer-hd: Hold down timer period (seconds).

Examples

This command sets the hold down timer for VRRP group 100 on LAN0 to 120 seconds.

```
ECV-A (config) # interface lan0 vrrp 100 timers holddown 120
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 1 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        :
```

```
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : master
  State Uptime              : 0 days 0 hrs 11 mins 55 secs 932 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:64
ECV-A (config) #
```

interface vrrp version

The **interface vrrp version** command configures the VRRP protocol version that is implemented for a specified VRRP group. VRRP versions 2 and 3 are supported.

VRRP version 2 (VRRPv2) supports IPv4 addresses. VRRPv3 supports IPv4 and IPv6 addresses, uses a different time unit to measure the interval between advertisement packets, and does not support authentication strings. A VRRP group is configured as VRRPv2 it is created.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

interface *intf-name* **vrrp** *vrrp-id* **version 2**
interface *intf-name* **vrrp** *vrrp-id* **version 3**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Value range is 1 through 255.

Examples

This command configures VRRPv3 on VRRP Group 65 located on LAN0 interface.

```
ECV-A (config) # interface lan0 vrrp 65 version 3
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 65
  Virtual IP address        : 10.19.157.65
  VRRP Version              : 3
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 128
  Advertisement interval    : 100 centi-secs
  Holddown Timer            : 60 secs
  Packet Trace Enabled      : no
  IP Address Owner          : no
  Current Priority          : 128
  Current State             : master
  State Uptime              : 4 days 0 hrs 5 mins 11 secs 55 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:41
ECV-A (config) #
```

ip default-gateway

Use the **ip default-gateway** command to set the default route to the specified next-hop or interface.

Use the **no** form of this command to remove the current default route or all the default routes.

Command Mode: Global Configuration mode

Syntax

ip default-gateway *next-hop-IP-address intf-name*
ip default-gateway *next-hop-IP-address intf-name metric* [*src*]

no ip default-gateway
no ip default-gateway *next-hop-IP-address* [*metric*]

Arguments

Parameter	Description
next-hop-IP-address	Specifies the IP address for the default gateway route.
intf-name	Either **mgmt0** or **mgmt1**. The interface named here forces the next-hop to use the named management interface, binding the next-hop.
metric	Specifies the metric of the subnet. Value must be between 0 and 100. When a peer has more than one tunnel with a matching subnet (for example, in a high availability deployment), it chooses the tunnel with the greater numerical value.
src	Specifies the Source IP to use in the header after the packet reaches the next hop.

Usage Guidelines

The complete command, **no ip default gateway**, removes all the default routes.

Examples

To set the default gateway to 10.10.4.5:

```
ECV (config) # ip default-gateway 10.10.4.5
```

ip domain-list

Use the **ip domain-list** command to add a domain name to use when resolving hostnames.

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

Command Mode: Global Configuration mode

Syntax

ip domain-list *domain-name*
no ip domain-list *domain-name*

Arguments

Parameter	Description
domain-name	Defines a domain name. For example, *silver-peak*.

Examples

To add the domain name, “silver-peak”:

```
ECV (config) # ip domain-list silver-peak
```

ip host

Use the **ip host** command to configure a static hostname or IP address mapping.

Use the **no** form of this command to remove static hostname or IP address mapping.

Command Mode: Global Configuration mode

Syntax

ip host *host-name IP-addr*
no ip host *host-name IP-addr*

Arguments

Parameter	Description
host-name	Defines a static host name for the IP host.
IP-addr	Specifies an IP address for the IP host.

Usage Guidelines

Useful for a URL definition where you want to use a name instead of an IP address.

Examples

To be able to use the name, “redshoes”, for the IP address, 10.10.10.4:

```
ECV (config) # ip host redshoes 10.10.10.4
```

ip mgmt-ip

The **ip mgmt-ip** command configures the source IP address for gateway management services. The source IP must be previously configured on a physical or virtual network interface with its Interface Type set to LAN. Management services include HTTPS, Orchestrator, DHCP Relay, NTP, NetFlow, RADIUS/TACACS+, SNMP, SSH, and Syslog. This setting only takes effect when the mgmt0 interface is down or does not exist.

This command does not apply to Cloud Portal reachability and websocket connections. These connections are established using the source IP address of the interface from which the Cloud Portal and websocket reachability tests are successful.

When Routing Segmentation (VRF) is disabled, this command specifies the source IP address for all management services.

When Routing Segmentation (VRF) is enabled, this command is deprecated by the Management Services feature available on Orchestrator. Therefore, this command only affects the source IP address for management services assigned to the default segment and have their interface set to any.

The **no ip mgmt-ip** command removes the gateway management services configuration from the gateway.

Command Mode: Global Configuration mode

Syntax

ip mgmt-ip *IP-addr*
no ip mgmt-ip

Arguments

Parameter	Description
IP-addr	Specifies an IP address for the IP host.

Defaults

The **ip mgmt-ip** command function is not configured by default.

Examples

None

ip multicast route group

The **ip multicast route group** command either adds an IP address to an existing static multicast group or creates a multicast group that includes a specified IP address. A multicast route group is a collection of hosts that receive a single data stream from a multicast source.

The **no ip multicast route group** command deleted a specified IP address from a static multicast group.

Command Mode: Privileged EXEC mode

Syntax

ip multicast route group *group-addr* **src-ip** *src-addr* **iif** *intf-list*
ip multicast route group *group-addr* **src-ip** *src-addr* **ipeer** *peer-list*

no ip multicast group *group-addr* **src-ip** *src-addr* **iif** *intf-list*
no ip multicast route group *group-addr* **src-ip** *src-addr* **iif** *intf-list*

Parameters

group-addr: IP address of the mulitcast group.

src-addr: IP address added to the multicast group.

intf-list: List of interfaces

peer-list: List of peers.

Examples

This command adds 15.1.1.1 to the multicast group at 12.1.1.1.

```
ECV (config) # ip multicast route group 12.1.1.1 src-ip 15.1.1.1 iif wan0
ip multicast route group * * * * *
grp ip 12.1.1.1
src ip 15.1.1.1
iif wan0
/cn/mrtrd/config/static/grp/12.1.1.1/sip/15.1.1.1/iif
ECV (config) #
```

ip name-server

Use the **ip name-server** command to add a DNS server.

Use the **no** form of this command to remove a DNS server.

Command Mode: Global Configuration mode

Syntax

ip name-server *IP-addr*
no ip name-server *IP-addr*

Arguments

Parameter	Description
IP-addr	Specifies an IP address for the DNS server.

Usage Guidelines

The system allows a maximum of three DNS servers and tells you when you try to request more.

The appliance tries to access DNS servers, as needed, in the order they were configured. Also, if you remove the first host in a list of three, the second host becomes the first host. A newly added host always goes to the bottom of the list.

Examples

To add a Domain Name Server with the IP address, 172.30.56.89:

```
ECV (config) # ip name-server 172.30.56.89
```

ip route

Use the **ip route** command to add a static route. Static routes help the appliance route management traffic out of the appliance to different subnets.

Use the **no** form of this command to remove a static route.

Command Mode: Global Configuration mode

Syntax

ip route *network-prefix mask-length next-hop-IP-addr intf-name* [*metric*]
ip route *network-prefix mask-length next-hop-IP-addr intf-name metric* [*src*]
no ip route *network-prefix mask-length* [*next-hop-IP-addr*]
no ip route *network-prefix mask-length next-hop-IP-addr* [*intf-name*]
no ip route *network-prefix mask-length next-hop-IP-addr intf-name* [*metric*]

Arguments

Parameter	Description
network-prefix	Specifies a network prefix to the IP route. This has the format, nnn.nnn.nnn.0.
mask-length	Specifies a mask length in slash notation.
next-hop-IP-addr	Specifies the next-hop IP address for the IP route.
next-hop-IP-addr intf-name	Binds the next-hop to the named interface, in this case, either **mgmt0** or **mgmt1**.
metric	Specifies the metric of the subnet. Value must be between 0 and 100. When a peer has more than one tunnel with a matching subnet (for example, in a high availability deployment), it chooses the tunnel with the greater numerical value.
src	Specifies the Source IP to use in the header after the packet reaches the next hop.

Examples

None

ip-tracking

The **ip-tracking** command configures IP tracking on the appliance.

The **no ip-tracking** commands disable specified IP tracking objects.

Command Mode: Global Configuration mode

Syntax

ip-tracking action *action-name* **attributes** *text-string*
no ip-tracking action *action-name*

ip-tracking manager *manager-name* { **attributes** *text-string* | **comment** *comment-text* | **disable** | **enable** }
no ip-tracking manager *manager-name*

ip-tracking operation *operation-name* **attributes** *text-string*
no ip-tracking operation *operation-name*

Arguments

Parameter	Description
action *action-name*	Creates an IP Tracking action object.
manager *manager-name*	Creates an IP Tracking manager object.
operation *operation-name*	Creates an IP Tracking operation object.
attributes *text-string*	Configures attributes for an object.
comment-text	Adds comment text.
enable	Enables the IP Tracking manager.
disable	Disables the IP Tracking manager.

Examples

None

license

Use the **license** command to install or remove a license key.

Command Mode: Global configuration mode

Syntax

license delete *license-number*
license install *license-key*
no license install

Arguments

Parameter	Description
delete *license-number*	Removes a license key by ID number.
key *license-key*	Installs a new license key. Use the **no** form of the command to remove license keys.

Examples

None

lldp holdtime

The **lldp holdtime** command configures the Link Layer Discovery Protocol (LLDP) hold time. The hold time is period that the receiver retains LLDP packet information.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.

Command Mode: Global Configuration mode

Syntax

lldp holdtime *hold-period*

Parameters

hold-period: LLDP packet information retention period (seconds). Value range is 10 through 255. Default is 120 seconds.

Examples

This command sets the LLDP hold time to 240 seconds.

```
ECV-A (config) # lldp holdtime 240
ECV-A (config) # show lldp
Global LLDP information:
        Sending LLDP packets every 174 seconds
        Sending a holdtime value of 240 seconds
        Sending LLDPv1 advertisements is enabled
ECV-A (config) #
```

lldp timer

The **lldp timer** command command configures the LLDP timer. The LLDP timer is the interval between the transmission of LLDP packets.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.

Command Mode: Global Configuration mode

Syntax

lldp timer *lldp-rate*

Parameters

lldp-rate: LLDP packet transmission interval (seconds per packet). Range is 5 through 254. Default is 60 seconds per packet.

Examples

This command sets the LLDP timer to 90 seconds.

```
ECV-A (config) # lldp timer 90
ECV-A (config) # show lldp
Global LLDP information:
        Sending LLDP packets every 90 seconds
        Sending a holdtime value of 240 seconds
        Sending LLDPv1 advertisements is enabled
ECV-A (config) #
```

logging

Use the **logging** command to configure event logging to a specific syslog server.

Use the **no** form of this command to abstain from sending event log messages to this server.

Command Mode: Privileged EXEC mode

Syntax

logging *IP-addr*
no logging *IP-addr*

logging *IP-addr* **facility** { *facility-level* | **all** }
no logging *IP-addr* **facility** { *facility-level* | **all** }

logging *IP-addr* **trap** *severity-level*

Arguments

Parameter	Description
logging *IP-addr*	Specifies the IP address to which you want to log events.
facility *facility-level*	Specifically sets the facility for messages to this syslog server to one of the following: Local 0, Local 1, Local 2, Local 3, Local 4, Local 5, Local 6, *or* Local 7
facility all	Specifies all facilities.
trap *severity-level*	Sets the minimum severity of log messages saved to this syslog server. You can choose from the following severity options: **none** Disables logging **emerg** Emergency: system is unusable **alert** Action must be taken immediately **crit** Critical conditions **err** Error conditions **warning** Warning conditions **notice** Normal but significant condition **info** Informational messages **debug** Debug-level messages

Examples

To configure the server, 10.10.4.4, to not receive any event logs:

```
(config) # no logging 10.10.4.4
```

logging facility

Use the **logging facility** command to configure event logging to a specific syslog server.

Command Mode: Global configuration mode

Syntax

logging facility auditlog *facility-level*
logging facility flow *facility-level*
logging facility node { **local0** | **local1** | **local2** | **local3** | **local4** | **local5** | **local6** | **local7** }
logging facility system *facility-level*

Arguments

Parameter	Description
facility-level	Specifically sets the facility for messages to this syslog server to one of the following: Local 0, Local 1, Local 2, Local 3, Local 4, Local 5, Local 6, *or* Local 7
auditlog	Specifies the log facility setting for audit log.
flow	Specifies the log facility setting for flow.
node	Specifies the log facility setting for the node.
system	Specifies the log facility setting for the system.

Examples

None

logging files

Use the **logging files** command to configure settings for local log files.

Command Mode: Global configuration mode

Syntax

logging files rotation criteria frequency { **daily** | **weekly** | **monthly** }
logging files rotation criteria size *size-megabytes*
logging files rotation criteria size-pct *size-percent*
logging files rotation force
logging files rotation max-num *number-files*
logging files upload *filename URL or scp://username:password@hostname/path/filename*
logging files upload cancel

Arguments

Parameter	Description
rotation criteria frequency	Rotates log files on a fixed, time-based schedule: **daily** = once per day at midnight **weekly** = once per week **monthly** = on the first day of every month
rotation criteria size *size-megabytes*	Rotates log files when they surpass a size threshold, in megabytes.
rotation criteria size-pct *size-percent*	Rotates log files when they surpass a specified percentage of /var partition size per log file.
rotation force	Forces an immediate rotation of the log files.
rotation max-num *number-files*	Specifies the maximum amount of log files to keep. The value must be between 0 and 4294967295.
upload *filename*	Specifies which log file to upload to a remote host.
upload *URL* or *scp://username:password@hostname/path/filename*	Determines the path for a remote host. Optionally, you can specify a new destination filename.
upload cancel	Cancels the current asynchronous file upload.

Examples

To delete the four oldest local log files:

```
ECV (config) # logging files delete oldest 4
```

To keep the most recent 350 local log files:

```
ECV (config) # logging files rotation max-num 350
```

To upload the log file, “messages” to an account at the remote SCP host, “ocean”, and rename the file to “messages_April2007”:

```
ECV (config) # logging files upload messages scp://root:seminole@ocean/tmp/
    messagee_April2007
```

To upload the log file, “messages.2.gz” to the URL, www.catchall.com/tmp/, and keep the original file name:

```
ECV (config) # logging files upload messages.2.gz www.catchall.com/tmp/
```

To rotate the log files when the /var partition surpasses 85% per log file:

```
ECV (config) # logging files rotation criteria size-pct 85
```

logging local

The **logging local** command sets minimum severity of log messages saved on the local disk.

Use the **no** form of this command to negate writing event log messages to the local disk.

Command Mode: Global configuration mode

Syntax

logging local *severity-level*
no logging local

Arguments

Parameter	Description
local *severity-level*	Sets the minimum severity of log messages saved on the local disk. You can choose from the following severity options: **none** Disables logging **emerg** Emergency: system is unusable **alert** Action must be taken immediately **crit** Critical conditions **err** Error conditions **warning** Warning conditions **notice** Normal but significant condition **info** Informational messages **debug** Debug-level messages

Examples

To disable local logging of all events related to system resources, use one of the following two commands:

```
ECV (config) # logging local override class system priority none
```

```
ECV (config) # no logging local override class system
```

logging trap

Use the **logging trap** to set the minimum severity of log messages sent to **all** syslog servers.

Use the **no** form of this command to negate sending events to all syslog servers.

Command Mode: Global configuration mode

Syntax

logging trap *severity-level*
no logging trap

Arguments

Parameter	Description
trap *severity-level*	Specifies the minimum severity of log messages sent to all syslog servers. You can choose from the following severity options: **none** Disables logging **emerg** Emergency: system is unusable **alert** Action must be taken immediately **crit** Critical conditions **err** Error conditions **warning** Warning conditions **notice** Normal but significant condition **info** Informational messages **debug** Debug-level messages

Examples

To set the minimum severity level of log messages sent to all syslog servers to “critical”:

```
(config) # logging trap crit
```

monitor

Use the **monitor** command to monitor interface bandwidth statistics.

Command Mode: EXEC mode

Syntax

monitor *intf* [*intf*] [*intf*] [*intf*] [**-t**]

Arguments

Parameter	Description
intf	Specifies the interface name. You can specify up to 4 interfaces.
-t	Optional timestamp

Usage Guidelines

Once you execute the command, the output updates every second. To discontinue, use *Ctrl* **+** *C*.

The available interfaces include:

- wan0
- lan0
- mgmt0
- mgmt1
- wan1
- lan1

Examples

To monitor the lan0 and wan0 interfaces:

```
ECV (config) # monitor lan0 wan0
```

mtr

The **mtr** command probes and reports on routers and their response time on an individual route path.

Command Mode: EXEC mode

Syntax

mtr [-hvrctglspniu46] [–help] [–version] [–report] [–report-wide] [–report-cycles COUNT] [–curses] [–split] [–raw] [–no-dns] [–gtk] [–address IP.ADD.RE.SS] [–interval SECONDS] [–psize BYTES | -s BYTES] HOSTNAME [PACKETSIZE]

Arguments

Parameter	Description
mtr-options	Specifies the type of **mtr**. Select one of the following options: **-h** *help*. Print the summary of command line argument options. **-v** *version*. Print the installed version of **mtr**. **-r** *report*. This option puts mtr into report mode. In this mode, **mtr** runs for the number of cycles specified by the **-c** option, prints statistics, and exit. This mode is useful for generating statistics about network quality. Each running instance of **mtr** generates a significant amount of network traffic. Using **mtr** to measure the quality of your network may result in decreased network performance. **-w** *report-wide*. This option puts **mtr** into wide report mode. When in this mode, **mtr** will not cut hostnames in the report. **-c** *report-cycles COUNT*. Use this option to set the number of pings sent to determine both the machines on the network and the reliability of those machines. Each cycle lasts one second. **-s** *BYTES*, *-psize BYTES*, *-PACKETSIZE*. These options or a trailing PACKETSIZE on the command line sets the packet size used for probing. It is in bytes inclusive IP and ICMP headers. If set to a negative number, every iteration use a different, random packetsize up to that number. **-t** *curses*. Use this option to force **mtr** to use the curses based terminal interface (if available). **-n** *no-dns*. Use this option to force **mtr** to display numberic IP numbers and not try to resolve the host names. **-o** *fields order*. Use this option to specify the fields and their order when loading **mtr**. Example: -o “LSD NBAW” **-g** *gtk*. Use this option to force **mtr** to use the GTK+ based X11 window interface (if available). GTK+ must have been available on the system when **mtr** was built for this to work. See the GTK+ web page at *http://www.gimp.org/gtk/* for more information about GTK+

Parameter	Description
	-p *split*. Use this option to set **mtr** to spit out a format that is suitable for a split-user interface. **-l** *raw*. Use this option to tell **mtr** to use the raw output format. This format is better suited for archival of the measurement results. It could be parsed to be presented into any of the other display methods. **-a** *address IP.ADD.RE.SS*. Use this option to bind outgoing packets' socket to specific interface, so that any packet will be sent through this interface. NOTE that this options doesn't apply to DNS requests (which could be and could not be what you want). **- i** *interval SECONDS*. Use this option to specify the positive number of seconds between ICMP ECHO requests. The default value for this parameter is one second. **-u** Use UDP diagrams instead of ICMP ECHO. **-4** Use IPv4 only. **-6** Use IPv6 only.

Usage Guidelines

mtr combines the functionality of traceroute and ping in a single network diagnostic tool.

mtr probes routers on the route path by limiting the number of hops that individual packets may traverse, and listening to responses of their expiry. It regularly repeats this process, usually once per second, and keep track of the response times of the hops along the path.

mtr combines the functionality of the **traceroute** and **ping** programs in a single network diagnostic tool.

[*from Linux man page*] As **mtr** starts, it investigates the network connection between the host **mtr** runs on and **HOSTNAME**. by sending packets with purposely low TTLs. It continues to send packets with low TTL, noting the response time of the intervening routers. This allows **mtr** to print the response percentage and response times of the internet route to **HOSTNAME**. A sudden increase in packet loss or response time is often an indication of a bad (or simply overloaded) link.

Examples

```
ECV (config) # mtr
                                    My traceroute  [v0.75]
ECV (0.0.0.0)                                                  Tue Sep 21
    02:03:12 2010
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
                                 Packets               Pings
 Host                          Loss%   Snt   Last   Avg  Best  Wrst StDev
 1. localhost                  0.0%    66    0.0   0.0   0.0   0.0   0.0
```

multicast enable / disable

The **multicast enable** command enables multicast on the appliance. Multicast is the transmission of data packets simultaneously to multiple hosts through a common IP address. EdgeConnect appliances support multicast through IGMP and PIM.

The **multicast disable** command disables multicast on the appliance. By default, multicast is disabled on appliances.

Internet Group Management Protocol (IGMP) is a layer 3 protocol that manages multicast group memberships in IPv4 networks for the purpose of directing multicast transmissions to hosts that request them.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Global Configuration mode

Syntax

multicast enable
multicast disable

Examples

This command enables multicast on the appliance.

```
ECV (config) # multicast enable
ECV (config) #
```

multicast filtername

The **multicast filtername** command specifies the address group that is allowed to participate in multicast transmissions. The specified filtername must be configured as an address group.

An address group defines a set of IP addresses and is configured from the Orchestrator UI by navigating to **Configuration > Templates & Policies > ACLs > Address Groups**.

The **no multicast filtername** command removes the specified address group from multicast transmission participation.

Command Mode: Global Configuration mode

Syntax

multicast filtername *filter-name*
no multicast filtername

Parameters

filter-name: The name of the address group assigned as the multicast filter.

Examples

```
ECV (config) # multicast filtername filter-1
ECV (config) #
```

nat-map

The appliance can perform *source network address translation* (Source NAT or SNAT) on inbound or outbound traffic.

Two use cases illustrate the need for NAT:

Inbound NAT. The appliance automatically creates a source NAT map when retrieving subnet information from the Silver Peak Cloud portal. This ensures that traffic destined to SaaS servers has a return path to the appliance from which that traffic originated.

Outbound NAT. The appliance and server are in the cloud, and the server accesses the internet. For example, a Citrix thin client accesses its cloud-based server, and the server accesses the internet.

For deployments in the cloud, **best practice is to NAT all traffic** — either inbound (WAN-to-LAN) or outbound (LAN-to-WAN), depending on the direction of initiating request. This avoids black-holing that can result from cloud-specific IP addressing requirements.

Enabling NAT on inbound traffic applies NAT policies to pass-through traffic as well as optimized traffic, ensuring that black-holing doesn't occur. Enabling NAT on outbound traffic only applies to pass-through traffic.

If Fallback is enabled, the appliance moves to the next IP (if available) when ports are exhausted on the current NAT IP.

In general, when applying NAT policies, configure separate WAN and LAN interfaces to ensure that NAT works properly. You can do this by deploying the appliance in Router mode in-path with two (or four) interfaces.

There are two types of NAT policies:

Dynamic – created automatically by the system for inbound NAT when the **SaaS Optimization** feature is enabled and SaaS service(s) are selected for optimization. The appliance polls the Silver Peak Unity Cloud Intelligence service for a directory of SaaS services, and NAT policies are created for each of the subnets associated with selected SaaS service(s), ensuring that traffic destined for servers in use by those SaaS services has a return path to the appliance.

Manual – created by the administrator for specific IP addresses / ranges or subnets. When assigning priority numbers to individual policies within a NAT map, first view dynamic policies to ensure that the manual numbering scheme doesn't interfere with dynamic policy numbering (that is, the manually assigned priority numbers cannot be in the range: 40000-50000). The default (no-NAT) policy is numbered 65535.

NAT maps are comprised of ordered entries. Each map entry consists of a *match* statement paired with a *set* action. Set actions are specific to the type of map.

A NAT map entry can match traffic that satisfies either a pre-defined ACL or any of the following attributes:

- ICMP or IP Protocol
- Source IP Address / Subnet
- Destination IP Address / Subnet
- Application (standard or user-defined, or a user-defined application group)
- Source Port Number

- Destination Port Number
- DSCP value
- VLAN

If you want to reuse the same match criteria in more than one map, you can pre-define ACLs, which are, essentially, reusable match statements.

Set actions are specific to the type of map. A NAT map has set actions for the following features:

- NAT type
- NAT direction
- NAT IP
- Fallback

Map entries are ordered according to their assigned *priorities*. Priorities identify, as well as order, entries within a map. Across entries, all priority values must be unique (in other words, no two *entries* in a given map can have the same priority value). match

In the following example, we'll add a new entry, with a priority of *50*, to the default map, *map1*. The first statement matches all traffic associated with the application, *AOL*. The second statement causes the source address and the source port to change in the IP header of that inbound traffic:

```
ECV (config) # nat-map map1 50 match app aol
ECV (config) # nat-map map1 50 set nat-type source-nat direction inbound
```

If you enter a new priority statement for an existing map, the CLI adds that entry to the map. However, if the map already has a *match* or *set* statement with the same priority, the new entry overwrites the previous one (and the CLI does not provide a warning).

If you want to create a new map, the CLI creates the map the first time you name it in a match statement.

Every map automatically includes a default entry with the priority, 65535, the highest possible number.

By default, one map is always active. You can change the active map at any time, simply by activating a different map.

nat-map (no)

Use the **no nat-map** command to delete a Network Address Translation (NAT) map or a specific priority entry from a NAT map.

Command Mode: Global Configuration mode

Syntax

no nat-map *map-name*
no nat-map *map-name priority-value*

Arguments

Parameter	Description
map-name	Specifies which NAT map.
priority-value	Designates a priority value for the NAT map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.

Defaults

None

Usage Guidelines

You can only delete a NAT map if it's inactive. Therefore, to delete the active NAT map, you must first activate a different NAT map. For example:

```
ECV (config) # nat-map map3 activate
ECV (config) # no nat-map map3
```

You can also delete a specific entry in a NAT map by using the **no nat-map** command and specifying a priority value. For example, the following statement deletes the priority *100* entry (*match* and *set* statements) from the NAT map, *fred*:

```
ECV (config) # no nat-map fred 100
```

nat-map activate

Use the **nat-map activate** command to activate an inactive NAT map.

Command Mode: Global Configuration mode

Syntax

nat-map *map-name* **activate**

Arguments

Parameter	Description
map-name	Specifies which existing, inactive NAT map.

Usage Guidelines

Only one NAT map can be active at a time. The Silver Peak appliance has a default NAT map, **map1**, that's active until you create and activate a new NAT map.

Examples

None

nat-map comment

Use the **nat-map comment** command to add a comment for a specified NAT map entry.

Command Mode: Global Configuration mode

Syntax

nat-map *map-name priority-value* **comment** *comment-text*

Arguments

Parameter	Description
map-name	Specifies the name of the NAT map.
priority-value	Designates a priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.
comment-text	Specifies the text used for the comment.

Examples

None

nat-map match

Use the **nat-map match** command to create a NAT map entry that uses match criteria to delineate traffic. Also use this command to change the matching conditions associated with an existing entry.

Command Mode: Global Configuration mode

Syntax

nat-map *map-name priority-value* **match acl** *ACL-name*

nat-map *map-name priority-value* **match app** *app-name*

nat-map *map-name priority-value* **match dscp** { **any** | *dscp-value* }

nat-map *map-name priority-value* **match matchstr** *match-string*

nat-map *map-name priority-value* **match protocol icmp** { *source-IP-addr-mask* | **any** | **any-ipv4** | **any-ipv6** } { *dest-IP addr-mask* | **any** | **any-ipv4** | **any-ipv6** } [**dscp** { **any** | *dscp-value* }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

nat-map *map-name priority-value* **match protocol ip** { *source-IP-addr-mask* | **any** | **any-ipv4** | **any-ipv6** } { *dest-IP addr-mask* | **any** | **any-ipv4** | **any-ipv6** } [**app** *app-name*] [**dscp** { **any** | *dscp-value* }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

nat-map *map-name priority-value* **match vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }

Arguments

Parameter	Description
map-name	Specifies the name of the NAT map.
priority-value	Designates a priority value for the map entry. Acceptable values are from *1* to *65534*. By default, the appliance reserves *65535* for the default entry.
match acl *ACL-name*	Creates an entry that uses an existing ACL to match traffic. Also use this command to change the ACL associated with an existing entry.
match app *app-name*	Creates an entry that uses a built-in or user-defined application—or an application group—to match traffic. Also use this command to change the application associated with an existing entry.
match dscp { *dscp-value* | **any** }	Creates or modifies an entry that matches traffic with a specific DSCP marking. You can use any of the following values:af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, *or* ef. **any** is a wildcard.

Parameter	Description
match matchstr *match-string*	Creates or modifies a NAT map that matches a string.
match protocol icmp { *source-IP-addr-mask* | **any** | **any-ipv4** | **any-ipv6** }	Creates or modifies a NAT map that matches the ICMP protocol. **any** matches any IPv4 or IPv6 address **any-ipv4** matches any IPv4 address **any-ipv6** matches any IPv6 address
match protocol ip { *source-IP-addr-mask* | **any** | **any-ipv4** | **any-ipv6** }	Creates or modifies a NAT map that matches the IP protocol. **any** matches any IPv4 or IPv6 address **any-ipv4** matches any IPv4 address **any-ipv6** matches any IPv6 address
match vlan { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }	Creates or modifies an entry that matches an interface and 802.1q VLAN tag. The available values include: *1..4094* the number assigned to a VLAN *intf.tag* as in **lan0.10** *any.tag* as in **any.10** *intf.any* as in **lan0.any** *intf.native* as in **lan0.native** **any** is a wildcard
source-IP-addr-mask	Specifies the source IP address and netmask in slash notation. For example, 192.1.2.0/24 or 2001:db8::/32
dest-IP addr-mask	Specifies the destination IP address and netmask in slash notation. For example, 192.1.2.0/24 or 2001:db8::/32.

Examples

None

nat-map modify-priority

Use the **nat-map modify-priority** commands to modify an existing NAT map priority value.

Command Mode: Global Configuration mode

Syntax

nat-map *map-name current-priority-value* **modify-priority** *new-priority-value*

Arguments

Parameter	Description
map-name	Specifies an existing NAT map.
current-priority-value	Specifies the current priority value for the entry you want to change.
modify-priority *new-priority-value*	Designates the new priority for this entry. This new priority value must be unique and between 1 to 65534.

Defaults

None

Usage Guidelines

If you try renumber the entry to a priority number that already exists, the CLI informs you that that's the case and that you can't make that modification.

Examples

To change the priority of entry 40 to be 60 for the map, *map1*:

```
ECV (config) # nat-map map1 40 modify-priority 60
```

nat-map set

Use the **nat-map set** command specifies or modifies an entry's action. You cannot create a **set** command for an entry until you first issue a **match** command.

Command Mode: Global Configuration mode

Syntax

nat-map *map-name priority-value* **set nat-type source-nat direction** { **inbound** | **outbound** | **none** }

nat-map *map-name priority-value* **set nat-type source-nat direction inbound nat-ip** { *intf-IP-addr* | **auto** | **tunnel_endpoint** } **fallback** { **enable** | **disable** }

nat-map *map-name priority-value* **set nat-type source-nat direction outbound nat-ip** { *intf-IP-addr* | **auto** } **fallback** { **enable** | **disable** }

nat-map *map-name priority-value* **set nat-type source-nat direction none nat-ip** { *intf-IP-addr* | **auto** } **fallback** { **enable** | **disable** }

nat-map *map-name priority-value* **set nat-type no-nat direction inbound nat-ip** { *intf-IP-addr* | **auto** | **tunnel_endpoint** } **fallback** { **enable** | **disable** }

nat-map *map-name priority-value* **set nat-type no-nat direction outbound nat-ip** { *intf-IP-addr* | **auto** } **fallback** { **enable** | **disable** }

nat-map *map-name priority-value* **set nat-type no-nat direction none nat-ip** { *intf-IP-addr* | **auto** } **fallback** { **enable** | **disable** }

Arguments

Parameter	Description
nat-map *map-name*	Specifies the name of the NAT map.
priority-value	Designates a priority value for the map entry. Acceptable values are from *1* to *65534*. By default, the appliance reserves *65535* for the default entry.
set	Configures the NAT map with the arguments that follow.
nat-type	Specifies the NAT type.
source-nat	Specifies the Source NAT on traffic coming into the LAN.
no-nat	Disables NAT on all traffic.
direction	Specifies the NAT direction: **inbound** Applies NAT to traffic coming into LAN. **outbound** Applies NAT to traffic going out into WAN. **none** Disables NAT.

Parameter	Description
nat-ip *intf-IP-addr*	Specifies the NAT IP address. To display the existing interface addresses, you can type, **nat-ip ?**
nat-ip { **auto** | **tunnel_endpoint** }	Specifies how the system should choose the NAT IP address.
fallback enable	Specifies fallback to the next available NAT IP address upon port exhaustion with the current NAT IP address.
fallback disable	Specifies not to fallback to the next available NAT IP address upon port exhaustion.

Defaults

The default is **no** network address translation.

Usage Guidelines

You cannot create a **set** command for an entry until you first issue a **match** command. And, until you create a **set** command, no Set Actions exist for that entry's priority.

Examples

None

ntp

Use the **ntp** commands to configure Network Time Protocol (NTP) on the appliance.

Use the **no** forms of the command to negate certain NTP options.

Command Mode: Privileged EXEC (ntp status command)
Command Mode: Global configuration mode (all other ntp commands)

Syntax

ntp { **disable** | **enable** }
no ntp { **disable** | **enable** }

ntp server *IP-addr*
no ntp server *IP-addr*

ntp status *<remote> <refid> <st> <t> <when> <poll> <reach> <delay> <offset> <jitter>*

ntp server *IP-addr* **version** *ver-number*

ntp server *IP-addr* **disable**
no ntp server *IP-addr* **disable**

ntp status

Arguments

Parameter	Description
disable	Disables NTP on the appliance.
enable	Enables NTP on the appliance.
server *IP-addr*	Configures the NTP server node with the default NTP version number.Use the **no** form of this command to remove this NTP server.
ntp status *<remote> <refid> <st> <t> <when> <poll> <reach> <delay> <offset> <jitter>*	Checks the connectivity of this NTP server.
server *IP-addr* **version** *ver-number*	Configures the NTP server node and specifies the NTP version number of this server.
server *IP-addr* **disable**	Temporarily disables this NTP server.The **no** command form reenables theNTP server.
status	Shows the status of NTP servers.

Usage Guidelines

Use the **no** form of **ntp enable** and **ntp disable** to negate the NTP option. In other words, to disable NTP, you can use the **no ntp enable**; to enable NPT, use the **no ntp disable**.

To remove an NTP server with the address, 170.10.10.4:

```
ECV (config) # no ntp server 170.10.10.4
```

Examples

None

ntpdate

Use the **ntpdate** command to set the system clock once from a remote server using Network Time Protocol (NTP).

Command Mode: Privileged EXEC mode

Syntax

ntpdate *IP-addr*

Arguments

Parameter	Description
IP-addr	Specifies the IP address of the remote NTP server.

Examples

To synchronize the server to the NTP server, 216.27.190.202:

```
ECV (config) # ntpdate 216.27.190.202
```

opt-map

The Silver Peak appliance allows you to configure how your traffic is optimized by creating *optimization maps*. Optimization maps make it easy for you to explicitly filter for the traffic you want to optimize, and then apply an action to that flow.

Optimization maps are made up of ordered entries. Each entry consists of a **match** statement paired with a **set** action. Set actions are specific to the type of map.

A map entry can match traffic that satisfies either a pre-defined ACL or any of the following attributes:

- Protocol
- Source IP Address / Subnet
- Destination IP Address / Subnet
- Source Port Number
- Destination Port Number
- Application (standard or user-defined, or a user-defined application group)
- DSCP value
- VLAN

If you want to reuse the same match criteria in more than one map, you can pre-define ACLs, which are, essentially, reusable match statements.

Set actions are specific to the type of map. An optimization map has set actions related to optimization and compression features:

- Network Memory
- IP header compression
- Payload compression
- TCP acceleration
- Protocol acceleration (CIFS, SSL, SRDF)

Map entries are ordered according to their assigned *priorities*. Priorities identify, as well as order, entries within a map. Across entries, all priority values must be unique (in other words, no two *entries* in a given map can have the same priority value).

In the following example, we'll add a new entry, with a priority of *50*, to the default map, *map1*. The first statement matches all traffic associated with the application, *AOL*. The second statement enables CIFS acceleration as the action for that traffic:

```
ECV (config) # opt-map map1 50 match app aol
ECV (config) # opt-map map1 50 set cifs enable
```

If you enter a new priority statement for an existing optimization map, the CLI adds that entry to the optimization map. However, if the map already has a *match* or *set* statement with the same priority, the new entry overwrites the previous one (and the CLI does not provide a warning).

If you want to create a new optimization map, the CLI creates the map the first time you name it in a match statement.

Every optimization map automatically includes a default entry with the priority, 65535, the highest possible number. That default entry applies all the optimization and compression features to all traffic subject to the optimization map.

By default, optimization maps have additional entries that enable protocol-specific optimizations for CIFS, SSL, iSCSI, SRDF, Citrix, and their common ports.

By default, one optimization map is always active. You can change the active map at any time, simply by activating a different map.

opt-map (no)

Use the **no opt-map** command to delete an optimization map or a specific priority entry from an optimization map.

Command Mode: Global Configuration mode

Syntax

no opt-map *map-name*
no opt-map *map-name priority-value*

Arguments

Parameter	Description
map-name	Specifies which optimization map.
priority-value	Designates a priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.

Usage Guidelines

You can only delete an optimization map if it's inactive. Therefore, to delete the active optimization map, you must first activate a different optimization map. For example:

```
ECV (config) # opt-map ginger activate
ECV (config) # no opt-map ginger
```

You can also delete a specific entry in an optimization map by using the **no opt-map** command and specifying a priority value. For example, the following statement deletes the priority *100* entry (*match* and *set* statements) from the optimization map, *fred*:

```
ECV (config) # no opt-map fred 100
```

opt-map activate

Use the **opt-map activate** command to activate an inactive optimization map.

Command Mode: Global Configuration mode

Syntax

opt-map *map-name* **activate**

Arguments

Parameter	Description
map-name	Specifies which existing, inactive optimization map.

Usage Guidelines

Only one optimization map can be active at a time. The Silver Peak appliance has a default optimization map, *map1*, that's active until you create and activate a new optimization map.

Examples

To activate the new optimization map, *rambo*:

```
ECV (config) # opt-map rambo activate
```

opt-map comment

Use the **opt-map comment** command to add a comment for a specified NAT map entry.

Command Mode: Global Configuration mode

Syntax

opt-map *map-name priority-value* **comment** *comment-text*

Arguments

Parameter	Description
map-name	Specifies the name of the optimization map.
priority-value	Designates a priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.
comment-text	Specifies the text used for the comment.

Examples

None

opt-map match

Use the **opt-map match** command to create an optimization map entry that uses match criteria to delineate traffic. Also use this command to change the matching conditions associated with an existing entry.

Command Mode: Global Configuration mode

Syntax

opt-map *map-name priority-value* **match acl** *ACL-name*

opt-map *map-name priority-value* **match app** { *app-name* | *app-group* }

opt-map *map-name priority-value* **match dscp** { *dscp-value* | **any** }

opt-map *map-name priority-value* **match matchstr** *match-string*

opt-map *map-name priority-value* **match protocol** *IP-protocol-number-name* { *source-ip-addr-netmask* | **any** } { *dest-ip-addr-netmask* | **any** } [**dscp** { *dscp-value* | **any** }] [**vlan** {**any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

opt-map *map-name priority-value* **match protocol ip** { *source-ip-addr-netmask* | **any** } { *dest-ip-addr-netmask* | **any** } [**app** { *app-name* | **any** }] [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

opt-map *map-name priority-value* **match protocol** { **tcp** | **udp** } { *source-ip-addr-netmask* | **any** } { *dest-ip-addr-netmask* | **any** } [{ **source-port-number** | **any** } { *dest-port-number* | **any** }] [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

opt-map *map-name priority-value* **match vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }

Arguments

Parameter	Description
opt map *map-name*	Specifies the optimization map. If the name does not exist, the CLI creates it.
priority-value	Designates a priority value for the optimization map. Value range is *1* to *65534*. By default, *65535* is reserved for the default entry.
match acl *ACL-name*	Creates an entry that uses an existing ACL to match traffic. This command can also change the ACL associated with an existing entry.

Parameter	Description
match app *app-name*	Creates an entry that uses a built-in or user-defined application (or application group) to match traffic. Command also changes the application associated with an existing entry.
match dscp { *dscp-value* | **any** }	Creates or modifies an entry that matches traffic with a specific DSCP marking. Valid dscp-values include: af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, *or* ef. **any** is a wildcard.
match matchstr *match-string*	Creates or modifies an opt map that matches a string.
any	**any** is a wildcard.
match protocol *IP-protocol-number-name*	Creates or modifies an entry that matches traffic with a specific protocol that is **NOT** named specifically as *ip*, *tcp*, or *udp*.
match protocol ip	Creates or modifies an entry that matches specific IP addresses. When you specify **protocol ip**, you are allowing *any* IP protocol. In that case, you also need to specify an application (or application group). If you don't, the CLI defaults to specifying **any** application.If you don't choose to specify a DSCP value in the full command, then the CLI defaults to specifying **any** DSCP value in the policy entry.
match protocol { tcp | **udp** }	Creates or modifies an entry that matches specific TCP or UDP addresses. If you don't choose to specify source and destination ports in the full command, then the CLI defaults to specifying **0:0** (any source port and any destination port) in the policy entry.
match vlan { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }	Creates or modifies an entry that matches an interface and 802.1q VLAN tag. The available values include: *1..4094* the number assigned to a VLAN *intf.tag* as in **lan0.10** *any.tag* as in **any.10** *intf.any* as in **lan0.any** *intf.native* as in **lan0.native** **any** is a wildcard
source-ip-addr-netmask	Specifies the source IP address and netmask in slash notation. For example, *10.2.0.0 0.0.255.255* should be entered as *10.2.0.0/16.*
dest-ip-addr-netmask	Specifies the destination IP address and netmask in slash notation. For example, *10.2.0.0/16.*

Usage Guidelines

You can specify one of the (built-in) applications (alphabetically left to right):

For each **opt-map match** command with a given priority, you must create an **opt-map set** command(s) with the same priority. But, you cannot create the **set** command without having first created the **match** command.

Examples

To create a match criteria with a priority of “100” for the map, “express”, that filters for all traffic coming from the LAN with a DSCP marking of “best effort”:

```
ECV (config) # opt-map express 100 match dscp be
```

To create a match criteria with a priority of “70” for the map, “express”, that filters for the application group, “secure”:

```
ECV (config) # opt-map express 70 match app secure
```

To create a match criteria with a priority of “20” for “map2” that filters for all AOL traffic that’s headed from the LAN to 172.34.8.0:

```
ECV (config) # opt-map map2 20 match protocol ip any 172.34.8.0 aol
```

Since you haven’t specified a DSCP value, the criteria will include all DSCP values, as if you had written it as follows:

```
ECV (config) # opt-map map2 20 match protocol ip any 172.34.8.0 aol any
```

To create a match criteria with a priority of “30” for the map, “arthouse” that filters for all UDP traffic coming from port 41 and having a destination of 122.33.44.0/24:

```
ECV (config) # opt-map arthouse 30 match protocol udp any 122.33.4.0/24 41:0
```

Since you haven’t specified a DSCP value, the criteria will include all DSCP values, as if you had written it as follows:

```
ECV (config) # opt-map arthouse 30 match protocol udp any 122.33.4.0/24 41:0 any
```

To create a match criteria with a priority of “10” for the map, “waldo” that filters for all Interior Gateway Protocol (IGP) traffic that has a DSCP marking of “af11”:

```
ECV (config) # opt-map waldo 10 match protocol igp any any dscp af11
```

opt-map modify-priority

Use **opt-map modify-priority** command to modify the priority value of an existing entry in the optimization map.

Command Mode: Global Configuration mode

Syntax

opt-map *map-name current-priority-value* **modify-priority** *new-priority-value*

Arguments

Parameter	Description
map-name	Specifies an existing optimization map.
current-priority-value	Specifies the current priority value for the entry you want to change.
modify-priority *new-priority-value*	Designates the new priority for this entry. This new priority value must be unique and between 1 to 65534.

Usage Guidelines

If you try renumber the entry to a priority number that already exists, the CLI informs you that that's the case and that you can't make that modification.

Examples

To change the priority of entry 40 to be 60 for the map, *wiser*:

```
ECV (config) # opt-map wiser 40 modify-priority 60
```

opt-map set

The **opt-map set** command specifies or modifies an entry's set action. You cannot create a **set** command for an entry until you first issue a **match** command.

Command Mode: Global Configuration mode

Syntax

opt-map *map-name priority-value* **set header** { **enable** | **disable** }

opt-map *map-name priority-value* **set network-memory** { **disable** | **balanced** | **min-latency** | **max-reduction** }

opt-map *map-name priority-value* **set payload** { **enable** | **disable** }

opt-map *map-name priority-value* **set tcp** { **enable** | **disable** }

opt-map *map-name priority-value* **set protocol-specific** { **none** | **cifs** | **ssl** | **srdf** | **citrix** | **iscsi** } [**network-memory** { **disable** | **balanced** | **min-latency** | **max-reduction** }]

opt-map *map-name priority-value* **set protocol-specific** { **none** | **cifs** | **ssl** | **srdf** | **citrix** | **iscsi** } **network-memory** { **disable** | **balanced** | **min-latency** | **max-reduction** } **payload** { **enable** | **disable** } **header** { **enable** | **disable** } **tcp** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp adjust-mss-to-mtu** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp auto-reset-flows** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp congestion-control** { **standard** | **optimized** | **aggressive** }

opt-map *map-name priority-value* **set advanced-tcp e2e-fin-handling** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp ip-black-listing** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp keep-count** *threshold*

opt-map *map-name priority-value* **set advanced-tcp keep-idle** *seconds*

opt-map *map-name priority-value* **set advanced-tcp keep-interval** *seconds*

opt-map *map-name priority-value* **set advanced-tcp lanside-wsfclamp** *threshold*

opt-map *map-name priority-value* **set advanced-tcp max-l2w-buffer** *Kbytes*

opt-map *map-name priority-value* **set advanced-tcp max-w2l-buffer** *Kbytes*

opt-map *map-name priority-value* **set advanced-tcp persist-drop** *seconds*

opt-map *map-name priority-value* **set advanced-tcp preserve-pkt-boundary** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp propagate-syn** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp reset-to-default**

opt-map *map-name priority-value* **set advanced-tcp route-policy-override** { **enable** | **disable** }

opt-map *map-name priority-value* **set advanced-tcp slow-lan-defense** *threshold*

opt-map *map-name priority-value* **set advanced-tcp slowlan-windowpenalty** *threshold*

opt-map *map-name priority-value* **set advanced-tcp window-scale-factor** *threshold*

Arguments

Parameter	Description
opt map *map-name*	Specifies which optimization map.
priority-value	Specifies an existing priority value for the optimization map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.
set	Configures the optimization map with the arguments that follow.
header { **enable** | **disable** }	Enables or disables header compression.
network-memory { **disable** | **balanced** | **min-latency** | **max-reduction** }	Sets the type of network memory for matched traffic. The options are: **disable** Disables Network Memory. **balanced** Sets Network Memory for a balance between minimum latency and maximum reduction. **min-latency** Sets Network Memory for minimum latency. **max-reduction** Sets Network Memory for maximum reduction.
payload { **enable** | **disable** }	Enables or disables payload compression for matched traffic.
protocol-specific { **none** | **cifs** | **ssl** | **srdf** | **citrix** | **iscsi** }	For the named protocol (CIFS, SSL, SRDF, Citrix, ISCSI) enables acceleration for matched traffic.To disable acceleration for all five protocols, use **none**.
tcp { **enable** | **disable** }	Enables or disables TCP acceleration for matched traffic.
advanced-tcp	Sets advanced TCP acceleration options.
adjust-mss-to-mtu { **enable** | **disable** }	Enables or disables the adjustment of the MSS to the tunnel MTU.
auto-reset-flows { **enable** | **disable** }	Enables or disables the auto-reset of TCP flows.

Parameter	Description
congestion-control { **enable** | **disable** }	Enables or disables congestion control for WAN.
e2e-fin-handling { **enable** | **disable** }	Enables or disables end-to-end FIN handling.
ip-black-listing { **enable** | **disable** }	Enables or disables IP blacklisting.
keep-count *threshold*	Specifies the maximum number of TCP keep-alive probes.
keep-idle *seconds*	Specifies the TCP keep-alive time, in seconds, to the first probe.
keep-interval *seconds*	Specifies the time interval between TCP keep-alive probes.
lanside-wsfclamp	For the LAN-side Window Scale Factor clamp, specifies the window scale factor value (**1**... **14**). To disable, use **0**.
max-l2w-buffer *Kbytes*	Specifies the maximum LAN-to-WAN buffer size, in kilobytes.
max-w2l-buffer *Kbytes*	Specifies the maximum WAN-to-LAN buffer size, in kilobytes.
persist-drop *seconds*	Specifies the maximum TCP persist timeout.
preserve-pkt-boundary { **enable** | **disable** }	Enables or disables the preserving of packet boundaries.
propagate-syn { **enable** | **disable** }	Enables or disables the Propagate SYN feature.
reset-to-default	Resets all advanced TCP options to default values.
route-policy-override { **enable** | **disable** }	Enables or disables the route policy override feature.
slow-lan-defense *threshold*	Sets the slow LAN defense threshold value (0 .. 12, 0=Off).
slowlan-winpenalty *threshold*	For the Slow LAN Window Penalty, specifies the window scale factor value (**1**... **10**). To disable, use **0**.
window-scale-factor *threshold*	Set the window scale factor value (1 .. 14).

Defaults

By default, the optimization map entry enables protocol-specific acceleration for CIFS and SSL.

Usage Guidelines

You cannot create a **set** command for an entry until you first issue a **match** command. And, until you create a **set** command, no Set Actions exist for that entry's priority.

Examples

None

overlay

Use the **overlay** command to configure applications on the appliance.

Command Mode: Global Configuration mode

Syntax

overlay add *overlay-name overlay-id*
overlay common internal-subnets *list-subnets*
overlay delete *overlay-name*
overlay *overlay-name* **bonding-policy** { **high-availability** | **high-quality** | **high-throughput** | **raw** }
overlay *overlay-name* **brownout-thres** { **jitter** *jitter-ms* | **latency** *latency-ms* | **loss** *loss-percent* }
overlay *overlay-name* **comment** *comment-overlay*
overlay *overlay-name* **internet-traffic policy local-breakout** { **backup** *Internet-traffic-backuptunnels* | **primary** *Internet-traffic-primary-tunnels* }
overlay *overlay-name* **internet-traffic policy-list** *list-internet-traffic-policies*
overlay *overlay-name* **overlay-priority** *priority-number* **links** { **add** *link-name* | **delete** *link-name* }
overlay *overlay-name* **overlay-priority** *priority-number* **state** { **use-sla** | **use-active** }
overlay *overlay-name* **topology node-type** { **non-hub** | **hub** }

Arguments

Parameter	Description
overlay-name	Name of the overlay. For example: **voice** or **data**.
overlay-id	A numerical identifier for the overlay.
add	Adds an overlay.
bonding-policy	Configures threshold options for this overlay. The four options are: **high-availability** **high-quality** **high-throughput** **raw**
brownout-thres	Configures threshold options for this overlay.
comment *comment-overlay*	Adds your comment to the overlay.

Parameter	Description
common internal-subnets	Configures internal subnets for all overlays.
delete	Deletes the specified overlay.
internet-traffic	Configures internet traffic policy for this overlay.
jitter *jitter-ms*	Configures jitter threshold for this overlay.
latency *latency-ms*	Configures latency threshold for this overlay.
links { **add** | **delete** } *link-name*	Adds or deletes links in this bucket.
local-breakout	Configures the local breakout policy for this overlay. The two options are: **backup** *Internet-traffic-backup-tunnels* Configures the backup passthrough tunnel(s) for local-breakout policy. **primary** *Internet-traffic-primary-tunnels* Configures the primary passthrough tunnel(s) for local-breakout policy.
loss *loss-percent*	Configures loss threshold for this overlay.
overlay-priority *Priority-number*	Configures tunnels usage priority for this overly.
policy	Configures internet traffic policy
policy-list *list-internet-traffic-policies*	Configures internet traffic policy-list for this overlay.
state { **use-sla** | **use-active** }	Specifies how to detect a brownout condition on the tunnel: **use-sla** – Determines brownout when threshold is exceeded for loss, latency, or jitter. **use-active** – Determines brownout when tunnel is down.
topology node-type { **non-hub** | **hub** }	Configures topology role for appliance in this overlay.

Examples

None

pim interface dr-priority

The **pim interface dr-priority** command configures the DR Priority value the specified interface advertises. The DR Priority value is used to elect a Designated Router (DR). The host that advertises the highest DR Priority value becomes the Designated Router.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network. The Designated Router sends periodic Join/Prune messages toward a group-specific Rendezvous Point (RP) for each group which has active members. These messages inform other PIM routers about clients that want to become receivers (Join) or stop being receivers (Prune) for the group.

Command Mode: Global Configuration mode

Syntax

pim interface *intf-name* **dr-priority** *priority-value*

Parameters

intf-name: The interface that will advertise the DR Priority value.

priority-value: The DR Priority value assigned to the interface. Value range is 1 to 18000. Default value is 1.

Examples

This command assigns a DR Priority value of 20 to the WAN0 interface.

```
ECV (config) # pim interface wan0 dr-priority 20
ECV (config) # show pim interfaces
IfName   Interface-IP Address   DR-Priority  Generation ID   Designated-Router-IP
    Hello Interval   Join/Prune Interval
wan0     10.19.156.10           20           3534349093      10.19.156.10          30
                  30
pim0     169.254.124.1          1            2520518556      169.254.124.2         30
                  30
pim1     169.254.125.1          1            423562176       169.254.125.2         30
                  30
pim2     169.254.126.1          1            296423632       169.254.126.2         30
                  30
ECV (config) #
```

pim interface enable

The **pim interface enable** command enables Protocol Independent Multicast - Sparse Mode (PIM-SM) on a specified interface. PIM-enabled interfaces can join multicast groups and subsequently receive data packets addressed to the group.

The **no pim interface enable** command disables PIM-SM on the specified interface. PIM-SM is disabled on interfaces by default.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network. Sparse Mode (PIM-SM) is a PIM variant where multicast packets are delivered to receivers that explicitly request traffic rather than being flooded to all routers. PIM-SM is suitable for networks where a small percentage of nodes are interested in receiving multicast traffic.

Command Mode: Global Configuration mode

Syntax

pim interface *intf-name* **enable**
no pim interface *intf-name* **enable**
no pim interface *intf-name*

Parameters

intf-name: The interface where PIM is enabled.

Examples

This command enables PIM on WAN0 interface.

```
ECV (config) # pim interface wan0 enable
ECV (config) # show pim interfaces
IfName   Interface-IP Address   DR-Priority  Generation ID   Designated-Router-IP
    Hello Interval   Join/Prune Interval
wan0     10.19.156.10           1            3534349093      10.19.156.10          30
                 30
pim0     169.254.124.1          1            2520518556      169.254.124.2         30
                 30
pim1     169.254.125.1          1            423562176       169.254.125.2         30
                 30
pim2     169.254.126.1          1            296423632       169.254.126.2         30
                 30
ECV (config) #
```

pim interface hello-interval

The **pim hello-interval** command configures the transmission interval between PIM Hello messages originating from the specified interface. The default hello-interval value of 30 seconds is set for an interface when PIM is enabled on the interface.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Global Configuration mode

Syntax

pim interface *intf-name* **hello-interval** *int-hello*

Parameters

intf-name: The interface assigned the Hello Interval value.

int-hello: The Hello message transmission interval (seconds). Value range is 1 to 18000. Default value is 30.

Examples

This command assigns a Hello interval value of 200 to the WAN0 interface.

```
ECV (config) # pim interface wan0 hello-interval 200
ECV (config) # show pim interfaces
IfName   Interface-IP Address   DR-Priority  Generation ID   Designated-Router-IP
    Hello Interval   Join/Prune Interval
wan0     10.19.156.10           1            3534349093      10.19.156.10          200
                30
pim0     169.254.124.1          1            2520518556      169.254.124.2         30
                30
pim1     169.254.125.1          1            423562176       169.254.125.2         30
                30
pim2     169.254.126.1          1            296423632       169.254.126.2         30
                30
ECV (config) #
```

pim interface join-prune-interval

The **pim join-prune-interval** command configures the period between Join/Prune messages that the specified interface originates and sends to the upstream RPF (Reverse Path Forwarding) neighbor. The default Join/Prune Interval value of 30 seconds is set for an interface when PIM is enabled on the interface.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Global Configuration mode

Syntax

pim interface *intf-name* **join-prune-interval** *int-prune*

Parameters

intf-name: Interface assigned the Join/Prune Interval value.

int-prune: Join/Prune transmission interval (seconds). Value range is 1 to 18000. Default value is 30.

Examples

This command assigns an join/prune-interval value of 100 to the wan0 interface.

```
ECV (config) # pim interface wan0 join-prune-interval-interval 100
ECV (config) # show pim interfaces
IfName   Interface-IP Address   DR-Priority  Generation ID   Designated-Router-IP
    Hello Interval   Join/Prune Interval
wan0     10.19.156.10           1            3534349093      10.19.156.10          30
                  100
pim0     169.254.124.1          1            2520518556      169.254.124.2         30
                  30
pim1     169.254.125.1          1            423562176       169.254.125.2         30
                  30
pim2     169.254.126.1          1            296423632       169.254.126.2         30
                  30
ECV (config) #
```

pim rp ip

The **pim rp ip** command configures the IP address of the RP that the appliance accesses to receive multicast traffic. A Rendezvous Point (RP) is the common contact point for multicast data sources and receivers.

The **no pim rp** command configures the RP IP address of 0.0.0.0 for the appliance. This prevents the appliance from accessing any Rendezvous Point.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network. Sources send multicast traffic to the RP, which is forwarded to receivers down a shared distribution tree. When the first hop router of the receiver learns about the source, it sends a Join message directly to the source, creating a source-based distribution tree from the source to the receiver. This source tree does not include the RP unless the RP is located within the shortest path between the source and receiver.

Command Mode: Global Configuration mode

Syntax

pim rp ip *ip-addr*
no pim ip *ip-addr*

Parameters

ip-addr: The IP address configured as the appliance RP (dotted decimal notation). Default value is 0.0.0.0.

Examples

This command configures 192.172.11.11 as the appliance's RP IP address.

```
ECV (config) # pim rp ip 192.172.11.11
ECV (config) # show pim rp
Group :224.0.0.0, RP addr :192.172.11.11, RPF Interface :wan0

ECV (config) #   None
```

ping

Use the **ping** command to send Internet Control Message Protocol (ICMP) echo requests to a specified host.

Command Mode: EXEC mode

Syntax

ping *ping-options destination*

Arguments

Parameter	Description
ping-options	Specifies the type of ping. Select one of the following options: **-a** Audible ping. **-A** *Adaptive ping* Interpacket interval adapts to round-trip time, so that effectively not more than one (or more, if preload is set) unanswered probes present in the network. Minimal interval is 200 msec if not super-user. On networks with low rtt this mode is essentially equivalent to flood mode. **-b** Allow pinging a broadcast address. **-B** Do not allow ping to change source address of probes. The address is bound to the one selected when ping starts. **-c** *count* Stop after sending count ECHO_REQUEST packets. With deadline option, ping waits for count ECHO_REPLY packets, until the time-out expires. **-d** Set the SO_DEBUG option on the socket being used. This socket option is unused. **-F** *flow label* Allocate and set 20 bit flow label on echo request packets. If value is zero, kernel allocates random flow label. **-f** *Flood ping* For every ECHO_REQUEST sent a period "." is printed, while for ever ECHO_REPLY received a backspace is printed. This provides a rapid display of how many packets are being dropped. If interval is not given, it sets interval to zero and outputs packets as fast as they come back or one hundred times per second, whichever is more. Only the super-user may use this option with zero interval. **-i** *interval* Wait interval seconds between sending each packet. The default is to wait for one second between each packet normally, or not to wait in flood mode. Only super-user may set interval to values less 0.2 seconds. **-I***interface address* Set source address to specified interface address. Argument may be numeric IP address or name of device.

Parameter	Description
	-l *preload* If preload is specified, ping sends that many packets not waiting for reply. Only the super-user may select preload more than 3. **-L** Suppress loopback of multicast packets. This flag only applies if the ping destination is a multicast address. **-n** Numeric output only. No attempt will be made to lookup symbolic names for host addresses. **-p** *pattern* You may specify up to 16 “pad” bytes to fill out the packet you send. This is useful for diagnosing data-dependent problems in a network. For example, -p ff will cause the sent packet to be filled with all ones. **-Q** *tos* Set Quality of Service -related bits in ICMP datagrams. tos can be either decimal or hex number. Traditionally (RFC1349), these have been interpreted as: 0 for reserved (currently being redefined as congestion control), 1-4 for Type of Service and 5-7 for Precedence. Possible settings for Type of Service are: minimal cost: 0x02, reliability: 0x04, throughput: 0x08, low delay: 0x10. Multiple TOS bits should not be set simultaneously. Possible settings for special Precedence range from priority (0x20) to net control (0xe0). You must be root (CAP_NET_ADMIN capability) to use Critical or higher precedence value. You cannot set bit 0x01 (reserved) unless ECN has been enabled in the kernel. In RFC2474, these fields has been redefined as 8-bit Differentiated Services (DS), consisting of: bits 0-1 of separate data (ECN will be used, here), and bits 2-7 of Differentiated Services Codepoint (DSCP). **-q** *Quiet output* Nothing is displayed except the summary lines at startup time and when finished. **-R** *Record route* Includes the RECORD_ROUTE option in the ECHO_REQUEST packet and displays the route buffer on returned packets. Note that the IP header is only large enough for nine such routes. Many hosts ignore or discard this option. **-r** Bypass the normal routing tables and send directly to a host on an attached interface. If the host is not on a directly attached network, an error is returned. This option can be used to ping a local host through an interface that has no route through it provided the option -I is also used. **-s** *packetsize* Specifies the number of data bytes to be sent. The default is 56, which translates into 64 ICMP data bytes when combined with the 8 bytes of ICMP header data. **-S** *sndbuf* Set socket sndbuf. If not specified, it is selected to buffer not more than one packet.

Parameter	Description
	-t ttl Set the IP Time to Live. **-T** *timestamp option* Set special IP timestamp options. timestamp option may be either tsonly (only timestamps), tsandaddr (timestamps and addresses) or tsprespec host1 [host2 [host3 [host4]]] (timestamp prespecified hops). **-M** *hint* Select Path MTU Discovery strategy. hint may be either do (prohibit fragmentation, even local one), want (do PMTU discovery, fragment locally when packet size is large), or dont (do not set DF flag). **-U** Print full user-to-user latency (the old behavior). Normally ping prints network round trip time, which can be different f.e. due to DNS failures. **-v** Verbose output. **-V** Show version and exit. **-w** *deadline* Specify a timeout, in seconds, before ping exits regardless of how many packets have been sent or received. In this case ping does not stop after count packet are sent, it waits either for deadline expire or until count probes are answered or for some error notification from network.Specifies the IP address of the destination that you are pinging.

Examples

None

proxy-arp

The **proxy-arp** command enables Proxy ARP on the specified interface. By default, Proxy ARP is disabled on all interfaces

Proxy ARP is a method where ARP requests for an IP Address that is not on a given network is answered by a proxy server on that network. The proxy provides its MAC Address as the destination, then directs traffic directed to the proxy address to its intended destination.

The **no proxy-arp** command disables Proxy ARP on the specified interface.

Command Mode: EXEC mode

Syntax

proxy-arp *intf-name*
no proxy-arp *intf-name*

Arguments

Parameter	Description
intf-name	The interface upon which Proxy ARP is enabled. May be an interface name or interface label.

Defaults

Proxy ARP is disabled

Examples

This command enables Proxy ARP on WAN2 interface.

```
ECV (config) # proxy-arp wan2
ECV (config) # show proxy-arp wan2
interface name           proxy-arp enabled
--------------           -----------------
ECV (config) #
```

qos-map

The Silver Peak appliance allows you to configure the Quality of Service (QoS) for your traffic by creating *QoS maps*. QoS maps make it easy for you to explicitly match the traffic that you want to queue, and then (1) send that traffic to a particular queue, and (2) specify the DSCP markings for WAN and LAN packets.

You can create elaborate combinations of match criteria, using IP addresses, ports, protocol, and/or DSCP markings. You can also create more complex matches within ACLs. Or, you can choose to simplify your match criteria by using well-known or user-defined applications, or application groups. By default, one QoS map is always active, and you can change the active map at any time, simply by activating a different map.

Each QoS map may have multiple entries. A map entry consists of one or more **match** statements, which specifies packet fields to be matched, and one **set** statement, which specifies the traffic class, or queue, for the traffic. You can also specify DSCP markings for the LAN (inner) and WAN (outer, or tunnel) packets.

For example, in the following example, the first statement matches all traffic that is associated with the application, *AOL*. The second statement specifies a traffic class ID of 9 for that traffic:

```
ECV (conf) # qos-map fred 50 match app aol
ECV (conf) # qos-map fred 50 set traffic-class 9
```

You create a new QoS map with a single, default entry which serves as a catch-all. In this example, if the QoS map, *fred*, did not exist, the CLI would create it when you entered the match statement.

Entries in a map are ordered according to their assigned *priorities*. Priorities are used to identify, as well as to order entries within a map. All priority values must be unique (in other words, no two entries in a given map can have the same priority value). In the above example, the priority for the entries is *50*.

If you enter a new priority statement for an existing QoS map, the CLI adds that entry to the QoS map. However, if you enter a statement that has the same priority as one that already exists, the new entry overwrites the previous one (and the CLI does not provide a warning).

A QoS map entry can match traffic that satisfies either a pre-defined ACL or any of the following attributes:

- IP Protocol
- Source IP Address
- Destination IP Address
- Source Port Number
- Destination Port Number
- Application
- DSCP value
- VLAN

To edit the ten available traffic classes, use the **shaper** command.

qos-map (no)

Use the **no qos-map** command to delete a QoS map or a specific priority entry from a QoS map.

Command Mode: Global Configuration mode

Syntax

no qos-map *map-name*
no qos-map *map-name priority-value*

Arguments

Parameter	Description
map-name	Specifies which QoS map.
priority-value	Designates a priority value in the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry, which cannot be removed.

Usage Guidelines

You can only delete a QoS map if it's inactive. To delete the active QoS map, you must first activate a different QoS map. For example:

```
ECV (config) # qos-map ginger activate
ECV (config) # no qos-map ginger
```

You can also delete a specific entry in a QoS map by using the **no qos-map** command and specifying a priority value. For example, the following statement deletes the priority *100* entry (*match* and *set* statements) from the QoS map, *fred*:

```
ECV (config) # no qos-map fred 100
```

qos-map activate

Use the **qos-map activate** command to activate an inactive QoS map.

Command Mode: Global Configuration mode

Syntax

qos-map *map-name* **activate**

Arguments

Parameter	Description
map-name	Specifies which existing, inactive QoS map.

Usage Guidelines

Only one QoS map can be active at time. The Silver Peak appliance has a default QoS map, *map1*, that is active until you create and activate a new QoS map.

Examples

To activate the new QoS map, *houdini*:

```
ECV (config) # qos-map houdini activate
```

qos-map comment

Use the **qos-map comment** command to add a comment for a specified QoS map entry.

Command Mode: Global Configuration mode

Syntax

qos-map *map-name priority-value* **comment** *comment-text*

Arguments

Parameter	Description
map-name	Specifies the name of the QoS map.
priority-value	Designates a priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.
comment-text	Specifies the text used for the comment.

Examples

None

qos-map match

Use the **qos-map match** command to create a QoS map entry that uses match criteria to delineate traffic. Also use this command to change the matching conditions associated with an existing entry.

Command Mode: Global Configuration mode

Syntax

qos-map *map name priority-value* **match acl** *ACL-name*

qos-map *map name priority-value* **match app** { *app-name* | *app-group* }

qos-map *map name priority-value* **match dscp** { *dscp-value* | **any** }

qos-map *map name priority-value* **match matchstr** *match-string*

qos-map *map name priority-value* **match protocol** *IP-protocol-number-name* { *source-ip-addr-mask* | **any** } { *dest-ip-addr-mask* | **any** } [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

qos-map *map name priority-value* **match protocol ip** { *source-ip-addr-mask* | **any** } { *dest-ip-addr-mask* | **any** } [**app** { *app-name* | **any** }] [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

qos-map *map name priority-value* **match protocol** { **tcp** | **udp** } { *source-ip-addr-mask* | **any** } { *dest-ip-addr-mask* | **any** } [{ *source-port-number* | **any** } { *dest-port-number* | **any** }] [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

qos-map *map name priority-value* **match vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }

Arguments

Parameter	Description
qos map *map name*	Specifies which QoS map. If the name doesn't exist, the CLI creates it.
priority-value	Designates a priority value for the map entry. Acceptable values are from *1* to *65534*. By default, the appliance reserves *65535* for the default entry.
match acl *ACL-name*	Creates an entry that uses an existing ACL to match traffic. Also use this command to change the ACL associated with an existing entry.
match app *app-name*	Creates an entry that uses a built-in or user-defined application—or an application group—to match traffic. Also use this command to change the application associated with an existing entry.

Parameter	Description
match dscp { *dscp-value* | **any** }	Creates or modifies an entry that matches traffic with a specific DSCP marking. You can use any of the following values:af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, *or* ef.__any__ is a wildcard.
match matchstr *match-string*	Creates or modifies a QoS map that matches a string.
any	**any** is a wildcard.
match protocol *IP-protocol-number-name*	Creates or modifies an entry that matches traffic with a specific protocol that is **NOT** named specifically as *ip*, *tcp*, or *udp*.
match protocol ip	Creates or modifies an entry that matches specific IP addresses.When you specify **protocol ip**, the assumption is that you are allowing *any* IP protocol. In that case, you also need to specify an application (or application group). If you don't, the CLI defaults to specifying **any** application.If you don't choose to specify a DSCP value in the full command, then the CLI defaults to specifying **any** DSCP value in the policy entry.
match protocol { **tcp** | **udp** }	Creates or modifies an entry that matches specific TCP or UDP addresses.If you don't choose to specify source and destination ports in the full command, then the CLI defaults to specifying **0:0** (any source port and any destination port) in the policy entry.If you don't choose to specify a DSCP value in the full command, then the CLI defaults to specifying **any** DSCP value in the policy entry.
match vlan { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }	Creates or modifies an entry that matches an interface and 802.1q VLAN tag. The available values include: *1..4094* the number assigned to a VLAN* intf.tag* as in **lan0.10** *any.tag* as in **any.10** *intf.any* as in **lan0.any** *intf.native* as in **lan0.native** **any** is a wildcard
source-ip-addr-mask	Specifies the source IP address and netmask in slash notation. For example, *10.2.0.0 0.0.255.255* should be entered as *10.2.0.0/16.*
dest-ip-addr-mask	Specifies the destination IP address and netmask in slash notation. For example, *10.2.0.0/16.*

Usage Guidelines

For each **qos-map match** command with a given priority, you must create a **qos-map set** command with the same priority. But, you cannot create a **set** command without having first created the **match** command.

Examples

To create a match criteria with a priority of “100” for the map, “express”, that filters for all traffic coming from the LAN with a DSCP marking of “best effort”:

```
ECV (config) # qos-map express 100 match dscp be
```

To create a match criteria with a priority of “70” for the map, “express”, that filters for the application group, “secure”:

```
ECV (config) # qos-map express 70 match app secure
```

To create a match criteria with a priority of “20” for “map2” that filters for all AOL traffic that’s headed from the LAN to 172.34.8.0:

```
ECV (config) # qos-map map2 20 match protocol ip any 172.34.8.0 aol
```

Since you haven’t specified a DSCP value, the criteria will include all DSCP values, as if you had written it as follows:

```
ECV (config) # qos-map map2 20 match protocol ip any 172.34.8.0 aol any
```

To create a match criteria with a priority of “30” for the map, “arthouse” that filters for all UDP traffic coming from port 41 and having a destination of 122.33.44.0/24:

```
ECV (config) # qos-map arthouse 30 match protocol udp any 122.33.4.0/24 41:0
```

Since you haven’t specified a DSCP value, the criteria will include all DSCP values, as if you had written it as follows:

```
ECV (config) # qos-map arthouse 30 match protocol udp any 122.33.4.0/24 41:0 any
```

To create a match criteria with a priority of “10” for the map, “waldo” that filters for all Interior Gateway Protocol (IGP) traffic that has a DSCP marking of “af11”:

```
ECV (config) # qos-map waldo 10 match protocol igp any any dscp af11
```

qos-map modify-priority

Use **qos-map modify-priority** command to modify the priority value of an existing entry.

Command Mode: Global Configuration mode

Syntax

qos-map *map-name current-priority-value* **modify-priority** *new-priority-value*

Arguments

Parameter	Description
map-name	Specifies an existing QoS map.
current-priority-value	Specifies the current priority value for the entry you want to change.
new-priority-value	Designates the new priority for this entry. This new priority value must be unique and between 1 to 65534.

Usage Guidelines

If you try renumber the entry to a priority number that already exists, the CLI informs you that that's the case and that you can't make that modification.

Examples

To change the priority of entry 40 to be 60 for the map, *DesMoines*:

```
ECV (config) # opt-map DesMoines 40 modify-priority 60
```

qos-map set

The **qos-map set** command specifies or modifies the set statement in a QoS map entry. You cannot use a **set** command until you first issue a **match** command.

Command Mode: Global Configuration mode

Syntax

qos-map *map-name priority-value* **set traffic-class** *traffic-class-ID*

qos-map *map-name priority-value* **set traffic-class** *traffic-class-ID* **lan-qos** { **trust-lan** | *dscp-value* } **wan-qos** { **trust-lan** | *dscp-value* }

qos-map *map-name priority-value* **set lan-qos** { **trust-lan** | *dscp-value* }

qos-map *map-name priority-value* **set wan-qos** { **trust-lan** | *dscp-value* }

Arguments

Parameter	Description
qos-map *map-name*	Specifies which QoS map.
priority-value	Specifies an existing priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.
traffic-class *traffic-class-ID*	Specifies the traffic class, or queue, to which matched traffic is sent. Traffic classes are identified by integer values from *1* through *10*.
lan-qos { **trust-lan**	*dscp-value* }	With **lan-qos**, **trust-lan** indicates that the DSCP marking should not change. In other words, the DSCP setting in the inner, encapsulated packet that comes in is the same one that goes out.You can assign any of the following DSCP values:af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, *or* ef.
wan-qos { **trust-lan**	*dscp-value* }	With **wan-qos**, **trust-lan** indicates that the marking of the outer packet follows the marking of the inner packet.You can assign any of the following DSCP values:af11, af12, af13, af21, af22, af23, af31, af32, af33, af41, af42, af43, be, cs1, cs2, cs3, cs4, cs5, cs6, cs7, *or* ef.

Defaults

By default, the **set** part of the default optimization map entry (priority 65535) is:

```
qos-map set traffic-class 1 lan-qos trust-lan wan-qos trust-lan
```

Usage Guidelines

You cannot create a **set** command for an entry until you first issue a **match** command. And, until you create a **set** command, no Set Actions exist for that entry's priority.

- When creating an entry (priority) with the Appliance Manager Graphical User Interface, the QoS map defaults are:
  - Traffic class = 1
  - LAN QoS = trust-lan
  - WAN QoS = trust-lan
- When you create the first **qos-map set** command **for a priority** with the CLI and you use a syntax that doesn't specify all three Set Actions, the CLI automatically creates the rest as defaults in the background.

For example, if your first set command for priority "10" in "map1" is:

```
ECV (config) # qos-map map1 10 set lan-qos be
```

then, the CLI also creates the following two additional entries behind the scenes:

```
qos-map map1 10 set traffic-class 1
qos-map map1 10 set wan-qos trust-lan
```

You can verify these results by using the command, **show qos-map**.

For pass-through traffic, any **lan-qos** specification is ignored. Any **wan-qos** specification is placed in the ToS field of the packet.

Examples

None

radius-server

Use the **radius-server** command to configure RADIUS server settings for user authentication.

Command Mode: Global configuration mode

Syntax

radius-server host *IP-addr* [**auth-port** *port*] [**key** *string*] [**retransmit** *0...3*] [**timeout** *1...15*]
no radius-server host *IP-addr* [**auth-port** *port*]

radius-server { **key** *string* | **retransmit** *0...3* | **timeout** *1...15* }
no radius-server { **key** | **retransmit** | **timeout** }

Arguments

Parameter	Description
host *IP-addr*	Configures host, at specified IP address, to send RADIUS authentication requests. Use the **no** form of this command to stop sending RADIUS authentication requests to host.
auth-port *port*	Specifies the authentication port to use with this RADIUS server. Use the **no** form of this command to stop sending RADIUS authentication requests to the authentication port.
key *string*	Specifies the shared secret key to use with this RADIUS server. Use the **no** form of this command to remove the global RADIUS server key.
retransmit *0...3*	Specifies the maximum number of retries that can be made in the attempt to connect to this RADIUS server. The range is 0 to 3.Use the **no** form of this command to reset the global RADIUS server retransmit count to its default.
timeout *1...15*	Specifies the number of seconds to wait before the connection times out with this RADIUS server, because of keyboard inactivity. The range is 1 to 15 seconds. Use the **no** form of this command to reset the global RADIUS server timeout setting to its default.

Examples

To define the RADIUS shared secret as “mysecret”:

```
ECV (config) # radius-server key mysecret
```

To specify the RADIUS server's IP address as 208.20.20.4 with authentication port 500 and a timeout of 10 seconds:

```
ECV (config) # radius-server host 208.20.20.4 auth-port 500 timeout 10
```

To set the number of times the global RADIUS server retransmits to its default value:

```
ECV (config) # no radius-server retransmit
```

reboot

Use the **reboot** command to reboot or shutdown the system.

Command Mode: EXEC mode (reboot - without parameters)
Command Mode: Privileged EXEC mode (all other reboot commands)

Syntax

reboot { **clean** | **force** | **halt** | **halt noconfirm** | **noconfirm** }

Arguments

Parameter	Description
reboot	Reboots the system.
clean	Reboots the system and cleans out the Network Memory.
force	Forces an immediate reboot of the system, even if it's busy.
halt	Shuts down the system.
halt noconfirm	Shuts down the system without asking about unsaved changes.
noconfirm	Reboots the system without asking about unsaved changes.

Examples

None

reload

Use the **reload** command to reboot or shutdown the system.

Command Mode: Privileged EXEC mode

Syntax

reload { **clean** | **force** | **halt** | **halt noconfirm** | **noconfirm** }

Arguments

Parameter	Description
reload	Reboots the system.
clean	Reboots the system and cleans out the Network Memory.
force	Forces an immediate reboot of the system, even if it's busy.
halt	Shuts down the system.
halt noconfirm	Shuts down the system without asking about unsaved changes.
noconfirm	Reboots the system without asking about unsaved changes.

Examples

None

route-map

The Silver Peak appliance allows you to manage your packet flow by creating **route maps**. Route maps make it easy for you to identify exactly the traffic that you need to manage. You can create elaborate combinations of match criteria, using IP addresses, ports, protocol, and/or DSCP markings. You can also create more complex matches within ACLs. Or, you can choose to simplify your match criteria by using well-known or user-defined applications, or application groups. By default, one route map is always active, and you can change the active map at any time, simply by activating a different map.

Each route map may have multiple entries. A map entry consists of one or more *match* statements, which specifies packet fields to be matched, and one *set* statement, which takes action on the matched traffic, such as sending it to a tunnel or dropping it.

For example, in the following example, the first statement matches all traffic that is associated with the application, *AOL*. The second statement sends that AOL traffic through the tunnel named *Holland*:

```
ECV (conf) # route-map fred 50 match app aol
ECV (conf) # route-map fred 50 set tunnel Holland
```

You create a new route map with a single, default entry which serves as a catch-all. In this example, if the route map, *fred*, did not exist, the CLI would create it when you entered the match statement.

Entries in a map are ordered according to their assigned *priorities*. Priorities are used to identify, as well as to order entries within a map. All priority values must be unique (in other words, no two entries in a given map can have the same priority value). In the above example, the priority for the entries is *50*.

If you enter a new priority statement for an existing route map, the CLI adds that entry to the route map. However, if you enter a statement that has the same priority as one that already exists, the new entry overwrites the previous one (and the CLI does not provide a warning).

A route map entry can match traffic that satisfies either a pre-defined ACL or any of the following attributes:

- IP protocol
- Source IP address and subnet mask
- Destination IP address and subnet mask
- Source port number
- Destination port number
- Application
- DSCP value
- VLAN

route-map (no)

You can use the **no route-map** command to delete a route map or a specific priority entry from a route map.

Command Mode: Global Configuration mode

Syntax

no route-map *map-name*
no route-map *map-name priority-value*

Arguments

Parameter	Description
map-name	Specifies which existing route map.
priority-value	Designates a priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.

Usage Guidelines

You can only delete a route map if it's inactive. To delete the active route map, you must first activate a different route map. For example:

```
ECV (config) # route-map ginger activate
ECV (config) # no route-map ginger
```

You can also delete a specific entry in a route map by using the **no route-map** command and specifying a priority value. For example, the following statement deletes the priority *100* entry (*match* and *set* statements) from the route map, *fred*:

```
ECV (config) # no route-map fred 100
```

route-map activate

Use the **route-map activate** command to activate a route map.

Command Mode: Global Configuration mode

Syntax

route-map *map-name* **activate**

Arguments

Parameter	Description
map-name	Specifies which route map.

Usage Guidelines

Only one route map can be active at time. The Silver Peak appliance has a default route map, *map1*, that is active until you create and activate a new route map.

Examples

To activate the new route map, *whichway*:

```
ECV (config) # qos-map whichway activate
```

route-map comment

Use the **route-map comment** command to add a comment for a specified QoS map entry.

Command Mode: Global Configuration mode

Syntax

route-map *map-name priority-value* **comment** *comment-text*

Arguments

Parameter	Description
map-name	Specifies the name of the route map.
priority-value	Designates a priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.
comment-text	Specifies the text used for the comment.

Examples

None

route-map match

Use the **route-map match** command to create a route map entry that uses match criteria to delineate traffic. Also use this command to change the matching conditions associated with an existing entry.

Command Mode: Global Configuration mode

Syntax

route-map *map-name priority-value* **match acl** *ACL-name*

route-map *map-name priority-value* **match app** { *app-name* | *app-group* }

route-map *map-name priority-value* **match dscp** { *dscp-value* | **any** }

route-map *map-name priority-value* **match matchstr** *match-string*

route-map *map-name priority-value* **match protocol** *IP-protocol-number-name* { *source-ip-addr-mask* | **any** } { *dest-ip-addr-mask* | **any** } [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

route-map *map-name priority-value* **match protocol ip** { *source-ip-addr-mask* | **any** } { *dest-ip-addr-mask* | **any** } [**app** { *app-name* | **any** }] [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

route-map *map-name priority-value* **match protocol** { **tcp** | **udp** } { *source-ip-addr-mask* | **any** } { *dest-ip-addr-mask* | **any** } [{ *source-port-number* | **any** } { *dest-port-number* | **any** }] [**dscp** { *dscp-value* | **any** }] [**vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }]

route-map *map-name priority-value* **match vlan** { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }

Arguments

Parameter	Description
route map *map-name*	Specifies which route map. If the name doesn't exist, the CLI creates it.
priority-value	Designates a priority value for the map entry. Acceptable values are from *1* to *65534*. By default, the appliance reserves *65535* for the default entry.
match acl *ACL-name*	Creates an entry that uses an existing ACL to match traffic. Also use this command to change the ACL associated with an existing entry.
match app *app-name*	Creates an entry that uses a built-in or user-defined application—or an application group—to match traffic. Also use this command to change the application associated with an existing entry.

Parameter	Description
match dscp { *dscp-value* | **any** }	Creates or modifies an entry that matches traffic with a specific DSCP marking. You can use any of the following values: **af11**, **af12**, **af13**, **af21**, **af22**, **af23**, **af31**, **af32**, **af33**, **af41**, **af42**, **af43**, **be**, **cs1**, **cs2**, **cs3**, **cs4**, **cs5**, **cs6**, **cs7**, or **ef**. **any** is a wildcard.
match matchstr *match-string*	Creates or modifies a route map that matches a string.
any	**any** is a wildcard.
match protocol *IP-protocol-number-name*	Creates or modifies an entry that matches traffic with a specific protocol that is **NOT** named specifically as *ip*, *tcp*, or *udp*.
match protocol ip	Creates or modifies an entry that matches specific IP addresses. When you specify **protocol ip**, you allow *any* IP protocol. In that case, you need to specify an application (or application group). Otherwise, the CLI defaults to specifying **any** application. If you do not specify a DSCP value in the full command, then the CLI defaults to specifying **any** DSCP value in the policy entry.
match protocol { **tcp** | **udp** }	Creates or modifies an entry that matches specific TCP or UDP addresses. If you don't choose to specify source and destination ports in the full command, then the CLI defaults to specifying **0:0** (any source port and any destination port) in the policy entry. If you don't choose to specify a DSCP value in the full command, then the CLI defaults to specifying **any** DSCP value in the policy entry.
match vlan { **any** | *1..4094* | *intf.tag* | *any.tag* | *intf.any* | *intf.native* }	Creates or modifies an entry that matches an interface and 802.1q VLAN tag. The available values include: *1..4094* the number assigned to a VLAN *intf.tag* as in **lan0.10** *any.tag* as in **any.10** *intf.any* as in **lan0.any** *intf>.native* as in **lan0.native** **any** is a wildcard
source-ip-addr-mask	Specifies the source IP address and netmask in slash notation. For example, *10.2.0.0 0.0.255.255* should be entered as *10.2.0.0/16.*
dest-ip-addr-mask	Specifies the destination IP address and netmask in slash notation. For example, *10.2.0.0/16.*

Usage Guidelines

For each **route-map match** command with a given priority, a **route-map set** command with the same priority is required. However, you cannot create a **set** command before creating the **match** command.

Examples

To create a match criteria with a priority of “100” for the map, “vinnie”, that filters for all traffic coming from the LAN with a DSCP marking of “best effort”:

```
ECV (config) # route-map vinnie 100 match dscp be
```

To create a match criteria with a priority of “70” for the map, “vinnie”, that filters for the application group, “secure”:

```
ECV (config) # route-map vinnie 70 match app secure
```

To create a match criteria with a priority of “20” for “map2” that filters for all AOL traffic that’s headed from the LAN to 172.34.8.0:

```
ECV (config) # route-map map2 20 match protocol ip any 172.34.8.0 aol
```

Since you haven’t specified a DSCP value, the criteria will include all DSCP values, as if you had written it as follows:

```
ECV (config) # route-map map2 20 match protocol ip any 172.34.8.0 aol any
```

To create a match criteria with a priority of “30” for the map, “arthouse” that filters for all UDP traffic coming from port 41 and having a destination of 122.33.44.0/24:

```
ECV (config) # route-map arthouse 30 match protocol udp any 122.33.4.0/24 41:0
```

Since you haven’t specified a DSCP value, the criteria will include all DSCP values, as if you had written it as follows:

```
ECV (config) # route-map arthouse 30 match protocol udp any 122.33.4.0/24 41:0 any
```

To create a match criteria with a priority of “10” for the map, “autobahn” that filters for all Interior Gateway Protocol (IGP) traffic that has a DSCP marking of “af11”:

```
ECV (config) # route-map autobahn 10 match protocol igp any any dscp af112
```

route-map modify-priority

Use **route-map modify-priority** command to modify the priority value of an existing entry.

Command Mode: Global Configuration mode

Syntax

route-map *map-name current-priority-value* **modify-priority** *new-priority-value*

Arguments

Parameter	Description
map-name	Specifies the name of an existing route map.
current-priority-value	Specifies the current priority value for the entry you want to change.
new-priority-value	Designates the new priority for this entry. This new priority value must be unique and between 1 to 65534.

Usage Guidelines

If you try renumber the entry to a priority number that already exists, the CLI informs you that that's the case and that you can't make that modification.

Examples

To change the priority of entry 40 to be 60 for the map, *lunar*:

```
ECV (config) # route-map lunar 40 modify-priority 60
```

route-map set

The **route-map set** command specifies or modifies the SET part of an entry in a given route map. You cannot use a **set** command until you first issue a **match** command.

Command Mode: Global Configuration mode

Syntax

route-map *map-name priority-value* **set auto-opt-balance** [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** }]

route-map *map-name priority-value* **set auto-opt-low-latency** [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** }]

route-map *map-name priority-value* **set auto-opt-low-loss** [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** }]

route-map *map-name priority-value* **set auto-opt-overlay-id** *overlay-name* [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** }]

route-map *map-name priority-value* **set auto-opt-preferred-if** { *intf-name* | **wan0** }

route-map *map-name priority-value* **set auto-optimize** [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** }]

route-map *map-name priority-value* **set drop**

route-map *map-name priority-value* **set pass-through** { **shaped** | **unshaped** }

route-map *map-name priority-value* **set peer-balance** *peer-hostname* [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** | **continue** }]

route-map *map-name priority-value* **set peer-low-latency** *peer-hostname* [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** | **continue** }]

route-map *map-name priority-value* **set peer-low-loss** *peer-hostname* [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** | **continue** }]

route-map *map-name priority-value* **set tunnel** *tunnel-name* [**if-down** { **pass-through** | **pass-through-unshaped** | **drop** | **continue** }]

Arguments

Parameter	Description
route-map *map-name*	Specifies which route map.
priority-value	Specifies an existing priority value for the map entry. Acceptable values are from 1 to 65534. By default, the appliance reserves 65535 for the default entry.

Parameter	Description
set auto-opt-balance	Auto-routes (optimizes) the traffic, load balancing.
set auto-opt-low-latency	Auto-routes (optimizes) the traffic, select tunnel with lowest latency.
set auto-opt-low-loss	Auto-routes (optimizes) the traffic, select tunnel with lowest loss.
set auto-opt-overlay-id *overlay-name*	Auto-routes (optimizes) the traffic, select the named overlay.
set auto-opt-preferred-if	Auto-routes (optimizes) the traffic, select desired interface for auto-opt.
set auto-optimize	Auto-routes (optimizes) the traffic.
set tunnel *tunnel-name*	Specifies the name of an existing tunnel. Use the **route-map set tunnel** command when you send matched traffic to a tunnel or a pair of redundant tunnels.
if-down { **pass-through** | **pass-through-unshaped** | **drop** | **continue** }	Establishes what action the Silver Peak appliance takes if the primary tunnel (and its backup tunnel, if there is one) is down. You can specify the following options with **if-down**: **pass-through** Traffic is passed through with QoS shaping. **pass-through-unshaped** - Traffic is passed through with no QoS shaping. **drop** - The packets are dropped. **continue** - Continue processing next entry. The default option, if you don't specify one, is **pass-through** (shaped).
set pass-through { **shaped** | **unshaped** }	Use the **route-map set passthrough** command if you want matching traffic to pass through the Silver Peak appliance unaccelerated. To limit the bandwidth of the traffic according to the passthrough bandwidth settings of the shaper, choose **shaped**; otherwise, choose **unshaped**.
set peer-balance *peer-hostname*	Specifies that the appliance load balance with its named peer. To view a list of peers, enter a space and question mark at the end of this argument.
set peer-low-latency *peer-hostname*	When the appliance has a peer, use the one with the lowest latency.
set peer-low-loss *peer-hostname*	When the appliance has a peer, use the one with the lowest loss.
set drop	Use when you want to drop matched traffic.

Defaults

The default action for **if-down** is to send the traffic through as pass-through and shaped.

Usage Guidelines

- You cannot use a **set** command until you first issue a **match** command.
- By default, the set part of the default route map entry (with priority *65535*) is **auto-optimize**, which means that the appliances determine the appropriate, available tunnel for the traffic. You can modify this to drop or pass-through unshaped as follows:

  route-map *map-name* **65535 set drop**
  route-map *map-name* **65535 set pass-through-unshaped**

Examples

None

saas

Use **saas** command to configure the system SaaS (Software as a Service) options.

Command Mode: Global Configuration mode

Syntax

saas { **enable** | **disable** }
saas ping-src-interface *source-intf-SaaS-RTT-pings*
saas rtt-interval *seconds*
saas rtt-num-req-per-host *number*

Arguments

Parameter	Description
disable	Disables SaaS.
enable	Enables SaaS.
ping-src-interface *source-intf-SaaS-RTT-pings*	Configures a physical source interface for SaaS pings. For example, **wan0**.
rtt-interval *seconds*	Specifies the RTT (Round Trip Time) daemon interval in seconds.
rtt-num-req-per-host *number*	Specifies the number of requests to send to each host to calculate the average RTT.

Examples

None

selftest

Use the **selftest** command to run a self test and diagnostics.

Command Mode: Privileged EXEC mode

Syntax

selftest start disk
selftest stop disk

Arguments

Parameter	Description
start disk	Starts a disk self test operation.
stop disk	Stops a disk self test operation.

Usage Guidelines

When you enter

```
selftest start disk
```

the following message appears:

```
This is an intrusive self test. This test puts the system in bypass mode
and perform read/write operations on the disks. The system will not process
any network traffic for the duration of the test. At the end of the test, you
need to reboot the system. While the test is running, if you attempt to run
other commands, you will receive errors.

Do you want to proceed? (y/n) (If you don't proceed, the question times out.)

Disk self test has been canceled.
```

Examples

None

shaper inbound

Use **shaper inbound** command to shape individual WAN, LAN, or management interfaces, or to shape the aggregate WAN interface.

Use the **no** command to remove an inbound shaper.

Command Mode: Global Configuration mode

Syntax

shaper inbound *shaper-name* { **enable** | **disable** }
shaper inbound *shaper-name* **accuracy** *usec*
shaper inbound *shaper-name* **max-bandwidth** *kbps*
shaper inbound *shaper-name* **traffic-class** *1-10* **excess-weight** *weight*
shaper inbound *shaper-name* **traffic-class** *1-10* **flow-rate-limit** *kbps*
shaper inbound *shaper-name* **traffic-class** *1-10* **max-bandwidth** *percent-interface-bw*
shaper inbound *shaper-name* **traffic-class** *1-10* **max-wait** *ms*
shaper inbound *shaper-name* **traffic-class** *1-10* **min-bandwidth** *percent-interface-bw*
shaper inbound *shaper-name* **traffic-class** *1-10* **priority** *1-10*
no shaper inbound { *shaper-name* | **default** | **wan** }

Arguments

Parameter	Description
disable	Disables inbound shaper.
enable	Enables inbound shaper.
shaper-name	Refers to the shaper for a specific interface, such as **wan0**, **wan1**, **twan0**, **twan1**, **bwan0**, **lan0**, **lan1**, **tlan0**, **tlan1**, **blan0**, **mgmt0**, **mgmt1**. Use **wan** for shaping the aggregate WAN interface.
accuracy *usec*	Specifies shaper accuracy in microseconds.
excess-weight *weight*	Specifies the shaper traffic class excess weight. If there is remaining bandwidth after satisfying the minimum bandwidth, then the excess is distributed among the traffic classes in proportion to the weightings specified. Values range from 1 to 10,000.
flow-rate-limit *kbps*	Specifies the traffic class's flow rate limit.
max-bandwidth *percent-interface-bw*	Specifies the traffic class's maximum bandwidth in kilobits per second. You can limit the maximum bandwidth that a traffic class will use by specifying a percentage. The bandwidth usage for the traffic class never exceeds this value.

Parameter	Description
max-wait *ms*	Specifies the maximum wait time in milliseconds. Any packets waiting longer than the specified Max Wait Time are dropped.
min-bandwidth *percent-interface-bw*	Specifies the shaper's minimum bandwidth in kilobits per second. Each traffic class is guaranteed this percentage of bandwidth, allocated in the order of priority. However, if the sum of the percentages is greater than 100%, then lower-priority traffic classes might not receive their guaranteed bandwidth if it is all consumed by higher-priority traffic.
priority *1-10*	Specifies the shaper traffic class priority. This determines the order in which each class's minimum bandwidth is allocated - 1 is first, 10 is last.
traffic-class *1-10*	Specifies the shaper traffic class.

Usage Guidelines

The inbound Shaper provides a simplified way to globally configure QoS (Quality of Service) on the appliances.

- It shapes inbound traffic by allocating bandwidth across ten traffic classes.
- The system applies these QoS settings globally before decompressing all the inbound tunnelized and pass-through-shaped traffic — shaping it as it arrives from the WAN.

Examples

None

shaper outbound

Use **shaper outbound** command to shape individual WAN, LAN, or management interfaces, or to shape the aggregate WAN interface.

Use the **no** command to remove an outbound shaper.

Command Mode: Global Configuration mode

Syntax

shaper outbound *shaper-name* { **enable** | **disable** }
shaper outbound *shaper-name* **accuracy** *usec*
shaper outbound *shaper-name* **max-bandwidth** *kbps*
shaper outbound *shaper-name* **traffic-class** *1-10* **excess-weight** *weight*
shaper outbound *shaper-name* **traffic-class** *1-10* **flow-rate-limit** *kbps*
shaper outbound *shaper-name* **traffic-class** *1-10* **max-bandwidth** *percent-interface-bw*
shaper outbound *shaper-name* **traffic-class** *1-10* **max-wait** *ms*
shaper outbound *shaper-name* **traffic-class** *1-10* **min-bandwidth** *percent-interface-bw*
shaper outbound *shaper-name* **traffic-class** *1-10* **priority** *1-10*
no shaper outbound { *shaper-name* | **default** | **wan** }

Arguments

Parameter	Description
disable	Disables outbound shaper.
enable	Enables outbound shaper.
shaper-name	Refers to the shaper for a specific interface, such as **wan0**, **wan1**, **twan0**, **twan1**, **bwan0**, **lan0**, **lan1**, **tlan0**, **tlan1**, **blan0**, **mgmt0**, **mgmt1**. Use **wan** for shaping the aggregate WAN interface. Availability of the non-WAN interfaces (as arguments) is to facilitate preparations for migrating from one appliance model to another, or one deployment mode to another.
accuracy *usec*	Specifies shaper accuracy in microseconds.
excess-weight *weight*	Specifies the shaper traffic class excess weight. If there is remaining bandwidth after satisfying the minimum bandwidth, then the excess is distributed among the traffic classes in proportion to the weightings specified . Values range from 1 to 10,000.
flow-rate-limit *kbps*	Specifies the traffic class's flow rate limit.
max-bandwidth *percent-interface-bw*	Specifies traffic class maximum bandwidth (kilobits per second). You can limit the maximum bandwidth that a traffic class will use by specifying a percentage. The bandwidth usage for the traffic class never exceeds this value.

Parameter	Description
max-wait *ms*	Specifies the maximum wait time in milliseconds. Any packets waiting longer than the specified Max Wait Time are dropped.
min-bandwidth *percent-interface-bw*	Specifies shaper's minimum bandwidth (kilobits per second). Each traffic class is guaranteed this percentage of bandwidth, allocated in the order of priority. However, if the sum of the percentages is greater than 100%, then lower-priority traffic classes might not receive their guaranteed bandwidth if it is all consumed by higher-priority traffic.
priority *1-10*	Specifies the shaper traffic class priority. This determines the order in which each class's minimum bandwidth is allocated - 1 is first, 10 is last.
traffic-class *1-10*	Specifies the shaper traffic class.

Usage Guidelines

The Shaper provides a simplified way to globally configure QoS (Quality of Service) on the appliances.

- It shapes outbound traffic by allocating bandwidth as a percentage of the system bandwidth.
- The system applies these QoS settings globally after compressing (deduplicating) all the outbound tunnelized and pass-through-shaped traffic — shaping it as it exits to the WAN.
- Availability of the non-WAN interfaces (as arguments) is to facilitate preparations for migrating from one appliance model to another, or one deployment mode to another.

Examples

None

slogin

Use the **slogin** command to securely log into another system using Secure Shell (SSH).

Command Mode: EXEC mode

Syntax

slogin *slogin-options* [*user-text*] *hostname-text* [*command*]

Arguments

Parameter	Description
slogin-options	Specify one of the following SSH login options: **-a** Disables forwarding of the authentication agent connection. **-A** Enables forwarding of the authentication agent connection. This can also be specified on a per-host basis in a configuration file. Agent forwarding should be enabled with caution. Users with the ability to bypass file permissions on the remote host (for the agent's Unix-domain socket) can access the local agent through the forwarded connection. An attacker cannot obtain key material from the agent, however they can perform operations on the keys that enable them to authenticate using the identities loaded into the agent. -**b** *bind_address* Specify the interface to transmit from on machines with multiple interfaces or aliased addresses. **-c** *cipher_spec* Additionally, for protocol version 2 a comma-separated list of ciphers can be specified in order of preference. **-e ch | ^ch | none** Sets the escape character for sessions with a pty (default: ~). The escape character is only recognized at the beginning of a line. The escape character followed by a dot (.) closes the connection; followed by control-Z suspends the connection; followed by itself sends the escape character once. Setting the character to None fully transparent. **-f** Requests ssh to go to background just before command execution. Useful if ssh is asking for passwords or passphrases but the user wants it in the background. This implies -n. The recommended way to start X11 programs at a remote site is with something like ssh -f host xterm. **-g** Allows remote hosts to connect to local forwarded ports.

Parameter	Description
	-i *identity_file* Selects a file from which the private key for RSA or DSA authentication is read. Default is $HOME/.ssh/identity (protocol version 1) and $HOME/.ssh/id_rsa and $HOME/.ssh/id_dsa (protocol version 2). Identity files may also be specified on a per-host basis in the configuration file.Mmultiple -i options are permitted, along with multiple identities specified in configuration files. **-k** Disables forwarding of Kerberos tickets and AFS tokens. This may also be specified on a per-host basis in the configuration file. **-l** *login_name* Specifies the user to log in as on the remote machine. This also may be specified on a per-host basis in the configuration file. **-m** *mac_spec* For protocol version 2, a comma-separated list of MAC (message authentication code) algorithms can be specified in order of preference. **-n** Redirects stdin from **/dev/null** (actually, prevents reading from stdin). This must be used when ssh is run in the background. A common trick is to use this to run X11 programs on a remote machine. For example, ssh -n shadows.cs.hut.fi emacs and will start an emacs on shadows.cs.hut.fi, and the X11 connection will be automatically forwarded over an encrypted channel. The ssh program will be put in the background. (This does not work if ssh needs to ask for a password or passphrase; see also the **-f** option.) **-N** Do not execute a remote command. This is useful for just forwarding ports (protocol version 2 only). **-o** *option* Can be used to give options in the format used in the configuration file. This is useful for specifying options for which there is no separate command-line flag. **-p** *port* Port to connect to on the remote host. This can be specified on a per-host basis in the configuration file. **-q** *Quiet mode* All warning and diagnostic messages are suppressed. **-s** May be used to request invocation of a subsystem on the remote system. Subsystems are a feature of the SSH2 protocol which facilitate the use of SSH as a secure transport for other applications (for example, sftp). The subsystem is specified as the remote command. **-t** Force pseudo-tty allocation. This can be used to execute arbitrary screen-based programs on a remote machine, which can be very useful, for example, when implementing menu services. Multiple -t options force tty allocation, even if ssh has no local tty. **-T** Disable pseudo-tty allocation. **-v** *Verbose mode* Causes ssh to print debugging messages about its progress. Helpful in debugging connection, authentication, and configuration problems. Multiple **-v** options increase verbosity. Maximum is 3. **-V** Display the version number and exit. **-x** Disables X11 forwarding.

Parameter	Description
	-X Enables X11 forwarding. This can also be specified on a per-host basis in a configuration file. X11 forwarding should be enabled with caution. Users with the ability to bypass file permissions on the remote host (for the user's X authorization database) can access the local X11 display through the forwarded connection. An attacker may then be able to perform activities such as keystroke monitoring. **-Y** Enables trusted X11 forwarding. Trusted X11 forwardings are not subjected to the X11 SECURITY extension controls. **-C** Requests compression of all data (including stdin, stdout, stderr, and data for forwarded X11 and TCP/IP connections). The compression algorithm is the same used by gzip(1), and the *level* CompressionLevel option for protocol version 1. Compression is desirable on modem lines and other slow connections, but will only slow down things on fast networks. The default value can be set on a host-by-host basis in the configuration files. **-F** *configfile* Specifies an alternative per-user configuration file. If a configuration file is given on the command line, the system-wide configuration file (/etc/ssh/ssh_config) will be ignored. The default for the per-user configuration file is $HOME/.ssh/config. **-L** *port:host:hostport* Specifies that the given port on the local (client) host is to be forwarded to the given host and port on the remote side. This works by allocating a socket to listen to port on the local side, and whenever a connection is made to this port, the connection is forwarded over the secure channel, and a connection is made to host port hostport from the remote machine. Port forwardings can also be specified in the configuration file. Only root can forward privileged ports. IPv6 addresses can be specified with an alternative syntax: port/host/hostport **-R** *port:host:hostport* Specifies that the given port on the remote (server) host is to be forwarded to the given host and port on the local side. This works by allocating a socket to listen to port on the remote side, and whenever a connection is made to this port, the connection is forwarded over the secure channel, and a connection is made to host port hostport from the local machine. Port forwardings can also be specified in the configuration file. Privileged ports can be forwarded only when logging in as root on the remote machine. IPv6 addresses can be specified with an alternative syntax: port/host/hostport **-D** *port* Specifies a local dynamic This works by allocating a socket to listen to port on the local side, and whenever a connection is made to this port, the connection is forwarded over the secure channel, and the application protocol is then used to determine where to connect to from the remote machine. Currently the SOCKS4 protocol is supported, and ssh will act as a SOCKS4 server. Only root can forward privileged ports. Dynamic port forwardings can also be specified in the configuration file.

Parameter	Description
	-1 Forces ssh to try protocol version 1 only. **-2** Forces ssh to try protocol version 2 only. **-4** Forces ssh to use IPv4 addresses only. **-6** Forces ssh to use IPv6 addresses only.
user-text	Specifies the name of a user on the remote host.
hostname-text	Specifies the name or path of the remote host.
command	Specifies a command to execute on the remote system.

Examples

None

snmp-server

Use the **snmp-server** command to configure SNMP server options.

Command Mode: Global Configuration mode

Syntax

snmp-server community *community-name* [**ro**]
no snmp-server community

snmp-server contact *name-contact*
no snmp-server contact

snmp-server enable
no snmp-server enable

snmp-server enable traps
no snmp-server enable traps

snmp-server encrypt { **md5** | **sha** } { **plaintext** *pwd-plain* | **prompt** }

snmp-server host *IP-addr* [**disable**]
no snmp-server host *IP-addr* [**disable**]

snmp-server host *IP-addr* **traps version** { **1** | **2c** } *community-name*
snmp-server host *IP-addr* **traps version 3** *v3-username*

snmp-server listen enable
no snmp-server listen enable

snmp-server listen interface *intf*
no snmp-server listen interface *intf*

snmp-server location *system-location*
no snmp-server location

snmp-server traps event raise-alarm
no snmp-server traps event raise-alarm

Arguments

Parameter	Description
community *community-name* [**ro**]	Configures the name for the SNMP read-only community, which is required to make SNMP queries. Use the **no** form of this command to reset the community string to its default.
contact *name-contact*	Sets a value for the *syscontact* variable in MIB-II. Use the **no** form of this command to clear the contents of the syscontact variable.

Parameter	Description
enable	Enables the SNMP server. Use the **no** form of this command to disable the SNMP server.
enable traps	Enables the sending of SNMP traps from this system. Use the **no** form of this command to disable sending of SNMP traps from this system.
encrypt { **md5** | **sha** }	Generate the encrypted form of the password from plain text, using one of the following hash types: **md5** Message-Digest algorithm 5 (a hash function with a 128-bit hash value) **sha** Secure Hash Algorithm, SHA-1
host *IP-addr*	Configures the hosts to which to send SNMP traps. Use the **no** form of this command to stop sending SNMP traps to a specified host.
host *IP-addr* **disable**	Temporarily disables sending of traps to this host. Use the **no** form of this command to reenable sending of SNMP traps to a specified host.
host *IP-addr* **traps version 3** *v3-username*	Sends SNMP traps to the specified host. The community string noted here is the V3 username; it's used for particular trap destination hosts.
host *IP-addr* **traps version** { **1** | **2c** } *community-string*	Specifies the SNMP version of traps to send to this host: **1** is SNMPv1. **2c** is SNMPv2c. The community string noted here is also a community name (string name); it's used for particular trap destination hosts.
listen enable	Enables SNMP interface restriction access to this system. Use the **no** form of this command to disable SNMP interface restriction access to this system.
listen interface *intf*	Specifies the interface you want to add to the SNMP server access restriction list. The supported interfaces are **mgmt0** and **mgmt1**. Use the **no** form of this command to remove an interface to the SNMP server access restriction list.
location *system-location*	Specifies the value for the syslocation variable in MIB-II. Use the **no** form of this command to clear the contents of the syslocation variable.
plaintext *pwd-plain*	Specifies the plaintext password to be encrypted.
prompt	Asks to specify the password securely with the following prompt, at which the user will enter text.
traps event raise-alarm	Generates a trap for each alarm that is raised and cleared. Use the **no** form of this command to negate this setting.

Usage Guidelines

You need an SNMP manager application such as HP OpenViewTM to browse the MIB II data and receive traps. There are many shareware and freeware SNMP manager applications available from the internet.

Examples

None

snmp-server user v3

Use the **snmp-server user v3** command to configure SNMP access on a per-user basis for v3 security parameters.

Command Mode: Global Configuration mode

Syntax

snmp-server user { *v3-username* | **admin** }
snmp-server user { *v3-username* | **admin** } **v3** [**enable**]
no snmp-server user { *v3-username* | **admin** } **v3** [**enable**]
snmp-server user { *v3-username* | **admin** } **v3 auth** { **md5** | **sha** } *pwd*
snmp-server user { *v3-username* | **admin** } **v3 auth** { **md5** | **sha** } *pwd* **priv** { **des** | **aes-128** } [*pwd*]
snmp-server user { *v3-username* | **admin** } **v3 encrypted auth** { **md5** | **sha** } *pwd*
snmp-server user { *v3-username* | **admin** } **v3 encrypted auth** { **md5** | **sha** } *pwd* **priv** { __de__s | **aes-128** } [*pwd*]
snmp-server user { *v3-username* | **admin** } **v3 prompt auth** { **md5** | **sha** } *pwd*
snmp-server user { *v3-username* | **admin** } **v3 prompt auth** { **md5** | **sha** } *pwd* **priv** { **des** | **aes-128** } [*pwd*]

Arguments

Parameter	Description
auth	Configures SNMP v3 security parameters, specifying passwords in plain text on the command line. Passwords are always stored encrypted.
auth { **md5** | **sha** } *pwd*	Configures the use of either the MD5 or SHA-1 hash algorithm, and sets a plaintext password to use for authentication. If followed by a carriage return, it uses the default privacy algorithm, with the same privacy password as that specified here for authentication. The default privacy program is AES-128.
enable	Enables SNMP v3 access for this user. Use the **no** form of this command to disable this user's SNMP v3 access.
encrypted	Configures SNMP v3 security parameters, specifying passwords in encrypted form.
priv { **des** | **aes-128** } [*pwd*]	Configures the use of either DES or AES-128 encryption for privacy. If you don't specify a password, it uses the same privacy password as that specified for authentication. If you do specify a password, it is in plaintext.
prompt	Configures SNMP v3 security parameters, specifying passwords securely in follow-up prompt rather than on the command line.
v3	Configures SNMP v3 users.

Defaults

The default privacy (encryption) program is **AES-128**.

Usage Guidelines

- Only **admin** is allowed as an SNMP v3 user.
- Passwords must be at least eight (8) characters in length.

Examples

To configure the passwords for **admin's** SNMP v3 security parameters as a follow-up after entering the command:

```
ECV (config) # snmp-server user admin v3 prompt auth md5 priv des
Auth password: ________
      Confirm: ________
Privacy password: __________
         Confirm: __________
ECV (config) #
```

ssh client global host-key-check

The **ssh client global host-key-check** command specifies the Strict Hostkey Checking level. This level the method the appliance uses to verify a host key when a user attempts to connect. Three checking levels are defined:

- YES: A client connects only if a matching host key is already in the known hosts file.
- NO: A client always connects. New or changed host keys are accepted without checking. The default policy level is *NO*.
- ASK: Clients are prompted for a key. The appliance accepts new host keys and rejects connections when a known key does not match an entered key.

The **no ssh client global host-key-check** command selects the *NO* (default) policy level.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh client global host-key-check *POLICY*
no ssh client global host-key-check

Parameters

POLICY: Specifies the Strict Hostkey Checking level. Valid options include:

- **yes**: The YES policy level.
- **no**: The NO policy level.
- **ask**: The ASK policy level.

Examples

This command sets the Strict Hostkey Checking level to YES.

```
ECV-A (config) # ssh client global host-key-check yes
ECV-A (config) # show ssh client
SSH client Strict Hostkey Checking: yes

No SSH global known hosts configured.

No SSH user identities configured.
```

```
SSH authorized keys:
  User admin:
  No authorized keys for user admin.

  User joe:
    Key 2: dfghi
ECV-A (config) #
```

ssh client global known-host

The **ssh client global known-host** command stores a specified host key to the known hosts file.

The **no ssh client global known-host** command removes the host key for the specified host in the known hosts file.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices. The known hosts file stores public keys of the hosts accessed by a user.

Command Mode: Global configuration mode

Syntax

ssh client global known-host *host-key*
no ssh client global known-host *host*

Parameters

host-key: The host key stored to the file.

host: The host for which the key is removed from the file.

Examples

None

ssh client global known-hosts-file

The **ssh client global known-hosts-files** command creates and configures a known hosts file.

The **no ssh client global known-hosts-files** command deletes the known hosts file.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices. The known hosts file stores public keys of the hosts accessed by a user.

Command Mode: Global configuration mode

Syntax

ssh client global known-hosts-files *filename*
no ssh client global known-hosts-files

Parameters

filename: The name of the known hosts file created and configured by the command.

Examples

This command creates the known hosts file named *example1.txt*.

```
ECV-A (config) # ssh client global known-hosts-file example1.txt
ECV-A (config) #
```

ssh client user authorized-key

The **ssh client user authorized-key** command assigns an authorized public key to a specified user. An authorized key is a public key that SSH uses to grant login access through public key authentication. Multiple authorized keys can be assigned to each user. A key ID is associated with each key. Authorized keys are configured separately for each user.

The **no ssh client user authorized-key** command removes the authorized key from a specified user. By default, authorized keys are not assigned to users. When removing an authorized key, the command must refer to a key by its key ID.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh client user *USER* **authorized-key sshv2** *key-code*
no ssh client user *USER* **authorized-key sshv2** *key-id*

Parameters

USER: The name of an existing user of the appliance. Options include:

- *username-text*: The user ID of the user account to which the key is assigned.
- **admin**: Command assigns the key to the **admin** account.
- **monitor**: Command assigns the key to the **monitor** account.

key-code: The authorized key code assigned to the specified user.

key-id: The key identifier associated with an authorized key.

Examples

These commands assign two authorized keys to the *joe* user account. As shown by the show command, the key IDs assigned to the keys are **1** and **2**.

```
ECV-A (config) # ssh client user joe authorized-key sshv2 abcde
ECV-A (config) # ssh client user joe authorized-key sshv2 dfghi
ECV-A (config) # show ssh client
SSH client Strict Hostkey Checking: no

No SSH global known hosts configured.

No SSH user identities configured.
```

```
SSH authorized keys:
  User joe:
    Key 1: abcde
    Key 2: dfghi
ECV-A (config) #
```

This command removes key 1 (abcde) as an authorized key for the *joe* user account.

```
ECV-A (config) # no ssh client user joe authorized-key sshv2 1
ECV-A (config) # show ssh client
SSH client Strict Hostkey Checking: no

No SSH global known hosts configured.

No SSH user identities configured.

SSH authorized keys:
  User joe:
    Key 2: dfghi
ECV-A (config) #
```

ssh client user identity

The **ssh client user identity generate** command generates SSH client identity private and public keys for a specified user.

The **ssh client user identity private-key** command configures the private SSH client key for a specified user.

The **ssh client user identity public-key** command configures the public SSH client key for a specified user.

The **no ssh client user identiy** command deletes the public and private SSH client keys for a specified user. By default, public and private keys are not defined for any user.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh client user *USER* **identity** *KEY-TYPE* **generate**
ssh client user *USER* **identity** *KEY-TYPE* **private-key** *private-key-code*
ssh client user *USER* **identity** *KEY-TYPE* **public-key** *public-key-code*

no ssh client user *USER* **identity**
no ssh client user *USER* **identity** *KEY-TYPE*

Parameters

USER: Specifies the name of an existing user of the appliance. Options include:

- *username-text*: The user ID of the account for which keys are generated or configured.
- **admin**: Command generates or configures keys for the **admin** account.
- **monitor**: Command generates or configures keys for the **monitor** account.

KEY-TYPE: The algorithm used by the command for public key encryption. Options include:

- **rsa2**: The RSAv2 algorithm.

private-key-code: The private key code the command assigns to the specified SSH.
public-key-code: The public key code the command assigns to the specified SSH.

Examples

This command generates public and private SSH client keys for the **admin** account.

```
ECV-A (config) # ssh client user admin identity rsa2 generate
ECV-A (config) #
```

ssh client user known-host remove

The **ssh client user known-host remove** command removes the host from the user's known host file.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. The SSH known hosts file contains fingerprints generated from SSH keys of remote appliances the user previously logged into. A user that logs into an appliance for the first time through SSH is asked to save the host's fingerprint. This command removes the fingerprint of the specified host from the user's known hosts file.

Command Mode: Global configuration mode

Syntax

ssh client user *USER* **known-host** *known-host-text* **remove**

Parameters

USER: Specifies the name of an existing user of the appliance. Options include:

- *username-text*: The user ID of the user account for which the host key is removed.
- **admin**: Command removes the host key for the **admin** account.
- **monitor**: Command removes the host key for the **monitor** account.

known-host-text: Specifies the host whose key is removed from the user's known host file.

Examples

This command removes the key associated with the host at 10.3.2.3 from the known host file for the *joe* user account.

```
ECV-A (config) # ssh client user joe known-host 10.3.2.3 remove
ECV-A (config) #
```

ssh server enable

The **ssh server enable** command enables Secure Shell (SSH) access to the appliance. SSH access is enabled by default.

The **no ssh server enable** command disables SSH access to the appliance. All current SSH sessions are terminated when this command is executed. To enable SSH access, open the CLI through Orchestrator.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh server enable
no ssh server enable

Examples

This command enables SSH access to the appliance.

```
ECV-A (config) # ssh server enable
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22

Host Key Finger Prints:
    RSA host key: SHA256:UF4Jb84ZTt7kgn+InFrpgtRpvKzS90yyPeDxBl9Tjns
    ECDSA host key: SHA256:eXMvanESR+jKYZ2pws/usYyzwLCZuygvAy3p/nB1Fhg

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
```

ssh server encryption-algos

The **ssh server encryption-algos** command specifies the encryption algorithms the appliance uses to authenticate clients attempting to open an SSH session. All current SSH sessions are terminated when this command is executed.

The **no ssh server encryption-algos** command configures the appliance to authenticate clients using default encryption algorithms. Default algorithms are aes256-ctr, aes192-ctr, and aes128-ctr.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh server encryption-algos *ALGO-NAME-1* [*ALGO-NAME-2* ... *ALGO-NAME-N*]
no ssh server encryption-algos

Parameters

ALGO-NAME-X: Encryption algorithm name. Command can specify multiple algorithms. Options include:

- **aes128-cbc**
- **aes192-cbc**
- **aes256-cbc**
- **aes128-ctr**
- **aes192-ctr**
- **aes256-ctr**
- **aes128-gcm@openssh.com**
- **aes256-gcm@openssh.com**

Examples

This command configures the appliance to authenticate clients with *aes256-gcm@openssh.com*.

```
ECV-A (config) # ssh server encryption-algos aes256-gcm@openssh.com
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22
```

```
Host Key Finger Prints:
    RSA host key: SHA256:UF4Jb84ZTt7kgn+InFrpgtRpvKzS90yyPeDxBl9Tjns
    ECDSA host key: SHA256:eXMvanESR+jKYZ2pws/usYyzwLCZuygvAy3p/nB1Fhg

SSH server Ciphers: aes256-gcm@openssh.com
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
```

ssh server host-key

The **ssh server host-key generate** command generates SSH server host keys for the appliance using a specified algorithm.

The **ssh server host-key private-key** command specifies the private server host key for a specified algorithm.

The **ssh server host-key public-key** command specifies the public server host key for a specified algorithm.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh server host-key *KEY-TYPE* **generate**
ssh server host-key *KEY-TYPE* **private-key** *private-key-code*
ssh server host-key *KEY-TYPE* **public-key** *public-key-code*

Parameters

KEY-TYPE: Specifies the host key algorithm type. Options include:

- **rsa2**: RSAv2 algorithm
- **ecdsa**: Digital Signature Algorithm, version 2 (DSAv2).

private-key-code: The private key for the specified SSH.
public-key-code: Sets the public key for the specified SSH.

Examples

These commands display ssh server parameters before and after commands that generate SSH server host keys.

```
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22
```

```
Host Key Finger Prints:
    RSA host key: SHA256:UhCSH3cAVsTDQQhNKzQWEXy282c99e4tOrt9ljcD3EY
    ECDSA host key: SHA256:xenMIyUS/loiyOh6+Tqv/j4C946IdS/OQ9OOrWSUXP4

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
ECV-A (config) # ssh server host-key generate
ECV-A (config) #
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22


Host Key Finger Prints:
    RSA host key: SHA256:vstMGg1rWdmXS7Tp/BwfMMU2MwNz5Ky5gWOmXTAo+e8
    ECDSA host key: SHA256:shU/daAAZ5BZkbswO0zAnUQptyeS8XEEG0z7I4tqa9E

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
```

ssh server key-exchange-algos

The **ssh server key-exchange-algos** command specifies the key exchange algorithm for the appliance. An SSH key exchange algorithm is a method for securely exchanging a shared session key between two parties. This key is used for encryption and authentication.

The **no ssh server key-exchange-algos** command resets the key exchange algorithm to the default algorithm (**diffie-hellman-group14-sha1**).

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks.

Command Mode: Global configuration mode

Syntax

ssh server key-exchange-algos *ALGO-NAME-1* [*ALGO-NAME-2 … ALGO-NAME-N*]
no ssh server exchange-algos

Parameters

ALGO-NAME-X: The key exchange algorithm. A command may specify multiple algorithms. Options include:

- **diffie-hellman-group1-sha1**
- **diffie-hellman-group14-sha1**
- **diffie-hellman-group14-sha256**
- **diffie-hellman-group16-sha512**
- **diffie-hellman-group18-sha512**
- **diffie-hellman-group-exchange-sha1**
- **diffie-hellman-group-exchange-sha256**
- **ecdh-sha2-nistp256**
- **ecdh-sha2-nistp384**
- **ecdh-sha2-nistp521**
- **curve25519-sha256**
- **curve25519-sha256@libssh.org**

Examples

This command configures diffie-hellman-group1-sha1 as the key exchange algorithm.

```
ECV-A (config) # ssh server key-exchange-algos diffie-hellman-group1-sha1
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22
```

```
Host Key Finger Prints:
    RSA host key: SHA256:vstMGg1rWdmXS7Tp/BwfMMU2MwNz5Ky5gWOmXTAo+e8
    ECDSA host key: SHA256:shU/daAAZ5BZkbswO0zAnUQptyeS8XEEG0z7I4tqa9E

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group1-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
```

ssh server mac-algos

The **ssh server mac-algos** command specifies the Message Authentication Code (MAC) algorithm for the appliance.

The **no ssh server mac-algos** command resets the MAC algorithm to the default algorithms (**hmac-sha2-256** and **hmac-sha1**).

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh server mac-algos *ALGO-NAME-1* [*ALGO-NAME-2 … ALOG-NAME-N*]
no ssh server mac-algos

Parameters

ALGO-NAME-X: Specifies the MAC algorithm. A command may specify multiple algorithms. Options include:

- **hmac-sha1**
- **hmac-sha1-96**
- **hmac-sha2-256**
- **hmac-sha2-512**
- **hmac-sha2-256-etm@openssh.com**
- **hmac-sha2-512-etm@openssh.com**

Examples

This command configures the appliance to use *hmac-sha2-512* as the MAC algorithm.

```
ECV-A (config) # ssh server mac-algos hmac-sha2-512
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22


Host Key Finger Prints:
    RSA host key: SHA256:vstMGg1rWdmXS7Tp/BwfMMU2MwNz5Ky5gWOmXTAo+e8
    ECDSA host key: SHA256:shU/daAAZ5BZkbswO0zAnUQptyeS8XEEG0z7I4tqa9E

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-512
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
```

ssh server permit-scp-sftp

The **ssh server permit-scp-sftp** command enables SSH access to the appliance through SCP and SFTP protocols.

The **no ssh server permit-scp-sftp** command disables SCP and SFTP as methods for accessing SSH access to the applianced. By default, SCP and SFTP access is disabled.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh server permit-scp-sftp
no ssh server permit-scp-sftp

Examples

This command enables SSH access to the appliance through SCP and SFTP.

```
ECV-A (config) # ssh server permit-scp-sftp
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  22


Host Key Finger Prints:
    RSA host key: SHA256:vstMGg1rWdmXS7Tp/BwfMMU2MwNz5Ky5gWOmXTAo+e8
    ECDSA host key: SHA256:shU/daAAZ5BZkbswO0zAnUQptyeS8XEEG0z7I4tqa9E

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: yes
ECV-A (config) #
```

ssh server ports

The **ssh server ports** command specifies the ports through which the appliance can be accessed by SSH. Port 22 is the default SSH port.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Global configuration mode

Syntax

ssh server ports *port-num-1* [*port-num-2 . . . port-num-N*]

Parameters

port-num-X: Port number of the SSH access ports. A command can specify multiple port numbers.

Examples

This command configures the appliance to be accessed through SSH from ports 44 and 55.

```
ECV-A (config) # ssh server ports 44 55
ECV-A (config) # show ssh server
SSH server enabled: yes
SSH server ports:  44 55


Host Key Finger Prints:
    RSA host key: SHA256:vstMGg1rWdmXS7Tp/BwfMMU2MwNz5Ky5gWOmXTAo+e8
    ECDSA host key: SHA256:shU/daAAZ5BZkbswO0zAnUQptyeS8XEEG0z7I4tqa9E


SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A (config) #
```

ssl auth-certificate

Use the **ssl auth-certificate** command to configure SSL certificate authority parameters.

Command Mode: Privileged EXEC mode

Syntax

ssl auth-certificate delete all
ssl auth-certificate delete subject-name *cert-subject-name*
ssl auth-certificate install cert-file *cert-file-or-URL*
ssl auth-certificate install pfx-file *PFX-file-or-URL*
ssl auth-certificate install pfx-file *PFX-file-or-URL* **mac-password** *MAC-pwd*
ssl auth-certificate list [**brief** | **detail** | **subject-name** *cert-subject-name*]
ssl auth-certificate list subject-name *cert-subject-name* [**brief** | **detail**]
ssl auth-certificate list subject-name *cert-subject-name* **issuer-name** *cert-issuer-name* [**brief** | **detail**]

Arguments

Parameter	Description
delete all	Deletes all certificate authority data.
subject-name *cert-subject-name*	Specifies certificate subject name.
issuer-name *cert-issuer-name*	Specifies certificate issuer name.
install { **cert-file** *cert-file-or-URL* | **pfx-file** *PFX-file-or-URL* }	Installs the certificate authority data by using either a certificate file or a PFX file.
key-passphrase *private-key-file-or-URL*	Specifies the private key pass phrase.
mac-password *MAC-pwd*	Specifies the MAC password.
list	Lists the certificate authority data.
brief	Lists certificate authorities in brief format.
detail	Lists certificate authorities in detailed format.

Examples

None

ssl builtin-signing

Use the **ssl builtin-signing** command to configure the SSL host to use the built-in certificate to sign.

Command Mode: Global Configuration mode

Syntax

ssl builtin-signing { **enable** | **disable** }

Arguments

Parameter	Description
enable	Enables the SSL host to use the built-in certificate to sign.
disable	Disables the SSL host to use the built-in certificate to sign.

Examples

None

ssl cert-substitution

Use the **ssl cert-substitution** command to configure SSL certificate substitution.

Command Mode: Global Configuration mode

Syntax

ssl cert-substitution { **enable** | **disable** }

Arguments

Parameter	Description
enable	Enables the SSL certificate substitution.
disable	Disables the SSL certificate substitution.

Examples

None

ssl host-certificate

Use the **ssl host-certificate** command to configure SSL host certificate parameters.

Command Mode: Privileged EXEC mode

Syntax

ssl host-certificate delete all
ssl host-certificate delete subject-name *cert-subj*
ssl host-certificate delete subject-name *cert-subj* **issuer-name** *cert-issuer*
ssl host-certificate install cert-file *cert-file-or-URL* **key-file** *private-key-file-or-URL* [**key-passphrase** *private-key-file-or-URL*]
ssl host-certificate install pfx-file *PFX-file-or-URL*
ssl host-certificate install pfx-file *PFX-file-or-URL* **mac-password** *pwd-mac* [**crypt-password** *pwd-encrypt*]
ssl host-certificate list [**brief** | **detail** | **subject-name** *cert-subj*]
ssl host-certificate list subject-name *cert-subj* [**brief** | **detail**]
ssl host-certificate list subject-name *cert-subj* **issuer-name** *cert-issuer* [**brief** | **detail**]

Arguments

Parameter	Description
delete all	Deletes all host certificate data.
subject-name *cert-subj*	Specifies certificate subject name.
issuer-name *cert-issuer*	Specifies certificate issuer name.
install { **cert-file** *cert-file-or-URL* | **pfx-file** *PFX-file-or-URL* }	Installs the host certificate data by using either a certificate file or a PFX file.
key-file *private-key-file-or-URL*	Specifies the private key.
key-passphrase *private-key-file-or-URL*	Specifies the private key pass phrase.
mac-password *pwd-mac*	Specifies the MAC password
crypt-password *pwd-encrypt*	Specifies the encryption password
list	Lists the host certificate data.
brief	Lists certificate authorities in brief format.
detail	Lists certificate authorities in detailed format.

Examples

None

ssl signing-certificate

Use the **ssl signing-certificate** command to configure SSL signing certificate parameters.

Command Mode: Privileged EXEC mode

Syntax

ssl signing-certificate delete all
ssl signing-certificate delete subject-name *cert-subj*
ssl signing-certificate delete subject-name *cert-subj* **issuer-name** *cert-issuer*
ssl signing-certificate install cert-file *cert-file-or-URL* **key-file** *private-key-file-or-URL* [**key-passphrase** *private-key-file-or-URL*]
ssl signing-certificate install pfx-file *PFX-file-or-URL*
ssl signing-certificate install pfx-file *PFX-file-or-URL* **mac-password** *pwd-mac* [**crypt-password** *pwd-encrypt*]
ssl signing-certificate list [**brief** | **detail** | **subject-name** *cert-subj*]
ssl signing-certificate list subject-name *cert-subj* [**brief** | **detail**]
ssl signing-certificate list subject-name *cert-subj* **issuer-name** *cert-issuer* [**brief** | **detail**]

Arguments

Parameter	Description
delete all	Deletes all signing certificate data.
subject-name *cert-subj*	Specifies certificate subject name.
issuer-name *cert-issuer*	Specifies certificate issuer name.
install { **cert-file** *cert-file-or-URL* | **pfx-file** *PFX-file-or-URL* }	Installs the host certificate data by using either a certificate file or a PFX file.
key-file *private-key-file-or-URL*	Specifies the private key.
key-passphrase *private-key-file-or-URL*	Specifies the private key pass phrase.
mac-password *pwd-mac*	Specifies the MAC password
crypt-password *pwd-encrypt*	Specifies the encryption password
list	Lists the host certificate data.
brief	Lists certificate authorities in brief format.
detail	Lists certificate authorities in detailed format.

Examples

None

ssl subs-certificate

Use the **ssl subs-certificate** command to configure SSL substitute certificate parameters.

Command Mode: Privileged EXEC mode

Syntax

ssl subs-certificate list [**brief** | **detail** | **subject-name** *cert-subject-name*]
ssl subs-certificate list subject-name *cert-subject-name* [**brief** | **detail**]
ssl subs-certificate list subject-name *cert-subject-name* **issuer-name** *cert-issuer-name* [**brief** | **detail**]

Arguments

Parameter	Description
subject-name *cert-subject-name*	Specifies certificate subject name.
issuer-name *cert-issuer-name*	Specifies certificate issuer name.
list	Lists the host certificate data.
brief	Lists certificate authorities in brief format.
detail	Lists certificate authorities in detailed format.

Examples

None

subnet

Use the **subnet** command to configure subnets.

Use the **no** form of this command to remove a specific subnet.

Command Mode: Global Configuration mode

Syntax

subnet *ip-prefix/length* **advertize** { **enable** | **disable** }
subnet *ip-prefix/length* **advertize-bgp** { **enable** | **disable** }
subnet *ip-prefix/length* **advertize-ospf** { **enable** | **disable** }
subnet *ip-prefix/length* **comment**
subnet *ip-prefix/length* **exclude** { **enable** | **disable** }
subnet *ip-prefix/length* **local** { **enable** | **disable** }
subnet *ip-prefix/length* **metric** *0-100*
no subnet *ip-prefix/length*

Arguments

Parameter	Description
ip-prefix/length	Specifies IP address and subnet. For example, 10.0.10.0/24.
advertize	Subnet is okay to advertise.
advertize disable	Disables subnet advertising.
advertize enable	Enables subnet advertising.
advertize-bgp disable	Disables advertising to BGP peers.
advertize-bgp enable	Enables advertising to BGP peers.
advertize-ospf disable	Disables advertising to OSPF peers.
advertize-ospf enable	Enables advertising to OSPF peers.
comment	Adds a comments for a specified subnet entry.
exclude enable	Excludes a subnet from auto optimization.
exclude disable	Includes a subnet for auto optimization.
local	Subnet is local.
local disable	Disable local determination.
local enable	Enables local determination.
metric *0-100*	Specifies a subnet routing metric. Value can be between 0 and 100. Lower metric values have priority.

Usage Guidelines

Use these commands to build each appliance's subnet table.

Examples

None

system arp-table-size

Use the **system arp-table-size** command to configure the maximum system ARP table size.

Command Mode: Global Configuration mode

Syntax

system arp-table-size *max-arp-table-size*

Arguments

Parameter	Description
max-arp-table-size	Configure maximum ARP table size. The range is 1024 to 10240000 entries.

Examples

None

system auto-ipid

Use the **system auto-ipid** command to configure the auto IP ID feature.

Command Mode: Global Configuration mode

Syntax

system auto-ipid { **disable** | **enable** }

Arguments

Parameter	Description
disable	Disables the auto IP ID.
enable	Enables the auto IP ID.

Defaults

The default state is enabled.

Usage Guidelines

This command is part of three auto-discovery strategies: **auto IP ID**, **auto SYN**, and **auto-subnet**. All three are enabled by default.

Examples

None

system auto-mac-configure

Use the **system auto-mac-configure** command to configure the virtual appliance to auto-configure the MACs (Media Access Control).

Command Mode: Global Configuration mode

Syntax

system auto-mac-configure { **disable** | **enable** }

Arguments

Parameter	Description
disable	Allows user to manually map MACs to NIC interfaces on virtual appliances.
enable	Allows system to automatically map MACs to NIC interfaces on virtual appliances.

Examples

None

system auto-policy-lookup

Use the **system auto-policy-lookup** command to configure periodic policy lookups.

Command Mode: Global Configuration mode

Syntax

system auto-policy-lookup interval *0..65535*

Arguments

Parameter	Description
interval *0..65535*	Configures the interval for periodic policy lookups. The interval is expressed as the number of seconds between lookups.

Examples

None

system auto-subnet

Use the **system auto-subnet** command to configure the auto-subnet feature.

Command Mode: Global Configuration mode

Syntax

system auto-subnet add-local-lan { **disable** | **enable** }
system auto-subnet add-local-wan { **disable** | **enable** }
system auto-subnet bgp-redistribute { **disable** | **enable** }
system auto-subnet add-local metric *0 - 100*
system auto-subnet { **disable** | **enable** }

Arguments

Parameter	Description
add-local	Configures auto-subnet add-local capability.
add-local-lan	Configures auto-subnet add-local capability for LAN interfaces.
add-local-wan	Configures auto-subnet add-local capability for WAN interfaces.
add-local metric *0 - 100*	Configures the metric for automatically added local subnets.
bgp-redistribute	Configures the capability to redistribute BGP routes.
disable	Disables auto-subnet.
enable	Enables auto-subnet.

Defaults

The default state is enabled.

Examples

None

system auto-syn

Use the **system auto-syn** command to configure the auto SYN feature.

Command Mode: Global Configuration mode

Syntax

system auto-syn { **disable** | **enable** }

Arguments

Parameter	Description
disable	Disables auto SYN.
enable	Enables auto SYN.

Defaults

The default state is enabled.

Usage Guidelines

This command is part of three auto-discovery strategies: auto IP ID, auto SYN, and auto-subnet. All three are enabled by default.

Examples

None

system bandwidth

Use the **system bandwidth** command to configure appliance bandwidth.

Command Mode: Global Configuration mode

Syntax

system bandwidth max *kbps*
system bandwidth if-rx-target [**enable** | **disable**]

Arguments

Parameter	Description
max *kbps*	Configures maximum bandwidth for traffic transmitted to the WAN side in kilobits per second. This is a total of all tunnelized traffic and pass-through shaped traffic.
if-rx-target	Receive-side target bandwidth for the WAN interface.
disable	Disables Interface DRC (Dynamic Rate Control).
enable	Enables Interface DRC (Dynamic Rate Control).

Usage Guidelines

Receive-side bandwidth (also known as **Dynamic Rate Control**) is a feature that prevents one appliance from overwhelming another appliance as a result of sending it more data than the recipient can process.

Examples

To configure the appliance to transmit at a maximum bandwidth of 8000 kilobits per second:

```
ECV (config) # system bandwidth max 8000
```

system bonding

Use the **system bonding** command to configure the appliance etherchannel bonding option. When using a four-port Silver Peak appliance, you can bond pairs of Ethernet ports into a single port with one IP address per pair.

Command Mode: Global Configuration mode

Syntax

system bonding { **disable** | **enable** }

Arguments

Parameter	Description
disable	Deactivates system bonding mode (processes all incoming traffic).
enable	Activates system bypass mode (bypasses all incoming traffic).

Examples

None

system bypass

The **system bypass** command configures the appliance bypass option. The appliance mechanically isolates itself from the network, allowing traffic to flow without intervention.

Use the **no** form of this command to remove bypass capability when you've augmented and configured a virtual appliance's stock hardware with a Silicom BPVM or BPUSB card.

Command Mode: Global Configuration mode

Syntax

system bypass { **disable** | **enable** }
system bypass type { **bpvm** | **bpusb** } **mac address** *mac-addr*
no system bypass

Arguments

Parameter	Description
disable	Deactivates system bypass mode (processes all incoming traffic).
enable	Activates system bypass mode (bypasses all incoming traffic).
type { **bpvm** | **bpusb** } **mac address** *mac-addr*	Configures the Silicom virtual bypass card's interface MAC address: **bpvm** – Silicom PCI Ethernet bypass adapter **bpusb** – Silicom USB Ethernet bypass adapter

Usage Guidelines

Virtual appliances generally don't have a bypass card because they use stock hardware, like a Dell server. However, motivated customers can open up the server and add a Silicom card to get the same capabilities as one of Silver Peak's NX hardware appliances. Silicom calls this card BPVM.

As part of configuring the BPVM (part of a separate, documented procedure), you must indicate which network interface can be used to communicate with the card by specifying the MAC address.

Examples

To configure the appliance so that all traffic flows through the appliance without processing any of the traffic:

```
ECV (config) # system bypass enable
```

system cc enable / disable

The **system cc enable** command enables Common Criteria mode on the appliance. This command also enables FIPS mode and reboots the appliance. By default, Common Criteria mode is disabled.

The **system cc disable** command disables Common Criteria mode, disables FIPS mode, and reboots the appliance.

The **noconfirm** parameter prompts the CLI to provide command execution status up through the reboot of the appliance.

Common Criteria is an international standard for computer security certification. When Common Criteria mode is enabled, the appliance is Common Criteria compliant to a set of guidelines and certifications that ensure the appliance meets the security standard that includes PKI certificates, Online certificate status protocol, and enhanced logging.

Command Mode: Global Configuration mode

Syntax

system cc enable
system cc enable noconfirm

system cc disable
system cc disable noconfirm

Usage Guidelines

The **system cc enable** and **system cc disable** commands are only available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **cc enable** and **cc disable**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command enables Common Criteria on the appliance.

```
ECV (config) # system cc enable noconfirm

Enabling Common Criteria mode will automatically enable FIPS mode

This operation will cause a system reboot.

Additional security configurations will be applied and
any unsaved configuration changes will get saved.

Configuration changes saved, and cc mode enabled

The appliance is going to reboot...
```

```
ECV (config) #


System shutdown initiated -- logging off.

This will take a few minutes...

Connection to 10.80.171.181 closed.
[root@abcde ~]#
```

system contact

Use the **system contact** command to configure contact information for this appliance.

Command Mode: Global Configuration mode

Syntax

system contact *contact-info*

Arguments

Parameter	Description
contact-info	Defines the contact information for the appliance.

Usage Guidelines

If you want to include spaces in the contact information, wrap the entire phrase in quotes.

Examples

To configure Sherlock Holmes as the system contact:

```
ECV (config) # system contact “Sherlock Holmes”
```

system disk

Use the **system disk** command to insert or remove a disk from the RAID array.

Command Mode: Privileged EXEC mode

Syntax

system disk *disk-ID* { **insert** | **remove** }

Arguments

Parameter	Description
disk-ID	Designates the host name for the appliance.
insert	Insert disk into RAID array.
remove	Remove disk from RAID array.

Examples

To add disk 9 back into an NX-8500's RAID array:

```
ECV (config) # system disk 9 insert
```

system disk encryption

Use the **system disk encryption** command to encrypt the appliance disk.

Command Mode: Global Configuration mode

Syntax

system disk encryption { **disable** | **enable** }

Arguments

Parameter	Description
encryption disable	Disables disk encryption.
encryption enable	Enables disk encryption.

Examples

None

system dpc

Use the **system dpc** command to configure Dynamic Path Control (DPC) for this appliance.

Command Mode: Global Configuration mode

Syntax

system dpc *failover-behavior* { **disable** | **fail-back** | **fail-stick** }

Arguments

Parameter	Description
tunnel-fail-behavior *failover-behavior*	If there are parallel tunnels and one fails, then Dynamic Path Control determines where to send the flows. There are three failover behaviors.
disable	When the original tunnel fails, the flows aren't routed to another tunnel.
fail-back	When the failed tunnel comes back up, the flows return to the original tunnel.
fail-stick	When the failed tunnel comes back up, the flows don't return to the original tunnel. They stay where they are.

Examples

None

system eclicense

Use the **system eclicense** command to configure a Silver Peak EdgeConnect license.

Command Mode: Global configuration mode

Syntax

system eclicense boost bandwidth *bandwidth-limit-in-kbps*
system eclicense boost { **disable** | **enable** }
system eclicense plus { **disable** | **enable** }

Arguments

Parameter	Description
boost	EdgeConnect Boost portal license configuration
plus	EdgeConnect Plus portal license configuration
bandwidth *bandwidth-limit-in-kbps*	Sets the EdgeConnect Boost bandwidth limit.
disable	Disables EdgeConnect Boost license.
enable	Enables EdgeConnect Boost license.

Usage Guidelines

This command is only available for EdgeConnect appliances.

Examples

None

system fips enable / disable

The **system fips enable** command enables FIPS mode and reboots the appliance. By default, FIPs mode is disabled.

The **system fips disable** command disables FIPS mode.

The **noconfirm** parameter prompts the CLI to provide command execution status up through the reboot of the appliance.

Federal Information Processing Standards (FIPS) is a set of publicly announced standards that the National Institute of Standards and Technology (NIST) developed for use in non-military United States government agencies and contractor applications.

Command Mode: Global Configuration mode

Syntax

system fips enable
system fips enable noconfirm

system fips disable
system fips disable noconfirm

Usage Guidelines

The **system fips enable** and **system fips disable** commands are not available in ECOS version 9.4.3 and all later versions. Equivalent commands available in these versions are **fips enable** and **fips disable**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command enables FIPS mode on the appliance.

```
ECV (config) # system fips enable
This operation will cause a system reboot.
Do you want to proceed? [y/n] y
```

system fips secure erase

The **system fips secure erase** command renders the appliance non-functional by overwriting all data with either zeros or ones. Secure erase prevents unauthorized access to sensitive information when disposing of or selling an appliance. This command provides a zeroization function as required by ISO 24759 and FIPS 140-2 implementation guidance.

Federal Information Processing Standards (FIPS) is a set of publicly announced standards that the National Institute of Standards and Technology (NIST) developed for use in non-military United States government agencies and contractor applications.

Command Mode: Global Configuration mode

Syntax

system fips secure erase

Usage Guidelines

The **system fips secure erase** command is not available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **fips secure erase**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command renders the appliance non-functional.

```
ECV (config) # system fips secure erase
Note: This command zeroizes the drive, rendering the appliance non-functional; ECOS
    will no longer run.
The entire appliance must be sent back to Silver Peak (RMA).
```

system firmware

Use the **system firmware** command to manage the appliance firmware.

Command Mode: Global configuration mode

Syntax

system firmware update { **LCC** | **BIOS** | **SAS** | **NIC** }

Arguments

Parameter	Description
update { **LCC** | **BIOS** | **SAS** | **NIC** }	Updates the specified appliance firmware: **LCC** Lifecycle Controller Firmware **BIOS** BIOS Firmware **SAS** Disk Controller Firmware **NIC** NIC Firmware

Examples

None

system hostname

Use the **system hostname** command to configure host name for this appliance.

Command Mode: Global Configuration mode

Syntax

system hostname *hostname-text*

Arguments

Parameter	Description
hostname-text	Designates the host name for the appliance.

Usage Guidelines

Hostnames may contain letters, numbers, periods ('.'), and hyphens ('-'), but may not begin with a hyphen. Hostnames cannot contain spaces.

Examples

None

system int-hairpin

Use the **system int-hairpin** command to configure the internal hairpinning feature.

Command Mode: Global Configuration mode

Syntax

system int-hairpin { **disable** | **enable** }

Arguments

Parameter	Description
disable	Disables the internal hairpinning feature.
enable	Enables the internal hairpinning feature.

Usage Guidelines

Hairpinning redirects inbound LAN traffic back to the WAN.

Examples

None

system ip-broadcast enable

The **system ip-broadcast enable** command enables the internal hairpinning feature. Hairpinning is the method where a packet travels to an interface and proceeds towards the internet but makes a “hairpin turn” and returns on the same interface.

The **no system ip-broadcast enable** command disables the internal hairpinning feature.

Command Mode: Global Configuration mode

Syntax

system ip-broadcast enable
no system ip-broadcast enable

Defaults

The internal hairpinning feature is disabled by default.

Examples

This command enables the internal hairpinning feature.

```
ECV (config) # system ip-broadcast enable
ECV (config) #
```

system location

Use the **system location** command to configure location information for this appliance.

Command Mode: Global Configuration mode

Syntax

system location *location-info*

Arguments

Parameter	Description
location-info	Specifies the location information for the appliance.

Usage Guidelines

If you want to include spaces in the contact information, wrap the entire phrase in quotes.

Examples

To specify the appliance location as “Pittsburgh”:

```
ECV (config) # system location Pittsburgh
```

To specify the appliance location as Earth (specified as a phrase):

```
ECV (config) # system location “third rock from the sun”
```

system mode

Use the **system mode** command to configure the appliance's mode (bridge or router) and next-hop IP. When using a 4-port appliance, you can configure two next-hops (one for each WAN interface).

Use the **no** form of the command to reset the router or bridge mode setting to its default.

Command Mode: Global Configuration mode

Syntax

system mode bridge *intf* **inbound-max-bawndwidth** *bw-kbps*
system mode bridge *intf* **outbound-max-bandwidth** *bw-kbps*
system mode bridge ip *IP-addr mask-length* **nexthop** *IP-addr* [**second-ip** *IP-addr mask-length* **second-nexthop** *IP-addr*]
system mode router *intf* **inbound-max-bandwidth** *bw-kbps*
system mode router *intf* **outbound-max-bandwidth** *bw-kbps*
system mode router ip *IP-addr mask-length* **nexthop** *IP-addr* [**second-ip** *IP-addr mask-length* **second-nexthop** *IP-addr*]
system mode router *intf IP-addr mask-length* **nh** *IP-addr*
system mode router *intf IP-addr mask-length* **nh** *IP-addr intf IP-addr mask-length* **nh** *IP-addr*
system mode router *intf IP-addr mask-length* **nh** *IP-addr intf IP-addr mask-length* **nh** *IP-addr*
system mode router *intf IP-addr mask-length* **nh** *IP-addr intf IP-addr mask-length* **nh** *IP-addr intf IP-addr mask-length* **nh** *IP-addr intf IP-addr mask-length* **nh** *IP-addr*
system mode server
system mode server inbound-max-bandwidth *bw-kbps*
system mode server outbound-max-bandwidth *bw-kbps*
no system mode

Arguments

Parameter	Description
bridge	Configures Bridge (in-line) Mode
inbound-max-bandwidth *bw-kbps*	Configures the interface's inbound maximum bandwidth
ip *IP-addr*	Configures the appliance IP address.
mask-length	Configures the appliance netmask or mask length.
nexthop *IP-addr*	Specifies the IP address of the: (bridge mode) – WAN next-hop for virtual bridge (router mode) – router mode next-hop IP
nh	Configures the Route mode next-hop

Parameter	Description
outbound-max-bandwidth *bw-kbps*	Configures the interface's outbound maximum bandwidth
router	Configures Router (out-of-path) Mode
second-ip *IP-addr*	Configures the appliance's second IP address for tunnel traffic.
second-nexthop *IP-addr*	Specifies the next-hop IP address that's associated with second IP address.
server	Configures Server Mode (single interface)

Defaults

The default system mode is bridge (in-line) mode.

Examples

To configure an appliance with the IP address, 172.27.120.1 to be in router mode, with a net-mask of 255.255.255.0 and a next-hop IP address of 172.27.120.2:

```
ECV (config) # system mode router ip 172.27.120.1 /24 nexthop 172.27.120.2
```

To reset the system to the default (bridge) mode:

```
ECV (config) # no system mode
```

system nat-all-inbound

Use the **system nat-all-inbound** command to configure the inbound source NAT feature.

Command Mode: Global Configuration mode

Syntax

system nat-all-inbound disable
system nat-all-inbound nat-ip { *intf-IP-addr* | **auto** }
system nat-all-inbound nat-ip { *intf-IP-addr* | **auto** } **fallback** { **enable** | **disable** }

Arguments

Parameter	Description
disable	Disables inbound source NAT.
nat-ip { *intf-IP-addr* | **auto** }	Configures the inbound source NAT IP address.
fallback enable	Specifies fallback to the next available NAT IP address upon port exhaustion with the current NAT IP address.
fallback disable	Specifies not to fallback to the next available NAT IP address upon port exhaustion.

Examples

None

system nat-all-outbound

Use the **system nat-all-outbound** command to configure the inbound source NAT feature.

Command Mode: Global Configuration mode

Syntax

system nat-all-outbound disable
system nat-all-outbound nat-ip { *intf-IP-addr* | **auto** }
system nat-all-outbound nat-ip { *intf-IP-addr* | **auto** } **fallback** { **enable** | **disable** }

Arguments

Parameter	Description
disable	Disables outbound source NAT.
nat-ip { *intf-IP-addr* | **auto** }	Configures the outbound source NAT IP address.
fallback enable	Specifies fallback to the next available NAT IP address upon port exhaustion with the current NAT IP address.
fallback disable	Specifies not to fallback to the next available NAT IP address upon port exhaustion.

Examples

None

system network-memory

Use the **system network-memory** command to configure system network memory.

Command Mode: Privileged EXEC mode (system erase)
Command Mode: Global Configuration mode (system media)

Syntax

system network-memory erase
system network-memory media ram
system network-memory media ram-and-disk

Arguments

Parameter	Description
erase	Erases system network memory.
media	Configures data store usage for RAM or RAM-and-disk.
ram	Network Memory data stored in RAM only
ram-and-disk	Network Memory data stored in RAM and disk.

Defaults

The default Network Memory mode is 0.

Examples

None

system passthru-to-sender

Use the **system passthru-to-sender** command to configure passthrough L2 return to sender.

Command Mode: Global configuration mode

Syntax

system passthru-to-sender
system passthru-to-sender { **disable** | **enable** }

Arguments

Parameter	Description
disable	Disables passthrough L2 return to sender.
enable	Enables passthrough L2 return to sender.

Examples

None

system peer-list

Use the **system peer-list** command to assign a priority to a peer.

Use the **no** form of this command to remove the peer name from the priority list.

Command Mode: Global configuration mode

Syntax

system peer-list *peer-name weight*
no system peer-list *peer-name*

Arguments

Parameter	Description
peer-name	Specifies the peer appliance.
weight	Specifies the priority to assign to the peer.

Usage Guidelines

When an appliance receives a Subnet with the same Metric from multiple remote or peer appliances, it uses the Peer Priority list as a tie-breaker.

If a Peer Priority is not configured, then the appliance randomly distributes flows among multiple peers.

The lower the number, the higher the peer's priority.

Examples

None

system registration

Use the **system registration** command to register the appliance with the Silver Peak portal.

Use the **no** form of this command to remove Silver Peak portal registration data.

Command Mode: Global Configuration mode

Syntax

system registration *Account-Key Account-Name*
system registration *Account-Key Account-Name App-Group-Name*
system registration *Account-Key Account-Name App-Group-Name App-Site-Name*
no system registration

Arguments

Parameter	Description
Account-Key	Specifies the Account Key assigned by Silver Peak.
Account-Name	Specifies the Account Name assigned by Silver Peak.
App-Group-Name	Optional tag assigned by user for ease of identification.
App-Site-Name	Optional tag assigned by user for ease of identification.

Examples

None

system router

Use the **system router** command to configure in-line router mode.

Use the **no** form of this command to remove in-line router mode in whole or in part.

Command Mode: Global Configuration mode

Syntax

system router *router-name* **create interface** *intf* { **lan** | **wan** }
no system router *router-name*

system router *router-name* **dhcp**

system router *router-name* **dhcp vlan** *VLAN-ID* [**inbound-max-bw** *bw-kbps* | **label** *intf-label* | **outbound-max-bw** *bw-kbps* | **renew** | **security-mode** *security-mode-intf*]

system router *router-name* **ip** *IP-addr* [**inbound-max-bw** *bw-kbps* | **label** *intf-label* | **outbound-max-bw** *bw-kbps* | **security-mode** *security-mode-intf*]

system router *router-name* **ip** *IP-addr mask* **nexthop** *IP-addr* [**vlan** *VLAN-ID*]

system router *router-name* **pppoe** [*Unit-number*]

system router *router-name* **pppoe** *Unit-number* [**inbound-max-bw** *bw-kbps* | **label** *intf-label* | **outbound-max-bw** *bw-kbps* | **security-mode** *security-mode-intf*]

no system router *router-name* **dhcp** [**vlan** *VLAN-ID*]

no system router *router-name* **dhcp vlan** *VLAN-ID* **label**

no system router *router-name* **ip** *IP-addr* **label**

no system router *router-name* **pppoe** *Unit-number* [**label**]

Arguments

Parameter	Description
create interface *physical-intf*	Specifies whether to create **lan0**, **wan0**, **lan1**, **wan1**, etc.
dhcp	Adds DHCPv4.
inbound-max-bw *bw-kbps*	Specifies the VLAN inbound max bandwidth in kilobits per second.
ip *IP-addr*	Specifies the router IP address
label *intf-label*	Specifies the interface label.
nexthop *IP-addr*	Specifies the Router mode next-hop.

Parameter	Description
outbound-max-bw *bw-kbps*	Specifies the VLAN outbound max bandwidth in kilobits per second.
renew	Renews DHCP.
router *router-name*	Specifies the router name.
security-mode *security-mode-router-intf*	Choose a security mode for the interface: **0** Open **1** Harden **2** Stateful Firewall **3** Stateful Firewall with SNAT
security-mode *security-mode-PPPoE-intf*	Choose a security mode for the interface: **0** Open **1** Harden **2** Stateful Firewall
vlan *VLAN-ID*	Specifies the DHCPv4 VLAN ID.
{ **lan** | **wan** }	Refers to the LAN side or the WAN side.
mask	Specifies the netmask. For example, 255.255.255.0, or /24.
Unit-number	PPPoE Unit number

Examples

None

system routing

Use the **system routing** command to configure interface routing.

Use the **no** form of this command to reset system-level routing information.

Command Mode: Global Configuration mode

Syntax

system routing inline
system routing redundancy { **default** | **none** | **lan-native** | **lan-native-vlan** | **lan-and-wan** | **all** }
no system routing inline

Arguments

Parameter	Description
inline	Enables inline router mode.
redundancy	Configures redundancy of routes between interfaces.
default	LAN routing allowed between VLANs and native interfaces (equivalent to lan-native-vlan)
none	No routing allowed between interfaces
lan-native	LAN routing allowed between native interfaces (no routing allowed between VLANs)
lan-native-vlan	LAN routing allowed between VLANs and native interfaces
lan-and-wan	LAN and WAN routing allowed between native interfaces
all	LAN and WAN routing allowed between all interfaces (caveat: this may disrupt DPC)

Examples

None

system smb-signing

Use the **system smb-signing** command to enable or disable SMB signing.

Command Mode: Global Configuration mode

Syntax

system smb-signing { **disable** | **enable** }

Arguments

Parameter	Description
disable	Disables SMB Signing optimization.
enable	Enables SMB Signing optimization.

Defaults

The default is disabled.

Usage Guidelines

This command must be executed together with the **cifs signing delegation domain** command.

Examples

None

system ssl-ipsec-override

Use the **system ssl-ipsec-override** command to configure SSL IPSec override.

Command Mode: Global Configuration mode

Syntax

system ssl-ipsec-override { **disable** | **enable** }

Arguments

Parameter	Description
disable	Deactivates the SSL IPSec override feature.
enable	Activates the SSL IPSec override feature.

Defaults

This feature is disabled by default.

Examples

None

tacacs-server

Use the **tacacs-server** command to configure hosts TACACS+ server settings for user authentication.

Command Mode: Global configuration mode

Syntax

tacacs-server host *IP-addr* [**auth-port** *port*] [**auth-type** { **ascii** | **pap** }] [**key** *string*] [**retransmit** *0…3*] [**timeout** *1…15*]

tacacs-server { **key** *string* | **retransmit** *0..3* | **timeout** *1…15* }

no tacacs-server host *IP-addr* [**auth-port** *port*]

no tacacs-server { **key** | **retransmit** | **timeout** }

Arguments

Parameter	Description
host *IP-addr*	Configures host, at specified IP address, to send TACACS+ authentication requests. Use the **no** form of this command to stop sending TACACS+ authentication requests to host.
auth-port *port*	Specifies the authentication port to use with this TACACS+ server. Use the **no** form of this command to stop sending TACACS+ authentication requests to the authentication port.
auth-type { **ascii** | **pap** }	Specifies the authentication type to use with this TACACS+ server. The options are: **ascii** – ASCII authentication **pap** – PAP (Password Authentication Protocol) authentication
key *string*	Specifies the shared secret key to use with this TACACS+ server. Use the **no** form of this command to remove the global TACACS+ server key.
retransmit *0…3*	Specifies the maximum number of retries that can be made in the attempt to connect to this TACACS+ server. The range is 0 to 3. Use the **no** form of this command to reset the global TACACS+ server retransmit count to its default.
timeout *1…15*	Specifies the number of seconds to wait before the connection times out with this TACACS+ server, because of keyboard inactivity. The range is 1 to 15 seconds. Use the **no** form of this command to reset the global TACACS+ server timeout setting to its default.

Usage Guidelines

When you don't specify a host IP, then configurations for **host**, **key**, and **retransmit** are global for TACACS+ servers.

Examples

To define the TACACS+ shared secret as "mysecret":

```
ECV (config) # tacacs-server key mysecret
```

To specify that the TACACS+ server with the IP address of 10.10.10.10 uses PAP authentication and tries to retransmit a maximum of 9 times:

```
ECV (config) # tacacs-server host 10.10.10.10 auth-type pap retransmit 9
```

To reset, to its default, the number of seconds after which the TACACS+ server times out after keyboard inactivity:

```
ECV (config) # no tacacs-server timeout
```

tca

The **tca** command to set the parameters for threshold crossing alerts.

The **no** form of this command to return a special instance (that is, specific values for a named tunnel) to the **default** values. Use **no tca** *tca-name* **default** to delete the TCA instance.

Command Mode: Global configuration mode

Syntax

tca *tca-name* **default** { **rising** | **falling** } **raise-threshold** *value* **clear-threshold** *value* [**sample-count** *number-samples*]

tca *tca-name tunnel-name* { **rising** | **falling** } **raise-threshold** *value* **clear-threshold** *value* [**sample-count** *number-samples*]

tca *tca-name* { **pass-through** | **pass-through-unshaped** } { **rising** | **falling** } **raise-threshold** *value* **clear-threshold** *value* [**sample-count** *number-samples*]

no tca *tca-name* { **default** | *tunnel-name* }

no tca *tca-name* {**default** | *tunnel-name*} [**rising** | **falling**]

tca *tca-name* { **default** | *tunnel-name* } { **enable** | **disable** }

tca *tca-name* { **pass-through** | **pass-through-unshaped** } { **enable** | **disable** }

Arguments

Parameter	Description
tca *tca-name*	Specifies which threshold crossing alert to configure. Some apply to one or more types of traffic. Others only have default values.The options are: **file-system-utilization** How much of the file system space has been used, expressed as a percentage. **lan-side-rx-throughput** LAN–side Receive throughput, in kilobits per second (**kbps**). **latency** Tunnel latency, in milliseconds (**ms**). **loss-post-fec** Tunnel loss, as **tenths of a percent**, *after* applying Forward Error Correction (FEC). **loss-pre-fec** Tunnel loss, as **tenths of a percent**, *before* applying Forward Error Correction (FEC).

Parameter	Description
	oop-post-poc Tunnel out-of-order packets, as **tenths of a percent**, *after* applying Packet Order Correction (POC). **oop-pre-poc** Tunnel out-of-order packets, as **tenths of a percent**, *before* applying Packet Order Correction (POC). **optimized flows** Total number of optimized flows. **reduction** Tunnel reduction, in percent (**%**). **total-flows** Total number of flows. **utilization** Tunnel utilization, as a percent (**%**). **wan-side-tx-throughput** WAN–side transmit throughput, in kilobits per second (**kbps**).
default	Sets the **tca** *tca-name* argument values for any tunnels that weren't specifically named in configuring an argument. For example, if you configured latency values for **tunnel_1** but not for **tunnel_2** and **tunnel_3**, then configuring **default** would only apply values to **tunnel_2** and **tunnel_3**.
tunnel-name	For specifying an individual tunnel for threshold configuration.
falling	Specifies a threshold crossing alarm for when the stat value falls too low.
rising	Specifies a threshold crossing alarm for when the stat value rises too high.
raise-threshold *value*	Specifies at what value to raise an alert.
clear-threshold *value*	After an alarm has been raised, specifies at what value to clear the alert. For a **rising** alarm, the clear-threshold value is equal to or less than the raise-threshold. For a **falling** alarm, the clear-threshold value is equal to or more than the raise-threshold
sample-count *number-samples*	Sets the number of samples that the metric must sustain below (or above) the threshold in order to raise (or clear) the alert.
enable	Enables this threshold control alert instance.
disable	Disables this threshold control alert instance.

Usage Guidelines

This table lists the default state of each type of threshold crossing alert:

TCA	Type	Unit	Default [ON, OFF]	allow rising	allow falling
wan-side-throughput	system	kbps	OFF	4	4

TCA	Type	Unit	Default [ON, OFF]	allow rising	allow falling
lan-side-throughput	system	kbps	OFF	4	4
optimized-flows	system	flows	OFF	4	4
total-flows	system	flows	OFF	4	4
file-system-utilization	system	%	**ON**__1__	4	
latency	tunnel	msec	**ON**	4	
loss-pre-fec	tunnel	1/10th %	OFF	4	
loss-post-fec	tunnel	1/10th %	OFF	4	
oop-pre-poc	tunnel	1/10th %	OFF	4	
oop-post-poc	tunnel	1/10th %	OFF	4	
utilization	tunnel	%	OFF	4	4
reduction	tunnel	%	OFF		4

Examples

To raise an alert when the percent reduction for *tunnel_a* falls below 60% and to clear the alarm as soon as reduction reaches 70%:

```
ECV (config) # tca reduction tunnel_a falling raise-threshold 60 clear-threshold 70
```

tcpdump

Use the **tcpdump** command to display packets on a network.

Command Mode: Privileged EXEC mode

Syntax

tcpdump [*tcpdump-options*]

Arguments

Parameter	Description
tcpdump-options	Enter one of the following options: **-A** Print each packet (minus its link level header) in ASCII. Handy for capturing web pages. **-c** Exit after receiving count packets. **-C** Before writing a raw packet to a savefile, check whether the file is currently larger than file_size and, if so, close the current savefile and open a new one. Savefiles after the first savefile will have the name specified with the -w flag, with a number after it, starting at 1 and continuing upward. The units of file_size are millions of bytes (1,000,000 bytes, not 1,048,576 bytes). **-d** Dump the compiled packet-matching code in a human readable form to standard output and stop. **-dd** Dump packet-matching code as a C program fragment. **-ddd** Dump packet-matching code as decimal numbers (preceded with a count). **-D** Print the list of the network interfaces available on the system and on which tcpdump can capture packets. For each network interface, a number and an interface name, possibly followed by a text description of the interface, is printed. The interface name or the number can be supplied to the -i flag to specify an interface on which to capture. **-e** Print the link-level header on each dump line.

Parameter	Description
	-E Use spi@ipaddr algo:secret for decrypting IPsec ESP packets that are addressed to addr and contain Security Parameter Index value spi. This combination may be repeated with comma or newline separation. Note that setting the secret for IPv4 ESP packets is supported at this time. Algorithms may be des-cbc, 3des-cbc, blowfish-cbc, rc3-cbc, cast128-cbc, or None The default is des-cbc. The ability to decrypt packets is only present if tcpdump was compiled with cryptography enabled. *secret* is the ASCII text for ESP secret key. If preceded by 0x, then a hex value will be read. The option assumes RFC2406 ESP, not RFC1827 ESP. The option is only for debugging purposes, and the use of this option with a true 'secret' key is discouraged. By presenting IPsec secret key onto command line you make it visible to others, via ps(1) and other occasions. In addition to the above syntax, the syntax file name may be used to have tcpdump read the provided file in. The file is opened upon receiving the first ESP packet, so any special permissions that tcpdump may have been given should already have been given up. **-f** Print 'foreign' IPv4 addresses numerically rather than symbolically. **-F** Use file as input for the filter expression. An additional expression given on the command line is ignored. **-i** Listen on interface. If unspecified, tcpdump searches the system interface list for the lowest numbered, configured up interface (excluding loopback). Ties are broken by choosing the earliest match. **-l** Make stdout line buffered. Useful if you want to see the data while capturing it. For example, tcpdump -l | tee dat, or tcpdump -l > dat & tail -f dat **-L** List the known data link types for the interface and exit. **-m** Load SMI MIB module definitions from file module. This option can be used several times to load several MIB modules into tcp-dump. **-M** Use secret as a shared secret for validating the digests found in TCP segments with the TCP-MD5 option (RFC 2385), if present. **-n** Don't convert host addresses to names. This can be used to avoid DNS lookups. **-nn** Don't convert protocol and port numbers etc. to names either. **-N** Don't print domain name qualification of host names. For example, if you give this flag then tcpdump will print `nic` instead of `nic.ddn.mil`.

Parameter	Description
	-O Do not run the packet-matching code optimizer. This is useful only if you suspect a bug in the optimizer. **-p** Don't put the interface into promiscuous mode. Note that the interface might be in promiscuous mode for some other reason; hence, **-p** cannot be used as an abbreviation for 'ether host {local-hw-addr} or ether broadcast'. **-q** Quick (quiet?) output. Print less protocol information so output lines are shorter. **-R** Assume ESP/AH packets to be based on old specification (RFC1825 to RFC1829). If specified, tcpdump will not print replay prevention field. Since there is no protocol version field in ESP/AH specification, tcpdump cannot deduce the version of ESP/AH protocol. **-r** Read packets from file (which was created with the -w option). Standard input is used if file is "-". **-S** Print absolute, rather than relative, TCP sequence numbers. **-s** Snarf snaplen bytes of data from each packet rather than the default of 68 (with SunOS's NIT, the minimum is actually 96). 68 bytes is adequate for IP, ICMP, TCP, and UDP but may truncate protocol information from name server and NFS packets. Packets truncated because of a limited snapshot are indicated in the output with **[__|__proto]**, where **proto** is the name of the protocol level at which the truncation has occurred. Note that taking larger snapshots both increases the amount of time it takes to process packets and, effectively, decreases the amount of packet buffering. This may cause packets to be lost. You should limit snaplen to the smallest number that will capture the protocol information you're interested in. Setting snaplen to 0 means use the required length to catch whole packets. **-T** Force packets selected by "expression" to be interpreted the specified type. Currently known types are: **aodv** (Ad-hoc On-demand Distance Vector protocol) **cnfp** (Cisco NetFlow protocol) **rpc** (Remote Procedure Call) **rtp** (Real-Time Applications protocol) **rtcp** (Real-Time Applications control protocol) **snmp** (Simple Network Management Protocol) **tftp** (Trivial File Transfer Protocol) **vat** (Visual Audio Tool) **wb** (distributed White Board) **-t** Don't print a timestamp on each dump line. **-tt** Print an unformatted timestamp on each dump line. **-ttt** Print a delta (in micro-seconds) between current and previous line on each dump line. **-tttt** Print a timestamp in default format proceeded by date on each dump line. **-u** Print undecoded NFS handles.

Parameter	Description
	-U Make output saved via the **-w** option “packet-buffered”; that is, as each packet is saved, it will be written to the output file, rather than being written only when the output buffer fills. The -U flag will not be supported if tcpdump was built with an older version of libpcap that lacks the pcap_dump_flush() function. **-v** Parses and prints (slightly more) verbose output. For example, time to live, identification, total length, and options in IP packets are printed. Also enables additional packet integrity checks such as verifying the IP and ICMP header checksum. When writing to a file with the -w option, report, every 10 seconds, the number of packets captured. **-vv** Even more verbose output. For example, additional fields are printed from NFS reply packets, and SMB packets are fully decoded. **-vvv** Even more verbose output. For example, telnet SB… SE options are printed in full. With -X Telnet options are printed in hexl. **-w** Write the raw packets to file rather than parsing and printing them out. They can later be printed with the -r option. Standard output is used if file is “-”. **-W** Used in conjunction with the -C option, this will limit the number of files created to the specified number, and begin overwriting files from the beginning, thus creating a ‘rotating’ buffer. In addition, it will name the files with enough leading 0s to support the maximum number of files, allowing them to sort correctly. **-x** Print each packet (minus its link level header) in hex. The smaller of the entire packet or snaplen bytes will be printed. Note that this is the entire link-layer packet, so for link layers that pad (e.g. Ethernet), the padding bytes will also be printed when the higher layer packet is shorter than the required padding. **-xx** Print each packet, including its link level header, in hex. **-X** Print each packet (minus its link level header) in hex and ASCII. This is handy for analyzing new protocols. **-XX** Print each packet, including its link level header, in hex and ASCII. **-y** Set the data link type to use while capturing packets to datalinktype. **-Z** Drops privileges (if root) and changes user ID to user and the group ID to the primary group of user. This behavior can also be enabled by default at compile time.

Examples

None

tcptraceroute

Use the **tcptraceroute** command to record route information in environments where traditional ICMP traceroute is defeated by firewalls or other filters.

Command Mode: EXEC mode

Syntax

tcptraceroute [-nNFSAE] [-i *intf-name*] [-f *first-ttl*] [-l *packet-length*][-q number-queries*] [-t *tos*][-m *max-ttl*] [-pP] *source-port*] [-s *source-address*][-w *wait-time*] *host-text* [*dest-port*] [*packet-length*]

Arguments

Parameter	Description
tcptraceroute-options	Specifies the type of **tcptraceroute**. Select from the following options: **-n** Display numeric output, rather than doing a reverse DNS lookup for each hop. By default, reverse lookup is not attempted on RFC1918 address space, regardless of -n flag. **-N** Perform a reverse DNS lookup for each hop, including RFC1918 addresses. **-f** Set initial TTL used in first outgoing packet. Default is 1. **-m** Set the maximum TTL used in outgoing packets. Default is 30. **-p** Use the specified local TCP port in outgoing packets. The default is to obtain a free port from the kernel using **bind**. Unlike with traditional **traceroute**, this number will not increase with each hop. **-s** Set source address for outgoing packets. See -i flag. **-i** Use the specified interface for outgoing packets. **-q** Set the number of probes to be sent to each hop. Default is 3. **-w** Set the timeout, in seconds, to wait for a response for each probe. Default is 3. **-S** Set the TCP SYN flag in outgoing packets. This is the default, if neither -S or -A is specified. **-A** Set the TCP ACK flag in outgoing packets. By doing so, it is possible to trace through stateless firewalls which permit outgoing TCP connections. **-E** Send ECN SYN packets, as described in RFC2481.__-t **Set the IP TOS (type of service) to be used in outgoing packets. The default is not to set any TOS.** -F **Set the IP "don't fragment" bit in outgoing packets.** -l__ Set the total packet length to be used in outgoing packets. If the length is greater than the minimum size required to assemble the necessary probe packet headers, this value is automatically increased.

Parameter	Description
	-d Enable debugging, which may or may not be useful. **–dnat** Enable DNAT detection, and display messages when DNAT transitions are observed. DNAT detection is based on the fact that some NAT devices, such as some Linux 2.4 kernels, do not correctly rewrite the IP address of the IP packets quoted in ICMP time-exceeded messages tcptraceroute solicits, revealing the destination IP address an outbound probe packet was NATed to. NAT devices which correctly rewrite the IP address quoted by ICMP messages, such as some Linux 2.6 kernels, will not be detected. For some target hosts, it may be necessary to use –dnat in conjunction with –track-port. See the examples.txt file for examples. **–no-dnat** Enable DNAT detection for the purposes of correctly identifying ICMP time-exceeded messages that match up with outbound probe packets, but do not display messages when a DNAT transition is observed. This is the default behavior. **–no-dnat-strict** Do not perform DNAT detection. No attempt is made to match ICMP time-exceeded messages with outbound probe packets. When tracerouting through a NAT device that does not rewrite IP addresses of IP packets quoted in ICMP time-exceeded messages, some hops along the path may appear unresponsive. This option is not needed in the vast majority of cases, but may be utilized if it is suspected that the DNAT detection code is misidentifying ICMP time-exceeded messages.
host *dest-port length*	The destination port and the packet length.

Defaults

The probe packet length is **40**.

Usage Guidelines

- **tcptraceroute** is a traceroute implementation using TCP packets.
- The more traditional traceroute sends out either UDP or ICMP ECHO packets with a TTL of one, and increments the TTL until the destination has been reached. By printing the gateways that generate ICMP time exceeded messages along the way, it is able to determine the path packets are taking to reach the destination.
- The problem is that with the widespread use of firewalls on the modern Internet, many of the packets that **traceroute** sends out end up being filtered, making it impossible to completely trace the path to the destination.

  However, in many cases, if hosts sitting behind the firewall are listening for connections on specific ports, then these firewalls will permit inbound TCP packets to those ports.

By sending out TCP SYN packets instead of UDP or ICMP ECHO packets, **tcptraceroute** is able to bypass the most common firewall filters.

- It is worth noting that **tcptraceroute** never completely establishes a TCP connection with the destination host.

  If the host is not listening for incoming connections, it will respond with an RST indicating that the port is closed.

  If the host instead responds with a SYN|ACK, the port is known to be open, and an RST is sent by the kernel **tcptraceroute** is running on to tear down the connection without completing three-way handshake. This is the same half-open scanning technique that **nmap** uses when passed the **-sS** flag.

Examples

None

telnet

Use the **telnet** command to log into another system by using telnet.

Command Mode: EXEC mode

Syntax

telnet [*telnet-options*] *host* [*port*]

Arguments

Parameter	Description
telnet-options	Specifies the type of **tcptraceroute**. Select from the following options: **-8** Specify an 8-bit data path. This causes an attempt to negotiate the TELNET BINARY option on both input and output. **-E** Stop any character from being recognized as an escape character. **-F** Forward a forwardable copy of the local credentials to the remote system. **-K** Specify no automatic login to the remote system. **-L** Specify an 8-bit data path on output. This causes the BINARY option to be negotiated on output. **-S tos** Set the IP type-of-service (TOS) option for the telnet connection to the value tos, which can be a numeric TOS value (in decimal, or a hex value preceded by 0x, or an octal value preceded by a leading 0) or, on systems that support it, a symbolic TOS name found in the /etc/iptos file. **-X atype** Disable the atype type of authentication. **-a** Attempt automatic login. This sends the user name via the USER variable of the ENVIRON option, if supported by the remote system. The name used is that of the current user as returned by getlogin(2) if it agrees with the current user ID; otherwise it is the name associated with the user ID. **-c** Disable the reading of the user's .telnetrc file. **-d** Set the initial value of the debug flag to TRUE. **-e** escape char Set the initial telnet escape character to escape char. If escape char is omitted, then there will be no escape character. **-f** Forward a copy of the local credentials to the remote system. **-k realm** If Kerberos authentication is being used, request that telnet obtain tickets for the remote host in realm instead of the remote host's realm, as determined by krb_realmofhost(3). **-l user** If the remote system understands the ENVIRON option, the user is sent to the remote system as the value for the variable user. This option implies the -a option. This option may be used with the open command.

Parameter	Description
	-n tracefile Open tracefile for recording trace information. **-r** Specify a user interface similar to rlogin(1). In this mode, the escape character is set to the tilde (~) character, unless modified by the -e option. **-x** Turn on encryption of the data stream. When this option is turned on, telnet will exit with an error if authentication cannot be negotiated or if encryption cannot be turned on.
host	Specifies the name, alias, or Internet address of the remote host.
port	Specifies a port number (address of an application). If the port is not specified, the default telnet port (23) is used

Examples

None

terminal

Use the **terminal** command to set terminal parameters.

Command Mode: EXEC mode

Syntax

terminal length *number-lines*

terminal type *terminal-type*
no terminal type

terminal width *number-chars*

Arguments

Parameter	Description
terminal length *number-lines*	Sets the number of lines for this terminal.
terminal type *terminal-type*	Sets the terminal type. The options are **xterm**, **ansi**, and **vt100**.Use the **no** form of the command to clear the terminal type.
terminal width *number-chars*	Sets the number of maximum number of characters in a line (row) for this terminal.

Defaults

The default terminal length is 24 rows.
The default terminal width is 80 characters.
The default terminal type is **xterm**.

Examples

To set the line width to 120 characters for this terminal:

```
ECV (config) # terminal width 120
```

traceroute

Use the **traceroute** command to trace the route that packets take to a destination.

Command Mode: EXEC mode

Syntax

traceroute [*traceroute-options*] *host* [*packet-length*]

Arguments

Parameter	Description
traceroute-options	Enter one of the following options: **-4** Use IPv4. **-6** Use IPv6. **-A** Perform AS path lookups in routing registries and print results directly after the corresponding addresses. **-f** Set initial time-to-live for first outgoing probe packet. **-F** Set the “don’t fragment” bit. Tells intermediate routers not to fragment the packet when it’s too big for a network hop MTU. **-d** Enable socket level debugging. **-g** Specify a loose source route gateway (8 maximum). **-i** Specify network interface to obtain source IP address for outgoing probe packets. Only useful on a multi-homed host. See -s flag for alternative method. **-I** Use ICMP ECHO instead of UDP datagrams. **-l** Use specified flow_label for IPv6 packets. **-m** Set max time-to-live (number of hops) for outgoing probe packets. Default is 30 hops (same as used for TCP connections). **-n** Print hop addresses numerically rather than symbolically and numerically (saves a nameserver address-to-name lookup for each gateway found on the path). **-N** Number of probe packets sent simultaneously. Sending several probes concurrently can speed up traceroute. Default is 16. When routers and hosts use ICMP rate throttling, specifying too large number can lead to losing some responses. **-p** Set base UDP port number used in probes (default 33434). Traceroute hopes nothing is listening on UDP ports base to base + nhops - 1 at the destination host (so an ICMP PORT_UNREACHABLE message is returned to terminate route tracing). If something is listening on a port in the default range, this option can be used to pick an unused port range.

Parameter	Description
	-q nqueries **-r** Bypass normal routing tables and send directly to a host on an attached network. If the host is not on a directly-attached network, an error is returned. Use this option to ping a local host through an interface with no route through it (such as after the interface was dropped by routed (8C)). **-s** Use the specifiedIP address (usually given as an IP number, not a hostname) as the source address in outbound probe packets. On multi-homed hosts (those with more than one IP address), this option can be used to force the source address to a value other than the IP address of the interface the probe packet is sent on. If the IP address is not one of this machine's interface addresses, an error is returned and nothing is sent. (See the -i flag for another way to do this.) **-t** Set type-of-service in probe packets to specified value (default zero) which is a decimal integer between 0 to 255. This option determines if different types-of-service result in different paths. (If you are not running 4.4bsd, this may not matter since normal network services like telnet and ftp does not control TOS). Not all values of TOS are legal or meaningful - see IP spec for definitions. If TOS value is changed by intermediate routers, (TOS=<value>!) is printed once: value is the decimal value of the changed TOS byte. **-T** Use TCP SYN for tracerouting. **-U** Use UDP datagram (default) for tracerouting. **-V** Print version info and exit. **-w** Set wait time (seconds) for a response to a probe (default 5 sec.). **-z** Set the time (in milliseconds) to pause between probes (default 0). Some systems such as Solaris and routers such as Ciscos rate limit icmp messages. A good value to use with this is 500 (e.g. 1/2 second).
host	Specifies the name, alias, or Internet address of the remote host.
packet-length	Specifies the packet length in bytes.

Defaults

The default packet length is 40 bytes.

Examples

None

traffic-class

Use the **traffic-class** command to assign a name to a specific traffic class.

Use the **no** form of this command to remove a name from a traffic class.

Command Mode: Global Configuration mode

Syntax

traffic-class *1-10* **name** *tc-name*
no traffic-class *traffic-class-id*

Arguments

Parameter	Description
1-10	Specifies the number of the traffic class.
name *tc-name*	Specifies the name to assign to a traffic class.
traffic-class-id	Specifies the number of the traffic class.

Examples

None

username (no)

The **no username** command deletes a specified user account. The **admin** amd **monitor** accounts cannot be deleted.

Command Mode: Global configuration mode

Syntax

no username *username-text*

Parameters

username-text: The ID of the user account deleted by the command.

Examples

The **no username franklin** command deletes the *franklin* user account.

```
ECV (config) # show usernames
admin      Capability: admin     Password set
franklin   Capability: admin     Password set
monitor    Capability: monitor   Account disabled
ECV (config) #
ECV (config) # no username franklin
ECV (config) # show usernames
admin      Capability: admin     Password set
monitor    Capability: monitor   Account disabled
ECV (config) #
```

username capability

The **username capability** command grants a privilege level to a specified user account. The appliance supports two privilege levels:

- **monitor**: Account can read and monitor data. This is equivalent to CLI enable mode access.
- **admin**: Account has all monitor level privileges and can add, modify, and delete commands. This is equivalent to CLI configuration mode access.

The **no username capability** command resets the privilege level of a specified user account to the default value of **monitor**.

Command Mode: Global configuration mode

Syntax

username *username-text* **capability** *LEVEL*
no username *username-text* **capability**

Parameters

username-text The ID of the user account for which the privilege level is changed.

LEVEL Specifies access rights granted the specified account. Options include:

- **monitor**: Monitor level privilege (CLI enable mode access)
- **admin**: Admin level privilege (CLI configuration mode access)

Usage Guidelines

The privilege level of the **admin** and **monitor** accounts cannot be changed.

Examples

These commands grant admin privilege to the user account *carrie* and monitor privilege to the suer account *joe*.

```
ECV (config) # username carrie capability admin
ECV (config) # username joe capability monitor
ECV (config) # show usernames
admin      Capability: admin     Password set
carrie     Capability: admin     Password set
joe        Capability: monitor   Password set
monitor    Capability: monitor   Account disabled
ECV (config) #
```

username disable

The **username disable** command prevents a specified user from logging into the appliance.

The **no username disable** commands enables the specified user to log into the appliance.

Command Mode: Global configuration mode

Syntax

username *USED-ID* **disable**
no username *USED-ID* **disable**

Parameters

USED-ID: Specifies the account that command disables or enables. Options include:

- *username-text*: The user ID of the user account.
- **admin**: The system-provided admin account.
- **monitor**: The system-provided monitor account.

Examples

This command disables the *franklin* user account from logging into the gateway.

```
ECV (config) # username franklin disable
ECV (config) # show usernames
admin      Capability: admin     Password set
franklin   Capability: admin     Account disabled
monitor    Capability: monitor   Account disabled
ECV (config) #
```

This command enables the *franklin* user account.

```
ECV (config) # no username franklin disable
ECV (config) # show usernames
admin      Capability: admin     Password set
franklin   Capability: admin     Password set
monitor    Capability: monitor   Account disabled
ECV (config) #
```

username password

The **username password** command creates a user account and assigns a password to the account. If the specified user account already exists, the command modifies the existing account's password. The command can also assign a password to **admin** or **monitor** accounts.

A command that does not include password text will prompt for a password after the command is entered.

The **username password 0** command assigns a specified clear text string as the password.

The **username password 7** command assigns a specified encrypted text string as the password. After an encrypted password is entered, the original characters are not visible or available in the history or configuration file.

Command Mode: Global configuration mode

Syntax

username *USED-ID* **password**
username *USED-ID* **password 0** *pwd-clear*
username *USED-ID* **password 7** *pwd-encrypt*

Parameters

USED-ID Specifies the user account to which the password is assigned. Options include:

- *username-text*: The user ID of the user account.
- **admin**: Command assigns the password to the admin account.
- **monitor**: Command assigns the password to the monitor account.

pwd-clear: The clear text string assigned as the password to the user account.

pwd-encrypt: The encrypted string assigned as the password to the user account.

Defaults

The default username and the default password are both **admin**.

Usage Guidelines

Passwords require the following:

- at least eight characters.
- at least one lower case letter
- at least one upper case letter.
- at least one digit.

- at least one special character.
- cannot be a word found in the dictionary.

Examples

These commands create the user account *franklin* and assigns the password *Asdfg123#*, then displays a list of user accounts.

```
ECV (config) # username franklin password Asdfg123#
ECV (config) # show usernames
admin      Capability: admin     Password set
franklin   Capability: admin     Password set
monitor    Capability: monitor   Account disabled
ECV (config) #
```

vrrp vmac enable / disable

The **vrrp vmac enable** command enables global usage of the Virtual Router Redundancy Protocol (VRRP) virtual MAC address. A virtual MAC address is a shared MAC address used by a group of routers. The master router in the group assigns the virtual MAC address to all the group routers.

The **vrrp vmac disable** command disables global usage of the VRRP virtual MAC address. By default, global usage of the VRRP virtual MAC address is disabled.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Global Configuration mode

Syntax

vrrp vamc enable
vrrp vamc disable

Examples

This command enables the global use of the VRRP virtual MAC address.

```
ECV-A (config) # vrrp vmac enable
ECV-A (config) #
```

wccp

Use the **wccp** command to configure the Web Cache Communications Protocol (WCCP).

Use the **no** form of the command to remove a WCCP configuration.

Command Mode: Global Configuration mode

Syntax

wccp { **enable** | **disable** }
wccp multicast-ttl *1..15*
wccp *51..255* **admin** { **up** | **down** }
wccp *51..255* **assignment method** { **hash** | **mask** | **either** }
wccp *51..255* **assignment method** { **hash** | **mask** | **either** } **assignment-detail** { **lan-ingress** | **wan-ingress** }

wccp *51..255* **assignment method** { **hash** | **mask** | **either** } **assignment-detail custom hash-srcip** { **enable** | **disable** } **hash-dstip** { **enable** | **disable** } **hash-srcport** { **enable** | **disable** } **hash-dstport** { **enable** | **disable** } **mask-srcip** *32-bit-hex* **mask-dstcip** *32-bit-hex* **mask-srcport** *16-bit-hex* **mask-dstport** *16-bit-hex*

wccp *51..255* **compatibility-mode** { **ios** | **nexus** }
wccp *51..255* **force-l2-return** { **enable** | **disable** }
wccp *51..255* **forwarding-method** { **gre** | **l2** | **either** }
wccp *51..255* **password** *pwd-text*
wccp *51..255* **router** *IP-addr* **protocol** { **tcp** | **udp** } **interface** { **lan0** | **wan0** }

wccp *51..255* **router** *IP-addr* **protocol** { **tcp** | **udp** } **interface** { **lan0** | **wan0** } **priority** *0..255* [**forwarding-method** { **gre** | **l2** | **either** }]

wccp *51..255* **router** *IP-addr* **protocol** { **tcp** | **udp** } **interface** { **lan0** | **wan0** } **priority** *0..255* **forwarding-method** { **gre** | **l2** | **either** } [**weight** *0..65535*]

wccp *51..255* **router** *IP-addr* **protocol** { **tcp** | **udp** } **interface** { **lan0** | **wan0** } **priority** *0..255* **forwarding-method** { **gre** | **l2** | **either** } **weight** *0..65535* [**password** *pwd-text*]

wccp *51..255* **weight** *0..100*
no wccp *51..255*

Arguments

Parameter	Description
wccp *51..255*	Specifies a WCCP service group ID.
admin up	Enables a WCCP service group.
admin down	Disables a WCCP service group.

Parameter	Description
assignment-detail { **custom** | **lan-ingress** | **wan-ingress** }	Specifies the details of the service group assignment method. The options are: **custom** – Assignment by custom values **lan-ingress** – Assignment by hash default. Uses the source address for distribution **wan-ingress** – Assignment by mask default. Uses the destination address for distribution in the router/L3 switch table.
assignment-detail custom	Specifies the details of the service group assignment method. The options are: **hash-srcip** { **enable** | **disable** } – Enable/disable using the hash source IP **hash-dstip** { **enable** | **disable** } – Enable/disable using the hash destination IP **hash-srciport** { **enable** | **disable** } – Enable/disable using the hash source port **hash-dstport** { **enable** | **disable** } – Enable/disable using the hash destination port **mask-srcip** *32-bit-hex* – Specifies the mask source IP as a 32-bit hex value **mask-dstip** *32-bit-hex* – Specifies the mask destination IP as a 32-bit hex value **mask-srcport** *16-bit-hex* – Specifies the mask source port as a 16-bit hex value **mask-dstport** *16-bit-hex* – Specifies the mask destination port as a 16-bit hex value
assignment-method { **hash** | **mask** | **either** }	Modifies the service group assignment method. This relates to how load balancing (of what packets go to which appliance) is set up with the router. The options are: **hash** **mask** **either** The assignment method is either hash or mask. In other words, the appliances will accept packets of either method from the router.
compatibility-mode { **ios** | **nexus** }	If a WCCP group is peering with a router running Nexus OS, then the appliance must adjust its WCCP protocol packets to be compatible. By default, the appliance is IOS-compatible.
disable	Disables the WCCP feature.
enable	Enables the WCCP feature.

Parameter	Description
force-l2-return	Modifies the service group's force L2 return. When WCCP has negotiated L3 forwarding and return methods, Force L2 Return can be used to strip the WCCP GRE header from any packets returned to the router (that is, pass-through traffic). This feature is not applicable if the negotiated forwarding method is L2. **NOTE:** Routing loops may occur if L2 returned packets are forwarded again to the appliance by a WCCP group.
forwarding-method { **gre** | **l2** | **either** }	Modifies the service group's forwarding method. The options are: **GRE** forwarding method **L2** forwarding method **Either** forwarding method
interface { **lan0** | **wan0** }	Modifies service group interface.
multicast-ttl *1..15*	Sets the Time To Live (TTL) value. The range is 1–15.
password *pwd-text*	Sets a password for the WCCP service group.
service-grp *51..255*	Specifies a comma-delimited list of service group IDs.
router *IP-addr*	Use comma separator to specify more than one IP. Use the physical IP for L2 redirection. Use the loopback IP for L3 redirection.
protocol { **tcp** | **udp** }	Configures the WCCP service group protocol for this router IP address.
priority *0..255*	Specifies the WCCP service group's priority. Values range from *0* to *255*.
weight *0..100*	Specifies the WCCP service group weight. *100* is the highest weight. When there is more than one appliance in a group, weight is used to distribute hash or mask assignment buckets on the router in order to load balance flows.

Usage Guidelines

To generate output for the **assignment** and **detail** arguments, enable WCCP after configuration.

Examples

None

web

Use the **web** command to configure the Web-based management User Interface.

Command Mode: Global configuration mode

Syntax

web auto-logout *number-minutes*
no web auto-logout

web { **enable** | **disable** }
web http { **enable** | **disable** }
web https { **enable** | **disable** }

web session max *5…50*
no web session max

Arguments

Parameter	Description
auto-logout *number-minutes*	Sets the length of user inactivity before auto-logout in minutes. The acceptable range is 10 – 60 minutes. Use the **no** form of the command to reset the automatic logout feature for Web sessions to the default setting of 1000 minutes.
{ **enable** | **disable** }	Enables or disables the Web User Interface.
http { **enable** | **disable** }	Enables or disables HTTP access to the Web User Interface.
https { **enable** | **disable** }	Enables or disables HTTPS (secure HTTP) access to the Web User Interface.
session max *5…50*	The maximum number of simultaneous Web sessions (integer). Value range is 5 to 50. The **no** form of the command resets the maximum number of sessions to the default (10).

Defaults

The default auto-logout setting is 15 minutes.
Web HTTP is disabled.
Web HTTPS is enabled.
The default HTTP port is 80.
The default HTTPS port is 443.
The maximum number of simultaneous Web sessions for an appliance is 10.

Usage Guidelines

The acceptable range is between one minute and 1440 minutes (one day).

Examples

To set the maximum length of keyboard inactivity to 7 hours before automatic logout:

```
ECV (config) # web auto-logout 420
```

write

Use the **write** command to save or display the commands in the running configuration.

Command Mode: Privileged EXEC mode

Syntax

write memory
write terminal

Arguments

Parameter	Description
memory	Saves the running configuration to the active configuration file.
terminal	Displays the commands needed to recreate current running configuration.

Defaults

None

Usage Guidelines

When you execute **write terminal** command, the CLI displays commands in the following categories:

Network interface configuration
Routing configuration
Other IP configuration
Logging configuration
AAA configuration
System network configuration
Tunnel creation
Tunnel configuration
Pass-through configuration
Network management configuration

Examples

None

Display Commands

This section describes the display commands. These commands provide status and performance information.

show aaa

Use the **show aaa** command to display AAA authentication settings.

Command Mode: Privileged EXEC mode

Syntax

show aaa

Examples

```
ECV (config) # show aaa
AAA authorization:
    Default User: admin
    Map Order: remote-first
Authentication method(s):
    local

ECV (config) #
```

show access-list

Use the **show access-list** command to display all existing Access Control Lists (ACLs). You can also specify a particular ACL to display.

Command Mode: Privileged EXEC mode

Syntax

show access-list
show access-list *ACL-name*

Arguments

Parameter	Description
access-list	When followed by a carriage return, displays all ACLs.
access-list *ACL-name*	Displays the configuration for the specified ACL.

Examples

The following displays the rules in the ACL, *acl1*:

```
ECV (config) # show access-list acl1
ACL acl1 configuration

ID    Protocol Source              Destination         Action DSCP   Application
----- -------- ------------------- ------------------- ------ ------ -----------
   10 ip       any                 3.3.3.0/24          permit any    any
   20 ip       any                 any                 permit any    snowball
ECV (config) #
```

show alarms

Use the **show alarms** command to display the details for all outstanding alarms.

Command Mode: Privileged EXEC mode

Syntax

show alarms [*alarm-ID* | **outstanding** | **summary**]

Arguments

Parameter	Description
alarms *alarm-ID*	Specifies an alarm ID.
outstanding	Displays the outstanding alarm table.
summary	Shows a summary count of outstanding alarms.

Usage Guidelines

If you use the **show alarms** command without an argument, the CLI displays all outstanding alarms in detail.

Examples

To view a list of all alarm details:

```
ECV (config) # show alarms
Alarm Details List:

Alarm Id:       1
Severity:       MAJ
Type:           EQU
Sequence Id:    5
Name:           equipment_gateway_connect
Description:    Datapath Gateway Connectivity Test Failed
Source:         system
Time:           2007/06/11 17:40:19
Acknowledged:   no
Active:         yes
Clearable:      no
Service Affect: yes
```

```
Alarm Id:       2
Severity:       CRI
Type:           TUN
Sequence Id:    4
Name:           tunnel_down
Description:    Tunnel state is Down
Source:         HQ-to-BranchA
Time:           2007/06/11 17:38:22
Acknowledged:   no
Active:         yes
Clearable:      no
Service Affect: yes

Alarm Id:       3
Severity:       MAJ
Type:           EQU
Sequence Id:    2
Name:           equipment_if_link_down
Description:    Network Interface Link Down
Source:         wan0
Time:           2007/06/11 17:37:09
Acknowledged:   no
Active:         yes
Clearable:      yes
Service Affect: yes
ECV (config) #
```

To view a table of details for all outstanding alarms:

```
ECV (config) # show alarms outstanding
###   Seq  Date                Type  Sev A Source       Description
--- ---- ------------------- ----- --- - ------------ -----------------
  1    5 2007/06/22 18:53:38   EQU MAJ N system       Datapath Gateway Connectivity
     Test Failed
  2    3 2007/06/22 18:51:37   TUN CRI N HQ-to-Branch Tunnel state is Down
  3    2 2007/06/22 18:50:28   EQU MAJ N wan0         Network Interface Link Down
```

show application

Use the **show application** command to display custom (user-defined) applications, with their associated information for protocol, port(s), DSCP, and VLAN.

Command Mode: Privileged EXEC mode

Syntax

show application
show application *app-priority* [**flows** | **stats**]
show application [**brief** | **stats**]
show application name *app-name*

Arguments

Parameter	Description
app-priority	Displays the configuration for the application assigned this priority.
app-priority **flows**	Displays flows that match this application.
app-priority **stats**	Displays statistics for this application.
brief	Displays all user-defined applications.
name	Displays application by name.
stats	Displays statistics for all applications.

Examples

To display all user-defined applications:

```
ECV (config) # show application
Application rule 10 configuration
  Application:       one_more
  Protocol:          tcp
  Src IP Range:
                     any
  Dst IP Range:
                     any
  Src Port Range:
                     any
  Dst Port Range:
                     any
  DSCP:              be
  VLAN:              any.any
```

```
Application rule 20 configuration
  Application:       another_one
  Protocol:          etherip
  Src IP Range:
                     any
  Dst IP Range:
                     172.50.50.0/24
  Src Port Range:
                     any
  Dst Port Range:
                     any
  DSCP:              any
  VLAN:              any.any

ECV (config) #
```

To view the details of the user-defined application, *one-more*, only:

```
ECV (config) # show application name one_more
Application rule 10 configuration
  Application:       one_more
  Protocol:          tcp
  Src IP Range:
                     any
  Dst IP Range:
                     any
  Src Port Range:
                     any
  Dst Port Range:
                     any
  DSCP:              be
  VLAN:              any.any

ECV (config) #
```

show application-builtin

Use the **show application-builtin** command to display all of the appliance's built-in applications, along with their associated ports.

Command Mode: Privileged EXEC mode

Syntax

show application-builtin

Examples

```
ECV (config) # show application-builtin

Application    Ports
------------   ------
          aol  5191-5193
       aol_im  4443,5190
      backweb  370
     cifs_smb  139,445
 cisco_skinny  2000-2001
       citrix  1494,1604
      cuseeme  7648-7652,24032
          dns  53
```

Only a small portion of the returned results are shown above.

show application-group

Use the **show application-group** command to display a list of all application groups, or to display the contents of a specific application group.

Command Mode: Privileged EXEC mode

Syntax

show application-group
show application-group *app-group*
show application-group *app-group* **debug**

Arguments

Parameter	Description
application-group *app-group*	Specifies the name of an existing application group.
debug	Displays debug information for the specific application group named.

Usage Guidelines

To get a list of the available application groups, enter the following command:

```
ECV # show application-group ?
```

Examples

To display all existing application-groups within the appliance:

```
ECV (config) # show application-group
Application Group VoIP : cisco_skinny,h_323,sip
Application Group web : http,https
ECV (config) #
```

To display the applications included in a specific application group:

```
ECV (config) # show application-group VoIP
Application Group VoIP : cisco_skinny,h_323,sip
ECV (config) #
```

To display the debug information for the application group, *VoIP*:

```
ECV (config) # show application-group VoIP debug
Application-Group VoIP Debug Information

ECV (config) # h_323,sip,
ECV (config) #
```

show arp

Use the **show arp** command to display the contents of the ARP cache.

Command Mode: Privileged EXEC mode

Syntax

show arp [**static**]
show arp statistics

Arguments

Parameter	Description
static	Limits the returned results to all statically configured ARP entries, omitting the dynamic entries.
statistics	Displays all ARP cache statistics

Usage Guidelines

If you use the **show arp** command with no arguments, the CLI displays all static and dynamic entries in the ARP cache.

Examples

```
ECV (config) # show arp
10.0.40.33 dev mgmt0 lladdr 00:1b:d4:73:ce:bf REACHABLE
1.1.1.1 dev wan0  INCOMPLETE
```

show banner

Use **show banner** command to display the Message of the Day (MOTD) and Login message banners.

Command Mode: EXEC mode

Syntax

show banner

Examples

```
ECV (config) # show banner
Banners:
    MOTD: Time for a margarita
    Login: How about some coffee?
ECV (config) #
```

show bgp

Use the **show bgp** command to display BGP–related information.

Command Mode: Privileged EXEC mode

Syntax

show bgp neighbors
show bgp summary

Arguments

Parameter	Description
neighbors	Displays BGP neighbors.
summary	Displays summary of BGP global data.

Examples

None

show bootvar

Use **show bootvar** command to display installed system images and boot parameters.

Command Mode: EXEC mode

Syntax

show bootvar

Examples

```
ECV (config) # show bootvar
Installed images:
  Partition 1:
  hidalgo 2.0.0.0_15449 #1-dev 2007-05-30 06:12:39 x86_64 root@bigchief:unknown

  Partition 2:
  hidalgo 2.0.0.0_15619 #1-dev 2007-06-07 20:00:58 x86_64 root@bigchief:unknown

Last boot partition: 2
Next boot partition: 2
ECV (config) #
```

show bridge

Use the **show bridge** command to display bridge information.

Command Mode: Privileged EXEC mode

Syntax

show bridge
show bridge [**brief** | *bridge-info*]
show bridge interface { **lan0** | **wan0** | **lan1** | **wan1** }
show bridge mac-address-table [**address** *ip-addr* | **bridge** *bridge-info* | **interface** *intf*]

Arguments

Parameter	Description
brief	Displays bridge information in brief format.
interface { **lan0** | **wan0** | **lan1** | **wan1** }	Shows bridge port information.
mac-address-table	Shows bridge MAC address table.
address *ip-addr*	Shows bridge MAC address table information for a specific IP address.
bridge *bridge-info*	Shows bridge MAC address table information for a specific bridge (for example, **bvi0**).
interface *intf*	Shows bridge MAC address table information for a specific interface. The interface can be **lan0**, **wan0**, **lan1**, or **wan1**.

Usage Guidelines

MAC table information is not available in router mode.

Examples

To display bridge information for the *lan1* interface:

```
ECV (config) # show bridge mac-address-table interface lan1
MAC Address       Dst Port   Learned Port Type            Age (s)
----------------- ---------- ------------ --------------- -------
00:e0:ed:0c:19:69 lan1       same         local              0.00
```

show cc

The **show cc** command displays the Common Criteria enable mode status on the appliance.

Common Criteria is an international standard for computer security certification. When Common Criteria mode is enabled, the appliance is Common Criteria compliant to a set of guidelines and certifications that ensure the appliance meets the security standard that includes PKI certificates, Online certificate status protocol, and enhanced logging.

Command Mode: Privileged EXEC mode

Syntax

show cc

Usage Guidelines

The **show cc** command is not available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **show system cc**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command displays the Common Criteria status on a appliance where Common Criteria is enabled.

```
ECV # show cc

Common Criteria mode: Enabled
ECV #
```

show cdp

The **show cdp** command displays the CDP enabled status on a specified interfaces.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC mode

Syntax

show cdp

Examples

These commands enable CDP, then display CDP parameters.

```
ECV-A (config) # discoveryd enable
ECV-A (config) # show cdp
Global CDP information:
        Sending CDP packets every 90 seconds
        Sending a holdtime value of 240 seconds
        Sending CDPv1 advertisements is enabled
ECV-A (config) #
```

These commands disable CDP, then display CDP parameters.

```
ECV-A (config) # discovery disable
ECV-A (config) # show cdp
CDP is not enabled
ECV-A (config) #
```

show cdp neighbors

The **show cdp neighbors** command displays a summary of CDP neighbor entries that includes the system ID, local interface, port ID, and configuration data.

The **show cdp neighbors detail** command displays more extensive information about all neighboring devices discovered using CDP, including system name, system description, capabilities, and port details for each connected device.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC mode

Syntax

show cdp neighbors
show cdp neighbors detail

Examples

This command displays CDP neighbors.

```
ECV-1 (config) # show cdp neighbors
Capability Codes: O - Other, R - Repeater, B - Bridge
                  W - Wlan, RR - Router, T - Telephone, D - DOCSIS, S - Station
System ID         Local Intrfce   Holdtme  Capability      Port ID
00:0c:29:e1:25:62   lan0            99       B S           00:0c:29:e1:25:6c
ECV-1 (config) #
```

This command displays extensive information about CDP neighbors.

```
ECV-1 (config) # show cdp neighbors detail
--------------------------------
Chassis ID: 00:0c:29:e1:25:62
Chassis subtype: MAC-address
System Name: abcduvwxyz-tga
System Description: Fedora 32 (Thirty Two) Linux 5.11.22-100.fc32.x86_64 #1 SMP Wed
    May 19 18:58:25 UTC 2021 x86_64
Port ID (outgoing port): 00:0c:29:e1:25:6c
Port subtype : MAC-address
Capabilities: Source-Route-Bridge Switch
Mgmt. Address: fe:80:00:00:00:00
Interface: lan0
Holdtime : 106 sec
ECV-1 (config) #
```

show cdp traffic

The **show cdp traffic** command displays CDP data transmission information for the appliance.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC mode

Syntax

show cdp traffic

Examples

This command displays CDP traffic information.

```
ECV-A (config) # show cdp traffic
CDP counters:
        Total packets output: 60, Input: 0
        Hdr syntax: 0, No memory: 0
ECV-A (config) #
```

show cli

Use the **show cli** command to display Command Line Interface options.

Command Mode: EXEC mode

Syntax

show cli

Examples

```
ECV (config) # show cli
CLI current session settings
  Maximum line size:       8192
  Terminal width:          80 columns
  Terminal length:         24 rows
  Terminal type:           vt102
  Auto-logout:             2 hours 0 minutes 0 seconds
  Paging:                  disabled
  Show hidden config:      yes
  Confirm losing changes:  yes
  Confirm reboot/shutdown: no

CLI defaults for future sessions
  Auto-logout:             2 hours 0 minutes 0 seconds
  Paging:                  enabled
  Show hidden config:      yes
  Confirm losing changes:  yes
  Confirm reboot/shutdown: no
ECV (config) #
```

show clock

Use the **show clock** command to display system time and date.

Command Mode: EXEC mode

Syntax

show clock

Examples

```
ECV (config) # show clock
Time: 21:41:59
Date: 2007/06/16
Time zone: America North United_States Pacific
ECV (config) #
```

show cluster

Use the **show cluster** command to display cluster information.

Command Mode: Privileged EXEC mode

Syntax

show cluster
show cluster spcp

Arguments

Parameter	Description
cluster	Displays the cluster interface and the appliances in the cluster.
cluster spcp	Displays the Silver Peak Communication Protocol statistics.

Examples

None

show configuration

Use the **show configuration** command to display the commands necessary to recreate the active, saved configuration.

Command Mode: Privileged EXEC mode

Syntax

show configuration [**full**]
show configuration files [*filename*]
show configuration [**running** | **running full**]
show configuration [**download status** | **upload status**]

Arguments

Parameter	Description
download status	Displays the status of a configuration file being downloaded to the appliance from a remote host.
files	Displays the names of the active and saved configuration files.
files [*filename*]	Displays the contents of the specified configuration file.
full	Displays commands to recreate the active, saved configuration, and includes commands that set default values.
running	Displays commands to recreate the current running configuration.
running full	Displays commands to recreate the current running configuration, and includes commands that set default values.
upload status	Displays the status of a configuration file being saved from the appliance to a remote host.

Examples

To display the commands to recreate the active, saved configuration – **excluding** those commands that set default values:

```
ECV > show configuration
```

To display the commands to recreate the active, saved configuration – **including** the commands that set default values:

```
ECV > show configuration full
```

To display the commands to recreate the current, running configuration – **excluding** those commands that set default values:

```
ECV > show configuration running
```

To display the commands to recreate the current, running configuration – **including** the commands that set default values:

```
ECV > show configuration running full
```

To display a list of configuration files on the appliance:

```
ECV (config) # show configuration files
initial (active)
newBaseline
initial.bak
backup.1158658595322.287.NE
ECV (config) #
```

To display the contents of the configuration file, *newBaseline*:

```
ECV > show configuration files newBaseline
```

show edgeha hasync

The **show edgeha** command displays high availability sync information.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Privileged EXEC mode

Syntax

show edgeha hasync

Examples

This command displays High Availability high availability sync information

```
ECV-A # show edgeha hasync

EDGEHA Peer IP 0.0.0.0, State UNKNOWN(0)
Last Connected time:NA
Last Unreachable time:NA
Hello Sent: 0, Hello rxed 0
Successfully connected: 0, Unreachable 0
-----------------------------------------------
HAsync intf , fd -1, ifidx 0
HAsync conn state 0, server 0, server_fd -1
Local IP 0.0.0.0, peer IP 0.0.0.0, mask 30
HAread -1
peerid 0, Magic 0x4a4aface, socket-fd 0
Peer version 3, clpr_hello_tx_seq 0, clpr_hello_rx_seq 0
clpr_hello_consec 0, clpr_hello_rx_time 0
clpr_rx_bytes_recv 0, clpr_rx_bytes_left 0
clpr_tx_bytes_sent 0, clpr_tx_bytes_left 0, clpr_tx_q_msgs 0
hasync_hello_built 0, qed 0, read 0
hasync_connect_retries 0, hasync_congp_retries 0
hasync_invalid_sfd 0, hasync_invalid_fd 0, hasync_select_err 0, hasync_select_ret0 0
hasync_rx_err_read 0, hasync_rx_err_eof 0, hasync_rx_err_trunc 0, hasync_rx_err_hdr 0
hasync_rx_err_mem 0, hasync_tx_err 0, hasync_tx_qfull 0, hello_psent 0
hasync_lbid_mismatch 0
spcp_msg_unknown 0, spcp_msg_magic 0, spcp_msg_hello_ver 0, spcp_msg_invalid_len 0
hasync_peerip_update 0, hasync_port_update 0
[HELLO] spcp_msgs_tx 0 spcp_msgs_bytes 0
[EDGEHA] spcp_msgs_tx 0 spcp_msgs_bytes 0
ECV-A #
```

show excess-flow

Use the **show excess-flow** command to display information about flows exceeding the number that the appliance supports.

Command Mode: Privileged EXEC mode

Syntax

show excess-flow
show excess flow log

Arguments

Parameter	Description
log	Displays a log of the excess flows.

Examples

None

show files

Use the **show files** command to display a list of available files and/or display their contents.

Command Mode: EXEC mode (show files system command)
Command Mode: Privileged EXEC mode (all other show files commands)

Syntax

show files debug-dump [*filename*]
show files job upload status
show files stats [*filename*]
show files system
show files tcpdump
show files upload status

Arguments

Parameter	Description
debug-dump [*filename*]	Displays the list of debug-dump files. If you specify a filename, the CLI displays the contents of the file.Debug dump files have the suffix, **.tgz**.
job upload status	Displays job-output file upload status. You would use this when running the **file job upload** command.
stats	Displays a list of statistics reports. Debug dump files have the suffix, **.csv**.
system	Displays information on user-visible file systems.
tcpdump	Displays tcpdump output files.
upload status	Displays files upload status.

Usage Guidelines

If you use the **show files debug-dump** command without the argument, the CLI displays a list of available debug dump files.

Examples

To display a list of debug-dump files:

```
ECV (config) # show files debug-dump
sysdump-RDT-2612-2-20070814-101408.tgz
sysdump-RDT-2612-2-20070820-031350.tgz
```

```
tunbug-ECV-20090109.tar
sysdump-RDT-2612-2-20070822-231449.tgz
sysdump-RDT-2612-2-20070910-094351.tgz
tunbug-ECV-20090102.tar.gz
tunbug-ECV-20090103.tar.gz
tunbug-ECV-20090104.tar.gz
tunbug-ECV-20090105.tar.gz
tunbug-ECV-20090106.tar.gz
tunbug-ECV-20090107.tar.gz
tunbug-ECV-20090108.tar.gz
ECV (config) #
```

show flow-debug

Use the **show flow-debug** command to display the flow-debug summary for the specified flow.

Command Mode: Privileged EXEC mode

Syntax

show flow-debug
show flow-debug description
show flow-debug detail

Arguments

Parameter	Description
description	Displays the names of the statistics, along with their definitions.
detail	Displays the detailed state of the selected flow.

Usage Guidelines

If multiple flows fit the criteria for the configured and enabled **flow-debug** command, then only the first match displays.

Examples

None

show flow-export

Use the **show flow-export** command to display the NetFlow flow export configuration parameters.

Command Mode: Privileged EXEC mode

Syntax

show flow-export

Examples

```
ECV # show flow-export
Flow export v5 disabled:
  no valid collectors are configured.
  active-flow-timeout   : 1 m
  engine-id             : 1
  engine-type           : 1
  interface             : WANTX

  0 flows exported in 0 udp datagrams
ECV #
```

show flow-redirection

Use the **show flow-redirection** command to display the flow redirection state and statistics.

Command Mode: Privileged EXEC mode

Syntax

show flow-redirection

Examples

```
ECV # show flow-redirection

Flow Redirection is disabled

ECV #
```

show hosts

Use the **show hosts** command to display hostname, DNS (Domain Name Server) configuration, and static host mappings.

Command Mode: EXEC mode

Syntax

show hosts

Examples

```
ECV (config) # show hosts
Hostname: ECV
Name server: 172.2.2.2 (configured)
Name server: 10.50.98.4 (configured)
Name server: 134.55.66.77 (configured)
Domain name: silver-peak (configured)
Domain name: rotorrouter (configured)
Domain name: chacha (configured)
Domain name: airborne (configured)
Domain name: roger (configured)
IP 127.0.0.1 maps to hostname localhost
ECV (config) #
```

show iflabels

Use the **show iflabels** command to display the labels available for interfaces.

Command Mode: Privileged EXEC mode

Syntax

show iflabels [**lan-labels** | **wan-labels**]

Arguments

Parameter	Description
lan-labels	Displays LAN interface labels.
wan-labels	Displays WAN interface label.

Examples

To display information about the system images and boot parameters for the appliance, *Tallinn*:

```
ECV (config) # show iflabels
Interface Labels:
LAN interface Labels:
--------------------

Label    Display Name
4        Voice
5        Data

WAN interface Labels:
---------------------
Label    Display Name
1        MPLS
2        Internet
3        LTE
ECV (config) #
```

show igmp interfaces

The **show igmp interfaces** command displays interfaces that are enabled to send IGMP membership requests and the IP address associated with the interface.

Command Mode: Privileged EXEC mode

Syntax

show igmp interfaces

Examples

This command display interfaces where IGMP is enabled.

```
ECV # show igmp interfaces
IfName               Interface-IP Address
wan0                 10.19.156.10
ECV #
```

show image

Use the **show image** command to display information about system images and boot parameters.

Command Mode: EXEC mode

Syntax

show image [**status**]

Arguments

Parameter	Description
status	Displays system image installation status.

Examples

To display information about the system images and boot parameters for the appliance, *ECV*:

```
ECV (config) # show image
Installed images:
  Partition 1:
  hidalgo 2.0.0.0_15449 #1-dev 2007-05-30 06:12:39 x86_64 root@bigchief:unknown

  Partition 2:
  hidalgo 2.0.0.0_15619 #1-dev 2007-06-07 20:00:58 x86_64 root@bigchief:unknown

Last boot partition: 2
Next boot partition: 2
ECV (config) #
```

show interfaces

The **show interfaces** command displays the detailed running state for any or all interfaces.

Command Mode: Privileged EXEC mode

Syntax

show interfaces [**brief** | **configured**]
show interfaces [*intf-name*]
show interfaces *intf-name* [**brief** | **configured**]

Arguments

Parameter	Description
show interfaces	Displays the detailed running state for **all** interfaces.
interfaces brief	Displays the brief running state for **all** interfaces.
interfaces configured	Displays the configuration for **all** interfaces.
interfaces *intf-name*	Shows the detailed running state for the specified interface.

Usage Guidelines

For a list of all the available interfaces only, login in Privileged EXEC Mode or Global Configuration Mode, and enter the following command:

```
ECV # show interfaces ?
```

Examples

To show the detailed running state for **lan0**:

```
ECV (config) # show interfaces lan0
Interface lan0 state
   Admin up:           no
   Link up:            no
   IP address:
   Netmask:
   Speed:              UNKNOWN
   Duplex:             UNKNOWN
   Interface type:     ethernet
   MTU:                1500
   HW address:         00:0C:BD:00:7F:4B
```

```
   RX bytes:           0
   RX packets:         0
   RX mcast packets:   0
   RX discards:        0
   RX errors:          0
   RX overruns:        0
   RX frame:           0

   TX bytes:           0
   TX packets:         0
   TX discards:        0
   TX errors:          0
   TX overruns:        0
   TX carrier:         0
   TX collisions:      0
ECV (config) #
```

show interfaces cdp

The **show interfaces cdp** command displays CDP enabled status on a specified interface. When CDP is not enabled, this command indicates that CDP is disabled regardless of the the CDP configuration on the interface.

The **interface cdp** command enables or disables CDP on a specified interface.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC mode

Syntax

show interfaces *intf-name* **cdp**

Parameters

intf-name: Name of the interface for which data is displayed.

Examples

These commands disable CDP, then displays the CDP status on LAN0.

```
ECV-A (config) # discoveryd disable
ECV-A (config) # show interfaces lan0 cdp
CDP is not enabled
ECV-A (config) #
```

These commands enable CDP on the appliance, disable CDP on the LAN0 interface, then display the CDP status on LAN0.

```
ECV-A (config) # discoveryd enable
ECV-A (config) # interface lan0 cdp disable
ECV-A (config) # show interface lan0 cdp
CDP is disabled on interface lan0
ECV-A (config) #
```

These commands enable CDP on the LAN0, then display CDP status on LAN0.

```
ECV-A (config) # interface lan0 cdp enable
ECV-A (config) # show interface lan0 cdp
CDP is enabled on interface lan0
ECV-A (config) #
```

show interfaces cdp neighbors

The **show interfaces cdp neighbors** command displays a summary of CDP neighbor entries on a specified interface. Information includes the system ID, local interface, port ID, and configuration data.

The **show interfaces cdp neighbors detail** command displays more extensive information about all neighboring devices discovered using CDP on a specified interface. Information includes system name, system description, capabilities, and port details for each connected device.

Cisco Discovery Protocol (CDP) is a layer two protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use CDP. Shared information can include device configuration, capabilities, and identification. CDP is proprietary to Cisco devices.

Command Mode: Privileged EXEC mode

Syntax

show interfaces *intf-name* **cdp neighbors**
show interfaces *intf-name* **cdp neighbors detail**

Examples

This command displays CDP neighbors on LAN0.

```
ECV-1 (config) # show interfaces lan0 cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

Device ID        Local Intrfce  Holdtme  Capability  Platform        Port ID
00:0c:29:e1:25:62      lan0           116      B S             00:0c:29:e1:25:6c
ECV-1 (config)#
```

This command displays extensive information about CDP neighbors on LAN0.

```
ECV-1 (config)# show interfaces lan0 cdp neighbors detail
-------------------------------
Chassis ID: 00:0c:29:e1:25:62
Chassis subtype: MAC-address
System Name: rkalsangra-tga
System Description: Fedora 32 (Thirty Two) Linux 5.11.22-100.fc32.x86_64 #1 SMP Wed
    May 19 18:58:25 UTC 2021 x86_64
Port ID (outgoing port): 00:0c:29:e1:25:6c
Port subtype : MAC-address
Capabilities: Source-Route-Bridge Switch
Mgmt. Address: fe:80:00:00:00:00
Interface: lan0
Holdtime : 95 sec
ECV-1 (config)#
```

show interfaces lldp

The **show interfaces lldp** command displays the LLDP enabled status on a specified interface. When LLDP is not enabled, this command indicates that LLDP is disabled regardless of the LLDP configuration on the interface.

The **interface lldp** command enables or disables LLDP on a specified interface.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

show interfaces *intf-name* **lldp**

Parameters

intf-name: Name of the interface for which data is displayed.

Examples

These commands disable LLDP, then displays the LLDP status on LAN0.

```
ECV-A (config) # discoveryd disable
ECV-A (config) # show interfaces lan0 lldp
LLDP is not enabled
ECV-A (config) #
```

These commands enable LLDP on the appliance, disable LLDP on the LAN0, then display the LLDP status on LAN0.

```
ECV-A (config) # discoveryd enable
ECV-A (config) # interface lan0 lldp disable
ECV-A (config) # show interface lan0 lldp
LLDP is disabled on interface lan0
ECV-A (config) #
```

These commands enable LLDP on the LAN0, then display LLDP status on LAN0.

```
ECV-A (config) # interface lan0 lldp enable
ECV-A (config) # show interface lan0 lldp
LLDP is enabled on interface lan0
ECV-A (config) #
```

show interfaces lldp neighbors

The **show interfaces lldp neighbors** command displays a summary of LLDP neighbor entries on a specified interface. Information includes the system ID, local interface, port ID, and configuration data.

The **show interfaces lldp neighbors detail** command displays more extensive information about all neighboring devices discovered using LLDP on a specified interface. Information includes system name, system description, capabilities, and port details for each connected device.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.devices on the network that also use LLDP. Shared information can include device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

show interfaces *intf-name* **lldp neighbors**
show interfaces *intf-name* **lldp neighbors detail**

Examples

This command displays LLDP neighbors on LAN0.

```
ECV-1 (config) # show interfaces lan0 lldp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

Device ID        Local Intrfce  Holdtme  Capability  Platform        Port ID
00:0c:29:e1:25:62      lan0           116      B S             00:0c:29:e1:25:6c
ECV-1 (config)#
```

This command displays extensive information about LLDP neighbors on LAN0.

```
ECV-1 (config)# show interfaces lan0 lldp neighbors detail
------------------------------
Chassis ID: 00:0c:29:e1:25:62
Chassis subtype: MAC-address
System Name: rkalsangra-tga
System Description: Fedora 32 (Thirty Two) Linux 5.11.22-100.fc32.x86_64 #1 SMP Wed
    May 19 18:58:25 UTC 2021 x86_64
Port ID (outgoing port): 00:0c:29:e1:25:6c
Port subtype : MAC-address
Capabilities: Source-Route-Bridge Switch
Mgmt. Address: fe:80:00:00:00:00
Interface: lan0
Holdtime : 95 sec
ECV-1 (config)#
```

show interfaces pass-through

The **show interfaces pass-through** command displays pass-through traffic information.

Command Mode: Privileged EXEC mode

Syntax

show interfaces pass-through
show interfaces pass-through configured
show interfaces pass-through stats { **flow** [*t-class*] | **qos** [*DSCP-val*] | **traffic-class** }

Arguments

Parameter	Description
configured	Displays the pass-through traffic configuration.
stats flow	Displays pass-through traffic flow metrics for default traffic class.
stats flow *t-class*	Displays pass-through traffic flow metrics for specified traffic class. Value range is 1 to 10
stats qos	Displays default pass-through QoS statistics. Default DSCP value is **be** (best effort).
stats qos *DSCP-val*	Displays pass-through QoS statistics for specified DSCP value.
stats traffic-class	Displays pass-through traffic class statistics.

Usage Guidelines

This command's functionality is the same as **show pass-through** .

Examples

To display the detailed state of pass-through traffic:

```
ECV (config) # show interfaces pass-through
Pass-through traffic state
  Minimum Bw:    32
  Maximum Bw:    10000

  Tx Bytes:      258
  Tx Pkts:       2
ECV (config) #
```

To display the pass-through traffic configuration:

```
ECV (config) # show interfaces pass-through configured
Pass-through traffic configuration
  Minimum Bw:    32
  Maximum Bw:    10000

Traffic Class:
  ID  Priority  Min Bw  Max Bw  Weight
  1     5       500000  1000000   1
  2     10      0       1000000   1
  3     10      0       1000000   1
  4     10      0       1000000   1
  5     10      0       1000000   1
  6     10      0       1000000   1
  7     10      0       1000000   1
  8     10      0       1000000   1
  9     10      0       1000000   1
  10    10      0       1000000   1

Traffic Class Queue Max:
  ID  Packets   Bytes        Flow Pkts  Flow Bytes     Wait (ms)
  1     2000    3000000         2000    3000000         500
  2     500     500000           100    100000          500
  3     500     500000           100    100000          500
  4     500     500000           100    100000          500
  5     500     500000           100    100000          500
  6     500     500000           100    100000          500
  7     500     500000           100    100000          500
  8     500     500000           100    100000          500
  9     500     500000           100    100000          500
  10    500     500000           100    100000          500
ECV (config) #
```

To display statistics for pass-through traffic with a DSCP marking of Best Effort:

```
ECV (config) # show interfaces pass-through stats qos
Tunnel pass-through QOS be Statistics:
 RX bytes:              107077          TX bytes:              68360
 RX packets:            1081            TX packets:            692

 RX processed packets:  0
 RX process bytes:      0

 RX invalid packets:                      0
 RX lost packets:                         0
 RX duplicate packets:                    0
```

```
 RX error correcting packets:             0
 TX error correcting packets:             0

 RX error correcting bytes:               0
 TX error correcting bytes:               0

 RX packets lost before error correction: 0
 RX packets lost after error correction:  0

 RX reconstructed packets in order:       0
 RX reconstructed packets out of order:   0

 RX out of order packets accepted:        0
 RX out of order packets dropped:         0
 RX out of order packets reordered:       0

 RX packets with 1 packet:                0
 Tx packets with 1 packet:                0

 RX packets with 1 fragment:              0
 TX packets with 1 fragment:              0

 RX packets with > 1 packet no fragment:  0
 TX packets with > 1 packet no fragment:  0

 RX packets with > 1 packet and fragment: 0
 TX packets with > 1 packet and fragment: 0
ECV (config) #
```

show interfaces security

Use the **show interfaces security** command to display the security mode for interfaces.

Command Mode: Privileged EXEC mode

Syntax

show interfaces security

Examples

This command displays the security mode on interfaces.

```
ECV # show interfaces security

Interface Security configuration:
---------------------------------
Interface        Security mode
---------        -------------
lan0             Open
lan1             Open
lo               Open
mgmt0            Open
mgmt1            Open
wan0             Open
wan1             Open
ECV #
```

show interfaces tunnel

The **show interfaces tunnel** command displays running status for any and all tunnels.

Command Mode: Privileged EXEC mode

Syntax

show interfaces tunnel [**brief** | **configured** | **peers** | **summary**]
show interfaces tunnel *tunnel-name* [**brief** | **configured** | **fastfail** | **ipsec** [**status**] | **summary**]
show interfaces tunnel *tunnel-name* **stats flow** [*t-class_1-10*]
show interfaces tunnel *tunnel-name* **stats ipsec**
show interfaces tunnel *tunnel-name* **stats latency**
show interfaces tunnel *tunnel-name* **stats qos** [*DSCP-value*]
show interfaces tunnel *tunnel-name* **stats traffic-class**
show interfaces tunnel *tunnel-name* **traceroute**

Arguments

Parameter	Description
brief	Displays brief running state for the tunnel(s).
configured	Displays configuration for the tunnel(s).
fastfail	Displays Fastfail information. When multiple tunnels carry data between two appliances, this option determines the basis for disqualifying a tunnel from carrying data, and how quickly.
peers	Displays table summary information for tunnel peers.
redundancy	Displays redundancy information (regarding WCCP or VRRP) for the tunnel(s).
summary	Displays summary information for the tunnel(s).
tunnel *tunnel-name*	Displays detailed running state for this tunnel.
ipsec status	Displays specified tunnel's IPSec information.
stats flow	Displays flow metrics for the default traffic class in the designated tunnel.
stats flow *t-class	Displays flow metrics for specified traffic class in designated tunnel. Value range is 1 to 10
stats ipsec	Displays IPSec statistics for the designated tunnel.
stats latency	Displays latency metrics for the designated tunnel.

Parameter	Description
stats qos	Displays default QoS statistics for designated tunnel. Default DSCP value is **be** (best effort).
stats qos *DSCP-value*	Displays the QoS statistics for the specified DSCP value in the designated tunnel.
stats traffic-class	Displays traffic class statistics for a designated tunnel.
traceroute	Displays traceroute information for this tunnel.

Defaults

The default DSCP value for QoS is **be** (Best Effort).

Usage Guidelines

If you don't specify a tunnel, then the output includes information for **all** tunnels.
If you do specify a tunnel, then the output is limited to that tunnel.
This command is equivalent to the **show tunnel** command.

Examples

To display summary information for the tunnel, "HQ-to-Branch":

```
ECV (config) # show interfaces tunnel HQ-to-BranchA summary
Tunnel                          Admin Oper         Remote IP        Uptime
------------------------------- ----- ------------ ---------------- --------
HQ-to-BranchA                   up    Down         172.30.5.2       0s
ECV (config) #
```

To display the IPSec status information for the tunnel, "HQ-to-Branch":

```
ECV (config) # __show interfaces tunnel HQ-to-BranchA ipsec status__
Tunnel HQ-to-BranchA ipsec state
  Tunnel Oper:          Down
  IPSec Enabled:        no
  IPSec Oper:           Disabled
  Total IPSec SAs:      in:0 out:0
ECV (config) #
```

To display the traffic class statistics for the tunnel, "gms_dm-vx3000a_dm-vx3000b":

```
ECV (config) # show interfaces tunnel gms_dm-vx3000a_dm-vx3000b stats traffic-class
show request for tunnel gms_dm-vx3000a_dm-vx3000b
Tunnel gms_dm-vx3000a_dm-vx3000b traffic class statistics
```

```
  tc name          LAN RX  LAN RX   WAN TX   WAN TX  QOS Drops Misc.Drops
                 Packets   Bytes    Packets    Kbps    Packets    Packets
   1 default           0       0        0       0          0          0
   2 real-time         0       0        0       0          0          0
   3 interactive       0       0        0       0          0          0
   4 best-effort    2609   66888     2817   51199          0          0
   5                   0       0        0       0          0          0
   6                   0       0        0       0          0          0
   7                   0       0        0       0          0          0
   8                   0       0        0       0          0          0
   9                   0       0        0       0          0          0
   10                  0       0        0       0          0          0
ECV (config) #
```

To display the latency statistics for traffic in the tunnel, “tunnel-2-8504”:

```
ECV (config) # show interfaces tunnel tunnel-2-8504 stats latency
Tunnel tunnel-2-8504 QOS 0 Latency Metrics:
 Minimum Round Trip Time :               1
 Maximum Round Trip Time :               4
 Average Round Trip Time :               2
ECV (config) #
```

show interfaces virtual

Use the **show interfaces virtual** command to display virtual interface information.

Command Mode: Privileged EXEC mode

Syntax

show interfaces virtual

Examples

None

show interfaces vrrp

The **show interfaces vrrp** command displays the detailed running state for VRRP groups on a specified interface.

The **show interfaces vrrp brief** command displays brief running state data for VRRP groups on a specified interface.

The **show interfaces vrrp configured** command displays configured data for VRRP groups on a specified interface.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Privileged EXEC mode

Syntax

show interfaces *intf-name* **vrrp**
show interfaces *intf-name* **vrrp brief**
show interfaces *intf-name* **vrrp configured**
show interfaces *intf-name* **vrrp** *vrrp-id*
show interfaces *intf-name* **vrrp** *vrrp-id* **brief**
show interfaces *intf-name* **vrrp** *vrrp-id* **configured**

Parameters

intf-name: Interface where the VRRP group is located.

vrrp-id: VRRP group identifier (integer). Range is 1 through 255. When parameter is omitted, command displays data for all groups.

Usage Guidelines

This command and the **show vrrp** command displays identical information.

Examples

This command displays VRRP parameters for VRRP groups on the LAN0 interface.

```
ECV-A # show interface lan0 vrrp brief
Intf  Grp   Pre   Adv  Group Addr     Version State  Master Addr    Pri Own
lan0  65    yes   150  10.19.157.65    3     master  10.19.157.10   128 no
lan0  100   yes   2    10.19.157.100   2     master  10.19.157.10   200 no
ECV-A #
```

show ip

Use the **show ip** command to display IP-related information.

Command Mode: EXEC mode (show ip mgmt command)
Command Mode: Privileged EXEC mode (all other listed show ip commands)

Syntax

show ip
show ip datapath route
show ip default-gateway [**static**]
show ip mgmt-ip
show ip route [**static**]

Arguments

Parameter	Description
datapath route	Displays the datapath routing table.
default-gateway	Displays the active default route.
default-gateway static	Displays the configured default route.
mgmt-ip	Displays the management IP address
route	Displays the routing table.
route static	Displays the configured static routes.

Usage Guidelines

If you're using DHCP for **mgmt0**, then it displays:

Management IP address: <none>

Examples

To display the active default datapath route:

```
ECV (config) # show ip default-gateway
Active default gateway: 10.0.52.5
ECV (config) #
```

show ip multicast static routes

The **show ip multicast static routes** command displays configured multicast static routes on the appliance.

Command Mode: Privileged EXEC mode

Syntax

show ip multicast static routes

Examples

This command displays the active default datapath route.

```
ECV (config) # show ip multicast static routes

GroupIP       SourceIP    IncomingIntf   IncomingPeer   OutgoingIntfs   OutgoingPeers
3.3.3.3       4.4.4.4     44             5.5.5.5
12.1.1.1      15.1.1.1

ECV (config) #
```

show ip-tracking

Use the **show ip-tracking** command to display IP tracking (IPSLA) information.

Command Mode: Privileged EXEC mode

Syntax

show ip-tracking ipsla-debug
show ip-tracking ipsla-if-debug
show ip-tracking ipsla-ip-debug
show ip-tracking manager
show ip-tracking summary

Arguments

Parameter	Description
ipsla-debug	Displays IPSLA (Internet Protocol Service Level Agreement) debug information.
ipsla-if-debug	Displays IPSLA interface debug information.
ipsla-ip-debug	Displays IPSLA IP address debug information.
manager	Displays the IP Tracking manager table.
summary	Displays a summary of the IP Tracking component.

Examples

To view the IP Tracking manager table:

```
ECV (config) # show ip-tracking manager
IP Tracking Mgr Table: 0 active Manager entries
```

To view a summary of the IP Tracking component:

```
ECV (config) # show ip-tracking summary
Global IP Tracking information:
Process Status:           Active
Manager Count:            0
Managers Active:          0
Monitor Operation Count:  0
Action Count:             0
Monitor Requests Sent:    0
```

show licenses

Use the **show licenses** command to display the installed licenses and licensed features.

Command Mode: EXEC mode

Syntax

show licenses

Examples

```
ECV (config) # show licenses
No licenses have been configured.
ECV (config) #
```

show lldp

The **show lldp** command displays the LLDP enabled status on a specified interfaces.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

show lldp

Examples

These commands enable LLDP, then display LLDP parameters.

```
ECV-A (config) # discoveryd enable
ECV-A (config) # show lldp
Global LLDP information:
        Sending LLDP packets every 90 seconds
        Sending a holdtime value of 240 seconds
        Sending LLDPv1 advertisements is enabled
ECV-A (config) #
```

These commands disable LLDP, then display LLDP parameters.

```
ECV-A (config) # discovery disable
ECV-A (config) # show lldp
LLDP is not enabled
ECV-A (config) #
```

show lldp neighbors

The **show lldp neighbors** command displays a summary of LLDP neighbor entries that includes the system ID, local interface, port ID, and configuration data.

The **show lldp neighbors detail** command displays more extensive information about all neighboring devices discovered using LLDP, including system name, system description, capabilities, and port details for each connected device.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

show lldp neighbors
show lldp neighbors detail

Examples

This command displays LLDP neighbors.

```
ECV-1 (config) # show lldp neighbors
Capability Codes: O - Other, R - Repeater, B - Bridge
                  W - Wlan, RR - Router, T - Telephone, D - DOCSIS, S - Station
System ID         Local Intrfce   Holdtme  Capability      Port ID
00:0c:29:e1:25:62   lan0            99       B S           00:0c:29:e1:25:6c
ECV-1 (config) #
```

This command displays extensive information about LLDP neighbors.

```
ECV-1 (config) # show lldp neighbors detail
--------------------------------
Chassis ID: 00:0c:29:e1:25:62
Chassis subtype: MAC-address
System Name: abcduvwxyz-tga
System Description: Fedora 32 (Thirty Two) Linux 5.11.22-100.fc32.x86_64 #1 SMP Wed
    May 19 18:58:25 UTC 2021 x86_64
Port ID (outgoing port): 00:0c:29:e1:25:6c
Port subtype : MAC-address
Capabilities: Source-Route-Bridge Switch
Mgmt. Address: fe:80:00:00:00:00
Interface: lan0
Holdtime : 106 sec
ECV-1 (config) #
```

show lldp traffic

The **show lldp traffic** command displays LLDP data transmission information for the appliance.

Link Layer Discovery Protocol (LLDP) is a layer two open standard protocol that allows Ethernet network devices to advertise details about themselves to directly connected devices on the network that also use LLDP. Shared information can includes device configuration, capabilities, and identification.

Command Mode: Privileged EXEC mode

Syntax

show lldp traffic

Examples

This command displays LLDP traffic information.

```
ECV-A (config) # show lldp traffic
LLDP counters:
        Total packets output: 60, Input: 0
        Hdr syntax: 0, No memory: 0
ECV-A (config) #
```

show log

Use the **show log** command to view event log contents.

Command Mode: Privileged EXEC mode

Syntax

show log

show log alert
show log alert continuous
show log alert files [*file-number*]
show log alert files *file-number* [**matching** *reg-exp*]
show log alert matching *reg-exp*

show log continuous [**matching** *reg-exp*]
show log continuous not matching *reg-exp*

show log files [*file-number*]
show log files *file-number* **matching** *reg-exp*
show log files *file-number* **not matching** *reg-exp*

show log matching *reg-exp*
show log not matching *reg-exp*

Arguments

Parameter	Description
alert	Displays alert event logs.
continuous	Displays new log messages as they arrive.
files	Displays a listing of all available **archived** log files.
files *file-number*	Specifies which **archived** log file number to display.
matching *reg-exp*	Displays event logs that match a given regular expression. If the expression includes spaces, enclose the expression with quotation marks.
not matching *reg-exp*	Displays event logs that **do not** match a given regular expression. If the expression includes spaces, enclose the expression with quotation marks.

Defaults

- Without arguments, the command, **show log**, displays the **current event log**.

- The command, **show log alert**, displays the **current alerts log** .
- The appliance keeps up to 30 archived alert log files. The older the file, the higher the file number. The newest file has no number; the most recent archived file is numbered “1”.

Usage Guidelines

To see what archived logs are available, use one of the following:

```
ECV (config) # show log files ?
```

```
ECV (config) # show log alert files ?
```

Examples

To show a list of all available alert log files:

```
ECV (config) # show log files
1
2
ECV (config) #
```

To show all archived files that match the expression, “ping”, in any string:

```
ECV (config) # show log matching ping

r dumping
Jun 17 17:24:45 localhost rename_ifs: Mapping MAC: 00:0C:BD:00:7F:4A to interface name
    : wan0
Jun 17 17:24:45 localhost rename_ifs: Mapping MAC: 00:0C:BD:00:7F:4B to interface name
    : lan0
Jun 17 17:24:45 localhost rename_ifs: Mapping MAC: 00:E0:81:2F:85:98 to interface name
    : mgmt0
Jun 17 17:25:09 Tallinn sysd[798]: TID 1084225888: [sysd.NOTICE]: WDOG: Gateway
    datapath ping test disabled when in BYPASS.
Jun 17 17:29:09 Tallinn sysd[798]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAILED: 2
Jun 17 17:30:09 Tallinn sysd[798]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAILED: 2
Jun 17 17:33:09 Tallinn sysd[798]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAILED: 2
Jun 17 17:34:09 Tallinn sysd[798]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAILED: 2
Jun 17 17:34:24 Tallinn cli[2411]: [cli.NOTICE]: user admin: Executing command:
show log matching ping
/tmp/messages_filtered-rvzGgG lines 39947-39958/39958 (END)
```

To view new alert log messages as they arrive:

```
ECV (config) # show log continuous
```

To view the #3 archived alert log file:

```
ECV (config) # show log alert files 3
```

show log-files

Use the **show log-files** command to display the a specific log listing.

Command Mode: Privileged EXEC mode

Syntax

show log-files *file-number* [**list matching** *reg-exp*]

Arguments

Parameter	Description
log-files *file-number*	Specifies a file number for which to display a log listing.
list matching *reg-exp*	Lists selected log lines that match the given expression.

Examples

To see what log files are available:

```
ECV (config) # show log-files ?
<file number>
1
2
ECV (config) #
```

To list log lines in the archived log file, “1”, that match the expression “system”:

```
ECV (config) # show log-files 1 list matching system
Dec 14 19:38:53 Tallinn mgmtd[850]: [mgmtd.ALERT]: ALARM RAISE: WARN,SW,9,
    system_shutdown,System shutdown has been initiated,System,2006/12/14  19:38:53,1,
    no,no,yes,yes.
Dec 14 19:39:00 Tallinn shutdown: shutting down for system reboot
Dec 14 19:41:49 localhost kernel: SCSI subsystem initialized
Dec 14 19:41:49 localhost kernel: VFS: Mounted root (ext3 filesystem) readonly.
Dec 14 19:41:49 localhost mdinit: Running system image: hidalgo 2.0.0.0_13180  #1-dev
    2006-12-14 07:0
5:03 x86_64 root@bigchief:unknown
Dec 14 19:41:43 localhost rc.sysinit: Checking root filesystem succeeded
Dec 14 19:41:43 localhost rc.sysinit: Remounting root filesystem in read-write mode:
    succeeded
Dec 14 19:41:43 localhost fsck: Checking all file systems.
Dec 14 19:41:43 localhost rc.sysinit: Checking filesystems succeeded
Dec 14 19:41:43 localhost rc.sysinit: Mounting local filesystems:  succeeded
```

```
Dec 14 19:41:59 Tallinn mdinit: Shutting down system logger:
Dec 14 19:42:13 Tallinn mgmtd[849]: [mgmtd.ALERT]: ALARM RAISE: CRI,EQU,2,
    equipment_system_bypass, System BYPASS mode,System,2006/12/14 19:42:13,1,no,yes,no
    ,no. NIC fail-to-wire  mode - BYPASS
Dec 14 19:43:23 Tallinn mgmtd[849]: [mgmtd.ALERT]: ALARM CLEAR: CRI,EQU,4,
    equipment_system_bypass, System BYPASS mode,System,2006/12/14 19:42:13,2,no,yes,no
    ,no. NIC fail-to-wire  mode - NORMAL
Dec 14 19:44:23 Tallinn mgmtd[849]: [mgmtd.ALERT]: ALARM RAISE: MAJ,EQU,5,
    equipment_gateway_connect,Datapath Gateway Connectivity Test  Failed,system
    ,2006/12/14 19:44:23,1,no,yes,no,yes.  Datapath Gateway Connectivity Test Failed
Dec 26 15:45:21 Tallinn mgmtd[849]: [mgmtd.ALERT]: ALARM RAISE: WARN,SW,6,
    system_shutdown,System shutdown has been initiated,System,2006/12/26  15:45:21,1,
    no,no,yes,yes.
Dec 26 15:45:26 Tallinn shutdown: shutting down for system reboot
lines 1-16
```

show log-list matching

Use the **show log-list matching** command to list event log lines that match the specified expression.

Command Mode: Privileged EXEC mode

Syntax

show log-list matching *reg-exp*

Arguments

Parameter	Description
matching *reg-exp*	Lists selected log lines that match the given expression.

Examples

None

show logging

Use the **show logging** command to display the logging configuration.

Command Mode: EXEC mode

Syntax

show logging
show logging facilities
show logging files upload status
show logging tech-support

Arguments

Parameter	Description
facilities	Displays log facilities configuration.
files upload status	Displays progress of a logging file being saved to a remote host.
tech-support	Displays entries that the appliance creates for tech support.

Examples

To view the logging configuration:

```
ECV (config) # show logging
Local logging level: notice
Default remote logging level: notice
No remote syslog servers configured.
Allow receiving of messages from remote hosts: no
Number of archived log files to keep: 30
Log rotation size threshold: 50 megabytes
Log format: standard
Levels at which messages are logged:
  CLI commands: notice
ECV (config) #
```

To monitor the progress of a logging file as it is copied from the appliance to a remote host.

```
ECV (config) # show logging files upload status
File Upload Status
  Name:                -not set-
  Status:              Ready
  Last Upload Status:  The system is ready for upload
  Start time:          -not set-
```

```
  End time:            -not set-
  Total upload size:   0
  Transferred size:    0
  Transfer rate:       0 bps
  Percent complete:    0%
ECV (config) #
```

To view the information saved for tech support:

```
ECV (config) # show logging tech-support
Apr 22 01:15:15 Tallinn sysd[781]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAIL
ED: 2
Apr 22 01:15:20 Tallinn tunneld[779]: TID 182912294944: [tunneld.ERR]:
    cipsec_recovery_statemachine:
Took IPSec recovery action - tunnel:Tallinn_to_Helsinki still down..
Apr 22 01:16:10 Tallinn tunneld[779]: TID 182912294944: [tunneld.ERR]:
    cipsec_recovery_statemachine:
Took IPSec recovery action - tunnel:Tallinn_to_Helsinki still down..
Apr 22 01:16:15 Tallinn sysd[781]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAIL
ED: 2
Apr 22 01:17:00 Tallinn tunneld[779]: TID 182912294944: [tunneld.ERR]:
    cipsec_recovery_statemachine:
Took IPSec recovery action - tunnel:Tallinn_to_Helsinki still down..
Apr 22 01:17:15 Tallinn sysd[781]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAIL
ED: 2
Apr 22 01:17:50 Tallinn tunneld[779]: TID 182912294944: [tunneld.ERR]:
    cipsec_recovery_statemachine:
Took IPSec recovery action - tunnel:Tallinn_to_Helsinki still down..
Apr 22 01:18:15 Tallinn sysd[781]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAIL
ED: 2
Apr 22 01:18:40 Tallinn tunneld[779]: TID 182912294944: [tunneld.ERR]:
    cipsec_recovery_statemachine:
Took IPSec recovery action - tunnel:Tallinn_to_Helsinki still down..
Apr 22 01:19:15 Tallinn sysd[781]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAIL
ED: 2
Apr 22 01:19:30 Tallinn tunneld[779]: TID 182912294944: [tunneld.ERR]:
    cipsec_recovery_statemachine:
Took IPSec recovery action - tunnel:Tallinn_to_Helsinki still down..
Apr 22 01:20:15 Tallinn sysd[781]: TID 1084225888: [sysd.ERR]: WDOG: Gateway datapath
    ping test FAIL
lines 1-12
```

To view the log facilities configuration:

```
ECV (config) # show logging facilities
Log Facilities Configuration:
    audit:     local0
    system:    local1
    flow:      local2
ECV (config) #
```

show memory

Use the **show memory** command to display system memory usage.

Command Mode: EXEC mode

Syntax

show memory

Examples

```
ECV (config) # show memory
          Total       Used       Free
Physical  4061 MB   3481 MB    579 MB
Swap         0 MB      0 MB      0 MB
ECV (config) #
```

show nat-map

Use the **show nat-map** command to display a list of all the existing NAT maps. The CLI also indicates which NAT map is currently active.

Command Mode: Privileged EXEC mode

Syntax

show nat-map
show nat-map *map-name*
show nat-map *map-name priority*
show nat-map *map-name priority* **stats**

Arguments

Parameter	Description
nat-map	Displays all existing NAT maps.
nat-map *map-name*	Displays each priority (entry) for specified NAT map, along with their MATCH criteria and SET actions.
nat-map *map-name priority*	Displays the priority value for a specified NAT map.
stats	Displays statistics for the specified map. If the priority number is included in the command, then the match statistics are limited to that map entry.

Usage Guidelines

The default entry in any map is always priority 65535. The NAT map specifics are:

```
65535 match
          Protocol:             ip
          IP version:           any
          Source:               any
          Destination:          any
          Application:          any
          DSCP:                 any
          VLAN:                 any.any
        set
          NAT Type:             no-nat
          NAT direction:        None
          NAT IP:               auto
          Fallback:             disabled
```

Examples

None

show nat statistics

Use the **show nat statistics** command to display NAT-related statistics.

Command Mode: Privileged EXEC mode

Syntax

show nat statistics

Examples

```
ECV (config) # show nat statistics
NAT Statistics

        Total NAT Tcp flow    :0
        Total NAT Udp flow    :0
        Total NAT Icmp flow   :0
        NAT mid flow no alloc :0

ECV (config) #
```

show ntp

Use the **show ntp** command to display NTP settings.

Command Mode: EXEC mode

Syntax

show ntp

Examples

```
ECV (config) # show ntp
NTP enabled: no
No NTP peers configured.
No NTP servers configured.
ECV (config) #
```

show opt-map

Use the **show opt-map** command to display a list of all the existing optimization maps. The CLI also indicates which optimization map is currently active.

Command Mode: Privileged EXEC mode

Syntax

show opt-map
show opt-map *map-name*
show opt-map *map-name priority*
show opt-map *map-name priority* **advanced-tcp**
show opt-map *map-name priority* **flows**
show opt-map *map-name prioritye* **stats**

Arguments

Parameter	Description
opt-map	Displays all existing optimization maps.
opt-map *map-name*	Displays each priority (entry) for the optimization map, along with their MATCH criteria and SET actions.
opt-map *map-name priority*	Displays the priority value specified for the optimization map.
advanced-tcp	Displays advanced TCP options.
flows	Displays the flows that match the priority (entry) number specified.
stats	Displays statistics for the specified map. When the command includes the priority number, match statistics are limited to that map entry.

Usage Guidelines

The default entries in any new opt map are as follows:

```
ECV (config) # show opt-map map1
Opt map map1 configuration (ACTIVE)
  10000 match
          Protocol:            tcp
          Source:              any
          Destination:         any
          Source Port:         any
```

```
          Destination Port:    139
          DSCP:                any
          VLAN:                any.any
        set
          Network Memory:      balanced
          Payload Comp:        enable
          Proxy Type:          cifs

  10010 match
          Protocol:            tcp
          Source:              any
          Destination:         any
          Source Port:         any
          Destination Port:    445
          DSCP:                any
          VLAN:                any.any
        set
          Network Memory:      balanced
          Payload Comp:        enable
          Proxy Type:          cifs

  10020 match
          Protocol:            tcp
          Source:              any
          Destination:         any
          Source Port:         any
          Destination Port:    443
          DSCP:                any
          VLAN:                any.any
        set
          Network Memory:      balanced
          Payload Comp:        enable
          Proxy Type:          ssl

  65535 match
          Protocol:            ip
          Source:              any
          Destination:         any
          Application:         any
          DSCP:                any
          VLAN:                any.any
        set
          Network Memory:      balanced
          Payload Comp:        enable
          Proxy Type:          tcp-only

ECV (config) #
```

You can view an appliance's list of optimization maps—and determine which map is active—with the command, **show opt-map**:

```
ECV> # show opt-map
maryann
ginger            [ACTIVE]
```

Examples

To view a list of all the priorities included in the optimization map, “map1”, for this appliance:

```
ECV (config) # show opt-map map1 ?
<cr>                  Display this optimization map
<1..65535>
10
20
75
85
110
120
130
65535
ECV (config) #
```

To find out how many flows match priority “100” in the optimization map, “ginger” :

```
ECV (config) # show opt-map ginger 100 flows
Flows matching Optimization Map ginger prio:100:
6 (L->W) sip:10.2.1.128 dip:10.16.1.200 ports:0/0

Total flows:1
```

To view the specifics of priority 10 in “map1” of the appliance, Tallinn:

```
ECV (config) # show opt-map map1 10
  10    match
Protocol:            ip
          Source:              10.10.10.0/24
          Destination:         10.10.20.0/24
          Application:         any
          DSCP:                any
          VLAN:                any.any
        set
          Network Memory:      balanced
          Payload Comp:        enable
          Proxy Type:          tcp-only


ECV (config) #
```

To display statistics for the optimization map, “O-2-3500-2”, in the appliance,”eh-3500-1”:

```
ECV (config) # show opt-map O-2-3500-2 stats
Optimization Map O-2-3500-2 Lookup Statistics:

Priority 100:
 Match Succeeded:       38918
  Permits:      38918   Denies: 0
 Match Failed:  0
  Source IP Address:    0       Destination IP Address: 0
  Source Port:          0       Destination Port:       0
  Application:          0       DSCP Markings:  0       Protocol:       0
```

```
Priority 65535:
 Match Succeeded:       0
  Permits:      0       Denies: 0
 Match Failed:  0
  Source IP Address:    0       Destination IP Address: 0
  Source Port:          0       Destination Port:       0
  Application:          0       DSCP Markings:  0       Protocol:       0
ECV (config) #
```

show overlay

Use the **show overlay** command to display detailed information any or all overlays.

Command Mode: Privileged EXEC mode

Syntax

show overlay
show overlay *overlay-name*

Arguments

Parameter	Description
overlay-name	Displays the name of a specific overlay.

Examples

To display all existing overlays:

```
ECV (config) # show overlay

Overlay Name(ID):        Voice(1)
        Brownout Loss:           1.000000
        Brownout latency:        75
        Brownout Jitter:         50
        Bonding policy:          high-availability
        Tunnel Usage Policy Bucket: 1
                Condition:      use-sla
                Links:
                                MPLS-MPLS(1-1)
                                Internet-Internet(2-2)
                                Kate-Kate(6-6)

        Tunnel Usage Policy Bucket: 2
                Condition:      use-active
                Links:
                                MPLS-MPLS(1-1)
                                Internet-Internet(2-2)
                                Kate-Kate(6-6)
ECV (config) #
```

show overlay-common

Use the **show overlay-common** command to display common configuration for overlays.

Command Mode: Privileged EXEC mode

Syntax

show overlay-common internal-subnets

Arguments

Parameter	Description
internal-subnets	Displays internal subnets list.

Examples

```
ECV (config) # show overlay-common internal-subnets
Internal subnets:
-----------------
10.0.0.0/8
172.16.0.0/12
192.168.0.0/16
ECV (config) #
```

show pass-through

The **show pass-through** command displays information about pass-through traffic.

This command's functionality is the same as *show interfaces pass-through*

Command Mode: Privileged EXEC mode

Syntax

show pass-through
show pass-through configured
show pass-through stats { **flow** [*traffic-class_1-10*] | **qos** [*DSCP-value*] | **traffic-class** }

Arguments

Parameter	Description
configured	Displays pass-through traffic configuration.
stats flow	Displays pass-through traffic flow metrics.
stats qos	Displays pass-through QoS stats for the default DSCP value (**be**).
stats qos *DSCP-value*	Displays pass-through QoS stats for a specified DSCP value.
stats traffic-class	Displays pass-through traffic class statistics.

Defaults

The default traffic class is 1.

Usage Guidelines

Use the command without arguments to display a detailed state of pass-through traffic.

Examples

To display the pass-through QoS statistics:

```
ECV (config) # show pass-through stats qos
Tunnel pass-through QOS be Statistics:
 RX bytes:              0               TX bytes:              258
 RX packets:            0               TX packets:            2
```

```
 RX processed packets:  0
 RX process bytes:      0

 RX invalid packets:                     0
 RX lost packets:                        0
 RX duplicate packets:                   0

 RX error correcting packets:            0
 TX error correcting packets:            0

 RX error correcting bytes:              0
 TX error correcting bytes:              0

 RX packets lost before error correction: 0
 RX packets lost after error correction:  0

 RX reconstructed packets in order:      0
 RX reconstructed packets out of order:  0

 RX out of order packets accepted:       0
 RX out of order packets dropped:        0
 RX out of order packets reordered:      0

 RX packets with 1 packet:               0
 Tx packets with 1 packet:               0

 RX packets with 1 fragment:             0
 TX packets with 1 fragment:             0

 RX packets with > 1 packet no fragment:  0
 TX packets with > 1 packet no fragment:  0

 RX packets with > 1 packet and fragment: 0
 TX packets with > 1 packet and fragment: 0
ECV (config) #
```

show pim debug

The **show pim debug cmd nhops** command displays data about Nexthop IP addresses that send and receive PIM packets.

The **show pim debug cmd stats** command displays statistical information concerning PIM packets processing. This information includes Commands submitted, Failed command submission, Successful commands, and Failed commands.

The **show pim debug cmdQ** command displays the number of submitted PIM commands.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim debug cmd nhops
show pim debug cmd stats
show pim debug cmdQ

Examples

This command displays Nexthop IP address information.

```
ECV # show pim debug cmd nhops
total nexthops 2
1. Next-hop ip 192.172.11.11 nhop 10.19.156.1 state NHOP_IN_RTM route_intf wan0
    ifindex 3 RP ttl 6 marked for del NO index 0 version 2
2. Next-hop ip 3.3.3.3 nhop 10.19.156.1 state NHOP_IN_RTM route_intf wan0 ifindex 3 RP
     ttl 13 marked for del NO index 3 version 1
ECV #
```

show pim interfaces

The **show pim interfaces** displays interfaces where PIM is enabled and the PIM settings on each interface. Parameter settings displayed by the command includes the IP address of the interface, DR-Priority, Generation ID, IP address of the Designated Router, Hello Interval, and Join/Prune Interval.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim interfaces

Examples

This command displays PIM parameter information about all PIM-enabled interfaces on the appliance.

```
ECV (config) # show pim interfaces
IfName   Interface-IP Address   DR-Priority  Generation ID   Designated-Router-IP
    Hello Interval   Join/Prune Interval
wan0     10.19.156.10           1            3534349093      10.19.156.10          200
                 30
pim0     169.254.124.1          1            2520518556      169.254.124.2         30
                 30
pim1     169.254.125.1          1            423562176       169.254.125.2         30
                 30
pim2     169.254.126.1          1            296423632       169.254.126.2         30
                 30
ECV (config) #
```

show pim interfaces stats

The **show pim interfaces stats** command displays PIM packets sent from all PIM-enabled interfaces on the appliance.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim interfaces stats

Examples

This command display the PIM packets sent from PIM-enabled interfaces.

```
ECV (config) # show pim interfaces stats
interface wan0
num_sent_hello 5596, num_sent_join_prune 0, num_sent_assert 0, num_sent_bsm 0,
    num_err_hello 0, num_recv_unknown_nbr 0, num_unknown_hello_opt 0, num_filtered_out
     0, num_sent_graft 0, num_sent_graft_ack 0, num_sent_state_refresh 0,
    num_sent_df_election 0, num_recv_hello 0, num_recv_join_prune 0, num_recv_assert
    0, num_recv_unknown_type 0, num_recv_bad_checksum 0,
interface lan0
num_sent_hello 838, num_sent_join_prune 0, num_sent_assert 0, num_sent_bsm 0,
    num_err_hello 0, num_recv_unknown_nbr 0, num_unknown_hello_opt 0, num_filtered_out
     0, num_sent_graft 0, num_sent_graft_ack 0, num_sent_state_refresh 0,
    num_sent_df_election 0, num_recv_hello 0, num_recv_join_prune 0, num_recv_assert
    0, num_recv_unknown_type 0, num_recv_bad_checksum 0,
interface pim0
num_sent_hello 25589, num_sent_join_prune 0, num_sent_assert 0, num_sent_bsm 0,
    num_err_hello 0, num_recv_unknown_nbr 0, num_unknown_hello_opt 0, num_filtered_out
     0, num_sent_graft 0, num_sent_graft_ack 0, num_sent_state_refresh 0,
    num_sent_df_election 0, num_recv_hello 25589, num_recv_join_prune 0,
    num_recv_assert 0, num_recv_unknown_type 0, num_recv_bad_checksum 0,
interface pim1
num_sent_hello 25589, num_sent_join_prune 0, num_sent_assert 0, num_sent_bsm 0,
    num_err_hello 0, num_recv_unknown_nbr 0, num_unknown_hello_opt 0, num_filtered_out
     0, num_sent_graft 0, num_sent_graft_ack 0, num_sent_state_refresh 0,
    num_sent_df_election 0, num_recv_hello 25589, num_recv_join_prune 0,
    num_recv_assert 0, num_recv_unknown_type 0, num_recv_bad_checksum 0,
interface pim2
num_sent_hello 25589, num_sent_join_prune 0, num_sent_assert 0, num_sent_bsm 0,
    num_err_hello 0, num_recv_unknown_nbr 0, num_unknown_hello_opt 0, num_filtered_out
     0, num_sent_graft 0, num_sent_graft_ack 0, num_sent_state_refresh 0,
    num_sent_df_election 0, num_recv_hello 25589, num_recv_join_prune 0,
    num_recv_assert 0, num_recv_unknown_type 0, num_recv_bad_checksum 0,

ECV (config) #
```

show pim internal stats

The **show pim internal stats** command displays debug and diagnostic PIM protocol data. This command should only be run under the supervision of HPE Aruba Networking Support or Engineering.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim internal stats

Examples

This command displays debug and diagnostic PIM data.

```
ECV # show pim internal stats
num_sent_crp_advert 0, num_sent_register 0, num_sent_register_stop 0,
    num_recv_crp_advert 0, num_recv_register 0, num_recv_register_stop 0,
    num_err_crp_advert 0, num_err_register 0, num_err_register_stop 0,
    num_recv_ignored_type 0, num_recv_unknown_ver 0, num_recv_bad_checksum 0,
    num_recv_bad_length 0, num_crp_advert_filtered 0

ECV #
```

show pim mroute

The **show pim mroute** command displays the multicast routing table.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim mroute

Examples

This command displays the multicast routing table.

```
ECV # show pim mroute
(*,G)/(S,G)                              Incoming Intf        Outgoing interfaces
ECV #
```

show pim neighbors

The **show pim neighbors** command displays information about PIM neighbors discovered by Hello messages.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim neighbors

Examples

This command displays information about PIM neighbors discovered by Hello messages.

```
ECV # show pim neighbors
Neighbour-IP Address      IfName       Neighbour-DR-Priority     Neighbour-Generation-
    ID
169.254.124.2             pim0         1                         2520518556
169.254.125.2             pim1         1                         423562176
169.254.126.2             pim2         1                         296423632
ECV #
```

show pim neighbors stats

The **show pim neighbors stats** command displays PIM packets sent from all neighbors discovered by PIM Hello messages.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim neighbors stats

Examples

This command displays PIM packets sent from all neighbors discovered by PIM Hello messages.

```
ECV # show pim neighbors stats
Nbr-ip 0x0.055e16619108p-1022ab
 num_recv_hello 30613216, num_recv_join_prune 31492, num_recv_assert 0, num_recv_bsm
     0, num_err_join_prune 0, num_err_assert 0, num_err_bsm 0, num_recv_graft 0,
     num_err_graft 0, num_recv_graft_ack 0, num_err_graft_ack 0,
     num_recv_state_refresh 0, num_err_state_refresh 0, num_recv_df_election 0,
     num_err_df_election 0
Nbr-ip 0x0.055e16619108p-1022ab
 num_recv_hello 30613216, num_recv_join_prune 31492, num_recv_assert 0, num_recv_bsm
     0, num_err_join_prune 0, num_err_assert 0, num_err_bsm 0, num_recv_graft 0,
     num_err_graft 0, num_recv_graft_ack 0, num_err_graft_ack 0,
     num_recv_state_refresh 0, num_err_state_refresh 0, num_recv_df_election 0,
     num_err_df_election 0
Nbr-ip 0x0.055e16619108p-1022ab
 num_recv_hello 30613216, num_recv_join_prune 31492, num_recv_assert 0, num_recv_bsm
     0, num_err_join_prune 0, num_err_assert 0, num_err_bsm 0, num_recv_graft 0,
     num_err_graft 0, num_recv_graft_ack 0, num_err_graft_ack 0,
     num_recv_state_refresh 0, num_err_state_refresh 0, num_recv_df_election 0,
     num_err_df_election 0

ECV #
```

show pim rp

The **show pim rp** command displays the multicast group, rendezvous point (RP), and reverse-path forwarding (RPF) interface for the appliance. The RPF is the closest interface to the root of the multicast tree.

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim rp

Examples

This command displays PIM RP information for the appliance.

```
ECV # show pim rp
Group :224.0.0.0, RP addr :192.172.11.11, RPF Interface :wan0

ECV #
```

show pim rtm

The **show pim rtm** command displays the multicast routing table manager (RTM).

Protocol Independent Multicast (PIM) is a layer 3 networking protocol for sending traffic from a single source to multiple destinations across a network.

Command Mode: Privileged EXEC mode

Syntax

show pim rtm

Examples

This command displays the multicast routing table manager.

```
ECV # show pim rtm
Dest IP/Subnet Address      Nexthop              IfName(IfIndex)      Type
3.3.3.3                     10.19.156.1          wan0                 Static
10.4.4.0                    10.4.4.4             -                    Connected
10.4.4.4                    0.0.0.0              -                    Connected
10.5.5.0                    10.5.5.5             -                    Connected
10.5.5.5                    0.0.0.0              -                    Connected
10.19.156.0                 10.19.156.10         wan0                 Connected
10.19.156.10                0.0.0.0              -                    Connected
10.19.157.0                 10.19.157.10         lan0                 Connected
10.19.157.10                0.0.0.0              -                    Connected
10.19.158.0                 10.19.158.10         -                    Connected
10.19.158.10                0.0.0.0              -                    Connected
10.19.159.0                 10.19.159.10         -                    Connected
10.19.159.10                0.0.0.0              -                    Connected
10.81.71.0                  10.81.71.178         -                    Connected
10.81.71.178                0.0.0.0              -                    Connected
169.254.124.0               169.254.124.1        pim0                 Connected
169.254.124.1               0.0.0.0              -                    Connected
169.254.125.0               169.254.125.1        pim1                 Connected
169.254.125.1               0.0.0.0              -                    Connected
169.254.126.0               169.254.126.1        pim2                 Connected
169.254.126.1               0.0.0.0              -                    Connected
192.172.11.11               10.19.156.1          wan0                 Static
ECV #
```

show pimlite adjacencies

The **show pimlite adjacencies** command displays the number of adjacencies for a specified multicast group. An adjacency is a relationship formed between the multicast group and Designated Router.

The **show pimlite adjacencies all** command displays the number of adjacencies for all multicast groups on the appliance.

PIM-lite is a PIM variant where an appliance can manipulate PIM routers and the PIM-SM protocol to consider other remote EdgeConnect appliances as neighboring router even when there are many routers and hops between the appliances.

Command Mode: Privileged EXEC mode

Syntax

show pimlite adjacencies *group-addr*
show pimlite adjacencies all

Parameters

group-ip-addr: Multicast group IP address (dotted decimal notation).

Examples

This command displays the adjacencies to the Multicast group at 224.0.0.0.

```
ECV # show pimlite adjacencies 224.0.0.0
Dumping Multicast Table: number of adjacencies:0

ECV #
```

show pimlite mroutes

The **show pimlite mroutes** command displays the PIM-lite multicast routing table contents. Each table entry is a multicast route, also known as an mroute.

PIM-lite is a PIM variant where an appliance can manipulate PIM routers and the PIM-SM protocol to consider other remote EdgeConnect appliances as neighboring router even when there are many routers and hops between the appliances.

Command Mode: Privileged EXEC mode

Syntax

show pimlite mroutes

Examples

This command displays the PIM-lite multicast routing table contents.

```
ECV # show pimlite mroutes
No Multicast Routes.
ECV #
```

show pimlite oifs

The **show pimlite oifs** command displays information about the outgoing interface list (OIF). Each multicast route entry has an OIF list.

PIM-lite is a PIM variant where an appliance can manipulate PIM routers and the PIM-SM protocol to consider other remote EdgeConnect appliances as neighboring router even when there are many routers and hops between the appliances.

Command Mode: Privileged EXEC mode

Syntax

show pimlite oifs

Examples

This command displays the OIFs for the route entries in the multicast routing table.

```
ECV (config) # show pimlite oifs
No OIFs in library.
ECV (config) #
```

show pimlite stats

The **show pimlite stats** command displays multicast transmission statistics.

PIM-lite is a PIM variant where an appliance can manipulate PIM routers and the PIM-SM protocol to consider other remote EdgeConnect appliances as neighboring router even when there are many routers and hops between the appliances.

Command Mode: Privileged EXEC mode

Syntax

show pimlite stats

Examples

This command displays multicast transmissions statistics.

```
ECV # show pimlite stats
Multicast Statistics:  multicast is enabled RP:192.172.11.11
-------------------------------------------------------
Packets filtered:         142146   142146
Packets tx to kernel(e):   40618    40618
Packets tx to kernel(r):  101528   101528
Adj Miss to kernel     :       0        0
FHR Pkts sent to kernel:       0        0
Hit *,g to kernel      :       0        0
Hit s,g -oif to kernel :       0        0
Hit s,g -valid to kern :       0        0
Local net rxd:                 0        0
Other mcast proto rxd:         0        0
No adj found, pkts drop:       0        0
Feature disabled:              0        0

Igmp pkts(v1,v2) blocked:      0        0

Igmp groups(v3) blocked:    7664     7664

Igmp pkts csum fixed:       7664     7664

Pim pkts filtered:             0        0


DATA PACKETS
-------------------------------------------------------
Multicast (from LAN)  Statistics
-------------------------------------------------------
Packets Received:         142146   142146
Packets Drop:                  0        0
Packets No Match:              0        0
Packets GIP Match:             0        0
```

```
Packets s,g Match:              0        0
Packets s,g Accept:             0        0
Packets *,g Match:              0        0
Packets *,g Accept:             0        0
RPF Fail:                       0        0
 Bad Tunnel:                    0        0
 Bad Peer:                      0        0
 Bad if:                        0        0
 No intf:                       0        0
 RPF Succ on fhr                0        0
 RPF Succ on rp                 0        0
 RPF Succ on interface          0        0
 RPF Succ on tunnel             0        0
Packets Duplicated:             0        0
Packets Dup Recycled:           0        0
Packets Dup New:                0        0
Packets Dup Chain:              0        0
Packets Dup Chain Recyc:        0        0
Packets Fanout ChainIn:         0        0
Packets Fanout ChainOut:        0        0



Multicast Inner (to/from tunnel)  Statistics
-----------------------------------------------------------
Packets Received:               0        0
Packets TX to tunnel:           0        0

PIM PACKETS
-----------------------------------------------------------
PIM Packets Rxd                 0        0
PIM Protocol Packets       126818   126818
PIM All Routers IP              0        0
PIM Reg Pkts in Q:              0        0
PIM Reg Pkts tot processed:     0        0
PIM Reg Pkts decap skip:        0        0
PIM Reg Pkts decap fail:        0        0
PIM Reg Pkts decap succ:        0        0
PIM Reg Pkts wrong RP  :        0        0
PIM Reg total sent     :        0        0
PIM Reg total recv     :        0        0
PIM Reg total recv tnl :        0        0
PIM Bad pkt count      :        0        0

PIM RCMP dequeued               0        0
PIM RCMP Rxd                    0        0
PIM RCMP Txd                    0        0


Multicast Configuration handling Statistics
-----------------------------------------------------------
Packets Received:              35       35
Packets Enqueued:              35       35
Packets Dequeued:              35       35
Packets Queue Current:          0        5
Packets Queue Overflow:         0        0
Packets No Buffer:              0        0
Packets ssdp dropped:           0        0
ECV #
```

show proxy-arp

The **show proxy-arp** command displays the enabled Proxy ARP status of the specified interface.

Command Mode: EXEC mode

Syntax

show proxy-arp *intf-name*

Arguments

Parameter	Description
intf-name	The interface upon which the show command displays status.

Examples

This command enables Proxy ARP status on WAN2 interface.

```
ECV (config) # proxy-arp wan2
ECV (config) # show proxy-arp wan2
interface name           proxy-arp enabled
--------------           -----------------
ECV (config) #
```

show qos-map

Use the **show qos-map** command to display a list of all the existing QoS maps. The CLI also indicates which QoS map is currently active.

Command Mode: Privileged EXEC mode

Syntax

show qos-map
show qos-map *map-name*
show qos-map *map-name priority*
show qos-map *map-name priority* **flows**
show qos-map *map-name* [*priority*] **stats**

Arguments

Parameter	Description
qos-map	Displays all existing QoS maps.
qos-map *map-name*	Displays each priority (entry) for the specified QoS map, along with their MATCH criteria and SET actions.
qos-map *map-name priority*	Displays the priority specified for the designated QoS map.
flows	Displays the flows that match the priority (entry) number specified.
stats	Displays statistics for the specified map.If the priority number is included in the command, then the match statistics are limited to that map entry.

Usage Guidelines

The default entry in any map is always priority 65535. The QoS map specifics are:

```
65535 match
          Protocol:            ip
          Source:              any
          Destination:         any
          Application:         any
          DSCP:                any
        set
          Traffic Class:       1
          LAN QoS:             trust-lan
          WAN QoS:             trust-lan
```

The following example shows the a sample list of QoS maps:

```
ECV> # show qos-map
maryann
ginger            [ACTIVE]
```

Examples

To show all the priorities in the QoS map, “map1”:

```
ECV (config) # show qos-map  map1
QoS map map1 configuration (ACTIVE)
  10    match
          Protocol:            ip
          Source:              any
          Destination:         any
          Application:         web
          DSCP:                any
        set
          Traffic Class:       1
          LAN QoS:             be
          WAN QoS:             be

  20    match
          Protocol:            ip
          Source:              172.20.20.0/24
          Destination:         any
          Application:         any
          DSCP:                any
        set
          Traffic Class:       3
          LAN QoS:             af12
          WAN QoS:             trust-lan

  40    match
          Protocol:            ip
          Source:              any
          Destination:         any
          Application:         aol
          DSCP:                any
        set
          Traffic Class:       3
          LAN QoS:             trust-lan
          WAN QoS:             trust-lan

  60    match
          Protocol:            ip
          Source:              any
          Destination:         any
          Application:         any
          DSCP:                be
        set
```

```
  65535 match
          Protocol:            ip
          Source:              any
          Destination:         any
          Application:         any
          DSCP:                any
        set
          Traffic Class:       1
          LAN QoS:             trust-lan
          WAN QoS:             trust-lan

ECV (config) #
```

To display information similar about flows that match the conditions specified by priority 100 in the map, “ginger”:

```
ECV (config) # show qos-map ginger 100 flows
Flows matching QoS Map ginger prio:100:
6 (L->W) sip:10.2.1.128 dip:10.16.1.200 ports:0/0

Total flows:1
```

show radius

Use the **show radius** command to display RADIUS settings for user authentication.

Command Mode: Privileged EXEC mode

Syntax

show radius

Examples

To show any RADIUS settings for the appliance, Tallinn:

```
ECV (config) # show radius
RADIUS defaults:
    key:
    timeout: 3
    retransmit: 1
No RADIUS servers configured.
ECV (config) #
```

show route-map

Use the **show route-map** command to display a list of all the existing route maps. The CLI also indicates which route map is currently active.

Command Mode: Privileged EXEC mode

Syntax

show route-map
show route-map *route-map-name*
show route-map *route-map-name priority-value*
show route-map *route-map-name priority-value* **flows**
show route-map *route-map-name priority-value* **stats**

Arguments

Parameter	Description
route-map	Displays all existing route maps.
route-map *route-map-name*	Displays each priority (entry) for the specified route map, along with their MATCH criteria and SET actions.
route-map *route-map-name priority-value*	Displays the priority specified for the designated route map.
flows	Displays the flows that match the priority (entry) number specified.
stats	Displays statistics for the specified map.If the priority number is included in the command, then the match statistics are limited to that map entry.

Usage Guidelines

The default entry in any map is always priority 65535. The route map specifics are:

```
ECV (config) # show route-map map1 65535
  65535 match
          Protocol:             ip
          Source:               any
          Destination:          any
          Application:          any
          DSCP:                 any
        set
          Pass-through:         Shaped
```

The following example shows the a sample list of route maps:

```
ECV> # show route-map
maryann
ginger            [ACTIVE]
```

Examples

To show all the priorities in the route map, “map1”:

```
ECV (config) # show route-map map1

Route map map1 configuration (ACTIVE)
  10    match
          Protocol:             ip
          Source:               any
          Destination:          any
          Application:          citrix
          DSCP:                 any
        set
          Primary Tunnel:       HQ-to-BranchA
          Down Action:          pass-through

  20    match
          Protocol:             etherip
          Source:               10.10.10.0/24
          Destination:          10.10.20.0/24
          DSCP:                 any
        set
          Primary Tunnel:       HQ-to-BranchA
          Down Action:          pass-through

  65535 match
          Protocol:             ip
          Source:               any
          Destination:          any
          Application:          any
          DSCP:                 any
        set
          Pass-through:         Shaped

ECV (config) #
```

To show the statistics for priority 20 in the route map, R-2-3500-2:

```
ECV (config) # show route-map R-2-3500-2 20 stats
Route Map R-2-3500-2 Lookup Statistics:

Priority 20:
 Match Succeeded:       3212721
  Permits:      3212721 Denies: 0
 Match Failed:  483
  Source IP Address:    479     Destination IP Address: 4
  Source Port:          0       Destination Port:       0
  Application:          0       DSCP Markings:  0       Protocol:       0
ECV (config) #
```

To list all the current flows that match priority 20 for the route map, R-2-3500-2:

```
ECV (config) # show route-map R-2-3500-2 10 flows
Flows matching Route Map R-2-3500-2 prio:10:

Total flows:0
eh-3500-1 (config) # show route-map R-2-3500-2 20 flows
Flows matching Route Map R-2-3500-2 prio:20:
1155 (L->W) sip:3.3.3.132 dip:3.3.5.132 ports:54317/7079
954 (L->W) sip:3.3.3.60 dip:3.3.5.60 ports:46082/7078
5169 (L->W) sip:3.3.3.79 dip:3.3.5.79 ports:17516/37693
647 (L->W) sip:3.3.3.74 dip:3.3.5.74 ports:30370/62999
4200 (L->W) sip:3.3.3.19 dip:3.3.5.19 ports:48779/1720
4193 (L->W) sip:3.3.3.115 dip:3.3.5.115 ports:50455/63239
3395 (L->W) sip:3.3.3.103 dip:3.3.5.103 ports:48726/1720
640 (L->W) sip:3.3.3.101 dip:3.3.5.101 ports:53199/58066
1368 (L->W) sip:3.3.3.16 dip:3.3.5.16 ports:18124/7079
35468 (L->W) sip:3.3.3.160 dip:3.3.5.160 ports:5060/5060
4475 (L->W) sip:3.3.3.143 dip:3.3.5.143 ports:32129/10581
1219 (L->W) sip:3.3.3.101 dip:3.3.5.101 ports:22793/7078
162 (L->W) sip:3.3.3.77 dip:3.3.5.77 ports:18249/26865
680 (L->W) sip:3.3.3.134 dip:3.3.5.134 ports:31366/38078
4414 (L->W) sip:3.3.3.31 dip:3.3.5.31 ports:8352/28438
120 (L->W) sip:3.3.3.132 dip:3.3.5.132 ports:8972/57105
4325 (L->W) sip:3.3.3.88 dip:3.3.5.88 ports:36950/36893
2354 (L->W) sip:3.3.3.148 dip:3.3.5.148 ports:7078/41540
```

show running-config

Use the **show running-config** command to display the current running configuration.

Command Mode: Privileged EXEC mode

Syntax

show running-config [**full**]

Arguments

Parameter	Description
full	Do not exclude commands that set default values.

Examples

None

show selftest disk

Use the **show selftest disk** command to run a self test and diagnostics.

Command Mode: Privileged EXEC mode

Syntax

show selftest disk

Examples

To view disk self test results:

```
ECV (config) # show selftest disk

Disk self test results:
Disk read results:
Duration: 26 seconds
Read I/O operations per second (IOPS): 391
Read rate (MBytes/second): 97
Read IOPS compared to optimal: 391%
Read rate compared to optimal: 391%

Disk write results:
Duration: 60 seconds
Write I/O operations per second (IOPS): 169
Write rate (MBytes/second): 42
Write IOPS compared to optimal: 169%
Write rate compared to optimal: 169%

Overall result: PASS


A reboot is required after disk selftest. Do you want to restart the appliance? (y/n)
```

show shaper

Use the **show shaper** command to display the shaper statistics.

Command Mode: Privileged EXEC mode

Syntax

show shaper
show shaper [**configured** | **stats**]

Arguments

Parameter	Description
configured	Displays shaper configuration.
stats	Displays shaper debug stats.

Examples

To view the shaper configuration :

```
ECV (config) # show shaper configured
wan shaper
  Max rate       : 500000 kbps
  Accuracy       : 5000 us
  class                    prio   min%   max% excess   wait
   1 default                  5     30    100    100    500
   2 real-time                1     30    100   1000    100
   3 interactive              2     20    100   1000    200
   4 best-effort              8     20    100    100    500
   5 blah                     5     30    100    100    500
   6                          5     30    100    100    500
   7                          5     30    100    100    500
   8                          5     30    100    100    500
   9                          5     30    100    100    500
  10                          5     30    100    100    500
ECV (config) #
```

show snmp

Use the **show snmp** command to display SNMP settings.

Command Mode: EXEC mode

Syntax

show snmp [**engine ID** | **user**]

Arguments

Parameter	Description
engine ID	Displays the SNMP engine ID of the local system.
user	Displays the SNMP v3 user security settings.

Examples

To display the SNMP settings:

```
ECV (config) # show snmp
SNMP enabled: yes
System location: third rock from the sun
System contact: ET Fone-Hoam
Read-only community: public
Traps enabled: yes
Events for which traps will be sent:
  raise-alarm: System Alarm has been raised
Trap sinks:
   172.20.2.191
      Enabled: yes
      Type: traps version 1
      Community: textstring
Interface listen enabled: yes
No Listen Interfaces.
ECV (config) #
```

To display the local system's SNMP engine ID:

```
ECV (config) # show snmp engineID
Local SNMP engineID: 0x80005d3b043930623464363761323365 34
ECV (config) #
```

To display the SNMP v3 user security settings:

```
ECV (config) # show snmp user
```

```
User name: admin
   Enabled:                 no
   Authentication type:     sha
   Authentication password: (NOT SET; user disabled)
   Privacy type:            aes-128
   Privacy password:        (NOT SET; user disabled)
ECV (config) #
```

show ssh client

The **show ssh client** command displays SSH settings for appliace clients.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: Privileged EXEC mode

Syntax

show ssh client

Examples

This command displays SSH client settings for the appliance.

```
ECV-A (config) # show ssh client
SSH client Strict Hostkey Checking: yes

No SSH global known hosts configured.

No SSH user identities configured.

SSH authorized keys:
  User admin:
  No authorized keys for user admin.

  User joe:
    Key 2: dfghi
ECV-A (config) #
```

show ssh server

The **show ssh server** command displays SSH server settings for the appliance.

The **show ssh server host-keys** command displays SSH settings for the appliance and the configured host keys.

Secure Shell (SSH) is a transport layer network protocol that facilitates secure remote login, command execution, and secure file transfer over unsecured networks. SSH uses cryptography to authenticate and encrypt connections between devices.

Command Mode: EXEC mode

Syntax

show ssh server
show ssh server host-keys

Examples

This command displays SSH server settings for the appliance.

```
ECV-A # show ssh server
SSH server enabled: yes
SSH server ports:  22


Host Key Finger Prints:
    RSA host key: SHA256:UF4Jb84ZTt7kgn+InFrpgtRpvKzS90yyPeDxBl9Tjns
    ECDSA host key: SHA256:eXMvanESR+jKYZ2pws/usYyzwLCZuygvAy3p/nB1Fhg

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A #
```

This command displays SSH server host keys for the ECV appliance.

```
ECV-A # show ssh server host-keys

SSH server enabled: yes
SSH server ports:  22


Host Key Finger Prints:
    RSA host key: SHA256:UF4Jb84ZTt7kgn+InFrpgtRpvKzS90yyPeDxBl9Tjns
    ECDSA host key: SHA256:eXMvanESR+jKYZ2pws/usYyzwLCZuygvAy3p/nB1Fhg
```

```
Host Keys:
    RSA host key: "ECV-A ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQD ..."
          <content truncated from example>
    ECDSA host key: "ECV-A ecdsa-sha2-nistp256AAAAE2VjZHNhLXNoYT ..."
          <content truncated from example>

SSH server Ciphers: aes256-ctr,aes192-ctr,aes128-ctr
SSH server MACs: hmac-sha2-256,hmac-sha1
SSH server KexAlgos: diffie-hellman-group14-sha1
SSH server Permitscpsftp: no
ECV-A #
```

show ssl

Use the **show ssl** command to list host certificate data.

Command Mode: Privileged EXEC mode

Syntax

show ssl

Examples

```
ECV # show ssl
SSL Proxy Settings:
        Certificate Substitution: Disabled
        Built-in CA Signing: Enabled

ECV #
```

show stats

Use the **show stats** command to display various traffic statistics.

Command Mode: EXEC mode

Syntax

show stats app *app-name* { **optimized-traffic** | **pass-through-unshaped** | **pass-through** | **all-traffic** } [**pretty**]

show stats dscp *DSCP-value* { **optimized-traffic** | **pass-through-unshaped** | **pass-through** | **all-traffic** } [**pretty**]

show stats flow { **tcpacc** | **tcpnoacc** | **nontcp** } { **optimized-traffic** | **pass-through-unshaped** | **pass-through** | **all-traffic** } [**pretty**]

show stats ftype { **tcpacc** | **tcpnoacc** | **nontcp** } { **optimized-traffic** | **pass-through-unshaped** | **pass-through** | **all-traffic** } [**pretty**]

show stats tclass *traffic-class-number* { **optimized-traffic** | **pass-through-unshaped** | **pass-through** | **all-traffic** } [**pretty**]

Arguments

Parameter	Description
app *app-name*	Displays network traffic statistics by application.
dscp *DSCP-value*	Displays network statistics by DSCP marking.
tclass *traffic-class-number*	Displays network traffic statistics by traffic-class.
ftype { **tcpacc** | **tcpnoacc** | **nontcp** }	Displays flow type traffic statistics: **tcpacc** Accelerated TCP traffic **tcpnoacc** Non-accelerated TCP traffic **nontcp** Non-TCP traffic
flow { **tcpacc** | **tcpnoacc** | **nontcp** }	Displays flow statistics: **tcpacc** Accelerated TCP traffic **tcpnoacc** Non-accelerated TCP traffic **nontcp** Non-TCP traffic
all-traffic	Displays all optimized, pass-through, and pass-through-unshaped traffic.
optimized-traffic	Displays all optimized traffic.
pass-through	Displays pass-through traffic.
pass-through-unshaped	Displays pass-through unshaped traffic.
pretty	Displays in thousands, separated and right-aligned.

Examples

None

show stats tunnel

Use the **show stats tunnel** command to display tunnel traffic statistics.

Command Mode: EXEC mode

Syntax

show stats tunnel *tunnel-name*

show stats tunnel *tunnel-name* { **latency** | **qos-error** | **qos-error** *traffic-class-number* } [**pretty**]
show stats tunnel *tunnel-name* [**pretty**]

show stats tunnel default
show stats tunnel default { **latency** | **qos-error** } [**pretty**]
show stats tunnel default [**pretty**]

show stats tunnel pass-through { **latency** | **qos-error**} [**pretty**]
show stats tunnel pass-through [**pretty**]

show stats tunnel pass-through-unshaped { **latency** | **qos-error** } [**pretty**]
show stats tunnel pass-through-unshaped [**pretty**]

show stats tunnel all-traffic { **latency** | **qos-error** } [**pretty**]
show stats tunnel all-traffic [**pretty**]

show stats tunnel optimized-traffic { **latency** | **qos-error** } [**pretty**]
show stats tunnel optimized-traffic [**pretty**]

Arguments

Parameter	Description
tunnel-name	Specifies the name of the tunnel.
all-traffic	Displays all optimized, pass-through, and pass-through-unshaped traffic.
latency	Displays tunnel latency statistics.
optimized-traffic	Displays all optimized traffic.
pass-through	Displays pass-through traffic.
pass-through-unshaped	Displays pass-through unshaped traffic.
pretty	Displays in thousands, separated and right-aligned.
qos-error	Displays tunnel QoS error statistics on all traffic classes.
qos-error *traffic-class-number*	Displays tunnel QoS error statistics for the specified traffic class.

Examples

To view optimized traffic, formatted for easier reading:

```
ECV # show stats tunnel optimized-traffic pretty
                bytes_wtx:   714,823,758
                bytes_wrx:   729,500,245
                bytes_ltx: 5,739,117,443
                bytes_lrx: 3,231,002,684
                 pkts_wtx:       816,634
                 pkts_wrx:       977,866
                 pkts_ltx:     4,529,350
                 pkts_lrx:     2,731,216
                 comp_l2w:             0
                 comp_w2l:             0
         comp_noohead_l2w:             0
         comp_noohead_w2l:             0
                latency_s:             0
            latency_min_s:             0
             flow_ext_tcp:             1
          flow_ext_tcpacc:             0
             flow_ext_non:             0
                 flow_add:             0
                 flow_rem:             0
     loss_prefec_wrx_pkts:         1,308
    loss_postfec_wrx_pkts:             0
      loss_prefec_wrx_pct:             0
     loss_postfec_wrx_pct:             0
      ooo_prepoc_wrx_pkts:             0
     ooo_postpoc_wrx_pkts:            26
       ooo_prepoc_wrx_pct:             0
      ooo_postpoc_wrx_pct:             0
           ohead_wrx_pkts:     3,142,683
           ohead_wtx_pkts:     3,126,115
          ohead_wrx_bytes:   463,542,375
          ohead_wtx_bytes:   474,786,262
      ohead_wrx_hdr_bytes:   113,928,904
      ohead_wtx_hdr_bytes:   184,900,104
              bw_util_pct:             0
ECV #
```

show subif

Use the **show subif** command to display sub-interface information.

Command Mode: EXEC mode

Syntax

show subif

Examples

None

show subnet

Use the **show subnet** command to display subnet-related information.

Command Mode: Privileged EXEC mode

Syntax

show subnet
show subnet bgp [**ipv4**]
show subnet configured
show subnet debug { **module** | **peer** }
show subnet learned
show subnet ospf [**ipv4**]

Arguments

Parameter	Description
bgp [**ipv4**]	Displays BGP advertisable (ipv4) rules.
configured	Displays configured rules.
debug module	Displays subnet module state, as a debugging aid.
debug peer	Displays subnet peer state, as a debugging aid.
ospf [**ipv4**]	Displays OSPF advertisable (ipv4) rules.
learned	Displays learned rules.

Examples

To display configured rules:

```
ECV (config) # show subnet configured
Route Table: 1/20000 entries
prefix/len :     metric    peer id saas
details
10.1.153.0/24  :         50    1659809 0
automatic advertized BGP local
```

show system

Use the **show system** command to display system configuration information.

Command Mode: Privileged EXEC mode

Syntax

show system
show system arp-table-size
show system auto-mac-configure
show system bypass
show system disk [**brief** | **smart-data**]
show system firmware
show system network-memory media
show system [**nexthops** | **wan-next-hops**]
show system peer-list
show system registration
show system smb-signing
show system ssl-ipsec-override

Arguments

Parameter	Description
arp-table-size	Displays configured system ARP (Address Resolution Protocol) table size.
auto-mac-configure	Displays auto MAC-NIC configuration.
bypass	Displays system bypass information.
disk	Displays system disk information.
disk brief	Displays brief system disk information.
disk smart-data	Displays system disk SMART (Self-Monitoring Analysis and Reporting Technology) – statistics a disk collects about itself.
firmware	Displays system firmware information.
network-memory media	Displays the media used for the system's network memory.
nexthops	Displays system next-hops and their reachability and uptime.
peer-list	Displays peer list information.
registration	Displays system registration information.
smb-signing	Displays SMB signing option.
ssl-ipsec-override	Displays any SSL IPSec override.

Parameter	Description
wan-next-hops	Displays system configuration WAN next-hops, along with their configured state and current status.

Examples

To display the configured system ARP table size:

```
ECV (config) # show system arp-table-size
System Arp Table Size

    Configured maximum arp table size       :    10240
    System's current maximum arp table size :    10240
```

To display the system disk information:

```
ECV (config) # show system disk
RAID 0 Info:
Status:           OK
Type:             Software
Size:             216
Percent Complete: 100
Drives:           1,0
Configuration:    RAID_1
Disk ID 0
  Status:         OK
  Size:           232 GB
  Serial Number:  WD-WCAL73249872

Disk ID 1
  Status:         OK
  Size:           232 GB
  Serial Number:  WD-WCAL73275682

ECV (config) #
```

To display the brief system disk information:

```
ECV (config) # show system disk brief
RAID 0 Info:
Status:           OK
Type:             Software
Size:             216
Percent Complete: 100
Drives:           1,0
Configuration:    RAID_1
ID      Status  Size(GB)        Serial
0       OK      232             WD-WCAL73249872
1       OK      232             WD-WCAL73275682

ECV (config) ##
```

To display the type of media being used for Network Memory:

```
ECV # show system network-memory media
Network Memory Media: ram and disk
ECV #
```

show system cc

The **show system cc** command displays the Common Criteria enable mode status on the appliance.

Common Criteria is an international standard for computer security certification. When Common Criteria mode is enabled, the appliance is Common Criteria compliant to a set of guidelines and certifications that ensure the appliance meets the security standard that includes PKI certificates, Online certificate status protocol, and enhanced logging.

Command Mode: Privileged EXEC mode

Syntax

show system cc

Usage Guidelines

The **show system cc** command is not available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **show cc**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command displays the Common Criteria status on a appliance where Common Criteria is enabled.

```
ECV # show system cc

Common Criteria mode: Enabled
ECV #
```

show system fips

The **show system fips** command displays the FIPS enable mode status for the appliance.

Federal Information Processing Standards (FIPS) is a set of publicly announced standards that the National Institute of Standards and Technology (NIST) developed for use in non-military United States government agencies and contractor applications.

Command Mode: Privileged EXEC mode

Syntax

show system fips

Usage Guidelines

The **show system fips** command is not available in ECOS version 9.4.3 and all later versions. The equivalent command available in these versions is **fips show**.

The **show version** command displays the ECOS version currently running on the appliance.

Examples

This command displays the FIPS status on a appliance where FIPS is disabled.

```
ECV # show system fips

FIPS mosde: Disabled
ECV #
```

show tacacs

Use the **show tacacs** command to display TACACS+ settings.

Command Mode: Privileged EXEC mode

Syntax

show tacacs

Examples

```
ECV (config) # show tacacs
TACACS+ defaults:
    key:
    timeout: 3
    retransmit: 1
No TACACS+ servers configured.
ECV (config) #
```

show tca

Use the **show tca** command to display threshold crossing alert settings.

Command Mode: EXEC mode

Syntax

show tca
show tca *tca-name*

Arguments

Parameter	Description
tca *tca-name*	Specifies which threshold crossing alert to display. The options are: **file-system-utilization** How much of the file system space has been used, expressed as a percentage. **lan-side-rx-throughput** LAN–side Receive throughput, in kilobits per second (**kbps**). **latency** Tunnel latency, in milliseconds (**ms**). **loss-post-fec** Tunnel loss, as **tenths of a percent**, *after* applying Forward Error Correction (FEC). **loss-pre-fec** Tunnel loss, as **tenths of a percent**, *before* applying Forward Error Correction (FEC). **oop-post-poc** Tunnel out-of-order packets, as **tenths of a percent**, *after* applying Packet Order Correction (POC). **oop-pre-poc** Tunnel out-of-order packets, as **tenths of a percent**, *before* applying Packet Order Correction (POC). **optimized flows** Total number of optimized flows. **reduction** Tunnel reduction, in percent (**%**). **total-flows** Total number of flows. **utilization** Tunnel utilization, as a percent (**%**). **wan-side-tx-throughput** WAN–side transmit throughput, in kilobits per second (**kbps**).

Examples

To display a summary of what the defaults are for the various threshold crossing alerts (this information is static because it is **not** the same as reporting the current state of any alert):

```
ECV > show tca
file-system-utilization (File-system utilization):                enabled
lan-side-rx-throughput (LAN-side receive throughput):             disabled
```

```
latency (Tunnel latency):                                          enabled
loss-post-fec (Tunnel loss post-FEC):                              disabled
loss-pre-fec (Tunnel loss pre-FEC):                                disabled
oop-post-poc (Tunnel OOP post-POC):                                disabled
oop-pre-poc (Tunnel OOP pre-POC):                                  disabled
optimized-flows (Total number of optimized flows):                 disabled
reduction (Tunnel reduction):                                      disabled
total-flows (Total number of flows):                               disabled
utilization (Tunnel utilization):                                  disabled
wan-side-tx-throughput (WAN-side transmit throughput):             disabled
ECV > fil
```

To display how reduction is currently configured in the threshold crossing alerts:

```
ECV > show tca reduction
reduction - Tunnel reduction:
  default
    enabled:                  no
  A-to-B
    enabled:                  yes
    falling:
      raise-threshold:        20 %
      clear-threshold:        35 %
  pass-through
    enabled:                  no
  pass-through-unshaped
    enabled:                  no
ECV >
```

show terminal

Use the **show terminal** command to display the current terminal settings.

Command Mode: EXEC mode

Syntax

show terminal

Examples

```
ECV (config) # show terminal
CLI current session settings
  Terminal width:          80 columns
  Terminal length:         24 rows
  Terminal type:           vt102
ECV (config) #
```

show transceiver

The **show transceiver** command displays transceiver information for a specified transceiver, referenced by the port that the transceiver services.

The **show transceiver detail** command displays detailed state transceiver information for a specified transceiver.

Command Mode: EXEC mode

Syntax

show transceiver
show transceiver detail

show transceiver *intf-name*
show transceiver *intf-name* **detail**

Parameters

intf-name: Interface containing the target transceiver. Commands that omit this parameter returns information for all transceivers on the appliance.

Examples

This command displays a list of transceivers on the appliance.

```
ECV-A (config) # show transceiver
  Port  Type                  Part Number  Rev. Serial Number  Speed    Length
  ----  ----                  -----------  ---- -------------  -----    ------
  wan0  N/A                   74752-9742   09   MOC2021A6XT    Unkn     3m (Copper)
  wan1  SFP-10G-SR/SFP-1G-SX  EC-SFP-SR    A    N86BP5H        1G/10G   300m (OM3)

ECV-A (config) #
```

This command displays a hardware and status information for the transceiver assigned to the WAN0 interface.

```
ECV-A (config) # show transceiver wan0 detail
Transceiver detail for wan0

    Type            : N/A
    Part Number     : 74752-9742
    Revision        : 09
    Serial Number   : MOC2021A6XT
    Vendor          : CISCO-MOLEX
    Form Factor     : SFP
    Cable Type      : Copper
    Speed           : Unkn
```

```
    Length          : 3m (Copper)
    Diagnostics     : None
    Removable       : Yes
    Connector Type  : Copper pigtail

Status:

    Temperature     : 0.00 degrees C / 32.00 degrees F
    Voltage         : 0.0000 V
    Tx Bias         : 0.000 mA
    Tx Power        : 0.0000 mW
    Rx Power        : 0.0000 mW

Alarms:

None

ECV-A (config) #
```

show tunnel

Use the **show tunnel** command to display the detailed running state for all tunnels.

An equivalent command is **show interfaces tunnel**.

Command Mode: Privileged EXEC mode

Syntax

show tunnel [**brief** | **configured** | **peers** | **summary**]
show tunnel *t-name* [**brief** | **configured** | **fastfail** | **ipsec** [**status**] | **summary** | **traceroute**]
show tunnel *t-name* **stats flow** [*traffic-class_1-10*]
show tunnel *t-name* **stats ipsec**
show tunnel *t-name* **stats latency**
show tunnel *t-name* **stats qos** [*DSCP-value*]
show tunnel *t-name* **stats traffic-class**
show tunnel stats cifs
show tunnel stats ssl

Arguments

Parameter	Description
brief	Displays brief running state for the tunnel(s).
configured	Displays configuration for the tunnel(s).
fastfail	Displays Fastfail information. When multiple tunnels are carrying data between two appliances, this feature determines on what basis to disqualify a tunnel from carrying data, and how quickly.
ipsec status	Displays the specified tunnel's IPSec information.
peers	Displays table summary information for all tunnel peers.
redundancy	Displays redundancy information (regarding WCCP or VRRP)
stats cifs	Displays system-wide CIFS statistics.
stats flow	Displays the flow metrics for the default traffic class
stats flow [*t-class*]	Displays flow metrics for specified traffic class in the tunnel. Value range is 1 to 10.
stats ipsec	Displays the IPSec statistics for the designated tunnel.
stats latency	Displays the latency metrics for the designated tunnel.
stats qos	Displays default QoS statistics. Default DSCP value is **be** (best effort).
stats qos *DSCP*	Displays QoS statistics for a specified DSCP value in the tunnel.

Parameter	Description
stats ssl	Displays system-wide SSL statistics.
stats traffic-class	Displays traffic class statistics for a specified traffic class.
summary	Displays summary information for the tunnel(s).
traceroute	Displays traceroute information for this tunnel.
tunnel *t-name*	Displays the detailed running state for this tunnel.

Defaults

The default DSCP value for QoS is **be** (Best Effort).

Usage Guidelines

If you don't specify a tunnel, then the output includes information for **all** tunnels. If you do specify a tunnel, then the output is limited to that tunnel.

Examples

To display the IPSec status for the tunnel, "tunnel-2-7501", in appliance, "eh-3500-1":

```
ECV (config) # show tunnel tunnel-2-7501 ipsec status
Tunnel tunnel-2-7501 ipsec state
  Tunnel Oper:         Down
  IPSec Enabled:       no
  IPSec Oper:          Disabled
  Total IPSec SAs:     in:0 out:0
ECV (config) #
```

To display the statistics for Traffic Class 41 for "t1", in appliance, "eh-3500-1":

```
ECV (config) # show tunnel t1 stats traffic-class 4
Tunnel t1 Traffic Class 4 Statistics:
 RX bytes:            0         TX bytes:            0
 RX packets:          0         TX packets:          0
                                TX Invalid packets:  0

 LAN queue dropped packets
   Packet Overload:           0
   Byte Overload:             0
   Packet Overload on Flow:   0
   Byte Overload on Flow:     0
   Queue Time Exceeded:       0
ECV (config) #
```

To display the latency statistics for “tunnel-2-8504”, in appliance, “eh-3500-1”:

```
ECV (config) # show tunnel tunnel-2-8504 stats latency
Tunnel tunnel-2-8504 QOS 0 Latency Metrics:
 Minimum Round Trip Time :                0
 Maximum Round Trip Time :                4
 Average Round Trip Time :                0

   Byte Overload on Flow:     0
   Queue Time Exceeded:       0
ECV (config) #
```

show usernames

The **show usernames** command displays a list of user accounts.

Command Mode: Privileged EXEC mode

Syntax

show usernames

Examples

This command displays the user accounts on the appliance.

```
ECV (config) # show usernames
Chris      Capability: admin     Password set
admin      Capability: admin     Password set
monitor    Capability: monitor   Password set
ECV (config) #
```

show users

The **show users** command displays a list of users that are currently logged into the gateway.

Command Mode: EXEC mode

Syntax

show users

Examples

This command displays the users that are logged into the appliance.

```
ECV (config) # show users
Line       User            Host              Login Time            Idle
pts/0      admin           172.20.41.92      2009/01/12 12:37:47   0s
Total users: 1
```

show users history

The **show users history** command displays a list of user sessions for all user accounts.

The **show users history** command that includes a username parameter returns the list of sessions for a specified account.

Command Mode: EXEC mode

Syntax

show users history
show users history username *username-text*

Parameters

username-text: Username of account for which command displays login history.

Examples

This command displays the login history for the **admin** user account.

```
ECV (config) # show users history username admin
admin    ttyS0                        Thu Dec 11 13:50   still logged in
admin    ttyS0                        Thu Dec 11 12:47 - 13:50  (01:03)
admin    ttyS0                        Thu Dec 11 11:48 - 12:03  (00:15)
admin    ttyS0                        Wed Dec 10 17:13 - 18:14  (01:00)
admin    ttyS0                        Tue Dec  9 21:49 - 22:33  (00:44)
admin    ttyS0                        Tue Dec  9 20:31 - 20:56  (00:24)
wtmp begins Tue Dec  9 20:31:45 2024
```

show version

Use the **show version** command to display version information for current system image.

Command Mode: EXEC mode

Syntax

show version [**concise**]

Arguments

Parameter	Description
concise	Displays concise version information.

Usage Guidelines

To display verbose version information, enter **show version** without an argument.

Examples

To display version information for the current system image:

```
ECV (config) # show version
Product name:      NX Series Appliance
Product release:   2.0.0.0_15619
Build ID:          #1-dev
Build date:        2007-06-07 20:00:58
Build arch:        x86_64
Built by:          root@bigchief

Uptime:            24m 40s

Product model:     NX3500
System memory:     3469 MB used / 591 MB free / 4061 MB total
Number of CPUs:    1
CPU load averages: 0.39 / 0.20 / 0.19
ECV (config) #
```

To display concise version information for the appliance, “Tallinn”:

```
ECV (config) # show version concise
hidalgo 2.0.0.0_15619 #1-dev 2007-06-07 20:00:58 x86_64 root@bigchief:unknown
ECV (config) #
```

show vlan

Use the **show vlan** command to display VLAN information.

Command Mode: Privileged EXEC mode

Syntax

show vlan

Examples

This is in Standard 4-port mode with two IPs:

```
ECV# show vlan

Tag     Interface   IP Nexthop      Second Nexthop
----    ---------   -------------   ------------
206     bvi0.206    80.80.80.1/24   80.80.80.2
70      bvi0.70     70.70.70.1/24   70.70.70.2
```

show vrrp

The **show vrrp** command displays VRRP parameters for all VRRP groups on the appliance.

The **show vrrp brief** command displays operational state information for all VRRP groups on the appliance.

The **show vrrp configured** command displays configured information for all VRRP groups on the appliance.

Virtual Router Redundancy Protocol (VRRP) is a layer 3 networking protocol that supports redundancy by facilitating transparent failover. VRRP enables a group of gateways to share a single virtual IP address to form a single virtual gateway, ensuring successful failover and high availability of the virtual gateway.

Command Mode: Privileged EXEC mode

Syntax

show vrrp
show vrrp brief
show vrrp configured

Usage Guidelines

This command and the **show interface vrrp** command displays identical information.

Examples

This command displays VRRP parameters for VRRP groups on the appliance.

```
ECV-A (config) # show vrrp
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 200
  Advertisement interval    : 2 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
  IP Address Owner          : no
```

```
  Current Priority          : 200
  Current State             : master
  State Uptime              : 0 days 2 hrs 54 mins 47 secs 429 msecs
  Master State Transitions  : 1
  Master IP address         : 10.19.157.10
  Virtual Mac Address       : 00:00:5e:00:01:64
ECV-A (config) #
```

This command displays VRRP operational state parameters for VRRP groups on the appliance.

```
ECV-A (config) # show vrrp brief
Intf   Grp   Pre   Adv  Group Addr     Version State  Master Addr    Pri Own
lan0   100   yes   2    10.19.157.100  2       master  10.19.157.10   200 no
ECV-A (config) #
```

This command displays VRRP configuration information for VRRP groups on the appliance.

```
ECV-A (config) # show vrrp configured
VRRP Interface lan0 - Group 100
  Virtual IP address        : 10.19.157.100
  VRRP Version              : 2
  Admin                     : up
  Preemption Enabled        : yes
  Priority (configured)     : 200
  Advertisement interval    : 2 secs
  Holddown Timer            : 120 secs
  Authentication String     : __*
  Description String        :
  Packet Trace Enabled      : no
ECV-A (config) #
```

show wccp

Use the **show wccp** command to display Web Cache Communications Protocol (WCCP) settings.

Command Mode: Privileged EXEC mode

Syntax

show wccp
show wccp *51-255*
show wccp [**configured** | **detail**]
show wccp *51-255* [**assignment** | **configured** | **detail**]

Arguments

Parameter	Description
wccp *51-255*	Specifies a WCCP service group ID.
assignment	Displays the details of a WCCP service group.
configured	Displays a configured WCCP service group.
detail	Displays details for a configured WCCP service group.
view	Displays a configured WCCP service group in view.

Usage Guidelines

Use the **show wccp** command without an argument to display global WCCP information.

Examples

To show an appliance's global WCCP information:

```
ECV (config) # show wccp
Global WCCP information

    Appliance information:
        Appliance Identifier:       172.30.2.34
        Protocol Version:
        Multicast TTL:              5
        Admin State:                Disabled

% There are no configured WCCP service groups.
```

To display the configuration for the WCCP service group, 51:

```
ECV (config) # show wccp 51 configured
Service Identifier: 51
    Admin State:               up
    Interface:                 wan0
    Appliance Identifier:
    Router IP address:         10.10.10.7
    Protocol:                  tcp
    Weight:                    100
    Priority:                  128
    Policy Group:              300
    Password:

    Forwarding Method:         either
    Force-L2-Return:           no
    Assignment Method:         either
    Assignment Detail:         lan-ingress
      HASH Assignments
        hash-srcip:            yes
        hash-dstip:            no
        hash-srcport:          no
        hash-dstport:          no
      MASK Assignments
        mask-srcip:            0x00001741
        mask-dstip:            0x00000000
        mask-srcport:          0x0000
        mask-dstport:          0x0000

ECV (config) #
```

To show the compatibility mode of WCCP service group 98:

```
ECV (config) # show wccp 98 configured
Service Identifier: 98
    Admin State:               up
    Interface:                 wan0
    Appliance Identifier:      6.6.6.1
    Router IP address:         6.6.6.101
    Protocol:                  tcp
    Weight:                    100
    Priority:                  128
    Policy Group:              300
    Password:
    Compatibility Mode:        nexus

    Forwarding Method:         either
    Force-L2-Return:           no
    Assignment Method:         either
    Assignment Detail:         lan-ingress
      HASH Assignments
        hash-srcip:            yes
        hash-dstip:            no
        hash-srcport:          no
        hash-dstport:          no
```

```
      MASK Assignments
        mask-srcip:             0x00001741
        mask-dstip:             0x00000000
        mask-srcport:           0x0000
        mask-dstport:           0x000
ECV (config) #
```

show web

Use the **show web** command to display Web user interface configuration and status.

Command Mode: Privileged EXEC mode

Syntax

show web

Examples

```
ECV (config) # show web
Web User Interface enabled: yes
   HTTP port:     80
   HTTP enabled:  yes
   HTTPS port:    443
   HTTPS enabled: yes
   Inactivity timeout: 30 minutes
   Max Web user sessions: 10
   Active Web user sessions: 1
ECV (config) #
```

show whoami

Use the **show whoami** command to display the identity and capabilities of the current user.

Command Mode: EXEC mode

Syntax

show whoami

Examples

```
ECV > show whoami
Current user: admin
Capabilities: admin
ECV >
```

Hewlett Packard
Enterprise