Check Point 1100 and 600 Appliance

CLI and Advanced Routing

Administration Guide

9 September 2014

Classification: [Protected]
Important Information

Latest Software
We recommend that you install the most recent software release to stay up-to-date with the latest functional improvements, stability fixes, security enhancements and protection against new and evolving attacks.

Latest Documentation
The latest version of this document is at: (http://supportcontent.checkpoint.com/documentation_download?ID=26395)
To learn more, visit the Check Point Support Center (http://supportcenter.checkpoint.com).

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>09 September 2014</td>
<td>Updated set admin access (on page 41)</td>
</tr>
<tr>
<td>11 June 2014</td>
<td>Updated information in OSPF Areas (on page 136)</td>
</tr>
<tr>
<td>23 February 2014</td>
<td>Updated the option to set VLAN on DMZ/WAN interfaces for Static IP, DHCP, PPPoe, PPTP, and L2TP under add internet connection (on page 16)</td>
</tr>
<tr>
<td></td>
<td>Updated ADSL connection type commands and other information under set internet-connection (on page 53)</td>
</tr>
<tr>
<td></td>
<td>Added read-only permission information in Adding Administrator User (on page 25) and Setting Permissions for an Admin User (on page 71)</td>
</tr>
<tr>
<td></td>
<td>Added RADIUS server authentication commands - Set the Main WLAN (on page 76) and Set VAP (on page 76)</td>
</tr>
<tr>
<td>7 November 2013</td>
<td>Updated information in Redistributing Routes to BGP (on page 99)</td>
</tr>
<tr>
<td>16 September 2013</td>
<td>Updates made to the Operation Mode section of Wireless Connection Settings (on page 72)</td>
</tr>
<tr>
<td>14 August 2013</td>
<td>First release of this document</td>
</tr>
</tbody>
</table>

Feedback
Check Point is engaged in a continuous effort to improve its documentation.
Please help us by sending your comments (mailto:cp TECHPUB_FEEDBACK@checkpoint.com?subject=Feedback on Check Point 1100 and 600 Appliance CLI and Advanced Routing Administration Guide).
## Contents

<table>
<thead>
<tr>
<th>Important Information</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLI Reference</td>
<td>10</td>
</tr>
<tr>
<td>Using Command Line Reference</td>
<td>12</td>
</tr>
<tr>
<td>CLI Syntax</td>
<td>13</td>
</tr>
<tr>
<td>Running Clish Commands from Expert Mode</td>
<td>13</td>
</tr>
<tr>
<td>Supported Linux Commands</td>
<td>14</td>
</tr>
<tr>
<td>add admin access</td>
<td>14</td>
</tr>
<tr>
<td>add antispm_allowed_sender</td>
<td>14</td>
</tr>
<tr>
<td>add bridge</td>
<td>15</td>
</tr>
<tr>
<td>add host</td>
<td>15</td>
</tr>
<tr>
<td>add interface</td>
<td>15</td>
</tr>
<tr>
<td>add internet connection</td>
<td>16</td>
</tr>
<tr>
<td>Configuring Static IP</td>
<td>16</td>
</tr>
<tr>
<td>Configuring DHCP</td>
<td>16</td>
</tr>
<tr>
<td>Configuring PPTP and L2TP</td>
<td>17</td>
</tr>
<tr>
<td>Configuring PPPoE</td>
<td>18</td>
</tr>
<tr>
<td>Configuring PPeA</td>
<td>19</td>
</tr>
<tr>
<td>Configuring EoA</td>
<td>20</td>
</tr>
<tr>
<td>Configuring Cellular Modem</td>
<td>20</td>
</tr>
<tr>
<td>add net-obj</td>
<td>20</td>
</tr>
<tr>
<td>Single IP</td>
<td>21</td>
</tr>
<tr>
<td>Network</td>
<td>21</td>
</tr>
<tr>
<td>IP Range</td>
<td>21</td>
</tr>
<tr>
<td>add snmp</td>
<td>22</td>
</tr>
<tr>
<td>Adding SNMP v2 Traps Receiver</td>
<td>22</td>
</tr>
<tr>
<td>Adding SNMP v3 Traps Receiver</td>
<td>22</td>
</tr>
<tr>
<td>Adding SNMP User</td>
<td>22</td>
</tr>
<tr>
<td>add static-route</td>
<td>23</td>
</tr>
<tr>
<td>Adding a Route with a Specific Gateway IP Address</td>
<td>23</td>
</tr>
<tr>
<td>Adding a Route with a Specific Interface and Priority</td>
<td>24</td>
</tr>
<tr>
<td>add switch</td>
<td>24</td>
</tr>
<tr>
<td>add user</td>
<td>25</td>
</tr>
<tr>
<td>Adding Local User</td>
<td>25</td>
</tr>
<tr>
<td>Adding Administrator User</td>
<td>25</td>
</tr>
<tr>
<td>add wlan vap</td>
<td>26</td>
</tr>
<tr>
<td>backup settings</td>
<td>26</td>
</tr>
<tr>
<td>cphaprobus</td>
<td>27</td>
</tr>
<tr>
<td>cphastop</td>
<td>28</td>
</tr>
<tr>
<td>cpinfo</td>
<td>29</td>
</tr>
<tr>
<td>cpshell</td>
<td>29</td>
</tr>
<tr>
<td>cpstart</td>
<td>29</td>
</tr>
<tr>
<td>cpstat</td>
<td>30</td>
</tr>
<tr>
<td>cpstop</td>
<td>31</td>
</tr>
<tr>
<td>cpwd_admin</td>
<td>31</td>
</tr>
<tr>
<td>delete bridge</td>
<td>32</td>
</tr>
<tr>
<td>delete dhcp</td>
<td>32</td>
</tr>
<tr>
<td>Deleting Excluded IP Addresses</td>
<td>32</td>
</tr>
<tr>
<td>delete dns</td>
<td>33</td>
</tr>
<tr>
<td>delete domainname</td>
<td>33</td>
</tr>
<tr>
<td>delete host</td>
<td>33</td>
</tr>
<tr>
<td>delete interface</td>
<td>34</td>
</tr>
<tr>
<td>delete internet-connection</td>
<td>34</td>
</tr>
<tr>
<td>Command</td>
<td>Page</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>set interface</td>
<td>53</td>
</tr>
<tr>
<td>set inactivity</td>
<td>53</td>
</tr>
<tr>
<td>set dns</td>
<td>52</td>
</tr>
<tr>
<td>set antispam mode</td>
<td>51</td>
</tr>
<tr>
<td>set bridge</td>
<td>50</td>
</tr>
<tr>
<td>set dhcp server</td>
<td>49</td>
</tr>
<tr>
<td>Setting the IP Pool</td>
<td>49</td>
</tr>
<tr>
<td>Excluding IP Addresses</td>
<td>49</td>
</tr>
<tr>
<td>Enabling the DHCP Server</td>
<td>48</td>
</tr>
<tr>
<td>Configuring the Default Gateway</td>
<td>48</td>
</tr>
<tr>
<td>Configuring Host Name</td>
<td>47</td>
</tr>
<tr>
<td>Configuring the WINS Server</td>
<td>47</td>
</tr>
<tr>
<td>Automatically configuring the WINS Server</td>
<td>47</td>
</tr>
<tr>
<td>Configuring IP Lease Time</td>
<td>47</td>
</tr>
<tr>
<td>Configuring DNS Server</td>
<td>47</td>
</tr>
<tr>
<td>Configuring the NTP Server</td>
<td>47</td>
</tr>
<tr>
<td>Configuring a TFTP Server</td>
<td>46</td>
</tr>
<tr>
<td>Configuring the Path for a Bootstrap File</td>
<td>46</td>
</tr>
<tr>
<td>Configuring Call Manager</td>
<td>46</td>
</tr>
<tr>
<td>Configuring X-Windows Display</td>
<td>46</td>
</tr>
<tr>
<td>Configuring VoIP Phones</td>
<td>46</td>
</tr>
<tr>
<td>Configuring Custom DHCP Option</td>
<td>46</td>
</tr>
<tr>
<td>Deleting DHCP Custom Option Code</td>
<td>46</td>
</tr>
<tr>
<td>set dhcp relay</td>
<td>45</td>
</tr>
<tr>
<td>set dns</td>
<td>45</td>
</tr>
<tr>
<td>set dns proxy</td>
<td>45</td>
</tr>
<tr>
<td>set dns mode</td>
<td>45</td>
</tr>
<tr>
<td>set domainname</td>
<td>45</td>
</tr>
<tr>
<td>set dynamic-dns</td>
<td>45</td>
</tr>
<tr>
<td>Managing dynamic DNS details</td>
<td>44</td>
</tr>
<tr>
<td>set expert password</td>
<td>44</td>
</tr>
<tr>
<td>set external ports access</td>
<td>44</td>
</tr>
<tr>
<td>set host</td>
<td>44</td>
</tr>
<tr>
<td>set hostname</td>
<td>44</td>
</tr>
<tr>
<td>set inactivity-timeout</td>
<td>44</td>
</tr>
<tr>
<td>set interface</td>
<td>44</td>
</tr>
<tr>
<td>Managing Interfaces</td>
<td>44</td>
</tr>
<tr>
<td>Configuring Static IP</td>
<td>44</td>
</tr>
<tr>
<td>Configuring Advanced Interface Settings</td>
<td>44</td>
</tr>
<tr>
<td>Delete Internet Connection</td>
<td>34</td>
</tr>
<tr>
<td>Delete ICMP servers</td>
<td>34</td>
</tr>
<tr>
<td>delete net-obj</td>
<td>34</td>
</tr>
<tr>
<td>delete static-route</td>
<td>34</td>
</tr>
<tr>
<td>delete proxy</td>
<td>34</td>
</tr>
<tr>
<td>delete snmp</td>
<td>34</td>
</tr>
<tr>
<td>delete switch</td>
<td>34</td>
</tr>
<tr>
<td>delete user</td>
<td>34</td>
</tr>
<tr>
<td>delete wlan vap</td>
<td>34</td>
</tr>
<tr>
<td>dynamic objects</td>
<td>34</td>
</tr>
<tr>
<td>exit</td>
<td>34</td>
</tr>
<tr>
<td>fetch certificate</td>
<td>34</td>
</tr>
<tr>
<td>fetch license</td>
<td>34</td>
</tr>
<tr>
<td>fetch policy</td>
<td>34</td>
</tr>
<tr>
<td>fw Commands</td>
<td>34</td>
</tr>
<tr>
<td>reboot</td>
<td>34</td>
</tr>
<tr>
<td>restore default-settings</td>
<td>34</td>
</tr>
<tr>
<td>restore settings</td>
<td>34</td>
</tr>
<tr>
<td>revert to factory defaults</td>
<td>34</td>
</tr>
<tr>
<td>revert to saved image</td>
<td>34</td>
</tr>
<tr>
<td>set admin access</td>
<td>34</td>
</tr>
<tr>
<td>set antispam mode</td>
<td>34</td>
</tr>
<tr>
<td>set bridge</td>
<td>34</td>
</tr>
<tr>
<td>set date</td>
<td>34</td>
</tr>
<tr>
<td>set dhcp server</td>
<td>34</td>
</tr>
<tr>
<td>set dhcp relay</td>
<td>34</td>
</tr>
<tr>
<td>set dns</td>
<td>34</td>
</tr>
<tr>
<td>set dns proxy</td>
<td>34</td>
</tr>
<tr>
<td>set dns mode</td>
<td>34</td>
</tr>
<tr>
<td>set domainname</td>
<td>34</td>
</tr>
<tr>
<td>set dynamic-dns</td>
<td>34</td>
</tr>
<tr>
<td>Managing dynamic DNS details</td>
<td>34</td>
</tr>
<tr>
<td>set expert password</td>
<td>34</td>
</tr>
<tr>
<td>set external ports access</td>
<td>34</td>
</tr>
<tr>
<td>set host</td>
<td>34</td>
</tr>
<tr>
<td>set hostname</td>
<td>34</td>
</tr>
<tr>
<td>set inactivity-timeout</td>
<td>34</td>
</tr>
<tr>
<td>set interface</td>
<td>34</td>
</tr>
<tr>
<td>Managing Interfaces</td>
<td>34</td>
</tr>
<tr>
<td>Configuring Static IP</td>
<td>34</td>
</tr>
<tr>
<td>Configuring Advanced Interface Settings</td>
<td>34</td>
</tr>
<tr>
<td>Delete Internet Connection</td>
<td>33</td>
</tr>
<tr>
<td>Delete ICMP servers</td>
<td>33</td>
</tr>
<tr>
<td>delete net-obj</td>
<td>33</td>
</tr>
<tr>
<td>delete static-route</td>
<td>33</td>
</tr>
<tr>
<td>delete proxy</td>
<td>33</td>
</tr>
<tr>
<td>delete snmp</td>
<td>33</td>
</tr>
<tr>
<td>delete switch</td>
<td>33</td>
</tr>
<tr>
<td>delete user</td>
<td>33</td>
</tr>
<tr>
<td>delete wlan vap</td>
<td>33</td>
</tr>
<tr>
<td>dynamic objects</td>
<td>33</td>
</tr>
<tr>
<td>exit</td>
<td>33</td>
</tr>
<tr>
<td>fetch certificate</td>
<td>33</td>
</tr>
<tr>
<td>fetch license</td>
<td>33</td>
</tr>
<tr>
<td>fetch policy</td>
<td>33</td>
</tr>
<tr>
<td>fw Commands</td>
<td>33</td>
</tr>
<tr>
<td>reboot</td>
<td>33</td>
</tr>
<tr>
<td>restore default-settings</td>
<td>33</td>
</tr>
<tr>
<td>restore settings</td>
<td>33</td>
</tr>
<tr>
<td>revert to factory defaults</td>
<td>33</td>
</tr>
<tr>
<td>revert to saved image</td>
<td>33</td>
</tr>
<tr>
<td>set admin access</td>
<td>33</td>
</tr>
<tr>
<td>set antispam mode</td>
<td>33</td>
</tr>
<tr>
<td>set bridge</td>
<td>33</td>
</tr>
<tr>
<td>set date</td>
<td>33</td>
</tr>
<tr>
<td>set dhcp server</td>
<td>33</td>
</tr>
<tr>
<td>set dhcp relay</td>
<td>33</td>
</tr>
<tr>
<td>set dns</td>
<td>33</td>
</tr>
<tr>
<td>set dns proxy</td>
<td>33</td>
</tr>
<tr>
<td>set dns mode</td>
<td>33</td>
</tr>
<tr>
<td>set domainname</td>
<td>33</td>
</tr>
<tr>
<td>set dynamic-dns</td>
<td>33</td>
</tr>
<tr>
<td>Managing dynamic DNS details</td>
<td>33</td>
</tr>
<tr>
<td>set expert password</td>
<td>33</td>
</tr>
<tr>
<td>set external ports access</td>
<td>33</td>
</tr>
<tr>
<td>set host</td>
<td>33</td>
</tr>
<tr>
<td>set hostname</td>
<td>33</td>
</tr>
<tr>
<td>set inactivity-timeout</td>
<td>33</td>
</tr>
<tr>
<td>set interface</td>
<td>33</td>
</tr>
<tr>
<td>Managing Interfaces</td>
<td>33</td>
</tr>
<tr>
<td>Configuring Static IP</td>
<td>33</td>
</tr>
<tr>
<td>Configuring Advanced Interface Settings</td>
<td>33</td>
</tr>
</tbody>
</table>
Enabling/Disabling .................................................................................................................. 53
Configuring Static IP .................................................................................................................. 54
Configuring PPTP and L2TP ....................................................................................................... 54
Configuring PPPoE ...................................................................................................................... 55
Configuring PPPoA ...................................................................................................................... 56
Configuring EoA ........................................................................................................................ 57
Configuring VPI, VCI and Encapsulation .................................................................................. 57
Configuring Cellular Modem .................................................................................................... 58
Configuring ICMP ..................................................................................................................... 58
Configuring Monitoring ............................................................................................................ 59
Route Traffic .............................................................................................................................. 59
Connect on Demand .................................................................................................................. 59
Load Balancing ........................................................................................................................ 60
Configuring Advanced Connection Settings ......................................................................... 60
set internet mode ...................................................................................................................... 61
set nat ........................................................................................................................................ 61
set net-obj ................................................................................................................................ 61
Single IP .................................................................................................................................. 61
Network ................................................................................................................................... 62
IP Range ................................................................................................................................... 62
set proxy .................................................................................................................................... 62
Managing a Proxy Server ......................................................................................................... 62
Configuring a Proxy Server ...................................................................................................... 63
set ntp ....................................................................................................................................... 63
set snmp ..................................................................................................................................... 64
Managing SNMP Agent ........................................................................................................... 64
Setting SNMP Version ............................................................................................................. 64
Setting Community String ....................................................................................................... 64
Setting SNMP Host Information ............................................................................................. 64
Setting Host Location .............................................................................................................. 65
Managing SNMP Traps ............................................................................................................ 65
Setting SNMP v2 Receivers ..................................................................................................... 65
Configuring SNMP v3 Receivers ............................................................................................ 65
Configuring SNMP v3 Users ................................................................................................... 66
Setting a Single Trap ................................................................................................................ 66
set static-route ........................................................................................................................ 67
set property ............................................................................................................................... 68
set sic_init ................................................................................................................................. 68
set switch .................................................................................................................................. 69
set time ...................................................................................................................................... 69
set time-zone ............................................................................................................................ 69
set timezone-dst ....................................................................................................................... 70
set user ...................................................................................................................................... 70
Setting Password for a User .................................................................................................... 70
Setting Password Hash for a User ............................................................................................ 70
Setting Remote Access for a Local User ................................................................................ 70
Setting Comments for a Local User ......................................................................................... 71
Setting Permissions for an Admin User .................................................................................. 71
set wlan ..................................................................................................................................... 71
Enable/Disable Wireless Radio ................................................................................................. 71
Enable/Disable VAP .................................................................................................................. 72
Assign Wireless Network to a Switch/Bridge ......................................................................... 72
Set SSID ................................................................................................................................... 72
Wireless Connection Settings ................................................................................................. 72
Wireless Security ...................................................................................................................... 75
Advanced Wireless Settings .................................................................................................... 76
Set the Main WLAN .................................................................................................................. 76
Set VAP .................................................................................................................................... 76
shell/expert ............................................................................................................................... 77
show admin access ................................................................. 78
show backup settings .......................................................... 79
show bridge ........................................................................... 79
show clock ............................................................................ 79
show commands .................................................................... 80
show date .............................................................................. 80
show dhcp .............................................................................. 80
    Showing DHCP Settings .................................................. 80
    Showing DHCP for an Interface ...................................... 81
    Showing DHCP IP Pool ..................................................... 81
show diag .............................................................................. 81
show disk usage ..................................................................... 81
show dns .............................................................................. 82
show domainname ................................................................. 82
show dynamic-dns ................................................................. 83
show internet mode .............................................................. 83
show internet connections .................................................... 83
show host .............................................................................. 83
show hostname ..................................................................... 84
show internet connection icmp servers .................................. 84
show inactivity-timeout ........................................................ 84
show interface ...................................................................... 85
show interfaces .................................................................... 85
show license ......................................................................... 85
show logs .............................................................................. 85
show memory usage ............................................................. 86
show network object ............................................................ 86
show ntp .............................................................................. 86
    Show NTP Status ........................................................... 86
    Show NTP Servers .......................................................... 86
show proxy .......................................................................... 87
show restore settings log ...................................................... 87
show revert log ..................................................................... 87
show static-routes ............................................................... 87
show rule hits ....................................................................... 88
show saved image ............................................................... 88
show snmp ............................................................................ 88
    Show SNMP Agent .......................................................... 88
    Show SNMP Host Information ........................................ 89
    Show SNMP Trap Information ......................................... 89
    Show SNMP Users .......................................................... 89
show software version .......................................................... 90
show switch .......................................................................... 90
show time ............................................................................. 90
show timezone ..................................................................... 90
show timezone-dst ............................................................... 91
show upgrade log ............................................................... 91
show user ............................................................................ 91
show vpn tunnel ................................................................... 92
show wlan ........................................................................... 92
    Show wlan Settings ......................................................... 92
    Show VAPS .................................................................... 92
    Show Radio Configuration .............................................. 92
upgrade from usb or tftp server .............................................. 93
vpn ....................................................................................... 93
    Managing the VPN Driver .............................................. 93
    Launching TunnelUtil Tool .......................................... 93
    Debugging VPN ............................................................. 94
Chapter 1

CLI Reference

In This Section:

- Using Command Line Reference ................................................................. 12
- add admin access ...................................................................................... 14
- add antispam__allowed_sender ................................................................. 14
- add bridge ................................................................................................. 15
- add host ...................................................................................................... 15
- add interface .............................................................................................. 15
- add internet connection ........................................................................... 16
- add net-obj ................................................................................................. 20
- add snmp .................................................................................................... 22
- add static-route ......................................................................................... 23
- add switch ................................................................................................. 24
- add user ...................................................................................................... 25
- add wlan vap ............................................................................................ 26
- backup settings ......................................................................................... 26
- cpaprob ...................................................................................................... 27
- cpastop ........................................................................................................ 28
- cpinfo ......................................................................................................... 29
- cpshell ........................................................................................................ 29
- cpstart ....................................................................................................... 29
- cpstat .......................................................................................................... 30
- cpstop ........................................................................................................ 31
- cpwd_admin ............................................................................................... 31
- delete bridge ............................................................................................. 32
- delete dhcp ................................................................................................. 32
- delete dns .................................................................................................. 33
- delete domainname .................................................................................... 33
- delete host .................................................................................................. 33
- delete interface .......................................................................................... 34
- delete internet-connection ......................................................................... 34
- delete net-obj ............................................................................................. 34
- delete static-route ..................................................................................... 35
- delete proxy ............................................................................................... 35
- delete snmp ............................................................................................... 35
- delete switch ............................................................................................. 36
- delete user ................................................................................................. 36
- delete wlan vap ........................................................................................ 36
- dynamic objects ....................................................................................... 36
- exit .............................................................................................................. 37
- fetch certificate ......................................................................................... 38
- fetch license .............................................................................................. 38
- fetch policy ............................................................................................... 38
fw Commands ................................................................................................. 39
reboot ............................................................................................................. 39
restore default-settings ............................................................................... 40
restore settings ............................................................................................. 40
revert to factory defaults ............................................................................ 40
revert to saved image .................................................................................. 41
set admin access ........................................................................................... 41
set antispam mode ....................................................................................... 42
set bridge ....................................................................................................... 42
set date ........................................................................................................... 43
set dhcp server ............................................................................................. 43
set dhcp relay ............................................................................................... 48
set dns ............................................................................................................. 48
set dns proxy ................................................................................................. 49
set dns mode .................................................................................................. 49
set domainname .............................................................................................. 49
set dynamic-dns ............................................................................................ 49
set expert password ...................................................................................... 50
set external ports access ............................................................................. 51
set host .......................................................................................................... 51
set hostname ................................................................................................. 51
set inactivity-timeout ................................................................................... 52
set interface .................................................................................................. 52
set internet-connection ............................................................................... 53
set internet mode .......................................................................................... 61
set nat ............................................................................................................ 61
set net-obj ..................................................................................................... 61
set proxy ........................................................................................................ 62
set ntp ............................................................................................................. 63
set snmp ........................................................................................................ 64
set static-route ............................................................................................. 67
set property ................................................................................................... 68
set sic_init ..................................................................................................... 68
set switch ...................................................................................................... 69
set time .......................................................................................................... 69
set time-zone ............................................................................................... 69
set timezone-dst ......................................................................................... 70
set user .......................................................................................................... 70
set wlan ........................................................................................................ 71
shell/expert ................................................................................................... 77
sim .................................................................................................................. 77
show admin access ...................................................................................... 78
show backup settings .................................................................................... 79
show bridge ................................................................................................... 79
show clock .................................................................................................... 79
show commands ......................................................................................... 80
show date ..................................................................................................... 80
show dhcp ..................................................................................................... 80
show diag ..................................................................................................... 81
show disk usage ........................................................................................... 81
show dns ....................................................................................................... 82
You can make changes to your appliance with the WebUI or Command Line Interface (CLI). When using CLI note these aspects:

- The CLI default shell (clish) covers all the operations that are supported from the WebUI. It also supports auto-completion capabilities, similar to Gaia. For advanced operations that require direct access to the file system (such as redirecting debug output to a file), log in to Expert mode.

- SSH to the appliance is supported and is enabled through the WebUI.

- You can enable login directly to expert mode. To do this:
  - Login to Expert mode using the "Expert" password.
  - Run the command `bashUser on`
  - You will now always login directly to expert mode (this mode is not deleted during reboot)
  - To turn this mode off, run the command `bashUser off`

- SCP to the appliance is supported but you need to enable direct login to Expert mode. Note that SFTP that is commonly used by winSCP is not supported. For more information, see sk52763 (http://supportcontent.checkpoint.com/solutions?id=sk52763).
**CLISH Auto-completion**

All CLISH commands support auto-completion. Standard Check Point and native Linux commands can be used from the CLISH shell but do not support auto-completion. These are examples of the different commands:

- **CLISH** - `fetch, set, show`
- **Standard Check Point** - `cphaprob, fw, vpn`
- **Native Linux** - `ping, tcpdump, traceroute`

**CLI Syntax**

The CLI commands are formatted according to these syntax rules.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text without brackets</td>
<td>Items you must type as shown</td>
</tr>
<tr>
<td><code>&lt;Text inside angle brackets&gt;</code></td>
<td>Placeholder for which you must supply a value</td>
</tr>
<tr>
<td><code>[Text inside square brackets]</code></td>
<td>Optional items</td>
</tr>
<tr>
<td>Vertical pipe (</td>
<td>)</td>
</tr>
<tr>
<td><code>{Text inside curly brackets}</code></td>
<td>Set of required items; choose one</td>
</tr>
<tr>
<td>Ellipsis (…)</td>
<td>Multiple values or parameters can be entered</td>
</tr>
</tbody>
</table>

**Running Clish Commands from Expert Mode**

You can run clish commands from Expert mode.

**Syntax**

```
clish [ -A -i { -c Cmd | -f File -v } -h -C ]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-c Cmd</code></td>
<td>Single command to execute</td>
</tr>
<tr>
<td><code>-f File</code></td>
<td>File to load commands from</td>
</tr>
<tr>
<td><code>-v</code></td>
<td>Verbose</td>
</tr>
<tr>
<td><code>-i</code></td>
<td>Ignore cmd failure in batch mode and continue</td>
</tr>
<tr>
<td><code>-A</code></td>
<td>Run as admin</td>
</tr>
<tr>
<td><code>-C</code></td>
<td>List available commands</td>
</tr>
<tr>
<td><code>-h</code></td>
<td>Help (this message)</td>
</tr>
</tbody>
</table>

**Note** - If the default shell you logged in with was clish, and then you logged in to expert from it, you cannot run clish from expert (running clish->expert->clish does not work, but expert->clish works).
**Supported Linux Commands**
These standard Linux commands are also supported by the Check Point 600 Appliance CLI.

- arp
- netstat
- nslookup
- ping
- resize
- sleep
- tcpdump
- top
- traceroute
- uptime

### add admin access

**Description**
Adds a specific IPv4 address or a network IPv4 address from which the administrator can remotely access the appliance.

**Syntax**
```
add admin-access-ipv4-address
{single-ipv4-address|network-ipv4-address} <ip_addr> {subnet-mask <netmask>|mask-length <mask_length>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address</td>
</tr>
<tr>
<td>&lt;mask_length&gt;</td>
<td>Interface mask length, a value between 1 - 32</td>
</tr>
<tr>
<td>&lt;netmask&gt;</td>
<td>Interface IPv4 address subnet mask</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
add admin-access-ipv4-address network-ipv4-address 1.1.1.1 subnet-mask 255.255.255.0
```

**Output**
Success shows OK. Failure shows an appropriate error message.

### add antispam__allowed_sender

**Description**
Adds a new Anti-Spam exception.

**Syntax**
```
add antispam__allowed_sender {domain <domain>|ipv4_addr <ip_addr>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address</td>
</tr>
<tr>
<td>&lt;domain&gt;</td>
<td>Domain address</td>
</tr>
</tbody>
</table>

**Example**
```
add antispam__allowed_sender ipv4_addr 3.3.3.3
```

**Output**
Failure shows an appropriate error message.
add bridge

Description: Adds a new bridge.

Syntax: `add bridge [name <name>]`

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>Bridge name</td>
</tr>
</tbody>
</table>

Example: `add bridge`

Output: Failure shows an appropriate error message.

add host

Description: Adds a static host named <host> and IP address <ip_addr>.

Syntax: `add host name <host> ipv4-address <ip_addr> [exclude-from-dhcp {off|on} [exclude-from-dhcp {off | on} reserve-mac-address <mac_addr>]]`

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;host&gt;</td>
<td>The host name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>The host IPv4 address format</td>
</tr>
<tr>
<td>&lt;mac_addr&gt;</td>
<td>default or MAC address format, 00:1C:7F:21:05:BE</td>
</tr>
</tbody>
</table>

Return Value: 0 on success, 1 on failure

Example: `add host name John ipv4-address 1.1.1.1`

Output: Failure shows an appropriate error message.

add interface

Description: Adds VLAN <vlan> to interface <interface>.

Syntax: `add interface <interface> vlan <vlan>`

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;vlan&gt;</td>
<td>VLAN name - a value between 1 and 4094.</td>
</tr>
</tbody>
</table>

Return Value: 0 on success, 1 on failure

Example: `add interface LAN4 vlan 1`

Output: Failure shows an appropriate error message.
add internet connection

Manages the internet connection.

**Configuring Static IP**

**Description** The `add internet-connection` command can set parameters for different static IP interface types.

**Syntax**
```
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>New connection name</td>
</tr>
<tr>
<td>&lt;vlan-id&gt;</td>
<td>To add a virtual Internet interface, set a vlan-id. Set X to a value between 1 and 4094.</td>
</tr>
<tr>
<td>&lt;conn_time&gt;</td>
<td>Number of seconds before connection test timeout. A number between 0 and 999. A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
<tr>
<td>&lt;ip_mask&gt;</td>
<td>IP address for subnet mask</td>
</tr>
<tr>
<td>&lt;mask_length&gt;</td>
<td>Mask length</td>
</tr>
</tbody>
</table>

**Example**
- add internet-connection interface WAN type static ipv4-address 1.1.1.2 subnet-mask 255.255.255.0 default-gw 1.1.1.1 dns-primary 2.2.2.2 dns-secondary 3.3.3.3
- add internet-connection interface WAN type static ipv4-address 1.1.1.7 subnet-mask 255.255.255.0 default-gw 1.1.1.1 conn-test-timeout 1
- add internet-connection interface WAN use-connection-as-vlan vlan-id 1 type static ipv4-address 1.1.1.7 subnet-mask 255.255.255.0 default-gw 1.1.1.1 conn-test-timeout 1

**Output** Failure shows an appropriate error message.

**Configuring DHCP**

**Description** The `add internet-connection` command can set parameters for the DHCP internet interface type.

Obtains IP automatically using DHCP, can be set only for internet interfaces.

**Syntax**
```
add internet-connection [name <conn_name>] interface {WAN|DMZ}[use-connection-as-vlan vlan-id <vlan-id>] type dhcp [conn-test-timeout <conn_time>]
```
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;conn_time&gt;</td>
<td>Number of seconds before connection test timeout. A number between 0 and 999. A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
<tr>
<td>&lt;vlan-id&gt;</td>
<td>To add a virtual Internet interface, set a vlan-id. Set X to a value between 1 and 4094.</td>
</tr>
<tr>
<td>&lt;connection_name&gt;</td>
<td>New connection name.</td>
</tr>
</tbody>
</table>

### Example

```
add internet-connection interface WAN use-connection-as-vlan vlan-id 1 type dhcp conn-test-timeout 5
```

### Output

Failure shows an appropriate error message.

---

**Configuring PPTP and L2TP**

### Description

The `add internet-connection` command can set parameters for these Internet interface types:

- **PPTP**
- **L2TP**

The parameters include: user name `<user>`, password `<pass>`, and server `<server>`.

### Syntax

```
add internet-connection [name <connection_name>] interface (DMZ|WAN) [use-connection-as-vlan vlan-id <vlan-id>] type {pptp|l2tp} username <user> password <pass> server <server> [local-ipv4-address <auto|ip_addr>] [wan-ipv4-address <auto|<ip_addr>] [wan-subnet-mask <ip_mask>|wan-mask-length <mask-length>] default-gw <ip_addr>] [conn-test-timeout <conn_time>]
```
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>New connection name</td>
</tr>
<tr>
<td><code>&lt;vlan-id&gt;</code></td>
<td>To add a virtual Internet interface, set a vlan-id. Set X to a value between 1 and 4094.</td>
</tr>
<tr>
<td><code>&lt;ip_addr&gt;</code></td>
<td>IPv4 address format.</td>
</tr>
<tr>
<td><code>&lt;ip_mask&gt;</code></td>
<td>IP address for subnet mask</td>
</tr>
<tr>
<td><code>&lt;mask_length&gt;</code></td>
<td>Mask length</td>
</tr>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>ISP user login name.</td>
</tr>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><code>&lt;server&gt;</code></td>
<td>Server host name or IP address.</td>
</tr>
<tr>
<td><code>local-ipv4-address</code></td>
<td>Local tunnel IPv4 address assignment.</td>
</tr>
<tr>
<td></td>
<td>auto - Get the IPv4 address from ISP.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;ip_addr&gt;</code> - IP address for local tunnel.</td>
</tr>
<tr>
<td><code>wan-ipv4-address</code></td>
<td>WAN IPv4 address assignment</td>
</tr>
<tr>
<td></td>
<td>auto - Get the WAN IPv4 address from ISP.</td>
</tr>
<tr>
<td></td>
<td>This is the default setting.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;ip_addr&gt;</code> - IP address for WAN port.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;ip_mask&gt;</code> - IP address for subnet mask for WAN port.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;mask_length&gt;</code> - Mask length for WAN port.</td>
</tr>
<tr>
<td></td>
<td><code>default-gw</code> - Default gateway for WAN port.</td>
</tr>
<tr>
<td><code>&lt;conn_time&gt;</code></td>
<td>Number of seconds before connection test timeout. A number between 0 and 999.</td>
</tr>
<tr>
<td></td>
<td>A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
</tbody>
</table>

### Example

- add internet-connection interface WAN type pptp username John password verySecurePass server 1.1.1.1 local-ipv4-address 2.2.2.2 wan-ipv4-address 3.3.3.3 wan-subnet-mask 255.255.255.0 default-gw 4.4.4.4

- add internet-connection interface DMZ use-connection-as-vlan vlan-id 1 type l2tp username John password verySecurePass server 1.1.1.1 local-ipv4-address 2.2.2.2

### Output

Failure shows an appropriate error message.

### Configuring PPPoE

#### Description

Configures WAN, DMZ, and ADSL PPPoE interface type settings with user name `<user>` and password `<pass>`.

#### Syntax

```
add internet-connection [name <connection_name>] interface {WAN|DMZ|ADSL} [use-connection-as-vlan vlan-id <vlan-id>] type pppoe username <user> password <pass> [conn-test-timeout <conn_time>]
```
### CLI Reference

#### Check Point 1100 and 600 Appliance Administration Guide CLI and Advanced Routing

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>&lt;user&gt;</strong></td>
<td>ISP user login name.</td>
</tr>
<tr>
<td><strong>&lt;pass&gt;</strong></td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><strong>&lt;conn_time&gt;</strong></td>
<td>Number of seconds before connection test timeout. A number between 0 and 999. A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
<tr>
<td><strong>&lt;vlan-id&gt;</strong></td>
<td>To add a virtual Internet interface, set a vlan-id. Set X to a value between 1 and 4094. This is applicable only for WAN or DMZ interfaces.</td>
</tr>
<tr>
<td><strong>&lt;connection_name&gt;</strong></td>
<td>New connection name.</td>
</tr>
</tbody>
</table>

#### Example

- add internet-connection interface ADSL type pppoe username Alex password myPassword
- add internet-connection interface use-connection-as-vlan vlan-id 1 DMZ type pppoe username Alex password myPassword

#### Output

Failure shows an appropriate error message.

---

### Configuring PPPoA

#### Description

Configures ADSL PPPoA interface types settings with user name `<user>` and password `<pass>`.

#### Syntax

```
add internet-connection [name <connection_name>] interface ADSL type pppoa username <user> password <pass> [conn-test-timeout <conn_time>]
```

#### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>&lt;user&gt;</strong></td>
<td>ISP user login name.</td>
</tr>
<tr>
<td><strong>&lt;pass&gt;</strong></td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><strong>&lt;conn_time&gt;</strong></td>
<td>Number of seconds before connection test timeout. A number between 0 and 999. A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
<tr>
<td><strong>&lt;connection_name&gt;</strong></td>
<td>New connection name.</td>
</tr>
</tbody>
</table>

#### Example

```
add internet-connection interface ADSL type pppoa username Alex password myPassword
```

#### Output

Failure shows an appropriate error message.
Configuring EoA

Description
Configures the ADSL EoA interface type with default settings.

Syntax
```
add internet-connection [name <connection_name>] interface ADSL type eoa [conn-test-timeout <conn_time>]
```

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;conn_time&gt;</td>
<td>Number of seconds before connection test timeout. A number between 0 and 999.</td>
</tr>
<tr>
<td></td>
<td>A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
<tr>
<td>&lt;connection_name&gt;</td>
<td>New connection name</td>
</tr>
</tbody>
</table>

Example
```
add internet-connection interface ADSL type eoa
```

Output
Failure shows an appropriate error message.

Configuring Cellular Modem

Description
Adds a new cellular internet connection.

Syntax
```
add internet-connection [name <connection_name>] type cellular number <dialedNumber> [username <user> password <pass>] [apn <apn>] [conn-test-timeout <conn_time>]
```

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>New connection name</td>
</tr>
<tr>
<td>&lt;dialedNumber&gt;</td>
<td>ISP dialed number. Default is *99#</td>
</tr>
<tr>
<td>&lt;user&gt;</td>
<td>ISP user login name.</td>
</tr>
<tr>
<td>&lt;pass&gt;</td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>&lt;apn&gt;</td>
<td>IPS access point name.</td>
</tr>
<tr>
<td>&lt;conn_time&gt;</td>
<td>Number of seconds before connection test timeout. A number between 0 and 999.</td>
</tr>
<tr>
<td></td>
<td>A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
</tbody>
</table>

Example
```
add internet-connection type cellular number *99# username Johnny password myPassword
```

Output
Failure shows an appropriate error message.

add net-obj

Create network objects to use in the appliance's feature configuration.
**Single IP**

**Description**: Adds a new IP network object.

**Syntax**
```
add net-obj name <name> type single ipv4-address <ip_addr>
[dns-resolving {true|false}] [exclude-from-dhcp {off|on} on
[mac-reserved-in-dhcp
{off|on reserve-mac-address <mac_addr>}]}}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>Network object name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Valid IPv4 address</td>
</tr>
<tr>
<td>dns-resolving</td>
<td>Allow DNS server to resolve this object name</td>
</tr>
<tr>
<td>&lt;mac_addr&gt;</td>
<td>MAC address format</td>
</tr>
</tbody>
</table>

**Example**
- add net-obj name NetObj1 type single ipv4-address 192.168.20.3
dns-resolving true
- add net-obj name NetObj2 type single ipv4-address 192.168.20.4
  exclude-from-dhcp on

**Output**
Failure shows an appropriate error message.

**Network**

**Description**: Adds a new network object.

**Syntax**
```
add net-obj name <name> type network ipv4-network-address <ip_addr>
netmask {subnet-mask <ip_mask>|mask-length <mask-length>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>Network object name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Valid IPv4 address</td>
</tr>
<tr>
<td>dns-resolving</td>
<td>Allow DNS server to resolve this object name</td>
</tr>
<tr>
<td>&lt;mask&gt;</td>
<td>Subnet mask</td>
</tr>
<tr>
<td>&lt;mask_length&gt;</td>
<td>Mask length</td>
</tr>
</tbody>
</table>

**Example**
add net-obj name NetObj1 type network ipv4-network-address 3.3.3.0
  netmask mask-length 24

**Output**
Failure shows an appropriate error message.

**IP Range**

**Description**: Adds a new IP range network object.

**Syntax**
```
add net-obj name <name> type range start-ipv4 <ip_addr> end-ipv4
  <ip_addr> [exclude-from-dhcp {on|off}]
```

Check Point 1100 and 600 Appliance Administration Guide CLI and Advanced Routing
add SNMP

Add SNMP related parameters.

Adding SNMP v2 Traps Receiver

Description
Adds SNMPv2 traps receiver, <comm_string> is used for SNMP security and authentication.

Syntax
add snmp traps-receiver <ip_addr> version v2 community <comm_string>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Trap receiver IPv4 address.</td>
</tr>
<tr>
<td>&lt;comm_string&gt;</td>
<td>A password for v1 and v2 protocols. The value can be any word.</td>
</tr>
</tbody>
</table>

Example
add snmp traps-receiver 1.1.1.1 version v2 community abcd

Output
Failure shows an appropriate error message.

Adding SNMP v3 Traps Receiver

Description
Adds SNMPv3 traps receiver, security parameters that are defined for the <v3_user> are used.

Syntax
add snmp traps-receiver <ip_addr> version v3 usm user <v3_user>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Trap receiver IPv4 address.</td>
</tr>
<tr>
<td>&lt;v3_user&gt;</td>
<td>A string representing the name of the user to add.</td>
</tr>
</tbody>
</table>

Example
add snmp traps-receiver 1.1.1.1 version v3 user usm1

Output
Failure shows an appropriate error message.

Adding SNMP User

Description
Adds USM security user parameters for the <v3_user>

Syntax
add snmp user <user_name> security-level {authNoPriv|authPriv} auth-pass-type {MD5|SHA1} auth-pass-phrase <auth-phrase> privacy-pass-type {AES|DES} privacy-pass-phrase <priv-phrase>
Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;user_name&gt;</td>
<td>Name user that was added with add snmp.</td>
</tr>
<tr>
<td>security-level</td>
<td>Messages sent or received by this user are authenticated using the privacy and authentication passwords. Values for this field are: authNoPriv and authPriv.</td>
</tr>
<tr>
<td>auth-pass-type</td>
<td>Authentication decryption protocol. Available values for this field are: MD5 and SHA1.</td>
</tr>
<tr>
<td>&lt;auth-phrase&gt;</td>
<td>The localized secret key used by the authentication protocol for authenticating messages.</td>
</tr>
<tr>
<td>privacy-pass-type</td>
<td>Which privacy decryption protocol to use. Available values for this field are: AES and DES.</td>
</tr>
<tr>
<td>&lt;priv-phrase&gt;</td>
<td>The localized secret key used by the privacy protocol for encrypting and decrypting messages.</td>
</tr>
</tbody>
</table>

Example

add snmp user usm1 security-level authPriv auth-pass-type SHA1 auth-pass-phrase safeAuthPassPhrase privacy-pass-type AES privacy-pass-phrase safePrivacyPassPhrase

Output

Failure shows an appropriate error message.

add static-route

Use the add static-route command to create new static routes. If a priority is not specified for the route, a default value of zero is used.

Adding a Route with a Specific Gateway IP Address

Description

Adds a route with a destination <route_IP>, source <route_IP> and with gateway <gw_IP>. The rule is assigned the default priority of <metric>.

Syntax

add static-route [destination <route_IP>] [source <route_IP>] [service <service_name>] nexthop gateway ipv4-address <gw_IP> [metric <metric>]

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;route_IP&gt;</td>
<td>IP address and subnet bit number of the route. &lt;IPv4-address&gt;/&lt;Subnet-bit-number&gt;</td>
</tr>
<tr>
<td>&lt;service_name&gt;</td>
<td>The service protocol which to whom the route rule applies to. The default is to all protocols.</td>
</tr>
<tr>
<td>&lt;gw_IP&gt;</td>
<td>Gateway IP address.</td>
</tr>
<tr>
<td>&lt;metric&gt;</td>
<td>Priority (metric) of the route.</td>
</tr>
</tbody>
</table>

Example

add static-route destination 172.15.47.0/24 nexthop gateway ipv4-address 10.0.0.1
Adding a Route with a Specific Interface and Priority

Description
Adds a route with a destination `<route_IP>`, source `<route_IP>` and a gateway interface of `<interface>`. The rule will be assigned the default priority of `<metric>`.

Syntax
```
add static-route [destination <route_IP>] [source <route_IP>] [service <service_name>] nexthop gateway logical <interface> [metric <metric>]
```

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;route_IP&gt;</code></td>
<td>IP address and subnet bit number of the route. IPv4-address/Subnet-bit-number</td>
</tr>
<tr>
<td><code>&lt;service_name&gt;</code></td>
<td>The service protocol which to whom the route rule applies to. The default is to all protocols.</td>
</tr>
<tr>
<td><code>&lt;interface&gt;</code></td>
<td>Interface to which the gateway is connected.</td>
</tr>
<tr>
<td><code>&lt;metric&gt;</code></td>
<td>Priority (metric) of the route.</td>
</tr>
</tbody>
</table>

Return Value
0 on success, 1 on failure

Example
```
add static-route destination 172.15.47.0/24 nexthop gateway logical Internet1
```

Output
Failure shows an appropriate error message.

add switch

Description
Adds an interface `<interface>` to a LAN switch. If the LAN switch does not exist, it is created and inherits all settings from the LAN1 interface.

Syntax
```
add switch name <interface>
```

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;interface&gt;</code></td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

Return Value
0 on success, 1 on failure

Example
```
add switch name LAN4
```

Output
Failure shows an appropriate error message.

Comments
The interface that is added to the switch must be unassigned. When executing the command on an interface that has an IP address assigned to it, the following error message is printed: `Error: <interface> port has static IP address assigned.`

LAN1 is always a part of LAN switch.
add user

Adds a new local user or administrator user with two password options: standard and MD5 encrypted.

**Adding Local User**

**Description**
Adds a new local user named `<user>` with specified password `<pass>` or `<pass_hash>` and Remote Access permissions `<is_remote-access-always-on>`

**Syntax**
```
add user <user> type local {password <pass>| password-hash <pass_hash>} remote-access <is_remote-access-always-on> [comments <comments>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>User login name</td>
</tr>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>User password. Alphanumeric and special characters are allowed</td>
</tr>
<tr>
<td><code>&lt;remote-access-always-on&gt;</code></td>
<td>Always Enable Remote Access (RA permissions are defined for this user specifically). Should be true/false.</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
add user John type local password extremelySafePassword
```

**Output**
Failure shows an appropriate error message.

**Adding Administrator User**

**Description**
Adds a new administrator user named `<user>` with specified password `<pass>` or `<pass_hash>`. You can optionally set read-only permission for the administrator.

**Syntax**
```
add user <user> type admin {password <pass>| password-hash <pass_hash>} [permission R|RW]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>User login name</td>
</tr>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>User password. Alphanumeric and special characters are allowed</td>
</tr>
<tr>
<td><code>&lt;pass_hash&gt;</code></td>
<td>User password, MD5 string representation</td>
</tr>
<tr>
<td>R</td>
<td>Sets read-only permission.</td>
</tr>
<tr>
<td>RW</td>
<td>Sets read/write permission.</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
add user myAdmin type admin password extremelySafePassword permission R
```
add wlan vap

**Description**
Adds a VAP with name (SSID) `<ssid>`.

**Syntax**
```
add wlan vap ssid <ssid>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;ssid&gt;</code></td>
<td>Network name (SSID)</td>
</tr>
</tbody>
</table>

**Example**
```
add wlan vap ssid Guest_Wireless
```

**Output**
Failure shows an appropriate error message.

**Comments**
The VAP is created disabled.

backup settings

**Description**
Creates a backup file that contains the current settings for the appliance and saves them to a file. The file is saved to either a USB device or TFTP server. You can use these options when the backup file is created:
- Specific file name (The default file name contains the current image and a date and time stamp)
- Password encryption
- Backup policies
- Add a comment to the file

**Syntax**
```
backup settings to {usb|tftp server <serverIP>} [filename <filename>] [file-encryption {off|on password <pass>}] [backup-policy {on|off}] [add-comment <comment>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;comment&gt;</code></td>
<td>Comment that is added to the file.</td>
</tr>
<tr>
<td><code>&lt;filename&gt;</code></td>
<td>Name of the backup file.</td>
</tr>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>Password for the file. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><code>&lt;serverIP&gt;</code></td>
<td>IPv4 address of the TFTP server.</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
backup settings to usb file-encryption on password admin backup-policy on add-comment check_point_new_configuration
```

**Output**
Success prints OK. Failure shows an appropriate error message.

**Comments**
When saving the backup file to a USB device, the backup settings command fails if there are two USB devices connected to the appliance.
cphaprob

Description
Defines and manages the critical cluster member properties of the appliance. When a critical process fails, the appliance is considered to have failed.

Syntax
```
cphaprob [-i[a]] [-d <device>] [-s {ok|init|problem}] [-f <file>] [-p] [register|unregister|report|list|state|if]
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>register</td>
<td>Registers &lt;appliance&gt; as a critical process</td>
</tr>
<tr>
<td>-a</td>
<td>Lists all devices in the cluster</td>
</tr>
<tr>
<td>-d &lt;device&gt;</td>
<td>The name of the device as it appears in the output of the cphaprob list</td>
</tr>
<tr>
<td>-p</td>
<td>The configuration change is permanent and applies after the appliance reboots.</td>
</tr>
<tr>
<td>-t &lt;timeout&gt;</td>
<td>If &lt;device&gt; fails to contact ClusterXL in &lt;timeout&gt; seconds, &lt;device&gt; is considered to have failed. To disable this parameter, enter the value 0.</td>
</tr>
<tr>
<td>-s</td>
<td>Status to be reported. ok – &lt;appliance&gt; is alive init – &lt;appliance&gt; is initializing problem – &lt;appliance&gt; has failed</td>
</tr>
<tr>
<td>-f &lt;file&gt;</td>
<td>Option to automatically register several appliances. The file defined in the &lt;file&gt; field should contain the list of appliances with these parameters: &lt;device&gt; &lt;timeout&gt; Status</td>
</tr>
<tr>
<td>register</td>
<td>Unregisters &lt;device&gt; as a critical process.</td>
</tr>
<tr>
<td>report</td>
<td>Reports the status of the &lt;device&gt; to the gateway.</td>
</tr>
<tr>
<td>list</td>
<td>Displays that state of: -i – Internal (as well as external) devices, such as interface check and HA initialization. -e – External devices, such as devices registered by the user or outside the kernel. For example, fwd, sync, filter. -ia – All devices, including those used for internal purposes, such as note initialization and load-balance configuration.</td>
</tr>
<tr>
<td>state</td>
<td>Displays the state of all the gateways in the High Availability configuration.</td>
</tr>
<tr>
<td>if</td>
<td>Displays the state of interfaces.</td>
</tr>
</tbody>
</table>

Example
```
cphaprob -d $process -t 0 -s ok -p register
```
These are some typical scenarios for the cphaprob command.

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cphaprob -d &lt;device&gt; -t &lt;timeout(sec)&gt; -s &lt;ok</td>
<td>init</td>
</tr>
<tr>
<td>cphaprob -f &lt;file&gt; register</td>
<td>Register all the user defined critical devices listed in &lt;file&gt;.</td>
</tr>
<tr>
<td>cphaprob -d &lt;device&gt; [-p] unregister</td>
<td>Unregister a user defined &lt;device&gt; as a critical process. This means that this device is no longer considered critical.</td>
</tr>
<tr>
<td>cphaprob -a unregister</td>
<td>Unregister all the user defined &lt;device&gt;.</td>
</tr>
<tr>
<td>cphaprob -d &lt;device&gt; -s &lt;ok</td>
<td>init</td>
</tr>
<tr>
<td>cphaprob [-i[a]] [-e] list</td>
<td>View the list of critical devices on a cluster member, and of all the other machines in the cluster.</td>
</tr>
<tr>
<td>cphaprob state</td>
<td>View the status of a cluster member, and of all the other members of the cluster.</td>
</tr>
<tr>
<td>cphaprob [-a] if</td>
<td>View the state of the cluster member interfaces and the virtual cluster interfaces.</td>
</tr>
</tbody>
</table>

### cphastop

**Description**

Disables High Availability on the appliance. Running cphastop on an appliance that is a cluster member stops the appliance from passing traffic. State synchronization also stops.

**Syntax**

cphastop

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

**Example**

cphastop

**Output**

Success prints OK. Failure shows an appropriate error message.
cpinfo

Description: Creates a Check Point Support Information (CPinfo) file on a machine at the time of execution.

The file is saved to a USB drive or TFTP server.

The CPinfo output file enables Check Point's support engineers to analyze setups from a remote location.

Syntax: cpinfo {to-tftp <ipaddr>|to-usb}

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ipaddr&gt;</td>
<td>IPv4 address</td>
</tr>
</tbody>
</table>

Return Value: 0 on success, 1 on failure

Example: cpinfo to-usb

Output: Success prints Creating cpinfo.txt file. Failure shows an appropriate error message.

cpshell

Description: Starts cpshell.

Syntax: cpshell

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Return Value: 0 on success, 1 on failure

Example: cpshell

Output: None

Comments: Use the shell command to change to expert mode.

cpstart

Start all Check Point processes and applications running on a machine.

Description: Starts firewall services.

Syntax: cpstart

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Return Value: 0 on success, 1 on failure

Example: cpstart

Output: Success shows Starting CP products... Failure shows an appropriate error message.
# cpstat

## Description

Shows Check Point statistics for applications.

## Syntax

```
```

## Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-p &lt;port&gt;</td>
<td>Port number of the server. The default is the standard server port (18192).</td>
</tr>
<tr>
<td>-s &lt;SICname&gt;</td>
<td>Secure Internal Communication (SIC) name of the server.</td>
</tr>
<tr>
<td>-f &lt;flavor&gt;</td>
<td>The flavor of the output (as it appears in the configuration file). The default is the first flavor found in the configuration file.</td>
</tr>
<tr>
<td>-o &lt;polling&gt;</td>
<td>Polling interval (seconds) specifies the pace of the results. The default is 0, meaning the results are shown only once.</td>
</tr>
<tr>
<td>-c &lt;count&gt;</td>
<td>Specifies how many times the results are shown. The default is 0, meaning the results are repeatedly shown.</td>
</tr>
<tr>
<td>-e &lt;period&gt;</td>
<td>Specifies the interval (seconds) over which ‘statistical’ olds are computed. Ignored for regular olds.</td>
</tr>
<tr>
<td>-x</td>
<td>XML output mode</td>
</tr>
<tr>
<td>-j</td>
<td>Json output mode</td>
</tr>
<tr>
<td>-d</td>
<td>Debug mode.</td>
</tr>
<tr>
<td>&lt;flag&gt;</td>
<td>One of these applications is displayed: One of the following: fw — Firewall component of the Security Gateway vpn — VPN component of the Security Gateway fg — QoS (formerly FloodGate-1) ha — ClusterXL (High Availability) os — OS Status mg — for the Security Management server persistency - for historical status values polsrv uas svr cpsemd cpsead asm ls ca</td>
</tr>
</tbody>
</table>

## Return Value

0 on success, 1 on failure
The following flavors can be added to the application flags:

- **vpn** — "default", "product", "IKE", "ipsec", "traffic", "compression", "accelerator", "nic", "statistics", "watermarks", "all"
- **fg** — "all"
- **ha** — "default", "all"
- **mg** — "default"
- **persistency** — "product", "Tableconfig", "SourceConfig"
- **polsrv** — "default", "all"
- **uas** — "default"
- **svr** — "default"
- **cpsemd** — "default"
- **cpsead** — "default"
- **asm** — "default", "WS"
- **ls** — "default"
- **ca** — "default", "crl", "cert", "user", "all"

**cpstop**

**Description**

Stops firewall services and terminates all Check Point processes and applications running on the appliance.

**Syntax**

`cpstop`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

**Example**

`cpstop`

**Output**

Success shows **Uninstalling Security Policy**... Failure shows an appropriate error message.

**cpwd_admin**

**Description**

The **cpwd_admin** utility can be used to verify if a process is running and to stop and start a process if necessary.
Syntax:
```
cpwd_admin {del <name>|detach <name>|list|kill|exist|start_monitor|stop_monitor|monitor_list}
```

### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>del</td>
<td>Deletes process</td>
</tr>
<tr>
<td>detach</td>
<td>Detaches process</td>
</tr>
<tr>
<td>list</td>
<td>Print status of processes</td>
</tr>
<tr>
<td>kill</td>
<td>Stops cpWatchDog</td>
</tr>
<tr>
<td>exist</td>
<td>Checks if cpWatchDog is running</td>
</tr>
<tr>
<td>start_monitor</td>
<td>cpwd starts monitoring this machine</td>
</tr>
<tr>
<td>stop_monitor</td>
<td>cpwd stops monitoring this machine</td>
</tr>
<tr>
<td>monitor_list</td>
<td>Displays list of monitoring processes</td>
</tr>
<tr>
<td>&lt;name&gt;</td>
<td>Name of process</td>
</tr>
</tbody>
</table>

### Return Value

0 on success, 1 on failure

### Example

```
cpwd_admin start_monitor
```

### Output

Success shows `OK`. Failure shows an appropriate error message.

---

### delete bridge

**Description**  
Deletes the bridge named `<bridge_name>`.

**Syntax**  
```
delete bridge <bridge_name>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;bridge_name&gt;</td>
<td>Name of the bridge</td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

**Example**  
```
delete bridge br1
```

**Output**

Success shows `OK`. Failure shows an appropriate error message.

---

### delete dhcp

**Deleting Excluded IP Addresses**

**Description**  
Deletes IP address exclude range that was defined for interface `<interface>`.

**Syntax**  
```
delete dhcp server interface <interface> exclude-range
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure
Example delete dhcp server interface LAN2 exclude-range

Output Failure shows an appropriate error message.

delete dns

Description Deletes the specified DNS (Domain Name Server) server. The secondary DNS server is used when the primary DNS server does not respond. The tertiary DNS server is used when the primary and secondary DNS servers do not respond.

Syntax delete dns {primary|secondary|tertiary} ipv4-address

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Return Value 0 on success, 1 on failure

Example delete dns secondary ipv4-address

Output Failure shows an appropriate error message.

delete domainname

Description Removes the domain name of the system.

Syntax delete domainname

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Return Value 0 on success, 1 on failure

Example delete domainname

Output Failure shows an appropriate error message.

Comments To set a domain name for the system, use the set domainname command.

delete host

Description Deletes the static host named <host>.

Syntax delete host name <host> ipv4

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;host&gt;</td>
<td>Name of the static host</td>
</tr>
</tbody>
</table>

Return Value 0 on success, 1 on failure

Example delete host name cnn.com ipv4

Output Success shows OK. Failure shows an appropriate error message.
delete interface

**Description**
Deletes the VLAN/VAP named `<interface>`.

**Syntax**
delete interface `<interface>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;interface&gt;</code></td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
delete interface DMZ:2

**Output**
Failure shows an appropriate error message.

delete internet-connection

**Delete Internet Connection**

**Description**
Deletes the internet connection `<connection_name>`.

**Syntax**
delete internet-connection `<connection_name>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Valid connection name</td>
</tr>
</tbody>
</table>

**Example**
Delete internet-connection internet1

**Output**
Failure shows an appropriate error message.

Delete ICMP servers

**Description**
Deletes the ICMP server from the internet connection `<connection_name>`.

**Syntax**
delete internet-connection `<connection_name>` probe-icmp-servers {first|second|third}

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Valid connection name</td>
</tr>
</tbody>
</table>

**Example**
delete internet-connection Internet1 probe-icmp-servers second

**Output**
Failure shows an appropriate error message.

delete net-obj

**Description**
Deletes a route with a name `<name>`.

**Syntax**
delete net-obj `<name>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;name&gt;</code></td>
<td>Network object name.</td>
</tr>
</tbody>
</table>

**Example**
delete net-obj NetObj1
**delete static-route**

**Description**
Deletes a route with a route ID `<route_id>`.

**Syntax**
delete static-route `<route_id>`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;route_id&gt;</code></td>
<td>Route ID</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
</table>
delete static-route 1

<table>
<thead>
<tr>
<th>Output</th>
</tr>
</thead>
</table>
Failure shows an appropriate error message.

**delete proxy**

**Description**
Deletes a proxy server.

**Syntax**
delete proxy

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Return Value</th>
</tr>
</thead>
</table>
0 on success, 1 on failure

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
</table>
delete proxy

<table>
<thead>
<tr>
<th>Output</th>
</tr>
</thead>
</table>
Failure shows an appropriate error message.

**delete snmp**

**Description**
Deletes these SNMP parameters:
- SNMP trap receiver
- SNMP contact information
- SNMP location
- SNMP v3 user

**Syntax**
delete snmp {traps-receiver <ip_addr>|contact|location|user <v3_user>}

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;ip_addr&gt;</code></td>
<td>Trap receiver IPv4 address.</td>
</tr>
<tr>
<td><code>&lt;v3_user&gt;</code></td>
<td>A string representing the name of the user to delete.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Return Value</th>
</tr>
</thead>
</table>
0 on success, 1 on failure

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
</table>
delete snmp trap-receiver 1.1.1.1
delete snmp user usm1

<table>
<thead>
<tr>
<th>Output</th>
</tr>
</thead>
</table>
Failure shows an appropriate error message.
**delete switch**

**Description**
Deletes the switch named `<switch_name>`.

**Syntax**
delete switch `<switch_name>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;switch_name&gt;</code></td>
<td>Valid switch name.</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
delete switch MySwitch

**Output**
Failure shows an appropriate error message.

**Comments**
When executing the command, the port LAN1 inherits the LAN switch configuration.

**delete user**

**Description**
Deletes an existing user with login name `<user>`.

**Syntax**
delete user `<user>` type {admin|local}

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>Login name of user</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
delete user John type local

**Output**
Failure shows an appropriate error message.

**delete wlan vap**

**Description**
Deletes an existing VAP with SSID `<vlan_name>`.

**Syntax**
delete wlan vap `<vlan_name>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;vlan_name&gt;</code></td>
<td>VLAN name</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
delete wlan vap Guest1

**Output**
Failure shows an appropriate error message.

**dynamic objects**

Manages dynamic objects on the appliance. The `dynamic_objects` command specifies an IP address to which the dynamic object is resolved.

First, define the dynamic object in the SmartDashboard. Then create the same object with the CLI (`-n` argument). After the new object is created on the gateway with the CLI, you can use the `dynamic_objects` command to specify an IP address for the object.
Any change you make to dynamic objects' ranges are applied immediately to the objects. It is not necessary to reinstall the policy.

**Description**
Manages dynamic objects on the appliance.

**Syntax**
dynamic_objects -o <object> [-r <fromIP> <toIP> ...] [-a] [-d] [-l] [-n <object>] [-c] [-do <object>]

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-o</td>
<td>Name of the dynamic object that is being configured</td>
</tr>
<tr>
<td>-r</td>
<td>Defines the range of IP addresses that are being configured for this object</td>
</tr>
<tr>
<td>-a</td>
<td>Adds range of IP addresses to the dynamic object</td>
</tr>
<tr>
<td>-d</td>
<td>Deletes range of IP addresses from the dynamic object</td>
</tr>
<tr>
<td>-l</td>
<td>Lists dynamic objects that are used on the appliance</td>
</tr>
<tr>
<td>-n</td>
<td>Creates a new dynamic object</td>
</tr>
<tr>
<td>-c</td>
<td>Compare the objects in the dynamic objects file and in objects.C.</td>
</tr>
<tr>
<td>-do</td>
<td>Deletes the dynamic object</td>
</tr>
<tr>
<td>&lt;object&gt;</td>
<td>Name of dynamic object</td>
</tr>
<tr>
<td>&lt;fromIP&gt;</td>
<td>Starting IPv4 address</td>
</tr>
<tr>
<td>&lt;toIP&gt;</td>
<td>Ending IPv4 address</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
dynamic_objects -n sg80gw -r 190.160.1.1 190.160.1.40 -a

**Output**
Success shows Operation completed successfully. Failure shows an appropriate error message.

**exit**

**Description**
Exits from the shell.

**Syntax**
exit

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td>None</td>
</tr>
</tbody>
</table>

**Return Value**
None

**Example**
exit

**Output**
None
**fetch certificate**

**Description**
Establishes a SIC connection with the Security Management Server and fetches the certificate. You fetch the certificate from a specific appliance with the `gateway-name` parameter.

**Syntax**
```
fetch certificate mgmt-ipv4-address <ip_addr> [gateway-name <gw_name>]
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Management IPv4 address</td>
</tr>
<tr>
<td>&lt;gw_name&gt;</td>
<td>Appliance/Module name</td>
</tr>
</tbody>
</table>

**Example**
```
fetch certificate mgmt-ipv4-address 192.168.1.100 gateway-name SMB_Appliance
```

**Output**
Success shows `OK`. Failure shows an appropriate error message.

**fetch license**

**Description**
Fetches a license from one of these locations:
- Local gateway - There is an option to specify the file name with the `<file_name>` parameter.
- User Center at Check Point
- USB device - There is an option to specify the file name with the `<file_name>` parameter.

**Syntax**
```
fetch license {local [file <file_name>]|usercenter|usb [file <file_name>]
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;file_name&gt;</td>
<td>Name of the file that contains the license</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
fetch license usb file LicenseFile.xml
```

**Output**
Success shows `OK`. Failure shows an appropriate error message.

**fetch policy**

**Description**
Fetches a policy from the Security Management Server with IPv4 address `<ip_addr>` or from the local gateway.

**Syntax**
```
fetch policy {local|mgmt-ipv4-address <ip_addr>}
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address of the management server</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
fetch policy mgmt-ipv4-address 192.168.1.100
```

**Output**
Success shows `Done`. Failure shows an appropriate error message.
**fw Commands**

The fw commands are used for working with various aspects of the firewall. All fw commands are executed on the Check Point Security Gateway. For more about the fw commands, see the R75.20 Command Line Interface (CLI) Reference Guide (http://supportcontent.checkpoint.com/documentation_download?ID=12264).

fw commands can be found by typing fw [TAB] at a command line. For some of the CLI commands, you can enter the -h parameter to display all the relevant arguments and parameters. These commands are:

<table>
<thead>
<tr>
<th>fw command</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>fw accel [-h]</td>
<td>Turn acceleration on/off</td>
</tr>
<tr>
<td>fw activation [-h]</td>
<td>Activate license</td>
</tr>
<tr>
<td>fw avload [-h]</td>
<td>Load AV signatures to kernel</td>
</tr>
<tr>
<td>fw ctl [args]</td>
<td>Control kernel</td>
</tr>
<tr>
<td>fw debug [-h]</td>
<td>Turn debug output on or off</td>
</tr>
<tr>
<td>fw fetch</td>
<td>Fetch last policy</td>
</tr>
<tr>
<td>fw fetchdefault [-h]</td>
<td>Fetch default policy</td>
</tr>
<tr>
<td>fw fetchlocal [-h]</td>
<td>Fetch local policy</td>
</tr>
<tr>
<td>fw monitor [-h]</td>
<td>Monitor Check Point Check Point 600 Appliance traffic</td>
</tr>
<tr>
<td>fw pull_cert</td>
<td>Pull certificate from internal CA</td>
</tr>
<tr>
<td>fw sfwd</td>
<td>fw daemon</td>
</tr>
<tr>
<td>fw sic_init [-h]</td>
<td>Initialize SIC</td>
</tr>
<tr>
<td>fw sic_reset [-h]</td>
<td>Reset SIC</td>
</tr>
<tr>
<td>fw sic_test</td>
<td>Test SIC with management</td>
</tr>
<tr>
<td>fw stat [-h]</td>
<td>Display policy installation status of the Gateway. (Command is provided for backward compatibility.)</td>
</tr>
<tr>
<td>fw tab [-h]</td>
<td>Display kernel-table content</td>
</tr>
<tr>
<td>fw unloadlocal</td>
<td>Unload local policy</td>
</tr>
<tr>
<td>fw ver [-k]</td>
<td>Display version</td>
</tr>
</tbody>
</table>

**reboot**

**Description**

Reboots the system.

**Syntax**

reboot

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Return Value**

None
**Example**
```
reboot
```

**Output**
```
None
```

---

## restore default-settings

**Description**
Restores the default settings of the appliance without affecting the software image. All the custom user settings for the appliance are deleted.

**Syntax**
```
restore default-settings [preserve-sic {yes|no}] [preserve-license {yes|no}] [force {yes|no}]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>preserve-sic</td>
<td>Select whether to preserve your current SIC settings</td>
</tr>
<tr>
<td>preserve-license</td>
<td>Select whether to preserve your current license</td>
</tr>
<tr>
<td>force</td>
<td>Skip the confirmation question</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```
restore default-settings preserve-sic yes
```

**Output**
```
n/a
```

**Comments**
The appliance automatically reboots after the default settings are restored.

---

## restore settings

**Description**
Restores the appliance settings from a backup file. The backup file can be located on a USB device or on a TFTP server.

**Syntax**
```
restore settings from {usb|tftp server <serverIP>} filename <file_name>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;file_name&gt;</td>
<td>Name of the backup file.</td>
</tr>
<tr>
<td>&lt;serverIP&gt;</td>
<td>IPv4 address of the TFTP server.</td>
</tr>
</tbody>
</table>

**Example**
```
restore settings from tftp server 1.1.1.1 filename sg80
```

**Output**
```
n/a
```

**Comments**
The appliance automatically reboots after the settings are restored.

---

## revert to factory defaults

**Description**
Revert the appliance to the original factory defaults. This command deletes all data and software images from the appliance.

**Syntax**
```
revert to factory-defaults
```

---
Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example

revert to factory-defaults

Output

Success shows a warning message. Enter yes to continue. Failure shows an appropriate error message.

**revert to saved image**

Description

Reverts the appliance to the previous software image.

Syntax

revert to previous-image

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example

revert to previous-image

Output

Success shows OK. Failure shows an appropriate error message.

**set admin access**

Description

Configures administrator access parameters for remote management of the appliance.

Syntax

set admin-access [interfaces <interface>] [web-access-port <web_port>] [ssh-access-port <ssh_port>] [allowed-ipv4-addresses <any|specific|any-except-internet>]

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Configure from which interfaces admin access is allowed. These options can be used: any SWITCH+WAN WAN LAN SWITCH</td>
</tr>
<tr>
<td>&lt;web_port&gt;</td>
<td>Configures the web port for HTTPS access</td>
</tr>
<tr>
<td>&lt;ssh_port&gt;</td>
<td>Secure Shell (SSH) port</td>
</tr>
</tbody>
</table>

- any - Configures allowed admin access from all IPv4 addresses.
- specific - Only IPv4 addresses that are configured with the add admin access command can be used to access the appliance.
- any-except-internet - From internal networks and VPN admin access is allowed from all IPv4 addresses. From the Internet, only IPv4 addresses that are configured with the add admin access command can be used to access the appliance.
Example

```
set admin-access web-access-port 4434 allowed-ipv4-addresses specific
```

Output

Success shows OK. Failure shows an appropriate error message.

Comments

Your access to the appliance may be blocked (although your current session is retained).

### set antispam mode

**Description**

Configures the Anti-Spam mode.

**Syntax**

```
set antispam mode {on|detect|off} [detection_method <filter_type> log {none|log|alert} spam_content_action {block|flag_header|flag_subject} flag_subject_stamp <spam_stamp_text>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;filter_type&gt;</td>
<td>String which represents what Spam Filtering based on. This can have the following values:</td>
</tr>
<tr>
<td></td>
<td>AS_TYPE.CONTENT_BASED – For content-based filtering.</td>
</tr>
<tr>
<td></td>
<td>AS_TYPE.IP_REPUTATION - For IP Reputation filtering.</td>
</tr>
<tr>
<td>&lt;spam_stamp_text&gt;</td>
<td>Flag spam email which subject contains &lt;spam_stamp_text&gt;</td>
</tr>
</tbody>
</table>

**Example**

- set antispam mode off
- set antispam mode detect detection_method AS_TYPE.CONTENT_BASED log log spam_content_action block flag_subject_stamp spam

**Output**

Failure shows an appropriate error message.

### set bridge

**Description**

Configures a bridge <name>.

**Syntax**

```
set bridge <name> {stp <stp_mode>|add member <network_interface>|delete member <bridge_obj>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>The bridge’s name</td>
</tr>
<tr>
<td>&lt;stp_mode&gt;</td>
<td>Spanning Tree Protocol status. This can have the following values: on off</td>
</tr>
<tr>
<td>&lt;network_interface&gt;</td>
<td>Network Interface to add to the bridge</td>
</tr>
<tr>
<td>&lt;bridge_obj&gt;</td>
<td>Network Interface to remove from the bridge</td>
</tr>
</tbody>
</table>
### set date

**Description**
Sets system date in YYYY-MM-DD format.

**Syntax**
```
set date <date>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;date&gt;</td>
<td>Date in YYYY-MM-DD format</td>
</tr>
</tbody>
</table>

**Example**
```
set date 2013-06-27
```

**Output**
Success shows the date. Failure shows an appropriate error message.

### set dhcp server

The `set dhcp server` command configures a range of parameters for the DHCP (Dynamic Host Configuration Protocol) server.

#### Setting the IP Pool

**Description**
Sets the DHCP server IP address pool for interface `<interface>`.

**Syntax**
```
set dhcp server interface <interface> include-ip-pool <ip_range>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_range&gt;</td>
<td>IPv4 address range format</td>
</tr>
</tbody>
</table>

**Example**
```
set dhcp server interface LAN2 include-ip-pool 192.168.1.50-192.168.1.60
```

**Output**
Failure shows an appropriate error message.

#### Excluding IP Addresses

**Description**
Sets a range of IP addresses that cannot be assigned by the DHCP (Dynamic Host Configuration Protocol) server for interface `<interface>`.

**Syntax**
```
set dhcp server interface <interface> exclude-ip-pool <ip_range>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_range&gt;</td>
<td>IPv4 address range format</td>
</tr>
</tbody>
</table>

**Example**
```
set dhcp server interface LAN2 exclude-ip-pool 192.168.1.52-192.168.1.54
```
Enabling the DHCP Server

Description
Enables or disables the DHCP server.

Syntax
set dhcp server interface <interface> {enable|disable}

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

Example
set dhcp server interface LAN2 enable

Output
Failure shows an appropriate error message.

Configuring the Default Gateway

Description
Configures the default gateway for the DHCP clients.

Syntax
set dhcp server interface <interface> default-gateway {auto|<ip_addr>}

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

Example
set dhcp server interface LAN2 default-gateway auto

Output
Failure shows an appropriate error message.

Configuring Host Name

Description
Configures the host name for the DHCP clients.

Syntax
set dhcp server interface <interface> domain <host>

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;host&gt;</td>
<td>Host name</td>
</tr>
</tbody>
</table>

Example
set dhcp server interface LAN5 domain myHost.com

Output
Failure prints appropriate error message.

Configuring the WINS Server

Description
Configures the WINS (Windows Internet Name Service) server for the DHCP clients.
Syntax

```
set dhcp server interface <interface> wins primary <ip_addr>
[secondary <ip_addr>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

**Example**

```
set dhcp server interface LAN2 wins primary 192.168.1.50
```

**Output**

Failure shows an appropriate error message.

### Automatically configuring the WINS Server

**Description**

Configures automatically the WINS server for the DHCP clients.

**Syntax**

```
set dhcp server interface <interface> wins-mode <ip_addr>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

**Example**

```
set dhcp server interface LAN2 wins-mode 192.168.1.50
```

**Output**

Failure shows an appropriate error message.

### Configuring IP Lease Time

**Description**

Configures the time (in hours) that an IP address is leased to a DHCP client.

**Syntax**

```
set dhcp server interface <interface> lease-time <hours>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;hours&gt;</td>
<td>Lease time in hours</td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

**Example**

```
set dhcp server interface LAN2 lease-time 18
```

**Output**

Failure shows an appropriate error message.

### Configuring DNS Server

The `set dhcp server` command configures the IP address of the DNS (Domain Name System) server for the DHCP server.

**Description**

Configures the DNS server IP address to `<ip_addr>`.

**Syntax**

```
set dhcp server interface <interface> dns {auto|primary
<ip_addr>|secondary <ip_addr>|tertiary <ip_addr>}
```
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

### Example

```bash
set dhcp server interface LAN2 dns tertiary 192.168.1.50
```

### Output

Success shows **Configurations Saved Successfully. Failure shows an appropriate error message.**

---

### Configuring the NTP Server

**Description**

Configures the IP addresses of the NTP (Network Time Protocol) servers.

**Syntax**

```bash
set dhcp server interface <interface> ntp ip_addr> [secondary <ip_addr>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

### Example

```bash
set dhcp server interface LAN2 ntp 192.168.1.50 secondary 192.168.1.60
```

### Output

Failure shows an appropriate error message.

---

### Configuring a TFTP Server

**Description**

Configures a TFTP (Trivial File Transfer Protocol) server.

**Syntax**

```bash
set dhcp server interface <interface> tftp <tftp_server>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;tftp_server&gt;</td>
<td>TFTP server name</td>
</tr>
</tbody>
</table>

### Example

```bash
set dhcp server interface LAN2 tftp 19.168.5.20
```

### Output

Failure shows an appropriate error message.

---

### Configuring the Path for a Bootstrap File

**Description**

Configures the bootstrap file path.

**Syntax**

```bash
set dhcp server interface <interface> file {none|<boot_file>}
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;boot_file&gt;</td>
<td>Bootstrap file path</td>
</tr>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

### Example

```bash
set dhcp server interface LAN2 file none
```
### Configuring Call Manager

**Description**
Configures the call manager server IP addresses.

**Syntax**
```bash
set dhcp server interface <interface> callmgr <ip_addr> [secondary <ip_addr>]
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

**Example**
```bash
set dhcp server interface LAN2 callmgr 198.162.1.18 secondary 198.162.2.1
```

**Output**
Failure shows an appropriate error message.

### Configuring X-Windows Display

**Description**
Configures X-Windows Display Manager.

**Syntax**
```bash
set dhcp server interface <interface> xwin-display-mgr <ip_addr>
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

**Example**
```bash
set dhcp server interface LAN2 xwin-display-mgr 192.168.20.5
```

**Output**
Failure shows an appropriate error message.

### Configuring VoIP Phones

**Description**
Configures the Avaya, Nortel, or Thomson VoIP phone parameters.

**Syntax**
```bash
set dhcp server interface <interface> {avaya-voip|nortel-voip|thomson-voip} {none|<config_string>}
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;config_string&gt;</td>
<td>Configuration string used to configure VoIP phones</td>
</tr>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Example**
```bash
set dhcp server interface LAN2 nortel-voip none
```

**Output**
Success shows `Configurations Saved Successfully`. Failure shows an appropriate error message.

### Configuring Custom DHCP Option

**Description**
Configures a custom DHCP server option.
set dhcp server interface <interface> custom-option name <name> type <type> tag <tag> data <data>

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Example**
n/a

**Output**
Success shows Configurations Saved Successfully. Failure shows appropriate error message.

### Deleting DHCP Custom Option Code

**Description**
Deletes DHCP all custom option with code <code> that was defined for interface <interface>.

**Syntax**
set dhcp server interface <interface> delete custom-option <option_name>

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;option_name&gt;</td>
<td>The name of the DHCP custom option</td>
</tr>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Example**
set dhcp server interface LAN2 delete custom-option code 12

**Output**
Failure shows an appropriate error message.

### set dhcp relay

**Description**
Manages DHCP relay for the IP addresses of a specified interface.

**Syntax**
set dhcp server interface <interface> relay relay-to <ip_addr> relay-secondary <ip_addr>

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

**Example**
set dhcp server interface LAN2 relay relay-to 198.162.1.1

**Output**
Failure shows an appropriate error message.

### set dns

**Description**
Sets a primary, secondary or tertiary DNS server to an IP address <ip_addr>. The secondary and tertiary DNS servers are optional.

**Syntax**
set dns {primary|secondary|tertiary} ipv4-address <ip_addr>

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure
Example: set dns secondary ipv4-address 4.4.4.4

Output: Success shows OK. Failure shows an appropriate error message.

**set dns proxy**

**Description:** Enables/disables the DNS proxy server.

**Syntax:**

```
set dns proxy {disable|enable [resolving {on|off}]}  
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Return Value:**

0 on success, 1 on failure

**Example:** set dns proxy enable

Output: Success shows OK. Failure shows an appropriate error message.

**set dns mode**

**Description:** Sets the global or internet mode for the DNS server. In internet mode the DNS configuration is inherited from the internet connection. In global mode the manual settings are taken as the DNS configuration.

**Syntax:**

```
set dns mode <global|internet>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;global</td>
<td>internet&gt;</td>
</tr>
</tbody>
</table>

**Return Value:**

0 on success, 1 on failure

**Example:** set dns mode global

Output: Success shows OK. Failure shows an appropriate error message.

**set domainname**

**Description:** Sets the domain name for the system to be <domain>.

**Syntax:**

```
set domainname <domain>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;domain&gt;</td>
<td>Domain name for the system</td>
</tr>
</tbody>
</table>

**Example:** set domainname checkpoint.com

Output: Success prints OK. Failure prints appropriate error message.

**set dynamic-dns**

You can use the `set dynamic-dns` command to configure DDNS account details.
Managing dynamic DNS details

**Description**
Enables and disables the dynamic dns service with these options:

- `set dynamic-dns disable` - Disables the DDNS service.
- `set dynamic-dns enable` - Enables the DDNS service.
- `set dynamic-dns provider <provider>` - Sets the DDNS provider that you have already set up an account with.
- `set dynamic-dns provider <provider> domain <domain>` - Sets the domain name (sometimes called host name) within your account that the device will use.
- `set dynamic-dns provider <provider> user <user>` - Sets the user name of the account.
- `set dynamic-dns provider <provider> password <password>` - Sets the password of the account.

**Syntax**
```
set dynamic-dns {disable|enable|provider <provider>} [domain <domain>] [user <user>] [password <password>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;provider&gt;</td>
<td>DDNS providers. Available values for this field are: DynDns and no-ip.com</td>
</tr>
<tr>
<td>&lt;domain&gt;</td>
<td>the domain name within your account that the device will use.</td>
</tr>
<tr>
<td>&lt;user&gt;</td>
<td>the user name of the account</td>
</tr>
<tr>
<td>&lt;password&gt;</td>
<td>the password of the account</td>
</tr>
</tbody>
</table>

**Example**
```
set dynamic-dns enable provider DynDns domain domain-name user ddnss-user password ddnss-pass
```

**Output**
Failure shows an appropriate error message.

set expert password

**Description**
Sets the initial password or password hash for the expert shell

**Syntax**
```
set expert {password|password-hash} {<pass>|<pass_hash>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;pass&gt;</td>
<td>Password using alphanumeric and special characters</td>
</tr>
<tr>
<td>&lt;pass_hash&gt;</td>
<td>Password MD5 string representation</td>
</tr>
</tbody>
</table>

**Example**
```
set expert password-hash $1$fGT7pGX6$oo9LUBJTkLOGKLhjRQ2rw1
```

**Output**
Success shows OK. Failure shows an appropriate error message.

**Comments**
To generate a password-hash, you can use this command on any Check Point 600/1100 Appliance gateway (as an expert user).
```
cryptpw -a md5 <password string>
```
set external ports access

Description  
Enable or disable the external ports (USB/Express card/SD card).

Syntax  
```
set external ports access {disable|enable}
```

Parameters  
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example  
```
set external ports access disable
```

Output  
Success shows OK. Failure shows an appropriate error message.

set host

Description  
Sets the IPv4 address <ip_addr> and the dhcp configuring of the existing host name <host>.

Syntax  
```
set host <host> [ipv4-address <ip_addr>] [exclude-from-dhcp {off|on} [exclude-from-dhcp {off|on} mac-reserved-in-dhcp on [reserve-mac-address <mac_addr>]]]
```

Parameters  
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;host&gt;</td>
<td>The host's name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>The host IPv4 address format</td>
</tr>
<tr>
<td>&lt;mac_addr&gt;</td>
<td>default or MAC address format, 00:1C:7F:21:05:BE</td>
</tr>
</tbody>
</table>

Example  
- set host myHost exclude-from-dhcp off
- set host myHost exclude-from-dhcp on mac-reserved-in-dhcp off
- set host myHost exclude-from-dhcp on mac-reserved-in-dhcp on reserve-mac-address 00:F4:7A:2F:11:AS
- set host myHost ipv4-address 192.168.3.0 exclude-from-dhcp off
- set host myHost ipv4-address 192.168.3.0 exclude-from-dhcp on mac-reserved-in-dhcp off
- set host myHost ipv4-address 192.168.3.0 exclude-from-dhcp on mac-reserved-in-dhcp on reserve-mac-address 00:F4:7A:2F:11:AS

Output  
Failure shows an appropriate error message.

set hostname

Description  
Sets the host name of the machines to <host>.

Syntax  
```
set hostname <host>
```

Parameters  
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;host&gt;</td>
<td>Host name</td>
</tr>
</tbody>
</table>

Example  
```
set hostname SG80
```

Output  
Failure shows an appropriate error message.
set inactivity-timeout

Description  
Sets inactivity timeout for web UI and shells assigned to users (in minutes).

Syntax  
set inactivity-timeout <time_out>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| <time_out> | Inactivity timeout in minutes.  
Range: 1-999  
Default: 10 |

Example  
set inactivity-timeout 60

Output  
Failure shows an appropriate error message.

set interface

You can use the set interface command to manage and configure the interfaces.

Managing Interfaces

The set interface command can enable or disable the interface.

Description  
Manages the interfaces

Syntax  
set interface <interface> {unassigned|state {on|off}}

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

Example  
set interface LAN5 state on

Output  
Success shows OK. Failure shows an appropriate error message.

Configuring Static IP

Description  
Sets parameters for different static IP interface types.

Syntax  
set interface <interface> ipv4-address <ip_addr> {subnet-mask <ip_mask>|mask-length <mask-length>} default-gw <ip_addr> [dns-primary <ip_addr> [dns-secondary <ip_addr>]]
<table>
<thead>
<tr>
<th>Parameters</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&lt;conn_time&gt;</td>
<td>Number of seconds before connection test timeout. A number between 0 and 999. A value of 0 applies the configuration and skips the connection tests.</td>
</tr>
<tr>
<td></td>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td></td>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
<tr>
<td></td>
<td>&lt;mask&gt;</td>
<td>Subnet mask</td>
</tr>
<tr>
<td></td>
<td>&lt;mask_length&gt;</td>
<td>Mask length</td>
</tr>
</tbody>
</table>

**Example**

```plaintext
set interface LAN5 ipv4-address 1.1.1.1 subnet-mask 255.255.255.0
default-gw 192.168.2.1 dns-primary 2.2.2.2 dns-secondary 3.3.3.3
```

**Output**

Failure shows an appropriate error message.

## Configuring Advanced Interface Settings

**Description**

Sets these advanced interface preferences:

- Auto-negotiation
- MTU
- Speed

**Syntax**

```plaintext
set interface <interface> [auto-negotiation <on|off>] [mtu <mtu>] [link-speed <speed>]
```

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
<tr>
<td></td>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;mtu&gt;</td>
<td>MTU size - integer in range 68-1500</td>
</tr>
<tr>
<td></td>
<td>&lt;speed&gt;</td>
<td>10M/100M/1000M</td>
</tr>
</tbody>
</table>

**Example**

```plaintext
set interface LAN5 auto-negotiation on link-speed 10 mtu 200
```

**Output**

Failure shows an appropriate error message.

## set internet-connection

You can use the `set internet-connection` command to manage and configure the internet interfaces.

### Enabling/Disabling

**Description**

Enables or disables internet connection `<connection_name>`.

**Syntax**

```plaintext
set internet-connection <connection_name> {enable|disable}
```

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&lt;connection_name&gt;</td>
<td>Valid connection name</td>
</tr>
</tbody>
</table>

**Example**

```plaintext
set internet-connection Internet1 enable
```
Output

Failure prints appropriate error message.

**Configuring Static IP**

**Description**
Sets parameters for different static IP interface types.

**Syntax**
```
set internet-connection <connection_name> type static ipv4-address <ip_addr> 
(subnet-mask <ip_mask>|mask-length <mask_length>)
default-gw <ip_addr> [dns-primary <ip_addr>] [dns-secondary <ip_addr>] [dns-tertiary <ip_addr>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>IPv4 address format</td>
</tr>
<tr>
<td>&lt;ip_mask&gt;</td>
<td>IP address for subnet mask</td>
</tr>
<tr>
<td>&lt;mask_length&gt;</td>
<td>Mask length</td>
</tr>
</tbody>
</table>

**Example**
- `set internet-connection Internet1 type static ipv4-address 1.1.1.2 subnet-mask 255.255.255.0 default-gw 1.1.1.1 dns-primary 2.2.2.2 dns-secondary 3.3.3.3`
- `set internet-connection Internet1 type static ipv4-address 1.1.1.7 subnet-mask 255.255.255.0 default-gw 1.1.1.1`

**Output**

Failure shows an appropriate error message.

**Configuring PPTP and L2TP**

**Description**
Sets interface type (PPTP and L2TP) settings with user name `<user>`, password `<pass>`, and server `<server>`.

**Syntax**
```
set internet-connection <connection_name> type {pptp|l2tp} username <user> password <pass> server <server> 
[local-ipv4-address <auto|ip_addr>] 
[wan-ipv4-address <auto|<ip_addr> {wan-subnet-mask <ip_mask>|wan-mask-length <mask-length>}] default-gw <ip_addr>]
```


### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Connection name</td>
</tr>
<tr>
<td><code>&lt;ip_addr&gt;</code></td>
<td>IPv4 address format.</td>
</tr>
<tr>
<td><code>&lt;ip_mask&gt;</code></td>
<td>IP address for subnet mask</td>
</tr>
<tr>
<td><code>&lt;mask_length&gt;</code></td>
<td>Mask length</td>
</tr>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>ISP user login name.</td>
</tr>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><code>&lt;server&gt;</code></td>
<td>Server host name or IP address.</td>
</tr>
</tbody>
</table>

**local-ipv4-address**
- Local tunnel IPv4 address assignment.
  - `auto` - Get the IPv4 address from ISP.
  - `<ip_addr>` - IP address for local tunnel.

**wan-ipv4-address**
- WAN IPv4 address assignment
  - `auto` - Get the WAN IPv4 address from ISP. This is the default setting.
  - `<ip_addr>` - IP address for WAN port.
  - `<ip_mask>` - IP address for subnet mask for WAN port.
  - `<mask_length>` - Mask length for WAN port.
  - `default-gw` - Default gateway for WAN port.

### Example

```
set internet-connection Internet1 type pptp username John password verySecurePass server 1.1.1.1 local-ipv4-address 2.2.2.2
wan-ipv4-address 3.3.3.3 wan-subnet-mask 255.255.255.0 default-gw 4.4.4.4
```

```
set internet-connection Internet1 type l2tp username John password verySecurePass server 1.1.1.1 local-ipv4-address 2.2.2.2
```

### Output

Failure shows an appropriate error message.

### Configuring PPPoE

#### Basic Configurations

**Description**
Configures WAN, DMZ, and ADSL PPPoE interface type settings with user name `<user>` and password `<pass>`.

**Syntax**

```
set internet-connection <connection_name> type pppoe username <user> password <pass>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>ISP user login name.</td>
</tr>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Connection name.</td>
</tr>
</tbody>
</table>
**Local tunnel IPv4 address assignment**

**Description**
Configures the local tunnel IPv4 address assignment.

**Syntax**
```
set internet-connection <connection_name> type pppoe local-ipv4-address <auto|ip_addr> [method {auto|chap|pap}] [idle-time <idle_time>]
```

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;user&gt;</td>
<td>ISP user login name.</td>
</tr>
<tr>
<td>&lt;pass&gt;</td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name.</td>
</tr>
<tr>
<td>&lt;idle_time&gt;</td>
<td>Idle timeout in minutes before disconnect when using Connect on demand. Value is between 1 and 999 minutes.</td>
</tr>
</tbody>
</table>

**Example**
```
set internet-connection Internet1 type pppoe local-ipv4-address auto method pap idle-time 20
```

**Output**
Failure shows an appropriate error message.

**Configuring PPPoA**

**Basic configurations**

**Description**
Configures the ADSL PPPoA interface type settings with user name `<user>` and password `<pass>`.

**Syntax**
```
set internet-connection <connection_name> type pppoa username <user> password <pass>
```

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;user&gt;</td>
<td>ISP user login name.</td>
</tr>
<tr>
<td>&lt;pass&gt;</td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name.</td>
</tr>
</tbody>
</table>

**Example**
```
set internet-connection Internet1 type pppoa username Alex password myPassword
```

**Output**
Failure shows an appropriate error message.
Local tunnel IPv4 address assignment

Description Configures the local tunnel IPv4 address assignment.

Syntax

```
set internet-connection <connection_name> type pppoa
local-ipv4-address <auto|ip_addr> [method {auto|chap|pap}]
[idle-time <idle_time>]
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;user&gt;</td>
<td>ISP user login name.</td>
</tr>
<tr>
<td>&lt;pass&gt;</td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name.</td>
</tr>
<tr>
<td>&lt;idle_time&gt;</td>
<td>Idle timeout in minutes before disconnect when using Connect on demand. Value is between 1 and 999 minutes.</td>
</tr>
</tbody>
</table>

Example

```
set internet-connection Internet1 type pppoa local-ipv4-address auto method pap idle-time 20
```

Output Failure shows an appropriate error message.

Configuring EoA

Description Changes internet type to ADSL EoA.

Syntax

```
set internet-connection <connection_name> type eoa
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name</td>
</tr>
</tbody>
</table>

Example

```
set internet-connection Internet1 type eoa
```

Output Failure shows an appropriate error message.

Configuring VPI, VCI and Encapsulation

Description Sets parameters for Virtual Path Identifier (VPI) and Virtual Circuit Identifier (VCI) in ADSL connections.

Syntax

```
set internet-connection <connection_name> [vpi <vpi>] [vci <vci>]
[standard <dsl_standard>] [encapsulation {llc|vcmux}]
```
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name</td>
</tr>
<tr>
<td>&lt;vpi&gt;</td>
<td>VPI value for the ADSL connection</td>
</tr>
<tr>
<td>&lt;vci&gt;</td>
<td>VCI value for the ADSL connection</td>
</tr>
<tr>
<td>&lt;dsl_standard&gt;</td>
<td>The standard to support for the DSL line, as specified by your ISP. This can have the following values: adsl2 - ADSL2 adsl2plus - ADSL2+ gdmt - G.DMT glite - G.Lite multimode - Automatically detect G.DMT or T1.413 t1413 - T.1413 The default value is adsl2plus.</td>
</tr>
</tbody>
</table>

### Example

```bash
set internet-connection Internet1 vpi 8 vci 48 encapsulation llc
```

### Output

Failure shows an appropriate error message.

### Configuring Cellular Modem

**Description**

Configures the cellular internet connection.

**Syntax**

```bash
set internet-connection <connection_name> type cellular number <dialedNumber> [username <user> password <pass>] [apn <apn>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Connection name</td>
</tr>
<tr>
<td>&lt;dialedNumber&gt;</td>
<td>ISP dialed number. Default is *99#</td>
</tr>
<tr>
<td>&lt;user&gt;</td>
<td>ISP user login name.</td>
</tr>
<tr>
<td>&lt;pass&gt;</td>
<td>ISP user password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>&lt;apn&gt;</td>
<td>IPS access point name.</td>
</tr>
</tbody>
</table>

### Example

```bash
set internet-connection Internet1 type cellular number *99# username Johnny password myPassword
```

### Output

Failure shows an appropriate error message.

### Configuring ICMP

**Description**

Configures the ICMP (Internet Control Message Protocol) settings for the appliance.

**Syntax**

```bash
set internet-connection <connection_name> probe-icmp-servers first <ip_addr_or_host> [second <ip_addr_or_host>] [third <ip_addr_or_host>]
```
### Configuring Monitoring

**Description**  
Manages probe settings.

**Syntax**  
`set internet-connection <connection_name> probe-next-hop {true|false} [{probe-servers off|probing-method dns}]`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Valid connection name</td>
</tr>
</tbody>
</table>

**Example**  
`set internet-connection Internet1 probe-next-hop false`

**Output**  
Success shows OK. Failure shows an appropriate error message.

### Route Traffic

**Description**  
Routes traffic through connection `<connection_name>` by default.

**Syntax**  
`set internet-connection <connection_name> route-traffic-through-gateway {true|false}`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Valid connection name</td>
</tr>
</tbody>
</table>

**Example**  
`set internet-connection Internet1 route-traffic-through-gateway true`

**Output**  
Failure shows an appropriate error message.

### Connect on Demand

**Description**  
Routes traffic through connection `<connection_name>` by default.

**Syntax**  
`set internet-connection <connection_name> connect-on-demand {true|false}`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;connection_name&gt;</code></td>
<td>Valid connection name</td>
</tr>
</tbody>
</table>

**Example**  
`set internet-connection Internet1 connect-on-demand true`

**Output**  
Failure shows an appropriate error message.
**Load Balancing**

**Description**
Configures how the traffic to the internet is divided between available connections, according to their weights.

**Syntax**
```
set internet-connection <connection_name> ha-priority <priority> load-balancing-weight <weight>
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Valid connection name</td>
</tr>
<tr>
<td>&lt;priority&gt;</td>
<td>Priority of the connection. Should be number between 1 to 64.</td>
</tr>
<tr>
<td>&lt;weight&gt;</td>
<td>Traffic will be distributed automatically according to the configured load balancing percent weights.</td>
</tr>
</tbody>
</table>

**Example**
```
set internet-connection Internet1 auto-negotiation on linkspeed 10/full
```

**Output**
Failure shows an appropriate error message.

**Configuring Advanced Connection Settings**

**Description**
Sets advanced connection preferences.

**Syntax**
```
set internet-connection <connection_name> [auto-negotiation {on|off}] [link-speed <Speed_Duplex>] [mtu <MTU_setting>] [mac-addr <MAC>]
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection_name&gt;</td>
<td>Valid connection name</td>
</tr>
<tr>
<td>&lt;Speed_Duplex&gt;</td>
<td>Enter the link speed in Mbps and duplex status using one of these values:</td>
</tr>
<tr>
<td></td>
<td>100/full</td>
</tr>
<tr>
<td></td>
<td>100/half</td>
</tr>
<tr>
<td></td>
<td>10/full</td>
</tr>
<tr>
<td>&lt;MTU_Setting&gt;</td>
<td>Integer greater or equal to 68 (Default = 1500)</td>
</tr>
<tr>
<td>&lt;MAC&gt;</td>
<td>Manually enter the applicable hardware address</td>
</tr>
</tbody>
</table>

**Example**
```
- set internet-connection Internet1 auto-negotiation on linkspeed 10/full
- set internet-connection Internet1 mtu 1500 mac-addr 00:5A:21:D6:E1:55
```

**Output**
Failure shows an appropriate error message.
set internet mode

Description: Sets configuration parameters for internet High Availability mode when both internet connections are configured.

Syntax: `set internet mode {high-availability|load-balancing}`

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>high-availability</td>
<td>The appliance reverts to the primary Internet connection when it is available</td>
</tr>
<tr>
<td>load-balancing</td>
<td>The appliance does not change the Internet connection</td>
</tr>
</tbody>
</table>

Example: `set internet mode high-availability`

Output: Success shows OK. Failure shows an appropriate error message.

set nat

Description: Hides internal networks behind the Gateway's external IP address.

Syntax: `set nat hide-internal-behind-gateway {true|false}`

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example: `set nat hide-internal-behind-gateway true`

Output: Failure shows an appropriate error message.

set net-obj

Edit network objects that are used in the appliance's feature configuration.

Single IP

Description: Edits a single IP type network object.

Syntax: `set net-obj <name> type single [ipv4-address <ip_addr>] [dns-resolving {true|false}] [exclude-from-dhcp {off|on} on [mac-reserved-in-dhcp {off|on reserve-mac-address <mac_addr>}]]`

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>Network Object name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Valid IPv4 address</td>
</tr>
<tr>
<td>dns-resolving</td>
<td>Allow DNS server to resolve this object name</td>
</tr>
<tr>
<td>&lt;mac_addr&gt;</td>
<td>MAC address format</td>
</tr>
</tbody>
</table>
### Network

**Description**
Edits a network object network type.

**Syntax**
```
set net-obj <name> type network [ipv4-network-address <ip_addr>] [netmask {subnet-mask <ip_mask>|mask-length <mask-length>}]```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>Network Object name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Valid IPv4 address</td>
</tr>
<tr>
<td>dns-resolving</td>
<td>Allow DNS server to resolve this object name</td>
</tr>
<tr>
<td>&lt;mask&gt;</td>
<td>Subnet mask</td>
</tr>
<tr>
<td>&lt;mask_length&gt;</td>
<td>Mask length</td>
</tr>
</tbody>
</table>

**Example**
```
set net-obj NetObj1 type network ipv4-network-address 3.3.3.0 netmask mask-length 24
```

**Output**
Failure shows an appropriate error message.

### IP Range

**Description**
Edits an IP range type network object.

**Syntax**
```
set net-obj <name> type range [start-ipv4 <ip_addr>] [end-ipv4 <ip_addr>] [exclude-from-dhcp {on|off}]```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>Network Object name</td>
</tr>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Valid IPv4 address</td>
</tr>
</tbody>
</table>

**Example**
```
set net-obj NetObj3 type range start-ipv4 3.3.3.3 end-ipv4 3.3.3.6 exclude-from-dhcp off
```

**Output**
Failure shows an appropriate error message.

### set proxy

You can configure a proxy server that is used to fetch a license from Check Point User Center.

#### Managing a Proxy Server

**Description**
Enables or disables the proxy server used to fetch a license from the Check Point User Center.
Syntax
set proxy {enable|disable}

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example
set proxy disable

Output
Failure shows an appropriate error message.

Configuring a Proxy Server

Description
Sets the IP address and port number of the proxy server that is used to fetch a license from the Check Point User Center. Also enables the proxy server that is set.

Syntax
set proxy server <server> port <port>

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;server&gt;</td>
<td>Proxy server hostname or IPv4 address.</td>
</tr>
<tr>
<td>&lt;port&gt;</td>
<td>Valid port numbers are between 1 and 65535.</td>
</tr>
</tbody>
</table>

Example
set proxy server 197.30.66.25 port 8080

Output
Failure shows an appropriate error message.
Comments

If active is off or is not set, then NTP is disabled.

set snmp

You can use the set snmp command to manage and configure the SNMP settings. You must use the add
snmp command to configure the SNMP v2 or v3 parameters for these commands:

- set snmp traps receiver
- set snmp usm user

Managing SNMP Agent

Description
Enables and disables an SNMP agent.

Syntax
set snmp agent {on|off}

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{on</td>
<td>off}</td>
</tr>
</tbody>
</table>

Setting SNMP Version

Description
Sets the SNMP version.

Syntax
set snmp agent-version {any|v3-only}

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{any</td>
<td>v3-only}</td>
</tr>
</tbody>
</table>

Example
set snmp agent-version v3-only

Setting Community String

Description
Sets the SNMP agent community string.

Syntax
set snmp community <comm_string>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;comm_string&gt;</td>
<td>A password for v1 and v2 protocols. The value can be any word.</td>
</tr>
</tbody>
</table>

Example
set snmp community public

Setting SNMP Host Information

Description
Sets information about the host the SNMP agent is running on.

Syntax
set snmp contact <contact_string>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;contact_string&gt;</td>
<td>The value can be word.</td>
</tr>
</tbody>
</table>

Example
set snmp contact checkpoint
**Setting Host Location**

**Description**
Sets information about the location of the host on which the SNMP agent is running.

**Syntax**
```
set snmp location <location>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;location&gt;</td>
<td>The value can be any word.</td>
</tr>
</tbody>
</table>

**Example**
```
set snmp location lab
```

**Managing SNMP Traps**

**Description**
Enables and disables the SNMP traps.

**Syntax**
```
set snmp traps {enable|disable}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
set snmp traps enable
```

**Setting SNMP v2 Receivers**

**Description**
Sets a community string for SNMPv2 traps receiver.

**Syntax**
```
set snmp traps receiver <ip_addr> version v2 community <comm_string>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Trap receiver IPv4 address.</td>
</tr>
<tr>
<td>&lt;comm_string&gt;</td>
<td>A password for v1 and v2 protocols. The value can be any word.</td>
</tr>
</tbody>
</table>

**Example**
```
set snmp traps receiver 1.1.1.1 version v2 community anystring
```

**Output**
Failure shows an appropriate error message.

**Configuring SNMP v3 Receivers**

**Description**
Sets a USM user for SNMPv3 traps receiver.

**Syntax**
```
set snmp traps receiver <ip_addr> version v3 usm user <user_name>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip_addr&gt;</td>
<td>Trap receiver IPv4 address</td>
</tr>
<tr>
<td>&lt;user_name&gt;</td>
<td>Name user that was added with add snmp.</td>
</tr>
</tbody>
</table>

**Example**
```
set snmp traps receiver 1.1.1.1 version v3 user john
```

**Output**
Failure shows an appropriate error message.

**Comments**
Security parameters that were defined for the <v3_user> with the add snmp command are used.
**Configuring SNMP v3 Users**

**Description**  
Sets USM security user parameters for the `<v3_user>`. 

**Syntax**  
```
set snmp user <user_name> security-level {authNoPriv|authPriv}  
auth-pass-type {MD5|SHA1} auth-pass-phrase <auth-phrase>  
privacy-pass-type {AES|DES} privacy-pass-phrase <priv-phrase>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;user_name&gt;</code></td>
<td>Name user that was added with add snmp.</td>
</tr>
<tr>
<td>security-level</td>
<td>Messages sent or received by this user are authenticated using the privacy and authentication passwords. Values for this field are: authNoPriv and authPriv.</td>
</tr>
<tr>
<td>auth-pass-type</td>
<td>Authentication decryption protocol. Available values for this field are: MD5 and SHA1.</td>
</tr>
<tr>
<td><code>&lt;auth-phrase&gt;</code></td>
<td>The localized secret key used by the authentication protocol for authenticating messages.</td>
</tr>
<tr>
<td>privacy-pass-type</td>
<td>Which privacy decryption protocol to use. Available values for this field are: AES and DES.</td>
</tr>
<tr>
<td><code>&lt;priv-phrase&gt;</code></td>
<td>The localized secret key used by the privacy protocol for encrypting and decrypting messages.</td>
</tr>
</tbody>
</table>

**Example**  
```
set snmp user usm1 security-level authPriv auth-pass-type SHA1 auth-pass-phrase safeAuthPassPhrase privacy-pass-type AES privacy-pass-phrase safePrivacyPassPhrase
```

**Output**  
Failure shows an appropriate error message.

**Comments**  
You must add the user with the `add snmp` command.

**Setting a Single Trap**

**Description**  
Sets parameters for a single trap.

**Syntax**  
```
set snmp traps trap-name <trap_name> [enable {on|off}] threshold <threshold> severity {1|2|3|4} repetitions <reps> repetitions-delay <rep_delay>
```
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;reps&gt;</td>
<td>Number of repetitions for trap. Available values are: 1-10 or infinite for sending traps as long as the trap condition holds.</td>
</tr>
<tr>
<td>&lt;rep_delay&gt;</td>
<td>Delay time (seconds) between repetitions.</td>
</tr>
<tr>
<td>&lt;severity&gt;</td>
<td>Trap severity: (1) Low, (2) Medium, (3) High, (4) Critical.</td>
</tr>
<tr>
<td>&lt;threshold&gt;</td>
<td>Trap threshold, value must be a positive number</td>
</tr>
<tr>
<td>&lt;trap_name&gt;</td>
<td>Enter a valid trap-name value.</td>
</tr>
</tbody>
</table>

### Output

- Failure shows an appropriate error message.

### Comments

These are the valid trap-name values:

- interface-link-down
- interface-disconnected
- memory-utilization
- partition-free-space
- core-utilization
- core-interrupts-rate
- new-connections-rate
- concurrent-connections-rate
- bytes-throughput
- accepted-packet-rate
- temperature-sensor-reading
- voltage-sensor-reading
- cluster-member-state-changed
- cluster-block-state-error
- cluster-state-error
- cluster-problem-status
- cluster-interface-down
- connection-with-log-server-error
- connection-with-all-log-servers-error

### set static-route

#### Description

Edits a route with a route ID <route_id> to destination <route_IP>, source <route_IP> and gateway <gw_IP> or gateway interface of <interface>, and priority <metric>.

#### Syntax

```
set static-route <route_id> [destination <route_IP>] [source <route_IP>] [service <service_name>] [nexthop gateway {ipv4-address <gw_IP>|logical <interface>}] [metric <metric>]
```
### CLI Reference

#### set stat
c

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;route_id&gt;</td>
<td>Route ID</td>
</tr>
<tr>
<td>&lt;route_IP&gt;</td>
<td>IP address and subnet bit number of the route.</td>
</tr>
<tr>
<td>&lt;IPv4-address&gt;/&lt;Subnet-bit-number&gt;</td>
<td></td>
</tr>
<tr>
<td>&lt;service_name&gt;</td>
<td>The service protocol which to whom the route rule applies to. The default is to all protocols.</td>
</tr>
<tr>
<td>&lt;gw_IP&gt;</td>
<td>Gateway IP address.</td>
</tr>
<tr>
<td>&lt;interface&gt;</td>
<td>Interface to which the gateway is connected.</td>
</tr>
<tr>
<td>&lt;metric&gt;</td>
<td>Priority (metric) of the route.</td>
</tr>
</tbody>
</table>

**Example**

- set static-route 1 destination 172.15.47.0/24 nexthop gateway ipv4-address 10.0.0.1
- set static-route destination 172.15.47.0/24 nexthop gateway logical Internet1
- set static-route 1 metric 3

**Output**

Failure shows an appropriate error message.

#### set property
c

**Description**

Disables or enables first time configuration (from the USB autoplay configuration or the WebUI).

**Syntax**

```
set property {USB_auto_configuration {always|once|off} | first-time-wizard {always|once}}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

- set property USB_auto_configuration off
- set property first-time-wizard off

**Output**

Failure shows an appropriate error message.

#### set sic_init
c

**Description**

Sets the SIC password.

**Syntax**

```
set sic_init password <pass>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;pass&gt;</td>
<td>One-time password, as specified by the Security Management server administrator.</td>
</tr>
</tbody>
</table>

**Example**

set sic_init password verySecurePassword

**Output**

Success shows OK. Failure shows an appropriate error message.
set switch

Description
Configures the switch <name>.

Syntax
set switch <name> {add port <network_interface>|delete port <switch_obj>}

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;name&gt;</td>
<td>The switch’s name</td>
</tr>
<tr>
<td>&lt;network_interface&gt;</td>
<td>Network Interface to add to the switch</td>
</tr>
<tr>
<td>&lt;switch_obj&gt;</td>
<td>Network Interface to remove from the switch</td>
</tr>
</tbody>
</table>

Example
set switch LAN1_Switch delete port LAN3

Output
Failure shows an appropriate error message.

set time

Description
Sets system time in HH:MM format.

Syntax
set time <time>

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;time&gt;</td>
<td>Time in HH:MM format.</td>
</tr>
</tbody>
</table>

Example
set time 15:08

Output
Success shows time information. Failure shows an appropriate error message.

set time-zone

Description
Sets system time zone.

Syntax
set time-zone <timezone>

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;timezone&gt;</td>
<td>The time zone of your device</td>
</tr>
</tbody>
</table>

Example
set time-zone GMT+01:00 (Amsterdam/Berlin/Bern/Rome/Stockholm/Vien)

Output
Success shows time zone information. Failure shows an appropriate error message.

Comments
You can only use time zones that are pre-configured on the appliance. Use auto-completion to display the list of time zones.
set timezone-dst

Description: Automatically adjusts clock for daylight saving changes

Syntax: set timezone-dst automatic {on|off}

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example: set timezone-dst automatic on

Output: Success shows time zone information. Failure shows an appropriate error message.

set user

Set parameters for a specified user name.

**Setting Password for a User**

Description: Sets a password `<pass>` for an existing user `<user>`.

Syntax: set user `<user>` type {admin|local} password `<pass>`

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;pass&gt;</code></td>
<td>User password. Alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>User login name.</td>
</tr>
</tbody>
</table>

Example:
- set user John type admin password verySecurePassword
- set user bob type local password verySecurePassword

Output: Failure shows an appropriate error message.

**Setting Password Hash for a User**

Description: Sets a password hash `<pass_hash>` for an existing user `<user>`.

Syntax: set user `<user>` type {admin|local} password-hash `<pass_hash>`

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;pass_hash&gt;</code></td>
<td>Password MD5 string representation</td>
</tr>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>User login name</td>
</tr>
</tbody>
</table>

Example: set user John type admin password-hash $1$CTnQg69e$dwMJPcrB27XnAXUckPW7N0

Output: Failure shows an appropriate error message.

**Setting Remote Access for a Local User**

Description: Sets remote-access to on or off for an existing user.

Syntax: set user `<user>` type {admin|local} remote-access {0|1}
Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{0</td>
<td>1}</td>
</tr>
<tr>
<td>&lt;user&gt;</td>
<td>User login name</td>
</tr>
</tbody>
</table>

Example

set user John type local remote-access 0

Output

Failure shows an appropriate error message.

**Setting Comments for a Local User**

Description

Sets comments for an existing user `<user>`.

Syntax

`set user <user> type local comments <comments>`

Example

set user John type local comments local_user

Output

Failure shows an appropriate error message.

**Setting Permissions for an Admin User**

Description

Sets permissions for an admin user.

Syntax

`set user <user> type admin permission R|RW`

Example

set user John type admin permission R

Output

Failure shows an appropriate error message.

**set wlan**

You can use the `set wlan` command to manage and configure the wireless network.

**Enable/Disable Wireless Radio**

Description

Enables or disables all wireless networks.

Syntax

`set wlan {enable|disable}`

Example

n/a
Example set wlan enable
Output Failure shows an appropriate error message.

Enable/Disable VAP
Description Enables or disables a VAP.
Syntax set wlan vap <vap_name> {enable|disable}
Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;vap_name&gt;</td>
<td>VAP name</td>
</tr>
</tbody>
</table>
Example set wlan vap Guests1 enable
Output Failure shows an appropriate error message.

Assign Wireless Network to a Switch/Bridge
Description Assigns a wireless network to a switch or bridge. These settings apply only to the main wireless network or the chosen VAP.
Syntax set wlan [vap <vap_name>] assignment <switch_name>
Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;vap_name&gt;</td>
<td>VAP name</td>
</tr>
<tr>
<td>&lt;switch_name&gt;</td>
<td>Switch/Bridge name</td>
</tr>
</tbody>
</table>
Example
- set wlan assignment LAN1Sw
- set wlan vap Guests1 assignment LAN5Sw
Output Failure shows an appropriate error message.

Set SSID
Description Sets SSID to the main wireless network.
Syntax set wlan ssid <ssid>
Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ssid&gt;</td>
<td>Network name (SSID)</td>
</tr>
</tbody>
</table>
Example set wlan ssid My_Wireless
Output Failure shows an appropriate error message.

Wireless Connection Settings
Description Configures these wireless connection settings on your appliance:
- The wireless connection's country, operation mode, and channel
- Wireless transmitter power
These settings apply to all wireless networks:
- The primary WLAN
- All virtual access points (VAPs).
set wlan radio [country <country>] [operation-mode <operation_mode>] [channel <channel>] [channel-width <channel_width>] [transmitter-power <transmitter_power>]

Syntax
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;operation_mode&gt;</code></td>
<td>The operation mode. Can have the following values:</td>
</tr>
<tr>
<td></td>
<td>11b - Operates in the 2.4 GHz range and offers a maximum theoretical rate of 11 Mbps. When using this mode, only 802.11b stations will be able to connect.</td>
</tr>
<tr>
<td></td>
<td>11g - Operates in the 2.4 GHz range, and offers a maximum theoretical rate of 54 Mbps. When using this mode, only 802.11g stations will be able to connect.</td>
</tr>
<tr>
<td></td>
<td>11bg - Operates in the 2.4 GHz range, and offers a maximum theoretical rate of 54 Mbps. When using this mode, both 802.11b stations and 802.11g stations will be able to connect.</td>
</tr>
<tr>
<td></td>
<td>11n - Operates in the 2.4 GHz range, and offers a maximum theoretical rate of 300 Mbps. When using this mode, only 802.11n stations will be able to connect.</td>
</tr>
<tr>
<td></td>
<td>11ng - Operates in the 2.4 GHz range, and offers a maximum theoretical rate of 300 Mbps. When using this mode, 802.11g stations and 802.11n stations will be able to connect.</td>
</tr>
<tr>
<td><code>&lt;country&gt;</code></td>
<td>The country in which you are located.</td>
</tr>
<tr>
<td><code>&lt;channel&gt;</code></td>
<td>Integer or String. The radio frequency to use for the wireless connection. Can have the following values:</td>
</tr>
<tr>
<td></td>
<td>auto - The appliance automatically selects a channel. A specific channel between 1 and 14. The default value is auto.</td>
</tr>
<tr>
<td><code>&lt;channel_width&gt;</code></td>
<td>String. The channel width. This can have the following values:</td>
</tr>
<tr>
<td></td>
<td>auto - Automatically select the channel width: 20Mhz or 40Mhz. 20 - 20Mhz. Selecting auto can increase wireless performance, if a 40Mhz channel is available. However, in some cases it may interfere with other access points or wireless equipment in the area.</td>
</tr>
<tr>
<td><code>&lt;transmitter_power&gt;</code></td>
<td>The transmitter power. Can have the following values:</td>
</tr>
<tr>
<td></td>
<td>min - The minimum power</td>
</tr>
<tr>
<td></td>
<td>eighth - One-eighth of full power</td>
</tr>
<tr>
<td></td>
<td>quarter - One quarter of full power</td>
</tr>
<tr>
<td></td>
<td>half - One half of full power</td>
</tr>
<tr>
<td></td>
<td>full - Full power</td>
</tr>
<tr>
<td></td>
<td>Setting a higher transmitter power increases the access point's range. A lower power reduces interference with other access points in the vicinity. The default value is full. It is not necessary to change this value, unless there are other access points in the vicinity.</td>
</tr>
</tbody>
</table>

**Example**

```
set wlan radio operation-mode 11ng channel 2 channel-width 20
```
**Output**

Failure shows an appropriate error message.

---

**Wireless Security**

These settings apply only to the main wireless network, or the chosen VLAN.

**Security Type**

**Description**

Sets the wireless network security type.

**Syntax**

```
set wlan [vap <vap_name>] security-type <security_type>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;vap_name&gt;</td>
<td>VAP name</td>
</tr>
<tr>
<td>&lt;security_type&gt;</td>
<td>A string that can have these values: &lt;br/&gt; none &lt;br/&gt; WEP &lt;br/&gt; WPA2 &lt;br/&gt; WPA/WPA2</td>
</tr>
</tbody>
</table>

**Example**

```
set wlan vap Guest_Wireless security-type none
```

**Output**

Failure shows an appropriate error message.

---

**Security Settings**

**Description**

Enables to set password and use Hotspot.

**Syntax**

```
set wlan [vap <vap_name>] wpa-auth-type password <wlan_password> [hotspot {on|off}]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;vap_name&gt;</td>
<td>VAP name</td>
</tr>
<tr>
<td>&lt;wlan_password&gt;</td>
<td>The password for the wireless network</td>
</tr>
</tbody>
</table>

**Example**

```
set wlan vap Guest_Wireless wpa-auth-type password SuperSecurePassword hotspot on
```

**Output**

Failure shows an appropriate error message.

---

**WPA Encryption Type**

**Description**

Sets the WPA encryption type.

**Syntax**

```
set wlan [vap <vap_name>] wpa-encryption-type <encryption_type>
```
**CLI Reference**

**Check Point 1100 and 600 Appliance Administration Guide CLI and Advanced Routing**

### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;vap_name&gt;</code></td>
<td>VAP name</td>
</tr>
<tr>
<td><code>&lt;encryption_type&gt;</code></td>
<td>A string that can have these values: Auto, CCMP-AES, TKIP - Use TKIP (Temporal Key Integrity Protocol)</td>
</tr>
</tbody>
</table>

**Example**

```
set wlan vap Guest_Wireless wpa-encryption-type Auto
```

**Output**

Failure shows an appropriate error message.

### Advanced Wireless Settings

**Description**

Enables to hide or show the SSID, to allow Station-to-Station Traffic, and to enable or disable WDS. These settings apply only to the main wireless network, or the chosen VLAN.

**Syntax**

```
set wlan {advanced-settings|vap advanced-settings <vap_name>} [hide-ssid {on|off}] [station-to-station {allow|block}] [wds {on|off}]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;vap_name&gt;</code></td>
<td>VAP name</td>
</tr>
</tbody>
</table>

**Example**

```
set wlan advanced-settings hide-ssid off
```

**Output**

Failure shows an appropriate error message.

### Set the Main WLAN

**Description**

Set authentication to use a RADIUS server and optionally, use a hotspot. If hotspot is set to on, users are redirected to the hotspot portal before gaining access from this interface.

**Syntax**

```
set wlan wpa-auth-type radius [hotspot {on|off}]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

```
set wlan wpa-auth-type radius
```

**Output**

Failure shows an appropriate error message.

### Set VAP

**Description**

Set authentication to use a RADIUS server and optionally, use a hotspot. If hotspot is set to on, users are redirected to the hotspot portal before gaining access from this interface.

**Syntax**

```
set wlan vap <ssid> wpa-auth-type radius [hotspot {on|off}]
```
Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ssid&gt;</td>
<td>Network name (SSID)</td>
</tr>
</tbody>
</table>

Example  

set wlan vap VAP1 wpa-auth-type radius

Output  

Failure shows an appropriate error message.

gull/expert

The shell and expert commands switch between the shell and expert modes.

Description  

Changes to expert mode.

Syntax  

shell  

expert

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Return Value  

None

Example  

shell

Output  

None

Comments  

Use the cpshell command to start cpshell.

sim

SecureXL Implementation Module commands
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ver</td>
<td>get the version</td>
</tr>
<tr>
<td>if</td>
<td>get the interface list</td>
</tr>
<tr>
<td>tab [-s] [name]</td>
<td>print the table content (-s for summary)</td>
</tr>
<tr>
<td>ranges</td>
<td>print the range content</td>
</tr>
<tr>
<td>tab -d templates</td>
<td>print only templates in drop state</td>
</tr>
<tr>
<td>dbg &lt;options&gt;</td>
<td>set the sim debug flags</td>
</tr>
<tr>
<td>affinity</td>
<td>get/set affinity options</td>
</tr>
<tr>
<td>nonaccel [-s</td>
<td>-c] &lt;name(s)&gt;</td>
</tr>
<tr>
<td>feature &lt;feature&gt; [on</td>
<td>off]</td>
</tr>
<tr>
<td>tmplquota &lt;options&gt;</td>
<td>configure template quota feature</td>
</tr>
<tr>
<td>hlgos &lt;options&gt;</td>
<td>configure Heavy-Load CPU QOS feature</td>
</tr>
</tbody>
</table>

### Output

Success shows OK. Failure shows an appropriate error message.

### show admin access

**Description**

Shows admin access configuration information including interfaces and IPv4 addresses.

Use the `show admin-access-ipv4-addresses` command to only show the IP addresses from which the admin is allowed to remotely access the appliance.

**Syntax**

```
show {admin-access|admin-access-ipv4-addresses}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

**Example**

`show admin-access`

**Output**

Success shows admin access configuration information. Failure shows an appropriate error message.
### show backup settings

**Description**
Shows previous backup information of the appliance's settings.

**Syntax**
```
show backup-settings-{log|info {from tftp server <server> filename <file>|from usb filename <file>}}
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;server&gt;</td>
<td>IP address or host name of the TFTP server</td>
</tr>
<tr>
<td>&lt;file&gt;</td>
<td>Name of backup file</td>
</tr>
</tbody>
</table>

**Example**
- `show backup-settings-log`
- `show backup-settings-info from usb filename backup`

**Output**
Success shows backup settings information. Failure shows an appropriate error message.

### show bridge

**Description**
Shows these bridge settings:
- `show bridges` - Settings for all defined bridges
- `show bridge <bridge>` - Configuration for bridge <bridge>

**Syntax**
```
show [bridges |bridge <bridge>]
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;bridge&gt;</td>
<td>The name of the bridge</td>
</tr>
</tbody>
</table>

**Example**
`show bridge br0`

**Output**
Bridge settings and bridge members settings

### show clock

**Description**
Shows current system date and time.

**Syntax**
```
show clock
```

**Parameters**
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
`show clock`

**Output**
Success shows date and time. Failure shows an appropriate error message.
show commands

Description: Shows all available CLI commands.

Syntax: show commands

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example: show commands

Output: List of all available CLI commands.

show date

Description: Shows current date in DD-Month-YYYY format.

Syntax: show date

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example: show date

Output: Shows the current date.

show dhcp

Shows DHCP (Dynamic Host Configuration Protocol) settings.

Showing DHCP Settings

Description: Shows the current DHCP settings. The server parameter shows all the custom server options and advanced settings for all the interfaces.

Syntax: show dhcp [servers]

Parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example: show dhcp servers

Output: Shows a list of all DHCP related settings

Servers parameter - All of the custom DHCP server options and advanced settings for all interfaces.

Comments: These columns are included in the output table: Interface, Mode, IP Pool
**Showing DHCP for an Interface**

**Description**
Shows the DHCP server settings for a specific interface.

**Syntax**
```
show dhcp server interface <interface>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Example**
```
show dhcp server interface LAN2
```

**Output**

- Mode: Enabled
- Default gateway: auto
- Wins mode: Automatic
- Lease time: 72
- Dns mode: Automatic

**Showing DHCP IP Pool**

**Description**
Shows the range of IP addresses that are available to DHCP clients on a specific interface.

**Syntax**
```
show dhcp server interface <interface> [ip-pool]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;interface&gt;</td>
<td>Valid interface name</td>
</tr>
</tbody>
</table>

**Example**
```
show dhcp server interface LAN1_Switch ip-pool
```

**Output**

- Range: `<start_ip_address>` - `<end_ip_address>`

**Comments**
No output if an IP pool is not set for the interface.

**show diag**

**Description**
Shows information about your appliance, such as the current firmware version and additional details.

**Syntax**
```
show diag
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show diag
```

**Output**

Current system information.

**show disk usage**

**Description**
Shows the file system space used and space available.

**Syntax**
```
show disk-usage [-h|-m|-k]
```
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Human readable (e.g. 1K 243M 2G)</td>
</tr>
<tr>
<td>-m</td>
<td>1024*1024 blocks</td>
</tr>
<tr>
<td>-k</td>
<td>1024 blocks</td>
</tr>
</tbody>
</table>

### Example

```
show disk-usage -h
```

### Output

Current file system space used and space available.

---

### show dns

**Description**

Shows these DNS (Domain Name Settings) settings:

- `show dns` - Displays all DNS related parameters.
- `show dns mode` - Displays DNS mode (global or internet).
- `show dns primary` - Displays IP address of first DNS server.
- `show dns secondary` - Displays IP address of second DNS server.
- `show dns tertiary` - Displays IP address of third DNS server.
- `show dns proxy` - Displays DNS proxy status.

**Syntax**

```
show dns [mode|primary ipv4-address |secondary ipv4-address |tertiary ipv4-address |proxy]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

```
show dns primary
```

**Output**

- `show dns` - Table containing all DNS related parameters, these columns are displayed: DNS Mode, First Server, Second Server, Third Server, DNS Proxy.
- `show dns mode` - Global or Internet
- `show dns primary ipv4-address` - IPv4 address of the first DNS server.
- `show dns secondary ipv4-address` - IPv4 address of the second DNS server.
- `show dns tertiary ipv4-address` - IPv4 address of the third DNS server.
- `show dns proxy` - DNS proxy: Enabled
  - Resolve Network Objects: Enabled

---

### show domainname

**Description**

Shows the domain name of the system.

**Syntax**

```
show domainname
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

```
show domainname
```

**Output**

Domain name of the system.
show dynamic-dns

Description
Shows DDNS account details.

Syntax
show dynamic-dns

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example
show dynamic-dns

Output
DDNS account details.
The service status, service provider, domain, user name, password.

show internet mode

Description
Shows configuration parameters for internet High Availability mode.

Syntax
show internet mode

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example
show internet mode

Output
Success shows Internet mode. Failure shows an appropriate error message.

show internet connections

Description
Shows the internet connections settings.

Syntax
show internet-connections

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example
show internet-connections

Output
Success shows information for all the internet connections defined. Failure shows an appropriate error message.

show host

Description
Shows these host settings:
- show host names ipv4 - Shows configuration for all configured hosts
- show host name <host> - Shows configuration for host <host>

Syntax
show host [names ipv4\|name <host>]

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;host&gt;</td>
<td>The name of the host</td>
</tr>
</tbody>
</table>
**Return Value**
0 on success, 1 on failure

**Example**
```
show host name cnn.com
```

**Output**
```
show host - Table containing configuration for all configured hosts. Table columns are: Host Name, and IP Address.
show host name <host> - Table containing configuration for host <host>. Table columns are: Host Name, and IP Address, or Host does not exist if the host does not exist.
```

---

**show hostname**

**Description**
Shows the host name.

**Syntax**
```
show hostname
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show hostname
```

**Output**
```
Host name of the machine.
```

---

**show internet connection icmp servers**

**Description**
Shows a list of ICMP servers.

**Syntax**
```
show internet-connection <connection-name> icmp-servers
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;connection-name&gt;</td>
<td>The internet connection name</td>
</tr>
</tbody>
</table>

**Example**
```
show internet-connection Internet1 icmp-servers
```

**Output**
```
Success shows ICMP server parameters. Failure shows an appropriate error message.
```

---

**show inactivity-timeout**

**Description**
Shows inactivity timeout for web UI and shells assigned to users (in minutes).

**Syntax**
```
show inactivity-timeout
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show inactivity-timeout
```

**Output**
```
Inactivity-timeout: <X> minutes.
<X> is the inactivity timeout for web UI and shells assigned to users.
```
show interface
Description: Shows parameters and status of a specified interface <interface>.
Syntax: `show interface <interface> all`
Parameters:
- `<interface>`: Valid interface name
Example: `show interface WAN all`
Output: Detailed information about the interface.

show interfaces
Description: Shows all interfaces, their ipv4 addresses, and status in a table format.
Syntax: `show interfaces [all]`
Parameters: n/a
Example: `show interfaces`
Output: Shows for every interface: Name, IPv4 Address, Status

show license
Description: Shows current license state.
Syntax: `show license`
Parameters: n/a
Example: `show license`
Output: Current license state

show logs
Description: Shows system, kernel, and traffic logs.
Syntax: `show logs {system|kernel|traffic}`
Parameters: n/a
Example: `show logs kernel`
Output: Success shows log file. Failure shows an appropriate error message.
## show memory usage

**Description**
Shows the amount of memory that is being used.

**Syntax**
```
show memory-usage
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show memory-usage
```

**Output**
Success shows used memory. Failure shows an appropriate error message.

## show network object

**Description**
Shows network objects settings for all defined network objects or for a specified network object `<net-obj-name>`.

**Syntax**
```
show net-obj [names | name <net-obj-name>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;net-obj-name&gt;</code></td>
<td>The name of the network object</td>
</tr>
</tbody>
</table>

**Example**
```
show net-obj names
show net-obj name Network_Object1
```

**Output**
List of all network objects or a specified network object.

## show ntp

The `show ntp` command shows NTP (Network Time Protocol) settings.

### Show NTP Status

**Description**
Indicates if NTP is enabled or disabled.

**Syntax**
```
show ntp active
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show ntp active
```

**Output**
NTP activation: Enabled/Disabled

### Show NTP Servers

**Description**
Shows configured NTP servers.

**Syntax**
```
show ntp servers
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>
Example

show ntp servers

Output

Table with the configured NTP servers.

**show proxy**

**Description**
Shows the current proxy status (enabled or disabled).

**Syntax**
show proxy

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

show proxy

**Output**

The proxy status.

**show restore settings log**

**Description**
Shows the log file of previous restore settings to default operations. You can display these restore settings log files:

- `restore-settings-log` - Log file for restoring saved settings.
- `restore-default-settings-log` - Log file for restoring the default settings.

**Syntax**
show {restore-settings-log|restore-default-settings-log}

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

show restore-settings-log

**Output**

Success shows the restore settings log file. Failure shows an appropriate error message.

**show revert log**

**Description**
Shows the log file of previous revert operations.

**Syntax**
show revert-log

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

show revert-log

**Output**

Success shows the revert log file. Failure shows an appropriate error message.

**show static-routes**

**Description**
Shows the manual routes.

**Syntax**
show static-routes
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

### Example

```
show static-routes
```

### Output

```
Manual routing rules
```

## show rule hits

**Description**

Shows the top firewall policy rule hits.

**Syntax**

```
show rule-hits [top <rule>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;rule&gt;</td>
<td>Number of rules in the security policy that are displayed. Minimum value is 1.</td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

**Example**

```
show rule-hits top 3
```

**Output**

```
Success shows number of hits per rule. Failure shows an appropriate error message.
```

## show saved image

**Description**

Shows information about the saved backup image.

**Syntax**

```
show saved-image
```

**Example**

```
show saved-image
```

**Output**

```
Success shows information about the image. Failure shows an appropriate error message.
```

## show snmp

The `show snmp` command shows information about the SNMP settings on the appliance.

### Show SNMP Agent

**Description**

Shows SNMP agent status or version information. The `show snmp community` command shows the SNMP agent community string (a "password" for SNMP v1 and v2 protocols).

**Syntax**

```
show snmp {agent|agent-version|community}
```
### Show SNM P Host Information

**Description**
Shows information about the SNMP host:
- `show snmp contact` - Information about the host on which the SNMP agent is running.
- `show snmp location` - The location of the SNMP host.

**Syntax**
```
show snmp {contact|location}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show snmp location
```

**Output**
Failure shows an appropriate error message.

### Show SNM P Trap Information

**Description**
Shows information about the SNMP traps. These parameters are shown:
- `status` - Shows SNMP traps status.
- `enabled-traps` - Shows a list of all enabled SNMP traps.
- `receivers` - Shows SNMP trap receivers.

**Syntax**
```
show snmp traps {status|enabled-traps|receivers}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show snmp traps enabled-traps
```

**Output**
Success shows SNMP trap information. Failure shows an appropriate error message.

### Show SNM P Users

**Description**
Shows information about SNMP v3 users. These parameters are shown:
- `show snmp users` - Displays the list of all SNMP v3 users
- `show snmp user <user>` - Displays the information about a specific SNMP v3 user

**Syntax**
```
show {snmp users|snmp user <user>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;user&gt;</td>
<td>SNMP v3 user name</td>
</tr>
</tbody>
</table>

**Example**
```
show snmp user john
```

**Output**
Success shows information about SNMP v3 users. Failure shows an appropriate error message.
show software version

Description  Shows the version of the current software.

Syntax  show software-version | ver

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example  show software-version

Output  Success shows the software version of the appliance. Failure shows an appropriate error message.

show switch

Description  Shows these bridge settings:

- show switches:  Shows settings for all defined switches.
- show switch <switch-name> ports:  Shows the ports that are members of the <switch-name>.

Syntax  show [switches | switch <switch-name> ports ]

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;switch-name&gt;</td>
<td>The name of the bridge</td>
</tr>
</tbody>
</table>

Example  show switches

show switch LAN1_Switch ports

Output  Switch settings and switch ports

show time

Description  Shows the current date in HH-MM-SS format.

Syntax  show time

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example  show time

Output  Current time.

show timezone

Description  Shows the system time zone in format AREA REGION.

Syntax  show timezone

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

Example  show timezone
Output

Time zone in format AREA REGION.

Comments

AREA is a geographic area. REGION is a region inside a specific area.

### show timezone-dst

**Description**
Shows system Daylight Saving Time status.

**Syntax**
```
show timezone-dst
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show timezone-dst
```

**Output**

Automatically adjust clock for daylight saving changes: Disabled

Or

Automatically adjust clock for daylight saving changes: Enabled

### show upgrade log

**Description**
Shows upgrade log files.

**Syntax**
```
show upgrade-log
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show upgrade-log
```

**Output**

Success shows upgrade log files. Failure shows an appropriate error message.

### show user

**Description**
Shows these user related settings:

- `show users type {admin|local}`: Shows a table containing user related preferences for all users.
- `show user <user> type {admin|local} password-hash`: Shows password hash for user `<user>`.
- `show user <user> type local remote-access`: Shows remote access status for user `<user>`.

**Syntax**
```
show user <user> type {admin|local} {password-hash| remote-access}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;user&gt;</code></td>
<td>User login name</td>
</tr>
</tbody>
</table>

**Example**
```
show user John shell type admin
```
Output

```
show user <user> type {admin|local} - Table containing user related preferences for all users. The table contains these columns: Username, Password Hash, password, remote-access, and expert-mode.
```

Comments

```
password-hash is a password MD5 string representation.
```

**show vpn tunnel**

**Description**
Shows all IKE (Internet Key Exchange) and IPSec (Internet Protocol Security) SAs (Security Associations) for the VPN tunnel.

**Syntax**
```
show vpn-tunnel-info
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
show vpn-tunnel-info
```

**Output**
Success shows information about the VPN tunnel. Failure shows an appropriate error message.

**show wlan**

**Show wlan Settings**

**Description**
Shows the main wireless network or the chosen VLAN settings.

**Syntax**
```
show wlan [vap <vap_name>]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;vap_name&gt;</td>
<td>VAP name</td>
</tr>
</tbody>
</table>

**Example**
```
show wlan
```

**Output**
Failure shows an appropriate error message.

**Show VAPS**

**Description**
Shows all the VAPS settings.

**Syntax**
```
show wlan vaps
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

**Example**
```
Show wlan vaps
```

**Output**
Failure shows an appropriate error message.

**Show Radio Configuration**

**Description**
Shows the radio configuration.

**Syntax**
```
show wlan radio
```
### CLI Reference

#### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td></td>
</tr>
</tbody>
</table>

#### Example

`show wlan radio`

#### Output

Failure shows an appropriate error message.

### upgrade from usb or tftp server

**Description**

Upgrades the software image from a file on a USB drive or TFTP server.

**Syntax**

```
upgrade from {usb [file <usb_file>] | tftp server <server> filename <tftp_file>}
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;usb_file&gt;</td>
<td>Name of software image file on USB drive.</td>
</tr>
<tr>
<td>&lt;server&gt;</td>
<td>Host name or IP address of TFTP server.</td>
</tr>
<tr>
<td>&lt;tftp_file&gt;</td>
<td>Name of software image file on TFTP server.</td>
</tr>
</tbody>
</table>

#### Example

`upgrade from tftp server my-tftp-server filename my-new-software`

#### Output

Success shows info after the download succeeds. Failure shows an appropriate error message.

### vpn

The `vpn` command manages the VPN driver and helps to debug the VPN.

#### Managing the VPN Driver

**Description**

Installs the VPN kernel (vpnk) and connects to the firewall kernel (fwk), attaching the VPN driver to the Firewall driver.

**Syntax**

```
vpn drv <on|off>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
</tbody>
</table>

**Return Value**

0 on success, 1 on failure

#### Example

`vpn drv on`

#### Output

Success shows OK. Failure shows an appropriate error message.

### Launching TunnelUtil Tool

**Description**

Launches the VPN TunnelUtil tool to:

- List IKE and IPSec SAs
- Delete IKE and IPSec SAs

**Syntax**

```
vpn tunnelutil
```
### Debugging VPN

**Description**
Contains multiple utilities for troubleshooting VPN issues.

**Syntax**
```plaintext
vpn debug {on [TOPIC=level]|off} [ikeon|ikeoff] [trunc
[TOPIC=level]] [mon|moff]
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>on</td>
<td>off</td>
</tr>
<tr>
<td>[TOPIC=level]</td>
<td>Sets level of debugging for a particular topic. This argument can only be used after on or trunc.</td>
</tr>
<tr>
<td>ikeon</td>
<td>ikeoff</td>
</tr>
<tr>
<td>trunc</td>
<td>Writes both sfwd.elg and ike.elg, but first clears the files</td>
</tr>
<tr>
<td>mon</td>
<td>moff</td>
</tr>
</tbody>
</table>

**Return Value**
0 on success, 1 on failure

**Example**
```plaintext
vpn debug on
```

**Output**
Failure shows an appropriate error message.
Chapter 2

Introduction to Advanced Routing

Dynamic Routing is integrated into the embedded version of the Gaia operating system and command-line shell. BGP, OSPF, and RIP are supported.

Dynamic Multicast Routing is supported, using PIM (Sparse Mode (SM), Dense Mode (DM) and Source-Specific Multicast (SSM)) and IGMP.
Chapter 3

BGP

In This Section:

Support for BGP-4++ .................................................................................................................. 96
BGP Sessions (Internal and External) .......................................................................................... 97
BGP Path Attributes ...................................................................................................................... 98
BGP Multi-Exit Discriminator ....................................................................................................... 99
BGP Interactions with IGPs ........................................................................................................... 99
Inbound BGP Route Filters .......................................................................................................... 99
Redistributing Routes to BGP ....................................................................................................... 99
BGP Communities ....................................................................................................................... 100
Route Reflection ......................................................................................................................... 100
Confederations ............................................................................................................................ 101
EBGP Multihop Support ............................................................................................................... 101
BGP Route Dampening .................................................................................................................. 101
TCP MD5 Authentication ............................................................................................................. 102
Configuring BGP - CLI (bgp) ...................................................................................................... 102

Border Gateway Protocol (BGP) is an inter-AS protocol, meaning that it can be deployed within and between autonomous systems (AS). An autonomous system is a set of routers under a single technical administration. An AS uses an interior gateway protocol and common metrics to route packets within an AS; it uses an exterior routing protocol to route packets to other ASes.

Note - This implementation supports BGP version 4 and 4++.

BGP sends update messages that consist of network number-AS path pairs. The AS path contains the string of ASes through which the specified network can be reached. An AS path has some structure in order to represent the results of aggregating dissimilar routes. These update messages are sent over TCP transport mechanism to ensure reliable delivery. BGP contrasts with IGPs, which build their own reliability on top of a datagram service.

As a path-vector routing protocol, BGP limits the distribution of router reachability information to its peer or neighbor routers.

You can run BGP over a route-based VPN by enabling BGP on a virtual tunnel interface (VTI).

Support for BGP-4++

Gaia Embedded implements BGP-4++ to support multiprotocol extensions.

You must use an IPv4 address for the router ID (BGP identifier). After the BGP session is up, prefixes can be advertised and withdrawn by sending normal UPDATE messages that include either or both of the new multiprotocol attributes MP_REACH_NLRI (used to advertise reachability of routes) and MP_UNREACH_NLRI (used to withdraw routes).

The new attributes are backward compatible. If two routers have a BGP session and only one supports the multiprotocol attributes, they can still exchange unicast IPv4 routes.

On each peer you configure the type of routes (capability) that should be exchanged between peers. Choose this selection:

- IPv4 unicast (the default)
For peering to be established, the routers must share a capability.

**Note** - Do not use the route redistribution and inbound filter pages of the WebUI to configure routing policies for BGP-4++. Instead use the route map commands in the CLI.

**BGP Sessions (Internal and External)**

BGP supports two basic types of sessions between neighbors: internal (sometimes referred to as IBGP) and external (EBGP). Internal sessions run between routers in the *same* autonomous systems, while external sessions run between routers in *different* autonomous systems. When sending routes to an external peer, the local AS number is prepended to the AS path. Routes received from an internal neighbor have, in general, the same AS path that the route had when the originating internal neighbor received the route from an external peer.

BGP sessions might include a single metric (Multi-Exit Discriminator or MED) in the path attributes. Smaller values of the metric are preferred. These values are used to break ties between routes with equal preference from the same neighbor AS.

Internal BGP sessions carry at least one metric in the path attributes that BGP calls the local preference. The size of the metric is identical to the MED. Use of these metrics is dependent on the type of internal protocol processing.

BGP implementations expect external peers to be directly attached to a shared subnet and expect those peers to advertise next hops that are host addresses on that subnet. This constraint is relaxed when the multihop option is enabled in the BGP peer template during configuration.

Type internal groups determine the immediate next hops for routes. They do this by using the next hop received with a route from a peer as a forwarding address, and use this to look up an immediate next hop in IGP routes. Type internal groups support distant peers, but they need to be informed of the IGP whose routes they are using to determine immediate next hops.

Where possible, for internal BGP group types, a single outgoing message is built for all group peers based on the common policy. A copy of the message is sent to every peer in the group, with appropriate adjustments to the next hop field to each peer. This minimizes the computational load of running large numbers of peers in these types of groups.

**Preventing Private AS Numbers from Propagating**

An ISP can assign private AS numbers (64512 to 65535) to a customer in order to conserve globally unique AS numbers. When an ISP does so, a BGP update from a customer network to the ISP has the private AS number in its AS_PATH attribute. When the ISP propagates its network information to other ISPs, the private AS number would normally be included. To avoid this, you can configure Gaia Embedded to remove the private AS number from BGP update messages to external peers.

To configure Gaia Embedded to remove private AS numbers from BGP updates, enable the Remove Private AS option on the configuration page for an external peer.

If you enable this option, private AS numbers are removed from BGP updates according to the following rules:

- If the AS_PATH includes both public and private AS numbers, the private AS numbers are not removed.
- If the AS_PATH contains the AS number of the destination peer, private AS numbers are not removed.
- If the AS_PATH includes confederations and all the AS numbers in the AS_PATH are private, all the private AS numbers are removed.

**BGP Route Refresh**

Gaia Embedded supports the ability to dynamically request BGP route updates from peers and to respond to requests for BGP route updates. For example, if you change the inbound routing policy, you can request that a peer readvertise its previously advertised routes so that the routes can be checked against the new policy. This feature is often referred to as a soft reset because it provides the ability to refresh routes received from a peer without tearing down the established session.

To configure BGP route updates in the:
- **CLI** - Run these commands:
  ```
  set bgp external remote-as as_number peer ip_address send-route-refresh
  set bgp internal peer ip_address send-route-refresh
  ```
- **WebUI** - Click the appropriate buttons in the **Edit Peer** page, in the section **Advanced Settings > Route Refresh**.

These options work only with peers that support the same capabilities. Gaia Embedded systems can also peer with systems that do not support these options.

### BGP Path Attributes

A path attribute is a list of AS numbers that a route has traversed to reach a destination. BGP uses path attributes to provide more information about each route and to help prevent routing loops in an arbitrary topology. You can also use path attributes to determine administrative preferences.

BGP collapses routes with similar path attributes into a single update for advertisement. Routes that are received in a single update are readvertised in a single update. The churn caused by the loss of a neighbor is minimized, and the initial advertisement sent during peer establishment is maximally compressed.

BGP does not read information that the kernel forms message by message. Instead, it fills the input buffer. BGP processes all complete messages in the buffer before reading again. BGP also performs multiple reads to clear all incoming data queued on the socket.

**Note** - This feature might cause a busy peer connection to block other protocols for prolonged intervals.

The following table displays the path attributes and their definitions

<table>
<thead>
<tr>
<th>Path Attribute</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AS_PATH</strong></td>
<td>Identifies the autonomous systems through which routing information carried in an UPDATE message passed. Components of this list can be AS_SETs or ASSEQUENCES.</td>
</tr>
<tr>
<td><strong>NEXT_HOP</strong></td>
<td>Defines the IP address of the border router that should be used as the next hop to the destinations listed in the UPDATE message.</td>
</tr>
<tr>
<td><strong>MULTI_EXIT DISC</strong></td>
<td>Discriminates among multiple exit or entry points to the same neighboring autonomous system. Used only on external links.</td>
</tr>
<tr>
<td><strong>LOCAL_PREF</strong></td>
<td>Determines which external route should be taken and is included in all IBGP UPDATE messages. The assigned BGP speaker sends this message to BGP speakers within its own autonomous system but not to neighboring autonomous systems. Higher values of a LOCAL_PREF are preferred.</td>
</tr>
<tr>
<td><strong>ATOMIC_AGGREGATE</strong></td>
<td>Specifies to a BGP speaker that a less specific route was chosen over a more specific route. The BGP speaker attaches the ATOMIC_AGGREGATE attribute to the route when it reproduces it to other BGP speakers. The BGP speaker that receives this route cannot remove the ATOMIC_AGGREGATE attribute or make any Network Layer Reachability Information (NLRI) of the route more specific. This attribute is used only for debugging purposes.</td>
</tr>
</tbody>
</table>

All unreachable messages are collected into a single message and are sent before reachable routes during a flash update. For these unreachable announcements, the next hop is set to the local address on the connection, no metric is sent, and the path origin is set to incomplete. On external connections, the AS path in unreachable announcements is set to the local AS. On internal connections, the AS path length is set to zero.

Routing information shared between peers in BGP has two formats: announcements and withdrawals. A route announcement indicates that a router either learned of a new network attachment or made a policy decision to prefer another route to a network destination. Route withdrawals are sent when a router makes a new local decision that a network is no longer reachable.
BGP Multi-Exit Discriminator

Multi-exit Discriminator (MED) values are used to help external neighbors decide which of the available entry points into an AS are preferred. A lower MED value is preferred over a higher MED value and breaks the tie between two or more preferred paths.

Note - A BGP session does not accept MEDs from an external peer unless the Accept MED field is set for an external peer.

BGP Interactions with IGPs

All transit ASes must be able to carry traffic that originates from locations outside of that AS, is destined to locations outside of that AS, or both. This requires a certain degree of interaction and coordination between BGP and the Interior Gateway Protocol (IGP) that the particular AS uses. In general, traffic that originates outside of a given AS passes through both interior Gateways (Gateways that support the IGP only) and border Gateways (Gateways that support both the IGP and BGP). All interior Gateways receive information about external routes from one or more of the border Gateways of the AS that uses the IGP.

Depending on the mechanism used to propagate BGP information within a given AS, take special care to ensure consistency between BGP and the IGP, since changes in state are likely to propagate at different rates across the AS. A time window might occur between the moment when some border gateway (A) receives new BGP routing information (which was originated from another border gateway (B) within the same AS) and the moment the IGP within this AS can route transit traffic to the border gateway (B). During that time window, either incorrect routing or black holes can occur.

To minimize such routing problems, border gateway (A) should not advertise to any of its external peers a route to some set of exterior destinations associated with a given address prefix using border gateway (B) until all the interior Gateways within the AS are ready to route traffic destined to these destinations by using the correct exit border gateway (B). Interior routing should converge on the proper exit gateway before advertising routes that use the exit gateway to external peers.

If all routers in an AS are BGP speakers, no interaction is necessary between BGP and an IGP. In such cases, all routers in the AS already have full knowledge of all BGP routes. The IGP is then only used for routing within the AS, and no BGP routes are imported into the IGP. The user can perform a recursive lookup in the routing table. The first lookup uses a BGP route to establish the exit router, while the second lookup determines the IGP path to the exit router.

Inbound BGP Route Filters

BGP routes can be filtered, or redistributed by AS number or AS path regular expression, or both.

BGP stores rejected routes in the routing table with a negative preference. A negative preference prevents a route from becoming active and prevents it from being installed in the forwarding table or being redistributed to other protocols. This behavior eliminates the need to break and re-establish a session upon reconfiguration if importation policy is changed.

The only attribute that can add or modify when you import from BGP is the local preference. The local preference parameter assigns a BGP local preference to the imported route. The local preference is a 32-bit unsigned value, with larger values preferred. This is the preferred way to bias a routing subsystem preference for BGP routes.

Redistributing Routes to BGP

Redistributing to BGP is controlled by an AS. The same policy is applied to all firewalls in the AS. BGP metrics are 16-bit, unsigned quantities; that is, they range from 0 to 65535 inclusive, with zero being the most attractive. While BGP version 4 supports 32-bit unsigned quantities, routed does not.

Note - To define a redistribution policy, use routemaps in the CLI.
BGP Communities

BGP communities allow you to group a set of IP addresses and apply routing decisions based on the identity of the group or community.

To implement this feature, map a set of communities to certain BGP local preference values. Then you can apply a uniform BGP configuration to the community as a whole as opposed to each router within the community. The routers in the community can capture routes that match their community values.

Use community attributes to configure your BGP speaker to set, append, or modify the community of a route that controls which routing information is accepted, preferred, or distributed to other neighbors. The following table displays some special community attributes that a BGP speaker can apply.

<table>
<thead>
<tr>
<th>Community attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO_EXPORT (0xFFFFFF01)</td>
<td>Not advertised outside a BGP confederation boundary. A stand-alone autonomous system that is not part of a confederation should be considered a confederation itself.</td>
</tr>
<tr>
<td>NO_ADVERTISE (0xFFFFFF02)</td>
<td>Not advertised to other BGP peers.</td>
</tr>
<tr>
<td>NO_EXPORT_SUBCONFED (0xFFFFFF03)</td>
<td>Not advertised to external BGP peers. This includes peers in other members’ autonomous systems inside a BGP confederation.</td>
</tr>
</tbody>
</table>

For more about communities, see RFCs 1997 and 1998.

Route Reflection

Generally, all border routers in a single AS need to be internal peers of each other; all nonborder routers frequently need to be internal peers of all border routers. While this configuration is usually acceptable in small networks, it can lead to unacceptably large internal peer groups in large networks. To help address this problem, BGP supports route reflection for internal and routing peer groups (BGP version 4).

When using route reflection, the rule that specifies that a router can not readvertise routes from internal peers to other internal peers is relaxed for some routers called route reflectors. A typical use of route reflection might involve a core backbone of fully meshed routers. This means that all the routers in the fully meshed group peer directly with all other routers in the group. Some of these routers act as route reflectors for routers that are not part of the core group.

Two types of route reflection are supported. By default, all routes received by the route reflector that originate from a client are sent to all internal peers (including the client group but not the client). If the no-client reflect option is enabled, routes received from a route reflection client are sent only to internal peers that are not members of the client group. In this case, the client group must be fully meshed. In either case, all routes received from a non-client internal peer are sent to all route reflection clients.

Typically, a single router acts as the reflector for a set, or cluster, of clients; for redundancy, two or more routers can also be configured to be reflectors for the same cluster. In this case, a cluster ID should be selected to identify all reflectors serving the cluster, using the cluster ID keyword.

Note - Check Point recommends that you not use multiple redundant reflectors unnecessarily as it increases the memory required to store routes on the peers of redundant reflectors.

No special configuration is required on the route reflection clients. From a client perspective, a route reflector is a normal IBGP peer. Any BGP version 4 speaker should be able to be a reflector client.

For further details, refer to the route reflection specification document (RFC 2796 as of this writing).

AS1 has five BGP-speaking routers. With Router B working as a route reflector, there is no need to have all the routers connected in a full mesh.
Confederations

An alternative to route reflection is BGP confederations. As with route reflectors, you can partition BGP speakers into clusters where each cluster is typically a topologically close set of routers. With confederations, this is accomplished by subdividing the autonomous system into multiple, smaller ASes that communicate among themselves. The internal topology is hidden from the outside world, which perceives the confederation to be one large AS.

Each distinct sub-AS within a confederation is referred to as a routing domain (RD). Routing domains are identified by using a routing domain identifier (RDI). The RDI has the same syntax as an AS number, but as it is not visible outside of the confederation, it does not need to be globally unique, although it does need to be unique within the confederation. Many confederations find it convenient to select their RDIs from the reserved AS space (ASes 64512 through 65535 (see RFC 1930)). RDIs are used as the ASes in BGP sessions between peers within the confederation.

The confederation as a whole, is referred to by a confederation identifier. This identifier is used as the AS in external BGP sessions. As far as the outside world is concerned, the confederation ID is the AS number of the single, large AS. For this reason, the confederation ID must be a globally unique, normally assigned AS number.

Note - Do not nest confederations.

For further details, refer to the confederations specification document (RFC 1965 as of this writing).

AS1 has seven BGP-speaking routers grouped under different routing domains: RDI A, RDI B, and RDI C. Instead of having a full-mesh connection among all seven routers, you can have a full-meshed connection within just one routing domain.

EBGP Multihop Support

Connections between BGP speakers of different ASes are referred to as EBGP connections. BGP enforces the rule that peer routers for EBGP connections need to be on a directly attached network. If the peer routers are multiple hops away from each other or if multiple links are between them, you can override this restriction by enabling the EBGP multihop feature. TCP connections between EBGP peers are tied to the addresses of the outgoing interfaces. Therefore, a single interface failure severs the session even if a viable path exists between the peers.

EBGP multihop support can provide redundancy so that an EBGP peer session persists even in the event of an interface failure. Using an address assigned to the loopback interface for the EBGP peering session ensures that the TCP connection stays up even if one of the links between them is down, provided the peer loopback address is reachable. In addition, you can use EBGP multihop support to balance the traffic among all links.

Warning - Enabling multihop BGP connections is dangerous because BGP speakers might establish a BGP connection through a third-party AS. This can violate policy considerations and introduce forwarding loops.

Router A and Router B are connected by two parallel serial links. To provide fault tolerance and enable load-balance, enable EBGP multihop and using addresses on the loopback interface for the EBGP peering sessions.

BGP Route Dampening

Route dampening lessens the propagation of flapping routes. A flapping route is a route that repeatedly becomes available then unavailable. Without route dampening, autonomous systems continually send advertisement and withdrawal messages each time the flapping route becomes available or unavailable. As the Internet has grown, the number of announcements per second has grown as well and caused performance problems within the routers.

Route dampening enables routers to keep a history of the routes that are flapping and prevent them from consuming significant network bandwidth. This is achieved by measuring how often a given route becomes available and then unavailable. When a set threshold is reached, that route is no longer considered valid.
and is no longer propagated for a given period of time, usually about 30 minutes. If a route continues to flap even after the threshold is reached, the time out period for that route grows in proportion to each additional flap. Once the threshold is reached, the route is dampened or suppressed. Suppressed routes are added back into the routing table once the penalty value is decreased and falls below the reuse threshold.

Route dampening can cause connectivity to appear to be lost to the outside world but maintained on your own network because route dampening is only applied to BGP routes. Because of increasing load on the backbone network routers, most NSPs (MCI, Sprint, UUNet etc.) have set up route suppression.

**Note** - BGP route dampening is supported only for EBGP. It is not supported for IBGP.

### TCP MD5 Authentication

The Internet is vulnerable to attack through its routing protocols and BGP is no exception. External sources can disrupt communications between BGP peers by breaking their TCP connection with spoofed RST packets. Internal sources, such as BGP speakers, can inject bogus routing information from any other legitimate BGP speaker. Bogus information from either external or internal sources can affect routing behavior over a wide area in the Internet.

The TCP MD5 option allows BGP to protect itself against the introduction of spoofed TCP segments into the connection stream. To spoof a connection using MD5 signed sessions, the attacker not only has to guess TCP sequence numbers, but also the password included in the MD5 digest.

### Configuring BGP - CLI (bgp)

#### Syntax

**Show Commands:**

```plaintext
show bgp

show bgp
  errors
  groups
  memory
  show bgp paths
  peer VALUE advertise
  peer VALUE detailed
  peer VALUE received
  peers
  peers detailed
  peers established
  routemap
  stats
  summary
```
set bgp internal peer VALUE Commands

set bgp internal peer VALUE
  [ peer-type VALUE ] on
  accept-routes VALUE
  authtype md5 secret VALUE
  authtype none
  capability default
  capability ipv4-unicast VALUE
  graceful-restart-helper off
  graceful-restart-helper on
  graceful-restart-helper-stalepath-time VALUE
  holdtime VALUE
  ignore-first-ashop VALUE
  keepalive VALUE
  local-address VALUE off
  local-address VALUE on
  log-state-transitions VALUE
  log-warnings VALUE
  no-aggregator-id VALUE off
  outgoing-interface VALUE [ peer-type VALUE ] on
  passive-tcp VALUE
  route-refresh off
  route-refresh on
  send-keepalives VALUE
  send-route-refresh request all unicast
  send-route-refresh request ipv4 unicast
  send-route-refresh route-update all unicast
  send-route-refresh route-update ipv4 unicast
  throttle-count VALUE
  trace VALUE off
  trace VALUE on
  weight VALUE
Other Set Commands:

- `set bgp cluster-id VALUE`
- `set bgp communities VALUE`
- `set bgp confederation`
  - `aspath-loops-permitted VALUE`
  - `identifier VALUE`
- `set bgp dampening`
  - `keep-history VALUE`
  - `max-flap VALUE`
  - `off`
  - `on`
  - `reachable-decay VALUE`
  - `reuse-below VALUE`
  - `suppress-above VALUE`
  - `unreachable-decay VALUE`
- `set bgp default-med VALUE`
- `set bgp default-route-gateway VALUE`
- `set bgp external remote-as VALUE`
  - `description VALUE`
  - `export-routemap VALUE off`
  - `export-routemap VALUE preference VALUE [ family VALUE ] on`
  - `import-routemap VALUE off`
  - `import-routemap VALUE preference VALUE [ family VALUE ] on`
  - `local-address VALUE off`
  - `local-address VALUE on`
  - `off`
  - `on`
  - `outdelay VALUE`
  - `virtual-address VALUE`
- `set bgp internal`
  - `description VALUE`
  - `export-routemap VALUE off`
  - `export-routemap VALUE preference VALUE [ family VALUE ] on`
  - `import-routemap VALUE off`
  - `import-routemap VALUE preference VALUE [ family VALUE ] on`
  - `interface VALUE off`
  - `interface VALUE on`
  - `local-address VALUE off`
  - `local-address VALUE on`
  - `med VALUE`
  - `nexthop-self VALUE`
  - `off`
  - `on`
  - `outdelay VALUE`
  - `protocol VALUE off`
  - `protocol VALUE on`
  - `virtual-address VALUE`
- `set bgp routing-domain`
  - `aspath-loops-permitted VALUE`
  - `identifier VALUE`
- `set bgp synchronization VALUE`
set bgp external remote-as VALUE peer VALUE
accept-med VALUE
accept-routes VALUE
aspath-prepend-count VALUE
authtype md5 secret VALUE
authtype none
capability default
capability ipv4-unicast VALUE
graceful-restart-helper off
graceful-restart-helper on
graceful-restart-helper-stalepath-time VALUE
holdtime VALUE
ignore-first-ashop VALUE
kepalive VALUE
local-address VALUE off
local-address VALUE on
log-state-transitions VALUE
log-warnings VALUE
med-out VALUE
multihop VALUE
no-aggregator-id VALUE
off
on
outgoing-interface VALUE on
passive-tcp VALUE
removeprivateas VALUE
route-refresh off
route-refresh on
send-keepalives VALUE
send-route-refresh request all unicast
send-route-refresh request ipv4 unicast
send-route-refresh route-update all unicast
send-route-refresh route-update ipv4 unicast
suppress-default-originate VALUE
throttle-count VALUE
trace VALUE off
trace VALUE on
ttl VALUE

**BGP**

When you do initial configuration, set the router ID. You can also use the following commands to change the router ID.

```
set router-id default
set router-id ip_address
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>Selects the highest interface address when OSPF is enabled.</td>
</tr>
</tbody>
</table>
### Parameter | Description
--- | ---
**ip_address** | The Router ID uniquely identifies the router in the autonomous system. The router ID is used by the BGP and OSPF protocols. We recommend setting the router ID rather than relying on the default setting. This prevents the router ID from changing if the interface used for the router ID goes down. Use an address on a loopback interface that is not the loopback address (127.0.0.1).

**Note** - In a cluster, you must select a router ID and make sure that it is the same on all cluster members.

- **Range**: Dotted-quad.([0-255],[0-255],[0-255],[0-255]). Do not use 0.0.0.0
- **Default**: The interface address of one of the local interfaces.

Use the following group of commands to set and view parameters for BGP.

```plaintext
set as as_number
set as off
```

### Parameter | Description
--- | ---
**as as_number** | The local autonomous system number of the router. This number is mutually exclusive from the confederation and routing domain identifier. The router can be configured with either the autonomous system number or confederation number, not both.

Caution: When you change the autonomous system number, all current peer sessions are reset and all BGP routes are deleted.

**as off** | Disables the configured local autonomous system number.

### External BGP

Use the following commands to configure external sessions of the protocol, that is, between routers in different autonomous systems.
set bgp external remote-as as_number
   <on | off>
   aspath-prepend-count <1-25 | default>
   description text
   local-address ip_address <on | off>
   outdelay <0-65535>
   outdelay off

Arguments

as_number <on | off>  Specifies the autonomous system number of the external peer group. Enter an integer from 1-65535.

aspath-prepend-count <1-25 | default>  Specifies the number of times this router adds to the autonomous system path on external BGP sessions. Use this option to bias the degree of preference some downstream routers have for the routes originated by this router. Some implementations prefer to select paths with shorter autonomous system paths. Default is 1.

description text  You can enter a brief text description of the group.

local-address ip_address <on | off>  Specifies the address used on the local end of the tcp connection with the peer group. The local address must be on an interface that is shared with the peer or with the peer’s gateway when the gateway parameter is used.

outdelay <0-65535>  Specifies the amount of time in seconds that a route must be present in the routing database before it is redistributed to BGP. The configured value applies to all peers configured in this group. This feature dampens route fluctuation. The value zero (0) disables this feature.

   Default: 0

outdelay off  Disables outdelay.

BGP Peers

Use the following commands to configure BGP peers. Gaia Embedded supports IPv4 addresses for BGP peers.
set bgp external remote-as as_number peer ip_address
  <on | off>
  med-out <0–4294967294 | default>
  accept-med <on | off>
  multihop <on | off>
  no-aggregator-id <on | off>
  holdtime <6–65535 | default>
  keepalive <2–21845 | default>
  ignore-first-as-hop <on | off>
  send-keepalives <on | off>
  send-route-refresh [request | route-update][ipv4 All] [unicast]
  route-refresh <on | off>
  accept-routes <all | none>
  passive-tcp <on | off>
  removeprivateas <on | off>
  authtype none
  authtype md5 secret secret
  throttle-count <0–65535 | off>
  ttl <1–255 | default>
  suppress-default-originate <on | off>
  log-state-transitions <on | off>
  log-warnings <on | off>
  trace bgp_traceoption <on | off>
  capability <default | ipv4-unicast>
  graceful-restart-helper <on | off>
  graceful-restart-helper-stalepath-time seconds

Arguments

<on | off>           Specifies a specific peer <ip_address> for the group.

med-out <0–4294967294 | default>      Specifies the multi-exit discriminator (MED) metric used as the primary
                                           metric on all routes sent to the specified peer address. This metric
                                           overrides the default metric on any metric specified by the redistribute
                                           policy. External peers uses MED values to decide which of the available
                                           entry points into an autonomous system is preferred. A lower MED value
                                           is preferred over a higher MED value.
                                           4294967294

accept-med <on | off>          Specifies that MED be accepted from the specified peer address. If you
                                do not set this option, the MED is stripped from the advertisement before
                                the update is added to the routing table.

multihop <on | off>             Enables multihop connections with external BGP peers more than one
                                hop away. By default, external BGP peers are expected to be directly
                                connected. This option can also be used for external load-balancing.

no-aggregator-id <on | off>      Specifies the router’s aggregate attribute as zero (rather than the router
                                ID value). This option prevents different routers in an AS from creating
                                aggregate routes with different AS paths.

holdtime <6–65535 | default>   Specifies the BGP holdtime interval, in seconds, when negotiating a
                                connection with the specified peer. If the BGP speaker does not receive a
                                keepalive update or notification message from its peer within the period
                                specified in the holdtime field of the BGP open message, the BGP
                                connection is closed.
                                180 seconds
The keepalive option is an alternative way to specify a holdtime value in seconds when negotiating a connection with the specified peer. You can use the keepalive interval instead of the holdtime interval. You can also use both intervals, but the holdtime value must be 3 times the keepalive interval value.

60 seconds

Specifies to ignore the first autonomous system number in the autonomous system path for routes learned from the corresponding peer. Set this option only if you are peering with a route server in transparent mode, that is, when the route server is configured to redistribute routes from multiple other autonomous systems without prepending its own autonomous system number.

Specifies for this router always to send keepalive messages even when an update message is sufficient. This option allows interoperability with routers that do not strictly adhere to protocol specifications regarding updates.

Specifies that the router dynamically request BGP route updates from peers or respond to requests for BGP route updates.

Re-learns routes previously sent by the BGP peer or refreshes the routing table of the peer. The peer responds to the message with the current routing table. Similarly, if a peer sends a route refresh request the current routing table is re-sent. A user can also trigger a route update without having to wait for a route refresh request from the peer.

Specifies an inbound BGP policy route if one is not already configured. Enter all to specify accepting routes and installing them with an invalid preference. Depending on the local inbound route policy, these routes are then made active or inactive.

Enter none to delete routes learned from a peer. This option saves memory overhead when many routes are rejected because no inbound policy exists.

Specifies for the router to wait for the specified peer to issue an open message. No tcp connections are initiated by the router.

Specifies that private AS numbers be removed from BGP update messages to external peers.

Specifies not to use an authentication scheme between peers. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.

Specifies to use md5 authentication between peers. In general, peers must agree on the authentication configuration to and from peer adjacencies. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`throttle-count &lt;0-65535</td>
<td>off&gt;`</td>
</tr>
<tr>
<td>`ttl &lt;1-255</td>
<td>default&gt;`</td>
</tr>
<tr>
<td>`suppress-default-origin ate &lt;on</td>
<td>off&gt;`</td>
</tr>
<tr>
<td>`log-state-transitions &lt;on</td>
<td>off&gt;`</td>
</tr>
<tr>
<td>`log-warnings &lt;on</td>
<td>off&gt;`</td>
</tr>
<tr>
<td>`trace bgp_traceoption &lt;on</td>
<td>off&gt;`</td>
</tr>
<tr>
<td></td>
<td>- <code>packets</code>—Trace all BGP packets to this peer.</td>
</tr>
<tr>
<td></td>
<td>- <code>open</code>—Trace all BGP open messages to this peer.</td>
</tr>
<tr>
<td></td>
<td>- <code>update</code>—Trace all BGP update messages to this peer.</td>
</tr>
<tr>
<td></td>
<td>- <code>keepalive</code>—Trace all keepalive messages to this peer.</td>
</tr>
<tr>
<td></td>
<td>- <code>all</code>—Trace all message types.</td>
</tr>
<tr>
<td></td>
<td>- <code>general</code>—Trace message related to Route and Normal.</td>
</tr>
<tr>
<td></td>
<td>- <code>route</code>—Trace routing table changes for routes installed by this peer.</td>
</tr>
<tr>
<td></td>
<td>- <code>normal</code>—Trace normal protocol occurrences. Abnormal protocol occurrences are always traced.</td>
</tr>
<tr>
<td></td>
<td>- <code>state</code>—Trace state machine transitions in the protocol.</td>
</tr>
<tr>
<td></td>
<td>- <code>policy</code>—Trace application of the protocol and user-specified policy to routes being imported and exported.</td>
</tr>
<tr>
<td>`capability &lt;default</td>
<td>ipv4-unicast&gt;`</td>
</tr>
<tr>
<td>`graceful-restart-helper &lt;on</td>
<td>off&gt;`</td>
</tr>
<tr>
<td><code>graceful-restart-helper -stalepath-time seconds</code></td>
<td>Specifies the maximum amount of time that routes previously received from a restarting router are kept so that they can be revalidated. The timer is started after the peer sends an indication that it has recovered.</td>
</tr>
</tbody>
</table>
**BGP Confederation Commands**

Use the following commands to configure BGP confederations. You can configure a BGP confederation in conjunction with external BGP.

```
set bgp
    confederation identifier as_number
    confederation identifier off
    confederation aspath-loops-permitted <1-10>
    confederation aspath-loops-permitted default
    routing-domain identifier as_number
    routing-domain identifier off
    routing-domain aspath-loops-permitted <1-10>
    routing-domain aspath-loops-permitted default
    synchronization <on | off>
```

**Arguments**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>confederation identifier as_number</td>
<td>Specifies the identifier for the entire confederation. This identifier is used as the autonomous system number in external BGP sessions. Outside the confederation, the confederation id is the autonomous system number of a single, large autonomous system. Thus the confederation id must be a globally unique, typically assigned autonomous system number.</td>
</tr>
<tr>
<td>confederation identifier off</td>
<td>Disables the confederation identifier.</td>
</tr>
<tr>
<td>confederation aspath-loops-permitted &lt;1-10&gt;</td>
<td>Specifies the number of times the local autonomous system can appear in an autonomous system path for BGP-learned routes. If this number is higher than the number of times the local autonomous system appears in an autonomous system path, the corresponding routes are discarded or rejected.</td>
</tr>
<tr>
<td>confederation aspath-loops-permitted default routing-domain identifier as_number</td>
<td>Specifies a value of 1.</td>
</tr>
<tr>
<td>routing-domain identifier as_number</td>
<td>Specifies the routing domain identifier (RDI) for this router. You must specify the RDI if you are using BGP confederations. The RDI does not need to be globally unique since it is used only within the domain of the confederation.</td>
</tr>
<tr>
<td>routing-domain identifier off</td>
<td>Disables the routing-domain identifier</td>
</tr>
<tr>
<td>routing-domain aspath-loops-permitted &lt;1-10&gt;</td>
<td>Specifies the number of times the local autonomous system can appear in an autonomous system path for BGP-learned routes. If this number is higher than the number of times the local autonomous system appears in an autonomous system path, the corresponding routes are discarded or rejected.</td>
</tr>
<tr>
<td>routing-domain aspath-loops-permitted default</td>
<td>Specifies a value of 1.</td>
</tr>
<tr>
<td>synchronization &lt;on</td>
<td>off&gt;</td>
</tr>
</tbody>
</table>

**BGP Route Reflection Commands**

Use these commands to configure BGP route reflection. You can configure route reflection as an alternative to BGP confederations. Route reflection supports both internal and external BGP routing groups.
set bgp
  internal peer <ip_address> peer-type reflector-client
  internal peer <ip_address> peer-type none
  internal peer <ip_address> peer-type no-client-reflector
  cluster-id ip_address
  cluster-id off
  default-med <0-65535>
  default-med off
  default-route-gateway ip_address
  default-route-gateway off

Parameter | Description
--- | ---
internal peer <peer ip_address> peer-type reflector-client | The peer router <peer ip_address> is a reflector client of the local router.
internal peer <peer ip_address> peer-type none | The peer router <peer ip_address> is not a reflector client of the local router. This is the default.
internal peer <peer ip_address> peer-type no-client-reflector | An advanced option.
cluster-id ip_address | The cluster ID used for route reflection. The cluster ID default is that of the router id. Override the default if the cluster has more than one route reflector
cluster-id off | Disable the cluster ID.
default-med <0-65535> | The multi-exit discriminator (MED) metric used to advertise routes through BGP.
default-med off | Disable the specified MED metric.
default-route-gateway ip_address | The default route. This route has a higher rank than any configured default static route for this router. If you do not want a BGP peer considered for generating the default route, use the peer <ip_address> suppress-default-originate on command.
default-route-gateway off | Disables the configured default BGP route.

**BGP Route Dampening Commands**

Use the following commands to configure BGP route dampening. BGP route dampening maintains a history of flapping routes and prevents advertising these routes. A route is considered to be flapping when it is repeatedly transitioning from available to unavailable or vice versa.

set bgp dampening
  <on | off>
  suppress-above <2-32>
  suppress-above default
  reuse-below <1-32>
  reuse-below default
  max-flat <3-64>
  max-flat default
  reachable-decay <1-900>
  reachable-decay default
  unreachable-decay <1-2700>
  unreachable-decay default
  keep-history <2-5400>
  keep-history default
Note: BGP route dampening is only supported for External BGP (EBGP).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>suppress-above &lt;2-32&gt;</td>
<td>Specifies the value of the instability metric at which route suppression takes place. A route is not installed in the forwarding table or announced even if it reachable during the period that it is suppressed.</td>
</tr>
<tr>
<td>suppress-above default</td>
<td>Specifies an instability metric value for suppressing routes of 3.</td>
</tr>
<tr>
<td>reuse-below metric &lt;1-32&gt;</td>
<td>Specifies the value of the instability metric at which a suppressed route becomes unsuppressed if it is reachable but currently suppressed. The value assigned to the reuse-below metric must be lower than the suppress-above value.</td>
</tr>
<tr>
<td>reuse-below metric default</td>
<td>Specifies an instability metric value for announcing previously suppressed routes of 2.</td>
</tr>
<tr>
<td>max-flap &lt;3-64&gt;</td>
<td>Specifies the upper limit of the instability metric. The value must be greater than the suppress-above value plus 1. Each time a route becomes unreachable, 1 is added to the current instability metric.</td>
</tr>
<tr>
<td>max-flat default</td>
<td>Specifies the upper limit of the instability metric as 16.</td>
</tr>
<tr>
<td>reachable-decay &lt;1-900&gt;</td>
<td>Specifies the time for the instability metric to reach half of its value when the route is reachable. The smaller the value the sooner a suppressed route becomes reusable.</td>
</tr>
<tr>
<td>reachable-decay default</td>
<td>Specifies a value of 300.</td>
</tr>
<tr>
<td>unreachable-decay &lt;1-2700&gt;</td>
<td>Specifies the time for the instability metric to reach half its value when the route is NOT reachable. The value must be equal to or higher than the reachable-decay value.</td>
</tr>
<tr>
<td>unreachable-decay default</td>
<td>Specifies a value of 900</td>
</tr>
<tr>
<td>keep-history &lt;2-5400&gt;</td>
<td>Specifies the period for which route flapping history is maintained for a given route.</td>
</tr>
<tr>
<td>keep-history default</td>
<td>Specifies a value of 1800.</td>
</tr>
</tbody>
</table>

**Internal BGP**

Use the following commands to configure internal BGP sessions, that is, between routers within the same autonomous system.
set bgp internal
  <on | off>
  description text
  med <0-65535>
  med default
  outdelay <0-65535>
  outdelay off
  nexthop-self <on | off>
  local-address ip_address <on | off>
  interface [all | if_name] <on | off>
  protocol [all | bgp_internal_protocol] <on | off>
  graceful-restart-helper <on | off>
  graceful-restart-helper-stalepath-time seconds
  route-refresh <on | off>
set bgp internal peer ip_address
peert <on | off>
weight <0-65535>
weight off
no-aggregator-id <on | off>
holdtime <6-65535>
holdtime default
keepalive <2-21845>
keepalive default
ignore-first-ashop <on | off>
send-keepalives <on | off>
send-route-refresh [request | route-update] [unicast]
accept-routes all
accept-routes none
passive-tcp <on | off>
auth-type none
auth-type-md5-secret secret
throttle-count <0-65535>
throttle count off
log-state-transitions <on | off>
log-warnings <on | off>
trace bgp_traceoption <on | off>
<table>
<thead>
<tr>
<th><strong>Parameter</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>description text</td>
<td>You can enter a brief text description of the group.</td>
</tr>
<tr>
<td>med &lt;0-65535&gt;</td>
<td>Specifies the amount of time in seconds that a route must be present in the routing database before it is redistributed to BGP. The configured value applies to all peers configured in this group. This feature dampens route fluctuation. Zero (0), which means that this feature is disabled.</td>
</tr>
<tr>
<td>med default</td>
<td></td>
</tr>
<tr>
<td>outdelay &lt;0-65535&gt;</td>
<td>Specifies the amount of time in seconds that a route must be present in the routing database before it is redistributed to BGP. The configured value applies to all peers configured in this group. This feature dampens route fluctuation. Zero (0), which means that this feature is disabled.</td>
</tr>
<tr>
<td>outdelay off</td>
<td>Disables outdelay.</td>
</tr>
<tr>
<td>nexthop-self &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>Default: off</td>
<td></td>
</tr>
<tr>
<td>local-address ip_address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>Note: If running BGP in a cluster you must not configure the local address.</td>
<td></td>
</tr>
<tr>
<td>Default: off</td>
<td></td>
</tr>
<tr>
<td>interface [all</td>
<td>if_name] &lt;on</td>
</tr>
<tr>
<td>protocol [all bgp_internal_protocol] &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>peer ip_address peer_type &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>peer_ip_address weight &lt;0-65535&gt;</td>
<td>Specifies the weight associated with the specified peer. BGP implicitly stores any rejected routes by not mentioning them in a route filter. BGP explicitly mentions them within the routing table by using a restrict keyword with a negative weight. A negative weight prevents a route from becoming active, which prevents it from being installed in the forwarding table or exported to other protocols. This eliminates the need to break and reestablish a session upon reconfiguration if import route policy is changed.</td>
</tr>
<tr>
<td>peer_ip_address weight off</td>
<td>Disables the weight associated with the specified peer.</td>
</tr>
<tr>
<td>peer_ip_address aggregator id &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td><strong>Default</strong>: off</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies the BGP holdtime interval, in seconds, when negotiating a connection with the specified peer. If the BGP speaker does not receive a keepalive update or notification message from its peer within the period specified in the holdtime field of the BGP open message, the BGP connection is closed.</td>
</tr>
<tr>
<td>holdtime &lt;6-65535&gt;</td>
<td></td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies a holdtime of 180 seconds.</td>
</tr>
<tr>
<td>holdtime default</td>
<td></td>
</tr>
<tr>
<td>peer ip_address</td>
<td>The keepalive option is an alternative way to specify a holdtime value in seconds when negotiating a connection with the specified peer. You can use the keepalive interval instead of the holdtime interval. You can also use both interval, but the holdtime value must be 3 times the keepalive interval value.</td>
</tr>
<tr>
<td>keepalive &lt;2-21845&gt;</td>
<td></td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies a keepalive interval of 60 seconds.</td>
</tr>
<tr>
<td>keepalive default</td>
<td></td>
</tr>
<tr>
<td>ignore-first-ashop &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>send-keepalives &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>send-route-refresh [request</td>
<td>route-update] [unicast]</td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies an inbound BGP policy route if one is not already configured. Enter all to specify accepting routes and installing them with an invalid preference. Depending on the local inbound route policy, these routes are then made active or inactive.</td>
</tr>
<tr>
<td>accept-routes all</td>
<td></td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies an inbound BGP policy route if one is not already configured. Enter none to specify deleting routes learned from a peer. This option saves memory overhead when many routes are rejected because no inbound policy exists.</td>
</tr>
<tr>
<td>accept-routes none</td>
<td></td>
</tr>
<tr>
<td>passive-tcp &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Default: off</td>
</tr>
<tr>
<td>authtype none</td>
<td>Specifies not to use an authentication scheme between peers. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.</td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies to use md5 authentication between peers. In general, peers must agree on the authentication configuration to and from peer adjacencies. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.</td>
</tr>
<tr>
<td>authtype md5 secret secret</td>
<td></td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Specifies the number of BGP updates to send at one time. The throttle count option limits the number of BGP updates when there are many BGP peers.</td>
</tr>
<tr>
<td>throttle-count &lt;0-65535&gt;</td>
<td></td>
</tr>
<tr>
<td>peer ip_address</td>
<td>Disables the throttle count option.</td>
</tr>
<tr>
<td>throttle_count off</td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>peer ip_address log-state-transitions &lt;on</td>
<td>Enables or disables logging of state transitions for the peer router.</td>
</tr>
<tr>
<td>peer ip_address log-warnings &lt;on</td>
<td>Enables or disables logging of warnings for the peer router.</td>
</tr>
<tr>
<td>peer ip_address trace bgp_traceoption &lt;on</td>
<td>Enables or disables tracing options for the BGP trace option. Log messages are saved in the var/log/routed directory. Enter the following words to set each trace option. Enter packets to trace all BGP packets to this peer. Enter open to trace all the BGP open messages to this peer. Enter update to trace all the BGP update messages to this peer. Enter keepalive to trace all the keepalive messages to this peer. Enter all to trace all the message types. Enter general to trace message related to Route and Normal. Enter route to trace routing table changes for routes installed by this peer. Enter normal to trace normal protocol occurrences. Abnormal protocol occurrences are always traced. Enter state to trace state machine transitions in the protocol. Enter policy to trace application of the protocol and user-specified policy to routes being imported and exported.</td>
</tr>
<tr>
<td>graceful-restart-helper &lt;on</td>
<td>Enables or disables graceful restart helper for the peer router.</td>
</tr>
<tr>
<td>graceful-restart-helper-stalepath-time seconds</td>
<td>Specifies the maximum amount of time that routes previously received from a restarting router are kept before they are revalidated. The timer is started after the peer sends an indication that it has recovered.</td>
</tr>
<tr>
<td>route-refresh &lt;on</td>
<td>Relearns routes previously sent by the BGP peer or refreshes the routing table of the peer. The peer responds to the message with the current routing table. Similarly, if a peer sends a route refresh request the current routing table is re-sent. A user can also trigger a route update without having to wait for a route refresh request from the peer.</td>
</tr>
</tbody>
</table>

**BGP Communities Commands**

Use the following command to configure BGP communities. A BGP community is a group of destinations that share the same property. However, a community is not restricted to one network or autonomous system. Use communities to simplify the BGP inbound and route redistribution policies. Use the BGP communities commands together with inbound policy and route redistribution.

```
set bgp communities <on | off>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;on</td>
<td>Enables or disables BGP policy options based on communities.</td>
</tr>
</tbody>
</table>
**BGP Show Commands**

Use the following commands to monitor and troubleshoot your BGP implementation.

```
show bgp
show bgp
groups
memory
errors
paths
stats
peers
peers detailed
peer ip_address detailed
peers established
peer ip_address advertise
peer ip_address received
summary
```
Chapter 4

IGMP

In This Section:

Configuring IGMP - CLI (igmp) ................................................................. 121

Internet Group Management Protocol (IGMP) allows hosts on multiaccess networks to inform locally attached routers of their group membership information. Hosts share their group membership information by multicasting IGMP host membership reports. Multicast routers listen for these host membership reports, and then exchange this information with other multicast routers.

The group membership reporting protocol includes two types of messages: host membership query and host membership report. IGMP messages are encapsulated in IP datagrams, with an IP protocol number of 2. Protocol operation requires that a designated querier router be elected on each subnet and that it periodically multicast a host membership query to the all-hosts group.

Hosts respond to a query by generating host membership reports for each multicast group to which they belong. These reports are sent to the group being reported, which allows other active members on the subnet to cancel their reports. This behavior limits the number of reports generated to one for each active group on the subnet. This exchange allows the multicast routers to maintain a database of all active host groups on each of their attached subnets. A group is declared inactive (expired) when no report is received for several query intervals.

The IGMPv2 protocol adds a leave group message and uses an unused field in the IGMPv.1 host membership query message to specify a maximum response time. The leave group message allows a host to report when its membership in a multicast group terminates. Then, the IGMP querier router can send a group-directed query with a very small maximum response time to probe for any remaining active group members. This accelerated leave extension can reduce the time required to expire a group and prune the multicast distribution tree from minutes, down to several seconds.

The unicast traceroute program allows the tracing of a path from one device to another, using mechanisms that already exist in IP. Unfortunately, you cannot apply such mechanisms to IP multicast packets. The key mechanism for unicast traceroute is the ICMP TTL exceeded message that is specifically precluded as a response to multicast packets. The traceroute facility implemented within routed conforms to the traceroute facility for IP multicast draft specification.


Gaia Embedded supports these IGMP features:

- Multicast traceroute
- Configurability of protocol timers
- Support for interfaces with secondary addresses

Monitoring template using the WebUI, you can configure the following options:

- Version number
- Loss robustness
- Query interval
- Query response interval
- Last-member query interval

Additionally, you can enable and disable router alert.

Check Point supports IGMP in a gateway as part of the support for PIM. The support of IGMP ensures synchronization of IGMP state from master to members when a new member running PIM joins the cluster.
Configuring IGMP - CLI (igmp)

Use the IGMP commands to configure parameters for the internet group management protocol.

Configure Interfaces for IGMP

Use these commands to configure IGMP for specific interfaces.

```
set igmp interface if_name
   version <1 | 2 | 3>
   last-member-query-interval <1-25>
   last-member-query-interval default
   loss-robustness <1-255>
   loss-robustness default
   query-interval <1-3600>
   query-interval default
   query-response-interval <1-25>
   query-response-interval default
   router-alert <on | off>
   static-group address <on | off>
   local-group address <on | off>
```

Note -

IGMP version 2 runs by default.

In a gateway cluster, run commands on every cluster member. The configuration of each cluster member must be identical.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface if_name</td>
<td>The interface on which IGMP should be configured.</td>
</tr>
<tr>
<td>last-member-query-interval</td>
<td>The maximum response time (in seconds) inserted into IGMP group-specific queries. The last member query interval may be used to tune the &quot;leave latency.&quot; A smaller value results in a reduction in the time to detect the loss of the last member of a multicast group. This value must always be less than the query interval.</td>
</tr>
<tr>
<td>loss-robustness</td>
<td>Allows tuning for the expected packet loss on a subnet. If the subnet is expected to be highly lossy, then the &quot;loss robustness&quot; value may be increased. IGMP protocol operation is robust to (lossrobustness - 1) packet loss</td>
</tr>
<tr>
<td>query-interval</td>
<td>The interval (in seconds) between IGMP general queries sent by the querier router. This parameter can be used to tune the IGMP messaging overhead and has a secondary effect on the timeout of idle IP multicast groups.</td>
</tr>
<tr>
<td>query-interval default</td>
<td>A value of 125.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>query-response-interval &lt;1-25&gt;</td>
<td>The maximum response time (in seconds) inserted into the periodic IGMP general queries. The query response interval may be used to tune the burstiness of IGMP messages; a larger value spreads the host IGMP reports over a larger interval, reducing burstiness. This value must always be less than the query interval.</td>
</tr>
<tr>
<td>router-alert &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>local-group address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>static-group address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>version &lt;1</td>
<td>2</td>
</tr>
</tbody>
</table>

**Monitoring IGMP (show igmp)**

Use these commands to monitor and troubleshoot IGMP.

```
show igmp
  stats
    stats receive
    stats transmit
    stats error
    interfaces
    interfaces [if_address
    groups [interface logical_interface] [local | static]
    group [if_address
    if-stats
    if-stat [if_address
    summary
```
Chapter 5

RIP

In This Section:

- RIP 2 .................................................................................................................................................. 123
- RIP 1 .................................................................................................................................................. 124
- Configuring RIP - CLI (rip) .................................................................................................................. 124

The Routing Information Protocol (RIP) is one of the oldest, and still widely used, interior gateway protocols (IGP). RIP uses only the number of hops between nodes to determine the cost of a route to a destination network and does not consider network congestion or link speed. Other shortcomings of RIP are that it can create excessive network traffic if there are a large number of routes and that it has a slow convergence time and is less secure than other IGPs, such as OSPF.

Routers using RIP broadcast their routing tables on a periodic basis to other routers, whether or not the tables have changed. Each update contains paired values consisting of an IP network address and a distance to that network. The distance is expressed as an integer, the hop count metric. Directly connected networks have a metric of 1. Networks reachable through one other router are two hops, and so on. The maximum number of hops in a RIP network is 15 and the protocol treats anything equal to or greater than 16 as unreachable.

RIP 2

The RIP version 2 protocol adds capabilities to RIP. Some of the most notable RIP 2 enhancements follow.

**Network Mask**

The RIP 1 protocol assumes that all subnetworks of a given network have the same network mask. It uses this assumption to calculate the network masks for all routes received. This assumption prevents subnets with different network masks from being included in RIP packets. RIP 2 adds the ability to explicitly specify the network mask for each network in a packet.

**Authentication**

RIP 2 packets also can contain one of two types of authentication methods that can be used to verify the validity of the supplied routing data.

The first method is a simple password in which an authentication key of up to 16 characters is included in the packet. If this password does not match what is expected, the packet is discarded. This method provides very little security, as it is possible to learn the authentication key by watching RIP packets.

The second method uses the MD5 algorithm to create a crypto checksum of a RIP packet and an authentication key of up to 16 characters. The transmitted packet does not contain the authentication key itself; instead, it contains a crypto-checksum called the digest. The receiving router performs a calculation using the correct authentication key and discards the packet if the digest does not match. In addition, a sequence number is maintained to prevent the replay of older packets. This method provides stronger assurance that routing data originated from a router with a valid authentication key.
RIP 1

**Network Mask**

RIP 1 derives the network mask of received networks and hosts from the network mask of the interface from which the packet was received. If a received network or host is on the same natural network as the interface over which it was received, and that network is subnetted (the specified mask is more specific than the natural network mask), then the subnet mask is applied to the destination. If bits outside the mask are set, it is assumed to be a host; otherwise, it is assumed to be a subnet.

**Auto Summarization**

The Check Point implementation of RIP 1 supports auto summarization; this allows the router to aggregate and redistribute nonclassful routes in RIP 1.

**Configuring RIP - CLI (rip)**

**Description**

Use these commands to configure RIP properties for specific interfaces.

**Syntax**

**Set Commands:**

```
set rip auto-summary VALUE
set rip expire-interval VALUE
set rip export-routemap VALUE
   off
   preference VALUE on
set rip import-routemap VALUE
   off
   preference VALUE on
set rip interface VALUE
   [ version VALUE ] on
   accept-updates VALUE
   metric VALUE
   off
   send-updates VALUE
   transport VALUE
   virtual-address VALUE

set rip interface VALUE authtype
   md5 secret VALUE [ cisco-compatibility VALUE ]
   none
   simple VALUE
```

**Show Commands:**

```
show rip
show rip
   errors
   interface VALUE
   interfaces
   neighbors
   packets
   routemap
   summary
```

Use this group of commands to set and view parameters for RIP.
Note - Gaia Embedded does not have CLI commands for route filtering and redistribution. You must configure inbound routing policies and redistribution of routes through the WebUI. You can configure route maps and route aggregation using CLI commands. Route map configuration done through the CLI takes precedence over route filtering and redistribution configured in the WebUI. For example, if RIP uses route maps for inbound filtering, anything configured on the page for inbound route filters for RIP is ignored. You can still use the WebUI to configure route redistribution into RIP.

**RIP Interface Commands**

Use these commands to configure RIP properties that apply to a RIP interface.

```
set rip interface if_name
  <off |on>
  version <1 | 2> on
  metric <0-16>
  metric default
  accept-updates <on | off>
  send-updates <on | off>
  transport <multicast | broadcast>
  authtype none
  authtype simple password
  authtype md5 secret secret [cisco-compatibility] <on | off>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface if_name</td>
<td>Turn on or turn off RIP on the interface.</td>
</tr>
<tr>
<td>&lt;off</td>
<td>on&gt;</td>
</tr>
<tr>
<td>&lt;1</td>
<td>2&gt;</td>
</tr>
<tr>
<td>metric &lt;0-16&gt;</td>
<td>The RIP metric to be added to routes that are sent using the specified interface(s). The default is zero. This is used to make other routers prefer other sources of RIP routes over this router.</td>
</tr>
<tr>
<td>metric default</td>
<td>A value of 0.</td>
</tr>
<tr>
<td>accept-updates</td>
<td>Whether RIP packets from other routers using the interface are accepted or ignored. Ignoring an update may result in suboptimal routing.</td>
</tr>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>send-updates</td>
<td>Whether RIP packets should be sent via the interface. This causes the interface to be a passive RIP listener.</td>
</tr>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>transport</td>
<td>The transport mechanism.</td>
</tr>
<tr>
<td>&lt;multicast</td>
<td>broadcast&gt;</td>
</tr>
<tr>
<td></td>
<td>Selecting Multicast specifies that RIP version 2 packets should be multicast on this interface. This is the default.</td>
</tr>
<tr>
<td></td>
<td>Note - When you use RIP 2, always select multicast. We recommend that you do not operate RIP 1 and RIP 2 together.</td>
</tr>
<tr>
<td></td>
<td>Selecting Broadcast specifies that RIP version 1 packets that are compatible with version 2 should be broadcast on this interface.</td>
</tr>
</tbody>
</table>
### Parameter | Description
--- | ---
authtype none | There is no authentication scheme for the interface to accept routing information from neighboring routers. This option applies to RIP version 2 only. In general, routers on a given link must agree on the authentication configuration in order to form neighbor adjacencies. This is used to guarantee that routing information is accepted only from trusted routers.

authtype simple password | Implement a simple authentication scheme for the interface to accept routing information from neighboring routers. Enter the Simple Password, from 1 to 16 characters. Must contain alphanumeric characters only. This option applies to RIP version 2 only.

authtype md5 secret secret | Implement an authentication scheme that uses an MD5 algorithm for the interface to accept routing information from neighboring routers. Enter the password.

cisco-compatibility <on | off> | To ensure interoperability with Cisco routers running RIP MD5 authentication, enable Cisco Compatibility. By default, RIP MD5 is set to conform to the Check Point standard, and not for Cisco compatibility. **Default**: off

### RIP Global Commands

Use these commands to configure RIP properties that apply to all interfaces configured for RIP.

```
set rip
  auto-summary <on | off>
  update-interval <1-65535>
  update-interval default
  expire-interval <1-65535>
  expire-interval default
```

| Parameter | Description |
--- | --- |
auto-summary <on | off> | Automatically aggregates and redistributes non-classful RIP Version 1 into RIP. This applies only to RIP Version 1. If the Auto summarization field option is unchecked, you must do the aggregation and redistribution manually by using route aggregation and route redistribution. **Note** - Take care when you set this parameter, as RIP has no protocol mechanism to detect misconfiguration. **Default**: on

update-interval <1-65535> | The amount of time, in seconds, between regularly scheduled RIP updates. To prevent synchronization of periodic updates, RIP updates are actually sent at a time from the uniform distribution on the interval (0.5T, 1.5T) where T corresponds to the Update Interval value. **Note** - Take care when you set this parameter, as RIP has no protocol mechanism to detect misconfiguration.

update-interval default | A value of 30 seconds.

expire-interval <1-65535> | The amount of time, in seconds, that must pass without receiving an update for a given route before the route is considered to have timed out. This value should be 6 times the update interval in order to allow for the possibility that packets containing an update could be dropped by the network.

expire-interval default | A value of 180 seconds.
RIP Show Commands

Use these commands to monitor and troubleshoot RIP.

show rip

show rip
    interfaces
    interface <if_name>
    packets
    errors
    neighbors
    summary
Chapter 6

OSPF

In This Section:

Types of Areas ................................................................. 128
Area Border Routers .......................................................... 129
High Availability Support for OSPF ....................................... 129
Configuring OSPF - CLI (ospf) ............................................ 130

Open Shortest Path First (OSPF) is an interior gateway protocol (IGP) used to exchange routing information between routers within a single autonomous system (AS). OSPF calculates the best path based on true costs using a metric assigned by a network administrator. RIP, the oldest IGP protocol chooses the least-cost path based on hop count. OSPF is more efficient than RIP, has a quicker convergence, and provides equal-cost multipath routing where packets to a single destination can be sent using more than one interface. OSPF is suitable for complex networks with a large number of routers. It can coexist with RIP on a network.

Gaia Embedded supports OSPFv2, which supports IPv4 addressing.

You can run OSPF over a route-based VPN by enabling OSPF on a virtual tunnel interface (VTI).

Types of Areas

Routers using OSPF send packets called Link State Advertisements (LSA) to all routers in an area. Areas are smaller groups within the AS that you can design to limit the flooding of an LSA to all routers. LSAs do not leave the area from which they originated, thus increasing efficiency and saving network bandwidth.

You must specify at least one area in your OSPF network—the backbone area, which has the responsibility to propagate information between areas. The backbone area has the identifier 0.0.0.0.

You can designate other areas, depending on your network design, of the following types:

- **Normal Area** — Allows all LSAs to pass through. The backbone is always a normal area.
- **Stub Area** — Stub areas do not allow Type 5 LSAs to be propagated into or throughout the area and instead depends on default routing to external destinations. You can configure an area as a stub to reduce the number of entries in the routing table (routes external to the OSPF domain are not added to the routing table).
- **NSSA (Not So Stubby Area)** — Allows the import of external routes in a limited fashion using Type-7 LSAs. NSSA border routers translate selected Type 7 LSAs into Type 5 LSAs which can then be flooded to all Type-5 capable areas. Configure an area as an NSSA if you want to reduce the size of the routing table, but still want to allow routes that are redistributed to OSPF.

It is generally recommended that you limit OSPF areas to about 50 routers based on the limitations of OSPF (traffic overhead, table size, convergence, and so on).

All OSPF areas must be connected to the backbone area. If you have an area that is not connected to the backbone area, you can connect it by configuring a virtual link, enabling the backbone area to appear contiguous despite the physical reality.

**Note** - If you need to connect two networks that both already have backbone areas and you do not want to reconfigure one to something other than 0.0.0.0, you can connect the two backbone areas using a virtual link.

Each router records information about its interfaces when it initializes and builds an LSA packet. The LSA contains a list of all recently seen routers and their costs. The LSA is forwarded only within the area it
originated in and is flooded to all other routers in the area. The information is stored in the link-state database, which is identical on all routers in the AS.

Area Border Routers

Routers called Area Border Routers (ABR) have interfaces to multiple areas. ABRs compact the topological information for an area and transmit it to the backbone area. Check Point supports the implementation of ABR behavior as outlined in the Internet draft of the Internet Engineering Task Force (IETF). The definition of an ABR in the OSPF specification as outlined in RFC 2328 does not require a router with multiple attached areas to have a backbone connection. However, under this definition, any traffic destined for areas that are not connected to an ABR or that are outside the OSPF domain is dropped. According to the Internet draft, a router is considered to be an ABR if it has more than one area actively attached and one of them is the backbone area. An area is considered actively attached if the router has at least one interface in that area that is not down.

Rather than redefine an ABR, the Check Point implementation includes in its routing calculation summary LSAs from all actively attached areas if the ABR does not have an active backbone connection, which means that the backbone is actively attached and includes at least one fully adjacent neighbor. You do not need to configure this feature; it functions automatically under certain topographies.

OSPF uses the following types of routes:

- **Intra-area**—Have destinations within the same area.
- **Interarea**—Have destinations in other OSPF areas.
- **Autonomous system external (ASE)**—Have destinations external to the autonomous system (AS). These are the routes calculated from Type 5 LSAs.
- **NSSA ASE Router**—Have destinations external to AS. These are the routes calculated from Type 7 LSAs.

All routers on a link must agree on the configuration parameters of the link. All routers in an area must agree on the configuration parameters of the area. A separate copy of the SPF algorithm is run for each area. Misconfigurations prevent adjacencies from forming between neighbors, and routing black holes or loops can form.

High Availability Support for OSPF

Gaia Embedded supports the OSPF protocol in clusters configured via ClusterXL.

In this configuration, the cluster becomes a Virtual Router, which is seen by neighboring routers as a single router that has an IP address that is the same as the virtual IP address of the cluster. Each member of the cluster runs the OSPF task, but only the member which is designated as primary or master actively participates in the network and exchanges routing information with neighbor routers. When a failover occurs, the standby member of the cluster becomes the master and its OSPF task becomes the active participant in protocol exchanges with neighbor routers.

Gaia Embedded also supports the OSPF protocol over VPN tunnels which terminate in the ClusterXL cluster.

ClusterXL

Gaia Embedded ClusterXL advertises the virtual IP address of the ClusterXL Virtual Router. The OSPF routes database of the master is synchronized across all members of the cluster. The OSPF task of each standby member obtains routing state and information from the master and installs the routes in the kernel as the master does. On a failover, one of the standby members becomes the new master and then continues where the old master failed. During the time that the new master resynchronizes routes database with the neighbor routers, traffic forwarding continues using the old kernel routes until OSPF routes are fully synchronized and pushed into the kernel.
Configuring OSPF - CLI (ospf)

Use the following group of commands to set and view parameters for OSPF. This syntax is shown below for each set of commands.

**Note** - Gaia Embedded does not have CLI commands for route filtering and redistribution. You must configure inbound routing policies and redistribution of routes through the WebUI. You can configure route maps and route aggregation using CLI commands. Route map configuration done through the CLI takes precedence over route filtering and redistribution configured in the WebUI. For example if OSPF uses route maps for inbound filtering and redistribution configured in the WebUI page for inbound route filters for OSPF is ignored. You can still use the WebUI to configure route redistribution into OSPF.

When you do initial configuration, set the router ID. You can also use the following commands to change the router ID.

```plaintext
set router-id
  default
  ip_address
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>router-id default</td>
<td>Selects the highest interface address when OSPF is enabled.</td>
</tr>
<tr>
<td>router-id ip_address</td>
<td>Specifies a specific IP address to assign as the router ID. Do not use 0.0.0.0 as the router ID address. Check Point recommends setting the router ID rather than relying on the default setting. Setting the router ID prevents the ID from changing if the default interface used for the router ID goes down. The Router ID uniquely identifies the router in the autonomous system. The router ID is used by the BGP and OSPF protocols. We recommend setting the router ID rather than relying on the default setting. This prevents the router ID from changing if the interface used for the router ID goes down. Use an address on a loopback interface that is not the loopback address (127.0.0.1). <strong>Note</strong> - In a cluster, you must select a router ID and make sure that it is the same on all cluster members.</td>
</tr>
<tr>
<td></td>
<td><strong>Range</strong>: Dotted-quad.([0-255],[0-255],[0-255],[0-255]). Do not use 0.0.0.0</td>
</tr>
<tr>
<td></td>
<td><strong>Default</strong>: The interface address of one of the local interfaces.</td>
</tr>
</tbody>
</table>

**OSPF Global Settings**

Use the following commands to configure setting that apply to all configured OSPF areas, including the backbone and stub areas.

For OSPFv2 use the following commands:
Parameters

rfc1583-compatibility <on | off>
The Check Point implementation of OSPF is based on RFC 2178, which fixed some looping problems in an earlier specification of OSPF. If your implementation runs in an environment with OSPF implementations based on RFC 1583 or earlier, enable this option, which is on by default. Setting compatibility with RFC 1583 ensures backward compatibility. on

spf-delay <1-60>
Specifies the time, in seconds, to wait before recalculating the OSPF routing table after a change in the topology.

spf-delay default
Specifies an spf-delay time of 2 seconds.

spf-holdtime <1-60>
Specifies the minimum time, in seconds, between recalculations of the OSPF routing table.

spf-holdtime default
Specifies an spf-holdtime of 5 seconds.

default-ase-cost <1-6777215>
Specifies the cost assigned to routes from other protocols that are redistributed into OSPF as autonomous systems external. If the route has a cost already specified, that cost takes precedent.

1
default-ase-type <1 | 2>
Specifies the type assigned to routes from other protocols that are redistributed into OSPF as autonomous systems external. If the route has a type already specified, that type takes precedent.

1
graceful-restart-helper <on | off>
Specifies whether the Check Point system should maintain the forwarding state advertised by peer routers even when they restart to minimize the negative effects caused by peer routers restarting.

OSPF Interfaces

Use the following commands to configure a backbone and other areas, such as stub areas, for specified interfaces.

For OSPFv2 use the following commands:
set ospf
  area <backbone | ospf_area> range ip_prefix <on | off>
  area <backbone | ospf_area> range ip_prefix restrict <on | off>
  stub-network ip_prefix <on | off>
  stub-network ip_prefix stub-network-cost <1-677722>

set ospf interface if_name
  area <backbone | ospf_area> <on | off>
  hello-interval <1-65535>
  dead-interval <1-65535>
  retransmit-interval <1-65535>
  cost <1-65535>
  priority <0-255>
  passive <on | off>
  authtype none
  simple password
  md5 key authorization key id secret md5 secret
  md5 key authorization key id

Parameters

area <backbone | ospf_area> range ip_prefix <on | off>
  Specifies the OSPF area to which the specified interface range belongs. Select an area from the areas already configured. Any area can be configured with any number of address ranges. These ranges are used to reduce the number of routing entries that a given area transmits to other areas. If a given prefix aggregates a number of more specific prefixes within an area, you can configure an address range that becomes the only prefix advertised to other areas. Be careful when configuring an address range that covers part of a prefix that is not contained within an area. An address range is defined by an IP prefix and a mask length. If you mark a range as restrict, it is not advertised to other areas.

area <backbone | ospf_area> range ip_prefix restrict <on | off>
  Any area can be configured with any number of address ranges. These ranges are used to reduce the number of routing entries that a given area transmits to other areas. If a given prefix aggregates a number of more specific prefixes within an area, you can configure an address range that becomes the only prefix advertised to other areas. Be careful when configuring an address range that covers part of a prefix that is not contained within an area. An address range is defined by an IP prefix and a mask length. If you mark a range as restrict, it is not advertised to other areas.

stub-network ip_prefix <on | off>
  Specifies a stub network to which the specified interface range belongs. Configure a stub network to advertise reachability to prefixes that are not running OSPF. The advertised prefix appears as an OSPF internal route and is filtered at area borders with the OSPF area ranges. The prefix must be directly reachable on the router where the stub network is configured, that is, one of the router’s interface addresses must fall within the prefix range to be included in the router-link-state advertisement. Use a mask length of 32 to configure the stub host. The local address of a point-to-point interface can activate the advertised prefix and mask. To advertise reachability to such an address, enter an IP address for the prefix and a non-zero cost for the prefix.
stub-network  ip_prefix  
stub-network-cost <1-677722>

Configure a stub network to advertise reachability to prefixes that are not running OSPF. The advertised prefix appears as an OSPF internal route and is filtered at area borders with the OSPF area ranges. The prefix must be directly reachable on the router where the stub network is configured, that is, one of the router’s interface addresses must fall within the prefix range to be included in the router-link-state advertisement. Use a mask length of 32 to configure the stub host. The local address of a point-to-point interface can activate the advertised prefix and mask. To advertise reachability to such an address, enter an IP address for the prefix and a non-zero cost for the prefix.

interface if_name area 
<backbone  | ospf area> 
<on   | off>

Specifies the OSPF area to which the specified interface belongs.

interface if_name 
hello-interval <1-65535>

Specifies the interval, in seconds, between hello packets that the router sends on the specified interface. For a given link, this value must be the same on all routers or adjacencies do not form.

interface if_name 
hello-interval default

Specifies the default value for the hello interval, which is 10 seconds.

interface if_name 
dead-interval <1-65535>

Specifies the number of seconds after which a router stops receiving hello packets that it declares the peer down. Generally, you should set this value at 4 times the value of the hello interval. Do not set the value at 0. For a given link, this value must be the same on all routers or adjacencies do not form.

interface if_name 
dead-interval default

Specifies the default value for the dead interval, which is 40 seconds.

interface if_name 
retransmit-interval <1-65535>

Specifies the number of seconds between link state advertisement transmissions for adjacencies belonging to the specified interface. This value also applies to database description and link state request packets. Set this value conservatively, that is, at a significantly higher value than the expected round-trip delay between any two routers on the attached network.

interface if_name 
retransmit-interval default

Specifies the default for the retransmit interval, which is 5 seconds.

interface if_name 
cost <1-65535>

Specifies the weight of the given path in a route. The higher the cost, the less preferred the link. To use one interface over another for routing paths, assign one a higher cost.

interface if_name 
priority <0-255>

Specifies the priority for becoming the designated router (DR) on the specified link. When two routers attached to a network attempt to become a designated router, the one with the highest priority wins. This option prevents the DR from changing too often. The DR option applies only to a share-media interface, such as Ethernet or FDDI; a DR is not elected on a point-to-point type interface. A router with a priority of 0 is not eligible to become the DR.
interface if_name passive
<on | off>

Enabling this option puts the specified interface into passive mode; that is, hello packets are not sent from the interface. Putting an interface into passive mode means that no adjacencies are formed on the link. This mode enables the network associated with the specified interface to be included in intra-area route calculation rather than redistributing the network into OSPF and having it function as an autonomous system external.

Default: off

interface if_name authtype none

Specifies not to use an authentication scheme for the specified interface.

interface if_name authtype simple password

Specifies to use simple authentication for the specified interface. Enter an ASCII string that is 8 characters long. Generally, routers on a given link must agree on the authentication configuration to form peer adjacencies. Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.

interface if_name authtype md5 key authorization key id secret md5 secret

Specifies to use MD5 authorization. Enter at least one key ID and its corresponding MD5 secret. If you configure multiple key IDs, the largest key ID is used for authenticating outgoing packets. All keys can be used to authenticate incoming packets. Generally, routers on a given link must agree on the authentication configuration to form peer adjacencies. Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.

OSPF Virtual Links

Use the following commands to configure OSPF virtual links. Configure a virtual link if the router is a border router that does not have interfaces in the backbone area. The virtual link is effectively a tunnel across an adjacent non-backbone area whose endpoint must be any of the adjacent area’s border routers that has an interface in the backbone area.

For OSPFv2 use the following commands:

```
set ospf area backbone virtual-link ip_address transit-area ospf_area
<on | off>
   transit-area ospf_area hello-interval <1-65535>
   transit-area ospf_area hello-interval default
   transit-area ospf_area dead interval <1-4294967295>
   transit-area ospf_area dead interval default
   transit-area ospf_area retransmit-interval <1-4294967295>
   transit-area ospf_area retransmit-interval default
   transit-area ospf_area authtype none
   transit-area ospf_area authtype simple password
   transit-area ospf_area authtype md5 key authorization key id secret md5 key
   transit-area ospf_area authtype md5 key authorization key id off
```

Parameters

```
ip_address transit-area ospf_area <on | off>
```

Specifies the IP address of the remote endpoint of the virtual link and transit area, which is a specified ospf area you configure using the set ospf area command. Configure the ospf area you are using as the transit area before you configure the virtual link. The transit area is the area shared by the border router on which you configure the virtual link and the router with an interface in the backbone area. Traffic between the endpoints of the virtual link flow through this area. The virtual link IP address functions as the router ID of the remote endpoint of the virtual link.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip_address transit-area ospf_area hello-interval (&lt;1-65535&gt;)</td>
<td>Specifies the interval, in seconds, between hello packets that the router sends on the specified interface. For a given link, this value must be the same on all routers or adjacencies do not form.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area hello-interval default</td>
<td>Specifies an interval of 10 seconds.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area dead-interval (&lt;1-4294967295&gt;)</td>
<td>Specifies the number of seconds after which a router stops receiving hello packets that it declares the neighbor down. Generally, you should set this value at 4 times the value of the hello interval. Do not set the value at 0. For a given link, this value must be the same on all routers or adjacencies do not form.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area dead-interval default</td>
<td>Specifies a value of 40 seconds.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area retransmit-interval (&lt;1-4294967295&gt;)</td>
<td>Specifies the number of seconds between link state advertisement transmissions for adjacencies belonging to the specified interface. This value also applies to database description and link state request packets. Set this value conservatively, that is, at a significantly higher value than the expected round-trip delay between any two routers on the attached network.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area retransmit-interval default</td>
<td>Specifies a value of 5 seconds.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area authtype none</td>
<td>Specifies not to use an authentication scheme for the specified interface.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area authtype simple password</td>
<td>Specifies to use simple authentication for the specified interface. Enter an ASCII string that is 8 characters long. Generally, routers on a given link must agree on the authentication configuration to form neighbor adjacencies. Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.</td>
</tr>
<tr>
<td>ip_address transit-area ospf_area authtype md5 key authorization key id secret MD5 secret</td>
<td>Specifies to use MD5 authorization. Enter at least one key ID and its corresponding MD5 secret. If you configure multiple key IDs, the largest key ID is used for authenticating outgoing packets. All keys can be used to authenticate incoming packets. Generally, routers on a given link must agree on the authentication configuration to form neighbor adjacencies. Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.</td>
</tr>
</tbody>
</table>
**OSPF Areas**

Use the following commands to configure OSPF areas, including the backbone and stub areas.

For OSPFv2, use the following commands.

```plaintext
set [instance instance_name] ospf area backbone <on | off>
set ospf area ospf_area
   <on | off>
   stub <on | off>
   stub default-cost <1-677215>
   stub summary <on | off>
   nssa <on | off>
   nssa default-cost <1-677215>
   nssa default-metric-type <1-2>
   nssa import-summary-routes <on | off>
   nssa translator-role <always | candidate>
   nssa translator-stability-interval <1-65535>
   nssa redistribution <on | off>
   nssa range ip_addr [restrict] <on | off>
```

**Arguments**

- `backbone <on | off>`
  - Specifies whether to enable or disable the backbone area. By default, the backbone area is enabled. You can disable the backbone area if the system does not have interfaces on the backbone area.

- `<on | off>`
  - Specifies the area ID for a new OSPF area. Check Point recommends that you enter the area ID as a dotted quad, but you can use any integer as the area ID. The area ID 0.0.0.0 is reserved for the backbone.

- `stub <on | off>`
  - Specifies the area ID for a stub area. Stub areas are areas that do not have AS external routes.
  - **Note:** The backbone area cannot be a stub area.

- `stub default-cost <1-677215>`
  - Specifies a default route into the stub area with the specified cost.

- `stub summary <on | off>`
  - Specifies the OSPF area as totally stubby, meaning that it does not have any AS external routes and its area border routers do not advertise summary routes.

- `nssa <on | off>`
  - Specifies the area ID for an NSSA.
  - **Note:** The backbone area cannot be an NSSA area.

- `nssa default-cost <1-677215>`
  - Specifies the cost associated with the default route to the NSSA.

- `nssa default-metric-type <1-2>`
  - Specifies the type of metric. The default, type 1, is equivalent to the `Default ASE Route Type` on the OSPF Voyager page. A type 1 route is internal and its metric can be used directly by OSPF for comparison. A type 2 route is external and its metric cannot be used for comparison directly.

- `nssa import-summary-routes <on | off>`
  - Specifies if summary routes (summary link advertisements) are imported into the NSSA.
nssa translator-role <always | candidate>

Specifies whether this NSSA border router will unconditionally translate Type-7 LSAs into Type-5 LSAs. When role is Always, Type-7 LSAs are translated into Type-5 LSAs regardless of the translator state of other NSSA border routers. When role is Candidate, this router participates in the translator election to determine if it will perform the translations duties.

nssa translator-stability-interval <1-65535>

Specifies how long in seconds this elected Type-7 translator will continue to perform its translator duties once it has determined that its translator status has been assumed by another NSSA border router. Default: 40 seconds.

nssa redistribution <on | off>

Specifies if both Type-5 and Type-7 LSAs or only Type-7 LSAs will be originated by this NSSA border router.

nssa range ip_addr [restrict] <on | off>

Specify the range of addresses to reduce the number of Type-5 LSAs for the NSSA border router. To prevent a specific prefix from being advertised, use the restrict argument.

---

**OSPF Show Commands**

Use the following commands to monitor and troubleshoot your OSPF implementation.

To view a summary of your OSPF implementation, including the number of areas configured and the number of interfaces configured within each area, use:

- For OSPFv2: `show ospf`

For OSPFv2 use the following commands:
show ospf
neighbors
neighbor ip_address
interfaces
interfaces stats
interfaces detailed
interface ifname
interface ifname stats
interface ifname detailed
packets
errors
errors dd
errors hello
errors ip
errors lsack
errors lsr
errors lsu
errors protocol
events
border-routers
database
database areas
database area ospf_area
database asbr-summary-lsa
database checksum
database database-summary
database detailed
database external-lsa
database network-lsa
database router-lsa
database summary-lsa
database type <1 | 2 | 3 | 4 | 5 | 7> [detailed]
database nssa-external-lsa [detailed]
summary

Parameters

neighbors Displays the IP addresses of neighboring interfaces, their priority and status, and the number of errors logged for each interface.

neighbor ip_address Displays the priority, status, and number of errors logged for the specified IP address.

interfaces Displays the names of all configured logical interfaces, their corresponding IP addresses, to area to which each interface is assigned, each interface’s status and the IP addresses of each logical interface’s designated router and backup designated router.

interfaces stats Displays the number of each type of error message logged for each OSPF interface as well as the number of link state advertisements sent by each interface.

interfaces detailed Displays detailed information about each OSPF interface, including the authentication type configured if any, the router IDs and IP addresses of the designated router and backup designated router, the timer intervals configured for hello wait, dead, and retransmit messages, and the number of neighbors for each interface.

interface if_name Displays the IP address, area ID, status, number of errors logged, and the IP address of the designated router and backup designated router for the specified interface.
interface if_name stats
Displays the number of each type of error message logged by the specified interface as well as the number of link-state advertisements sent by the specified interface.

interface if_name detailed
Displays detailed information about the specified interface, including the authentication type configured if any, the router IDs and IP addresses of the designated router and backup designated router, the timer intervals configured for hello wait, dead, and retransmit messages, and the number of neighbors for each interface.

packets
Displays the number of each type of packet sent, including hello packets, link-state update packets, and link-state acknowledgment and link-state request packets.

errors
Displays the number of each type of error message sent, including hello protocol errors, database description errors, protocol errors, link-state acknowledgment errors, link-state request errors, link-state update errors, and IP errors.

errors dd
Displays the number of each type of database-description error messages only.

errors hello
Displays the number of each type of hello-error message only.

errors ip
Displays the number of each type of IP-errors message only.

errors lsack
Displays the number of each type of link-state acknowledgment error message only.

errors lsu
Displays the number of each type of link-state update error message only.

errors lsr
Displays the number of each type of link-state request error messages only.

errors protocol
Displays the number of each type of protocol error message only.

border-routers
Displays the IP address of each area border router, the OSPF area of each border router, and the cost associated with each IP address.

database
Displays router-link state and network-link state statistics for each OSPF area. Also displays the checksum, sequence number, and link count of each OSPF interface.

database areas
Displays router-link state, network-link state, AS-border-router link state, AS-external link state, and summary-link state statistics for each OSPF area. Also displays the checksum, sequence number, and link count of each OSPF interface.

database area ospf_area
Displays router-link state, network-link state, AS-border-router-link state, AS-external-link state, and summary-link state statistics for the specified OSPF area. Also displays the checksum, sequence number, and link count of each IP address configured within the specified OSPF area.

database asbr-summary
Displays a summary of AS-border-router link state statistics for each OSPF area. For OSPFv2 only.

database external
Displays AS-external-link state statistics for each OSPF area.

database database-summary
Displays a summary of router-link-state, network-link state, summary-link-state, and AS-border-router-link state statistics.
database network  Displays network-link-state statistics, including the advertised router, sequence number, and checksum of each OSPF interface. For OSPFv2 only.

database nssa-external-lsa [detailed]  Displays type 7 LSAs (NSSA). This argument applies only to OSPF v2.

database router-lsa  Displays router-link-state statistics, including the advertised router, sequence number, checksum, and link count, of each OSPF interface. For OSPFv2 only.

database summary-lsa  Displays a summary of link-state statistics for each OSPF area. For OSPFv2 only.

database type <1 | 2 | 3 | 4 | 5 | 7> [detailed]  Displays link-state statistics associated with the specified number:

- 1—router-link-state statistics.
- 2—network-link-state statistics.
- 3—summary-link-state statistics.
- 4—AS-border-router-link-state statistics.
- 5—AS-external-link-state statistics.
- 7—NSSA. This option applies only to OSPF v2.

events  Displays the number of interface up/down events; virtual interface up/down events; designated router election events; router ID changes; area border router changes; AS border router changes, and link state advertisement messages.
Chapter 7

Route Aggregation

In This Section:

Configuring Route Aggregation - CLI (aggregate) .................................................. 141

Route aggregation allows you to take numerous specific routes and aggregate them into one encompassing route. Route aggregation can reduce the number of routes that a given protocol advertises. The aggregates are activated by contributing routes. For example, if a router has many interface routes subnetted from a class C and is running RIP 2 on another interface, the interface routes can be used to create an aggregate route (of the class C) that can then be redistributed into RIP. Creating an aggregate route reduces the number of routes advertised using RIP. You must take care when aggregating if the route that is aggregated contains holes.

Create an aggregate route by first specifying the network address and mask length. Second, a set of contributing routes must be provided. A contributing route is defined when a source (for example, a routing protocol, a static route, an interface route) and a route filter (a prefix) are specified. An aggregate route can have many contributing routes, but at least one of the routes must be present to generate an aggregate.

Aggregate routes are not used for packet forwarding by the originator of the aggregate route, only by the receiver. A router receiving a packet that does not match one of the component routes that led to the generation of an aggregate route responds with an ICMP network unreachable message. This message prevents packets for unknown component routes from following a default route into another network where they would be continually forwarded back to the border router until their TTL expires.

Configuring Route Aggregation - CLI (aggregate)

Create aggregate routes using these commands.

```
set aggregate ip_prefix
   contributing protocol <protocol> contributing-route
    <all | ip_prefix> <on | off>
    <ip_prefix> exact on
    <ip_prefix> refines on
    off
   contributing protocol <protocol> off
   rank default
   rank <0–255>
   weight default
   aspath-truncate <on | off>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>contributing protocol &lt;protocol&gt;</td>
<td>The IP address and mask length of the new aggregate route and the contributing protocol or interface route. To specify a protocol, enter direct, static, ospf2, ospf2ase, bgp, rip, igrp, rip, or aggregate. To specify a contributing route, enter all to contribute all the routes for a specific protocol or enter the IP address and mask length to contribute a specific route.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| `<ip_prefix> exact on` | The contributing route is limited to the specified IP address and mask length only.  
You cannot enable both `exact on` and `refines on` at the same time. If you enable `refines on` when `exact on` is enabled, `exact on` is automatically disabled. |
| `<ip_prefix> refines on` | The contributing route is based on addresses with a greater value than the specified mask length of the specified IP address.  
You cannot enable both `exact on` and `refines on` at the same time. If you enable `refines on` when `exact on` is enabled, `exact on` is automatically disabled. |
| rank default          | The rank to assign to the aggregate route when routes from different protocols to the same destination are present. For each route, the route from the protocol with the lowest rank is used.  
Each routing protocol has a different default rank value. Aggregate routes have a default rank of 130. |
| rank `<0-255>`        | The rank to assign to the aggregate route when routes from different protocols to the same destination are present. For each route, the route from the protocol with the lowest rank is used. Each routing protocol has a different default rank value.  
Each routing protocol has a different default rank value. Aggregate routes have a default rank of 130. |
| weight default        | A value that breaks a tie if select routes going to the same destination have the same rank value. The route with the highest weight is the active route. The active route is installed in the kernel forwarding table and redistributed to the other routing protocols.  
- **Range**: 0-65535.  
- **Default**: 0 |
| weight `<0-65535>`    | A value that breaks a tie if select routes going to the same destination have the same rank value. The route with the highest weight is the active route. The active route is installed in the kernel forwarding table and redistributed to the other routing protocols.  
- **Default**: 0 |
| aspath-truncate `<on | off>` | Specifies that the autonomous system (AS) path be truncated to the longest common AS path. The default behavior is to build an AS path that consists of sets and sequences of all contributing AS paths.  
- **Default**: off |
Chapter 8

Routing Options

In This Section:

Configuring Protocol Rank - CLI (protocol-rank)................................. 143

The *protocol rank* is the value that the routing daemon uses to order routes from different protocols to the same destination. It is an arbitrarily assigned value used to determine the order of routes to the same destination. Each route has only one rank associated with it, even though rank can be set at many places in the configuration. The route derives its rank from the most specific route match among all configurations.

The *active route* is the route installed into the kernel forwarding table by the routing daemon. In the case where the same route is contributed by more than one protocol, the one with the lowest rank becomes the active route.

Rank cannot be used to control the selection of routes within a dynamic interior gateway protocol (IGP); this is accomplished automatically by the protocol and is based on the protocol metric. Instead, rank is used to select routes from the same external gateway protocol (EGP) learned from different peers or autonomous systems.

Some protocols—BGP and aggregates—allow for routes with the same rank. To choose the active route in these cases, a separate tie breaker is used. This tie breaker is called *LocalPref* for BGP and *weight* for aggregates.

Configuring Protocol Rank - CLI (protocol-rank)

Rank is used by the routing system when there are routes from different protocols to the same destination. For each route, the route from the protocol with lowest rank number is used.

**Syntax**

```plaintext
set protocol-rank protocol
  bgp rank <0–255>
  bgp rank default
  rip rank <0–255>
  rip rank default
  ospf rank <0–255>
  ospf rank default
  ospfase rank <0–255>
  ospfase rank default
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rank &lt;0–255&gt;</td>
<td>The protocol rank value.</td>
</tr>
<tr>
<td>ospf rank default</td>
<td>The default rank value for OSPF is 10.</td>
</tr>
<tr>
<td>rip rank default</td>
<td>The default rank value for RIP is 100.</td>
</tr>
<tr>
<td>bgp rank default</td>
<td>The default rank value for BGP is 170.</td>
</tr>
<tr>
<td>ospfase rank default</td>
<td>The default rank value for OSPF ASE routes is 150.</td>
</tr>
</tbody>
</table>
Chapter 9

Route Maps

In This Section:

Configuring Route Map - CLI (routemap) ................................................................. 144

Route maps are used to control which routes are accepted and announced by dynamic routing protocols. Use route maps to configure inbound route filters, outbound route filters, and to redistribute routes from one protocol to another.

You can define route maps only using the CLI. Routemaps are not available in the WebUI.

Route maps support RIP, BGP, and OSPFv2. You can also use the Route Redistribution and Inbound Route Filters features that you configure using the WebUI. Route map for import policy corresponds to Inbound Route Filters; route map for export policy corresponds to Route Redistribution.

Note - Route maps offer more configuration options than the WebUI configuration for route redistribution and inbound route filters. They are not functionally equivalent.

Protocols can use route maps for redistribution and WebUI settings for inbound route filtering and vice versa. However, if one or more route maps are assigned to a protocol (for import or export), any corresponding WebUI configuration (for route redistribution or inbound route filters) is ignored.

Configuring Route Map - CLI (routemap)

Each route map includes a list of match criteria and set statements. You can apply route maps to inbound, outbound, or redistribution routes. Routes are compared to the match criteria, and all the actions specified in the set criteria are applied to those routes which meet all the match conditions. You can specify the match conditions in any order. If you do not specify any match conditions in a route map, the route map matches all routes.

You define route maps, then assign them to protocols for export or import policy for that protocol. Route maps take precedence over WebUI based configuration.

To create a route map, use CLI commands to specify a set of criteria that must be matched for the command to take effect. If the criteria are matched, then the system executes the actions you specify. A route map is identified by name and an identifying number, an Allow or Restrict clause, and a collection of match and set statements.

There can be more than one instance of a route map (same name, different ID). The lowest numbered instance of a route map is checked first. Route map processing stops when either all the match criteria of some instance of the route map are satisfied, or all the instances of the particular route map are exhausted. If the match criteria are satisfied, the actions in the set section are performed.

Routing protocols can use more than one route map when you specify distinct preference values for each. The appropriate route map with lowest preference value is checked first.
**Set Routemap Commands**

**To set a routemap:**

```plaintext
set routemap rm_name id <1-65535>
    <off|on>
    allow
    inactive
    restrict
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>routemap rm_name</td>
<td>The name of the routemap.</td>
</tr>
<tr>
<td>id &lt;1-65535&gt;</td>
<td>The ID of the routemap. You can enter the keyword default or the default value 10.</td>
</tr>
<tr>
<td>&lt;off</td>
<td>on&gt;</td>
</tr>
<tr>
<td></td>
<td>• off to delete a routemap.</td>
</tr>
<tr>
<td>allow</td>
<td>Allow routes that match the routemap.</td>
</tr>
<tr>
<td>inactive</td>
<td>Temporarily disable a routemap. To activate the routemap, use the allow or restrict arguments.</td>
</tr>
<tr>
<td>restrict</td>
<td>Routes that match the routemap are not allowed.</td>
</tr>
</tbody>
</table>

**To specify actions for a routemap:**

- **Note** - Some statements affect only a particular protocol (["Supported Route Map Statements by Protocol" on page 150](#)).
- The same parameter cannot appear both as a match and action statement in a routemap. These include Community, Metric, and Nexthop.

```plaintext
set routemap rm_name id id_number action
    aspath-prepend-count <1-25>
    community <append | replace | delete> [on|off]
    community <1-65535> as <1-65535> [on|off]
    community no-export [on|off]
    community no-advertise [on|off]
    community no-export-subconfed [on|off]
    community none [on|off]
    localpref <1-65535>
    metric <add|subtract> <1-16>
    metric igp [add | subtract] <1-4294967295>
    metric value <1-4294967295>
    nexthop <ip ipv4_address>
    precedence <1-65535>
    preference <1-65535>
    route-type <type-1 | type-2>
    remove action_name
    ospfautomatictag tag
    ospfmanualtag tag
    riptag tag
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>routemap rm_name</td>
<td>Specifies the name of the routemap.</td>
</tr>
<tr>
<td>id id_number</td>
<td>Specifies the ID of the routemap. You can enter the keyword default or the default value 10.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>aspath-prepend-count</td>
<td>Specifies to affix AS numbers at the beginning of the AS path. It indicates the number of times the local AS number should be prepended to the AS PATH before sending out an update. BGP only.</td>
</tr>
<tr>
<td>community &lt;append</td>
<td>replace</td>
</tr>
<tr>
<td>community &lt;1-65535&gt; as &lt;1-65535&gt; [on</td>
<td>off]</td>
</tr>
<tr>
<td>community no-export [on</td>
<td>off]</td>
</tr>
<tr>
<td>community no-advertise [on</td>
<td>off]</td>
</tr>
<tr>
<td>community no-export-subconfed [on</td>
<td>off]</td>
</tr>
<tr>
<td>community none [on</td>
<td>off]</td>
</tr>
<tr>
<td>localpref &lt;1-65535&gt;</td>
<td>Set the local preference for BGP route. BGP only.</td>
</tr>
<tr>
<td>metric [add</td>
<td>subtract] &lt;1-16&gt;</td>
</tr>
<tr>
<td>metric igp [add</td>
<td>subtract] &lt;1-4294967295&gt;]</td>
</tr>
<tr>
<td>metric value &lt;1-4294967295&gt;</td>
<td>Set the metric value. For RIP the metric is metric, for OSPF the metric is cost, and for BGP the metric is MED.</td>
</tr>
<tr>
<td>nexthop &lt;ip ipv4_address&gt;</td>
<td>Set IPv4 Nexthop Address. BGP only.</td>
</tr>
<tr>
<td>precedence &lt;1-65535&gt;</td>
<td>Sets the rank of the route. Precedence works across protocols. Use this setting to bias routes of one protocol over the other. The lower value has priority.</td>
</tr>
<tr>
<td>preference &lt;1-65535&gt;</td>
<td>Applies only to BGP. This is equivalent to the bgp weight (in Cisco terms) of the route. However, unlike Cisco, the route with lower value will be preferred. This value is only relevant for the local router.</td>
</tr>
<tr>
<td>route-type &lt;type-1</td>
<td>type-2&gt;</td>
</tr>
</tbody>
</table>
# Route Maps

## Check Point 1100 and 600 Appliance Administration Guide CLI and Advanced Routing

### Routing

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>remove action_name</td>
<td>Remove the specified action from the routemap. For community, it removes all community statements. Allowed values for action_name are:</td>
</tr>
<tr>
<td></td>
<td>aspath-regex</td>
</tr>
<tr>
<td></td>
<td>community</td>
</tr>
<tr>
<td></td>
<td>ifaddress</td>
</tr>
<tr>
<td></td>
<td>interface</td>
</tr>
<tr>
<td></td>
<td>metric</td>
</tr>
<tr>
<td></td>
<td>neighbor</td>
</tr>
<tr>
<td></td>
<td>network</td>
</tr>
<tr>
<td></td>
<td>nexthop</td>
</tr>
<tr>
<td></td>
<td>protocol</td>
</tr>
<tr>
<td></td>
<td>route-type</td>
</tr>
<tr>
<td>ospfautomatictag tag</td>
<td>Creates an automatic OSPF route tag.</td>
</tr>
<tr>
<td>ospfmanualtag tag</td>
<td>Creates a manual OSPF route tag.</td>
</tr>
<tr>
<td>riptag tag</td>
<td>Creates a RIP route tag.</td>
</tr>
</tbody>
</table>

To specify the criteria that must be matched for the routemap to take effect:

- **Note**: Some statements affect only a particular protocol ("Supported Route Map Statements by Protocol" on page 150).

The same parameter cannot appear both as a match and action statement in a routemap. These include Community, Metric, and Nexthop.

```plaintext
set routemap rm_name id <1-65535> match
  as <1-65535> [on | off]
  aspath-regex ["<regular_expression>" | empty] origin
  <any | igp | incomplete>
  community <1-65535> as <1-65535> [on|off]
  community exact [on|off]
  community no-export [on|off]
  community no-advertise [on|off]
  community no-export-subconfed [on|off]
  community none [on|off]
  ifaddress IPv4_addr [on | off]
  interface interface name [on | off]
  metric value <1-4294967295>
  neighbor IPv4_addr [on | off]
  network IPv4_network / masklength <all | exact | off | refines>
  network<IPv4_network / masklength between masklength and masklength
  nexthop IPv4_addr [on | off]
  protocol <ospf2 | ospf2ase | bgp | rip | static | direct | aggregate>
  route-type <type-1 | type-2 | inter-area | intra-area> [on | off]
  remove match_condition_name
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>as &lt;1-65535&gt; [on</td>
<td>off]</td>
</tr>
<tr>
<td>aspath-regex [&quot;&lt;regular-expression&gt;&quot;</td>
<td>empty] origin &lt;any</td>
</tr>
<tr>
<td></td>
<td>Note: Enter the regular expression in quotation marks. Use the empty keyword to match a null ASPath.</td>
</tr>
<tr>
<td>community &lt;1-65535&gt; as &lt;1-65535&gt; [on</td>
<td>off]</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>community exact [on</td>
<td>off]</td>
</tr>
<tr>
<td>community no-export [on</td>
<td>off]</td>
</tr>
<tr>
<td>community no-advertise [on</td>
<td>off]</td>
</tr>
<tr>
<td>community no-export-subconfed [on</td>
<td>off]</td>
</tr>
<tr>
<td>community none [on</td>
<td>off]</td>
</tr>
<tr>
<td>ifaddress IPv4_addr [on</td>
<td>off]</td>
</tr>
<tr>
<td>interface interface_name [on</td>
<td>off]</td>
</tr>
<tr>
<td>metric value &lt;1-4294967295&gt;</td>
<td>Match the specified metric value.</td>
</tr>
<tr>
<td>neighbor IPv4_addr [on</td>
<td>off]</td>
</tr>
</tbody>
</table>
| network IPv4_network / masklength | Use with the following keywords:  
  - all: Match all networks belonging to this prefix and masklength. This is a combination of exact and refines.  
  - between masklength and masklength: Specify a range of masklengths to be accepted for the specified prefix.  
  - exact: Match prefix exactly.  
  - off: Delete the network match statement.  
  - refines: Match networks with more specific mask lengths only. Matches only subnets.  
There can be multiple network match statements in a route map. |
| nexthop IPv4_addr [on | off] | Match the specified nexthop address. |
| protocol <ospf2 | ospf2ase | bgp | rip | static | direct | aggregate> | Match the specified protocol. Use this for route redistribution. |
### Route Maps

#### Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`route-type &lt;type-1</td>
<td>type-2</td>
</tr>
</tbody>
</table>

| `remove match_condition_name`     | Remove the specified match condition from the routemap. For match conditions which can have multiple match statements (such as network, neighbor), this argument removes all of them. |

#### Show Routemap Commands

- `show routemap rm_name <all | id VALUE>`
- `show routemaps`

#### Routemap Protocol Commands

**To assign routemaps to protocols:**

The preference value specifies which order the protocol will use each routemap.

- `set <ospf | rip > export-routemap rm_name preference VALUE on`
- `import-routemap rm_name preference VALUE on`

**To turn a routemap off:**

- `set <ospf | rip > export-routemap rm_name off`
- `import-routemap rm_name off`

**To view routemaps assigned to protocols:**

- `show <ospf | rip> routemap`

**To set BGP routemaps for export and import policies:**

- `set bgp external remote-as <1-65535> export-routemap rm_name off preference <1-65535> [family inet] on`
- `set bgp external remote-as <1-65535> import-routemap rm_name off preference <1-65535> [family inet] on`
- `set bgp internal export-routemap rm_name off preference <1-65535> [family inet] on`
- `set bgp internal import-routemap rm_name off preference <1-65535> [family inet] on`

**Note** - You cannot use routemaps in BGP confederations. To configure route filters and redistribution for BGP confederations, use the Inbound Route Filters and Route Redistribution pages in the WebUI.
Supported Route Map Statements by Protocol

Some statements affect only a particular protocol, for example, matching the Autonomous System Number is applicable only to BGP. If such a condition is in a routemap used by OSPF, the match condition is ignored. Any non-applicable match conditions or actions are ignored and processing is done as if they do not exist. A log message is generated in /var/log/messages for any such statements.

Note - The same parameter cannot appear both as a match and action statement in a routemap. These include Community, Metric, and Nexthop.

RIP

- Import Match conditions: Neighbor, Network, Interface, Ifaddress, Metric, Neighbor, Nexthop.
- Import Actions: Precedence, Metric Add/Subtract
- Export Match conditions when exporting from RIP - Interface, Ifaddress, Metric, Network, Nexthop
- Export Match Conditions when redistributing using Protocol match: According to the protocol from which route is being redistributed.
- Export Actions when exporting from RIP - Metric Add/Subtract
- Export Actions when redistributing - Metric Set

OSPFv2

- Import Match conditions: Network (Route Prefix)
- Import Actions: Precedence
- Export Match conditions when other protocols redistribute OSPF routes: Network, Interface, Ifaddress, Metric, Route-type, Nexthop
- Export Match conditions when OSPF redistributes routes from other protocols: Conditions supported by that protocol
- Export Actions when redistributing to AS External: Metric, Route-type

BGP

When you do initial configuration, set the router ID. You can also use the following commands to change the router ID.

```
set router-id default
set router-id ip_address
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>Selects the highest interface address when OSPF is enabled.</td>
</tr>
<tr>
<td>ip_address</td>
<td>The Router ID uniquely identifies the router in the autonomous system. The router ID is used by the BGP and OSPF protocols. We recommend setting the router ID rather than relying on the default setting. This prevents the router ID from changing if the interface used for the router ID goes down. Use an address on a loopback interface that is not the loopback address (127.0.0.1). Note - In a cluster, you must select a router ID and make sure that it is the same on all cluster members.</td>
</tr>
<tr>
<td>Range:</td>
<td>Dotted-quad.([0-255].[0-255].[0-255].[0-255]). Do not use 0.0.0.0</td>
</tr>
<tr>
<td>Default:</td>
<td>The interface address of one of the local interfaces.</td>
</tr>
</tbody>
</table>
Use the following group of commands to set and view parameters for BGP.

```
set as as_number
set as off
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>as as_number</td>
<td>The local autonomous system number of the router. This number is mutually exclusive from the confederation and routing domain identifier. The router can be configured with either the autonomous system number or confederation number, not both. Caution: When you change the autonomous system number, all current peer sessions are reset and all BGP routes are deleted.</td>
</tr>
<tr>
<td>as off</td>
<td>Disables the configured local autonomous system number.</td>
</tr>
</tbody>
</table>

**Redistributing Static, Interface, or Aggregate Routes**

When redistributing **static** routes into BGP, OSPFv2 or RIP the following match conditions are supported:

- Network Prefix,
- Nexthop
- Interface
- Ifaddress
- Protocol (proto = static)

When redistributing **interface/direct** routes into BGP, OSPFv2 or RIP the following match conditions are supported:

- Network Prefix
- Interface
- Ifaddress
- Protocol (proto = direct)

When redistributing **aggregate** routes into BGP, OSPFv2 or RIP the following match conditions are supported:

- Network Prefix
- Protocol (proto = aggregate)

**Route Map Examples**

**Example 1**

Redistribute interface route for eth3c0 into ospf, and set the ospf route-type to AS type-2 with cost 20.

```
set routemap direct-to-ospf id 10 on
set routemap direct-to-ospf id 10 match interface eth3c0
set routemap direct-to-ospf id 10 match protocol direct
set routemap direct-to-ospf id 10 action route-type type-2
set routemap direct-to-ospf id 10 action metric value 20
set ospf export-routemap direct-to-ospf preference 1 on
```

**Example 2**

Do not accept routes from RIP neighbor 192.0.2.3, accept routes from neighbor 192.0.2.4 as is, and for all other routes increment the metric by 2.

```
set ospf default-metric 2
```
set routemap rip-in id 10 on
set routemap rip-in id 10 restrict
set routemap rip-in id 10 match neighbor 192.0.2.3

set routemap rip-in id 15 on
set routemap rip-in id 15 match neighbor 192.0.2.4

set routemap rip-in id 20 on
set routemap rip-in id 20 action metric add 2

set rip import-routemap rip-in preference 1 on

### Example 3

Redistribute all static routes into BGP AS group 400. Set the MED value to 100, prepend our AS number to the aspath 4 times. If the route belongs to the prefix 192.0.2.0/8, do not redistribute. Send all BGP routes whose aspath matches the regular expression (100 200+) and set the MED value to 200.

set routemap static-to-bgp id 10 on
set routemap static-to-bgp id 10 restrict
set routemap static-to-bgp id 10 match protocol static
set routemap static-to-bgp id 10 match network 192.0.2.0/8 all

set routemap static-to-bgp id 15 on
set routemap static-to-bgp id 15 match protocol static
set routemap static-to-bgp id 15 action metric 100
set routemap static-to-bgp id 15 action aspath-prepend-count 4

set routemap bgp-out id 10 on
set routemap bgp-out id 10 match aspath-regex "(100 200+)" origin any
set routemap bgp-out id 10 action metric 200

set bgp external remote-as 400 export-routemap bgp-out preference 1 family inet on
set bgp external remote-as 400 export-routemap static-to-bgp preference 2 family inet on

📚 **Note** - There is no need for a match protocol statement for routes belonging to the same protocol.
Chapter 10

PIM

In This Section:

Configuring PIM - CLI (pim) ................................................................. 154

Protocol-Independent Multicast (PIM) gets its name from the fact that it can work with any existing unicast protocol to perform multicast forwarding. It supports two types of multi-point traffic distribution patterns: dense and sparse.

Dense mode is most useful when:

• Senders and receivers are in close proximity.
• There are few senders and many receivers.
• The volume of multicast traffic is high.
• The stream of multicast traffic is constant.

Dense-mode PIM resembles Distance Vector Multicast Routing Protocol (DVMRP). Like DVMRP, dense-mode PIM uses Reverse Path Forwarding and the flood-and-prune model.

Sparse mode is most useful when:

• A group has few receivers.
• Senders and receivers are separated by WAN links.
• The type of traffic is intermittent.

Sparse-mode PIM is based on the explicit join model; the protocol sets up the forwarding state for traffic by sending join messages. This model represents a substantial departure from flood-and-prune protocols, such as dense-mode PIM, which set up the forwarding state through the arrival of multicast data.

The implementation does not support enabling both dense mode and sparse mode or either mode of PIM and DVMRP on the same appliance. For more information about PIM, read the following Internet Engineering Task Force (IETF) drafts.


You can run PIM over a route-based VPN by enabling PIM on a virtual tunnel interface (VTI). You must use an unnumbered interface for the VTI.
Configuring PIM - CLI (pim)

Syntax

Set Commands:

To configure PIM parameters:

set pim mode
   dense
   sparse
   ssm

set pim
   assert-interval VALUE
   assert-limit VALUE
   assert-rank protocol VALUE rank VALUE
   bootstrap-candidate local-address VALUE
   bootstrap-candidate off
   bootstrap-candidate on
   bootstrap-candidate priority VALUE
   candidate-rp advertise-interval VALUE
   candidate-rp local-address VALUE
   candidate-rp multicast-group VALUE off
   candidate-rp multicast-group VALUE on
   candidate-rp off
   candidate-rp on
   candidate-rp priority VALUE
   cisco-compatibility VALUE
   data-interval VALUE
   ha-mode off
   ha-mode on
   hello-interval VALUE
   interface VALUE dr-priority VALUE
   interface VALUE local-address VALUE
   interface VALUE off
   interface VALUE on
   interface VALUE virtual-address off
   interface VALUE virtual-address on
   jp-delay-interval VALUE
   jp-interval VALUE
   jp-suppress-interval VALUE
   nat-mode off
   nat-mode on
   network VALUE dr-priority VALUE
   network VALUE off
   network VALUE on
   register-suppress-interval VALUE
   spt-threshold multicast VALUE threshold VALUE off
   spt-threshold multicast VALUE threshold VALUE on
   state-refresh off
   state-refresh on
   state-refresh-interval VALUE
   state-refresh-ttl VALUE
   static-rp off
   static-rp rp-address VALUE multicast-group VALUE off
   static-rp rp-address VALUE multicast-group VALUE on
   static-rp rp-address VALUE off
   static-rp rp-address VALUE on
Show Commands

show pim

show pim
  bootstrap
  candidate-rp
  group-rp-mapping VALUE
  interface VALUE
  interfaces
  joins
  memory
  neighbor VALUE
  neighbors
  network VALUE
  networks
  rps
  sparse-mode-stats
  stats
  summary
  timers

PIM Interfaces

After you set PIM to run dense mode, sparse mode or SSM, use the following commands to configure PIM for specific interfaces.

set pim interface if_name
  <on | off>
  local-address ip_address
  dr-priority <0-4294967295>
  dr-priority default

Sparse Mode PIM

Use the following commands to configure parameters for sparse mode PIM only.

set pim
  ha-mode <on | off>
  bootstrap-candidate <on | off>
  bootstrap-candidate local-address ip_address
  bootstrap-candidate priority <0-255>
  bootstrap-candidate priority default
  candidate-rp <on | off>
  candidate-rp local-address ip_address
  candidate-rp priority <0-255>
  candidate-rp priority default
  candidate-rp multicast group mcast_ip_prefix <on | off>
  static-rp off
  static-rp rp-address ip_address < on | off>
  static-rp rp-address ip_address multicast-group mcast_ip_prefix <on | off>
  register-suppress-interval <60-3600>
  register-suppress-interval default
  candidate-rp advertise-interval <1-3600>
  candidate-rp advertise-interval default
  cisco compatibility <on | off>
  spt-threshold multicast mcast_ip_prefix threshold <0-1000000> <on | off>
  spt-threshold multicast mcast_ip_prefix threshold infinity <on | off>
  nat-mode <on | off>
  state-refresh <on | off>
  state-refresh-interval <0 - 255>
  state-refresh-ttl <0 - 255>
**Timer and Assert Rank Parameters for Dense Mode and Sparse Mode**

Use these commands to change or restore default values for timers and assert ranks.

```plaintext
set pim
    hello-interval <1-21845>
    hello-interval default
    data-interval <11-3600>
    data-interval default
    assert-interval <1-3600>
    assert-interval default
    assert-limit <10-10000>
    assert-limit default
    assert-limit <0>
    jp-interval <1-3600>
    jp-interval default
    jp-delay-interval <1-3600>
    jp-delay-interval default
    jp-suppress-interval <2-3600>
    jp-suppress-interval default
    assert-rank protocol protocol name rank <0-255>
    assert-rank protocol protocol name rank default
```

**Arguments**

- `<dense | sparse | ssm>`: Specifies the mode PIM should use.

- `interface if_name <on | off>`: Specifies whether to enable or disable PIM on a specified interface.

- `local-address ip_address`: Specifies the local address used in all advertisements sent on the interface. This option is useful when multiple IP addresses are configured on the interface. If you enter an address other than one configured for that interface, PIM ignores your configured address and selects one of the addresses configured on the interface. Warning: If neighboring routers choose advertisement addresses that do not appear to be on a shared subnet, all messages from the neighbor will be rejected. Thus, a PIM router on a shared LAN must have at least one interface address with a subnet prefix shared by all neighboring PIM routers.

- `ha-mode <on | off>`: Specifies whether to enable or disable the High Availability (HA) mode. Enable the High-Availability (HA) mode when two routers are configured to back each other up to forward multicast traffic and sparse-mode PIM is implemented. When this option is enabled, all PIM-enabled interfaces are available only if each interface is up and has a valid address assigned. If any PIM-enabled interface goes down or all its valid addresses are deleted, then all PIM-enabled interfaces become unavailable and remain in that state until all interfaces are back up.

  The HA mode feature applies only to sparse-mode PIM. The HA mode feature does not affect the functioning of dense-mode PIM.

- `dr-priority <0-4294967295>`: Specifies the dr-priority advertised in the PIM hello messages sent on the corresponding interface. This value, which has a default of 1, is used for DR election on a LAN. The router with the highest priority and the highest IP address is elected the designated router. To break a tie, the DR is selected on the basis of the highest IP address. If even one router does not advertise a dr-priority value in its hello messages, the DR election is based on the IP address.
dr-priority default Specifies a value of 1.

bootstrap-candidate <on | off> Specifies that the platform is a candidate bootstrap router. The bootstrap router collects candidate rendezvous point information and disseminates rp-set information associated with each group prefix. To avoid a single point of failure, configure more than router in a domain as a candidate bootstrap router.

off

bootstrap-candidate local-address ip_address Specifies the IP address of the bootstrap router used in bootstrap messages. By default, the router picks an address from one of the interfaces on which PIM is enabled.

bootstrap-candidate priority <0-255> Specifies the value used to elect the bootstrap router from among the candidate bootstrap routers. The candidate bootstrap router with the highest priority value is elected bootstrap router for the domain. The highest priority value is 0, so the lower the value, the higher the priority.

bootstrap-candidate priority default Specifies a value of 0.

candidate-rp <on | off> Specifies that the platform is a candidate rendezvous point router.

off

candidate-rp local-address ip_address Specifies the IP address of the candidate rendezvous point router used in candidate rendezvous point messages. By default, the router picks an address from one of the interfaces on which PIM is enabled.

candidate-rp priority <0-255> Specifies the priority of the candidate rendezvous point included in the corresponding multicast group address. The higher the priority, the lower the value.

candidate-rp priority default Specifies a value of 0.

candidate-rp multicast-group mcast_ip_prefix <on | off> Specifies the multicast address advertised in the candidate rendezvous point advertisements. For the multicast IP prefix value, you must enter an IP address and mask length. If you do not specify a group multicast address, the candidate rendezvous point advertises itself as the rendezvous point for all multicast groups.

static-rp off Disables the static rendezvous point option.

static-rp rp-address ip_address <on | off> Specifies to enable or disable a static rendezvous point. If you do not specify an associated multicast group and prefix, the static-rp is considered to be responsible for all multicast groups (224.0.0.0/4).

static-rp rp-address ip_address multicast-group mcast_ip_prefix <on | off> Specifies the IP address associated with the static rendezvous point and the multicast IP address for which the rendezvous point is responsible. For the multicast IP prefix value, you must enter an IP address and mask length.

register-suppress-interval <60-3600> Specifies the mean interval between receiving a register-stop and allowing registers to be sent again. A lower value means more frequent register bursts at the rendezvous point, while a higher value means a longer join latency for new receivers.

register-suppress-interval default Specifies a value of 60.
candidate-rp advertise-interval
<1-3600>

Specifies the interval between which candidate-rendezvous point routers send candidate-rendezvous point advertisements.

candidate-rp advertise-interval
default

Specifies a value of 60.

cisco-compatibility
<on | off>

The checksum of the PIM register messages is calculated without including the multicast payload. Earlier releases of Cisco’s IOS calculate the checksum by including the multicast payload. If you experience difficulties having PIM register messages sent by your Check Point appliance being accepted by a Cisco router that is the elected rendezvous point (RP), configure this option. A Check Point appliance that is the elected RP, accepts register messages that calculate the checksum with or without the multicast payload, that is it accepts all register messages.

cisco-compatibility=off

spt-threshold multicast mcast_ip_prefix threshold
<0-1000000>

Specifies the multicast group address to apply to the shortest path tree (spt) threshold and the date rate in kbits/sec to trigger the spt switch over.

spt-threshold multicast mcast_ip_prefix threshold
infinity <on | off>

Specifies the data rate in kbits/sec to trigger the spt switch over as infinity.

hello interval <1-21845>

Specifies the interval, in seconds, at which PIM hello messages are sent on the LAN.

hello interval default

Specifies a value of 30.

data-interval <11-3600>

Specifies the interval, in seconds, after which multicast (S,G) state for a silent source is deleted.

data-interval default

Specifies a value of 210.

assert-interval <1-3600>

Specifies the interval between the last time an assert is received and the assert is timed out.

assert-interval default

Specifies a value of 180.

assert-limit <10-10000>

Specifies the number of asserts to send per second.

assert-limit default

Specifies a value of 10.

assert-limit <0>

Disables the limit placed on the number of asserts that can be sent per second.

jp-interval <1-3600>

Specifies the interval, in seconds, between which join/prune messages are sent.

jp-interval default

Specifies a value of 60.

jp-delay-interval
<1-3600>

Specifies maximum interval, in seconds, between the time when the RPF neighbor changes and a triggered Join/Prune message is sent.

jp-delay-interval default

Specifies a value of 5.
**Show PIM Commands**

Use these commands to monitor and troubleshoot PIM.

These commands apply to both dense-mode and sparse-mode PIM:

```
show pim
  interfaces
  interfaces if_address
  neighbors
  neighbor ip_address
  memory
  timers
  stats
  summary
```

These commands apply only to sparse-mode PIM:
Debugging PIM - CLI

Use these commands to debug PIM:

<table>
<thead>
<tr>
<th>Command</th>
<th>Shows</th>
</tr>
</thead>
<tbody>
<tr>
<td>show pim interface</td>
<td>Which interfaces are running PIM, their status, and the mode they are running. This command also displays the interface and its DR priority and the number of PIM neighbors on the interface.</td>
</tr>
<tr>
<td>show pim neighbors</td>
<td>The IP address of each PIM neighbor and the interface on which the neighbor is present. This command also displays the neighbor’s DR priority, generation ID, holdtime and the time the neighbor is set to expire based on the holdtime received in the most recent hello message.</td>
</tr>
<tr>
<td>show pim statistics</td>
<td>The number of different types of PIM packets received and transmitted and any associated errors.</td>
</tr>
<tr>
<td>show mfc cache</td>
<td>Multicast source and group forwarding state by prefix.</td>
</tr>
<tr>
<td>show mfc interfaces</td>
<td>Shows multicast source and group forwarding state by interface.</td>
</tr>
</tbody>
</table>

Use these commands to debug sparse-mode PIM (PIM-SM):

<table>
<thead>
<tr>
<th>Command</th>
<th>Shows</th>
</tr>
</thead>
<tbody>
<tr>
<td>show pim bootstrap</td>
<td>The IP address and state of the Bootstrap router.</td>
</tr>
<tr>
<td>show pim candidate-rp</td>
<td>The state of the Candidate Rendezvous Point state machine.</td>
</tr>
<tr>
<td>show pim joins</td>
<td>PIM’s view of the join-prune (*, G and S, G) state, including RP for the group, incoming, and outgoing interface(s), interaction with the multicast forwarding cache and the presence of local members. To view the equivalent information for dense-mode PIM, use the show mfc cache command.</td>
</tr>
<tr>
<td>show pim rps</td>
<td>The active RP-set, including the RP addresses, their type (or source of information about them) and the groups for which they are configured to act as RP.</td>
</tr>
<tr>
<td>show pim group-rp-mapping</td>
<td>The RP selected for a particular group based on information from the active RP-set.</td>
</tr>
<tr>
<td>show pim sparse-mode</td>
<td>Error statistics for multicast forwarding cache (MFC); Bootstrap Router (BSR) messages; Candidate Rendezvous Point (CRP) advertisements; and the Internet Group Management Protocol (IGMP).</td>
</tr>
<tr>
<td>statistics</td>
<td></td>
</tr>
</tbody>
</table>
Index

A

add admin access • 14
add antispam_allowed_sender • 14
add bridge • 15
add host • 15
add interface • 15
add internet connection • 16
add net-obj • 20
add snmp • 22
add static-route • 23
add switch • 24
add user • 25
add wlan vap • 26
Adding a Route with a Specific Gateway IP Address • 23
Adding a Route with a Specific Interface and Priority • 24
Adding Administrator User • 25
Adding Local User • 25
Adding SNMP User • 22
Adding SNMP v2 Traps Receiver • 22
Adding SNMP v3 Traps Receiver • 22
Advanced Wireless Settings • 76
Area Border Routers • 129
Assign Wireless Network to a Switch/Bridge • 72
Authentication • 123
Auto Summarization • 124
Automatically configuring the WINS Server • 45

B

backup settings • 26
Basic configurations • 56
Basic Configurations • 55
BGP • 96, 105, 150
BGP Communities • 100
BGP Communities Commands • 118
BGP Confederation Commands • 111
BGP Interactions with IGP • 99
BGP Multi-Exit Discriminator • 99
BGP Path Attributes • 98
BGP Peers • 107
BGP Route Dampening • 101
BGP Route Dampening Commands • 112
BGP Route Reflection Commands • 111
BGP Route Refresh • 97
BGP Sessions (Internal and External) • 97
BGP Show Commands • 119

C

CLI Reference • 10
CLI Syntax • 13
Confederations • 101
Configure Interfaces for IGMP • 121
Configuring a Proxy Server • 63
Configuring a TFTP Server • 46
Configuring Advanced Connection Settings • 60
Configuring Advanced Interface Settings • 53
Configuring BGP - CLI (bgp) • 102
Configuring Call Manager • 47
Configuring Cellular Modem • 20, 58
Configuring Custom DHCP Option • 47
Configuring DHCP • 16
Configuring DNS Server • 45
Configuring EoA • 20, 57
Configuring Host Name • 44
Configuring ICMP • 58
Configuring IGMP - CLI (igmp) • 121
Configuring IP Lease Time • 45
Configuring Monitoring • 59
Configuring OSPF - CLI (ospf) • 130
Configuring PIM - CLI (pim) • 154
Configuring PPPoE • 19, 56
Configuring PPPoE • 18, 55
Configuring PPTP and L2TP • 17, 54
Configuring Protocol Rank - CLI (protocol-rank) • 143
Configuring RIP - CLI (rip) • 124
Configuring Route Aggregation - CLI (aggregate) • 141
Configuring Route Map - CLI (route-map) • 144
Configuring SNMP v3 Receivers • 65
Configuring SNMP v3 Users • 66
Configuring Static IP • 16, 52, 54
Configuring the Default Gateway • 44
Configuring the NTP Server • 46
Configuring the Path for a Bootstrap File • 46
Configuring the WINS Server • 44
Configuring VoIP Phones • 47
Configuring VPI, VCI and Encapsulation • 57
Configuring X-Windows Display • 47
Connect on Demand • 59
cphaprob • 27
cphastop • 28
cpinfo • 29
cpshell • 29
cpstart • 29
cpstat • 30
cpstop • 31
cpwd_admin • 31

D

Debugging PIM - CLI • 160
Debugging VPN • 94
delete bridge • 32
delete dhcp • 32
delete dns • 33
delete domainname • 33
delete host • 33
Delete ICMP servers • 34
delete interface • 34
Delete Internet Connection • 34
delete internet-connection • 34
delete net-obj • 34
delete proxy • 35
delete snmp • 35
delete static-route • 35
delete switch • 36
delete user • 36
delete wlan vap • 36
Deleting DHCP Custom Option Code • 48
Deleting Excluded IP Addresses • 32
dynamic objects • 36
set wlan • 71
Setting a Single Trap • 66
Setting Comments for a Local User • 71
Setting Community String • 64
Setting Host Location • 65
Setting Password for a User • 70
Setting Password Hash for a User • 70
Setting Permissions for an Admin User • 71
Setting Remote Access for a Local User • 70
Setting SNMP Host Information • 64
Setting SNMP v2 Receivers • 65
Setting SNMP Version • 64
Setting the IP Pool • 43
shell/expert • 77
show admin access • 78
show backup settings • 79
show bridge • 79
show clock • 79
show commands • 80
show date • 80
show dhcp • 80
show diag • 81
show disk usage • 81
show dns • 82
show domainname • 82
show dynamic-dns • 83
show host • 83
show hostname • 84
show inactivity-timeout • 84
show interface • 85
show interfaces • 85
show internet connection icmp servers • 84
show internet connections • 83
show internet mode • 83
show license • 85
show logs • 85
show memory usage • 86
show network object • 86
show ntp • 86
Show NTP Servers • 86
Show NTP Status • 86
Show PIM Commands • 159
show proxy • 87
Show Radio Configuration • 92
show restore settings log • 87
show revert log • 87
Show Routemap Commands • 149
show rule hits • 88
show saved image • 88
show snmp • 88
Show SNMP Agent • 88
Show SNMP Host Information • 89
Show SNMP Trap Information • 89
Show SNMP Users • 89
show software version • 90
show static-routes • 87
show switch • 90
show time • 90
show timezone • 90
show timezone-dst • 91
show upgrade log • 91
show user • 91
Show VAPS • 92
show vpn tunnel • 92
show wlan • 92
Show wlan Settings • 92
Showing DHCP for an Interface • 81
Showing DHCP IP Pool • 81
Showing DHCP Settings • 80
sim • 77
Single IP • 21, 61
Sparse Mode PIM • 155
Support for BGP-4++ • 96
Supported Linux Commands • 14
Supported Route Map Statements by Protocol • 150
T
TCP MD5 Authentication • 102
Timer and Assert Rank Parameters for Dense Mode and Sparse Mode • 156
Types of Areas • 128
U
upgrade from usb or tftp server • 93
Using Command Line Reference • 12
V
vpn • 93
W
Wireless Connection Settings • 72
Wireless Security • 75
WPA Encryption Type • 75