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=13221)
To learn more, visit the Check Point Support Center (http://supportcenter.checkpoint.com).
For more about this release, see the R75.40 home page

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
</table>
| 20 January 2014    | • BGP clarification for clustered environments ("External BGP" on page 19).
|                    | • Policy Based Routing (on page 73) is supported from R75.45.              |
| 7 January 2013     | • Updated: BGP (on page 9) and OSPF (on page 45) can be used on both numbered and un-numbered interfaces. |
|                    | • Improved BGP Route Refresh (on page 11).                                |
|                    | • Updated: High Availability Support for OSPF (on page 46)                |
|                    | • Added: Policy Based Routing (on page 73)                                |
| 15 May 2012        | • Removed references to routing instances.                               |
|                    | • Improved formatting and document layout.                               |
| 6 May 2012         | Removed references to OSPFv3, RIPng and IP Clustering.                    |
| 21 February 2012   | First release of this document.                                          |

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 Gaia Advanced Routing R75.40 Administration Guide).
RIP Show Commands ................................................................. 44
OSPF ................................................................................................... 45
Types of Areas ......................................................................................... 45
Area Border Routers .................................................................................. 46
High Availability Support for OSPF .......................................................... 46
Configuring OSPF - CLI (ospf) .................................................................... 47
OSPF Global Settings .................................................................................. 47
OSPF Interfaces ......................................................................................... 48
OSPF Virtual Links ...................................................................................... 51
OSPF Areas ............................................................................................... 53
OSPF Show Commands .............................................................................. 54
Route Aggregation ..................................................................................... 58
Configuring Route Aggregation - CLI (aggregate) ........................................... 58
Routing Options ........................................................................................ 60
Configuring Protocol Rank - CLI (protocol-rank) ............................................. 60
Router Discovery ....................................................................................... 61
  How Router Discovery Works .................................................................... 61
  Configuring Router Discovery - CLI (rdisc) .................................................... 61
    ICMP Router Discovery Interfaces .............................................................. 62
    ICMP Router Discovery Show Commands .................................................. 63
Route Maps ............................................................................................... 64
  Configuring Route Map - CLI (routemap) ...................................................... 64
    Set Routemap Commands ........................................................................... 64
    Show Routemap Commands ........................................................................ 69
    Routemap Protocol Commands ................................................................. 69
    Supported Route Map Statements by Protocol ............................................ 70
    Route Map Examples ................................................................................... 71
Policy Based Routing .................................................................................. 73
  Configuring Policy Based Routing - WebUI .................................................. 73
    Action Table Parameters ......................................................................... 74
    Policy Rule Parameters ............................................................................. 75
  Configuring Policy Based Routing - CLI ....................................................... 75
    Monitoring Policy Based Routing .............................................................. 77
PIM ........................................................................................................ 78
  Configuring PIM - CLI (pim) ................................................................. 79
    PIM Interfaces ......................................................................................... 80
    Sparse Mode PIM .................................................................................... 80
    Timer and Assert Rank Parameters for Dense Mode and Sparse Mode ........... 81
    Show PIM Commands .............................................................................. 85
    Debugging PIM - CLI ............................................................................. 86
Index ........................................................................................................ 87
Chapter 1

Introduction to Gaia Advanced Routing

Dynamic Routing is fully integrated into the Gaia WebUI 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.
In This Chapter

Configuring DHCP Relay - CLI (bootp) 7

BOOTP/DHCP Relay extends Bootstrap Protocol (BOOTP) and Dynamic Host Configuration Protocol (DHCP) operation across multiple hops in a routed network. In standard BOOTP, all interfaces on a LAN are loaded from a single configuration server on the LAN. BOOTP Relay allows configuration requests to be forwarded to and serviced from configuration servers located outside the single LAN.

BOOTP/DHCP Relay offers the following advantages over standard BOOTP/DHCP:

- You can provide redundancy by configuring an interface on the Check Point system to relay client configuration requests to multiple servers. With this setup, configuration requests are relayed to all the listed servers simultaneously.
- You can provide load balancing by configuring multiple interfaces on the Check Point system to relay client configuration requests to different servers.
- It allows you to centrally manage client configuration across multiple LANs. This is particularly useful in large enterprise environments.

The Gaia implementation of BOOTP Relay is compliant with RFC 951, RFC 1542, and RFC 2131. BOOTP Relay supports Ethernet and IEEE 802 LANs by using canonical MAC byte ordering, that is, clients that specify Bootp htype=1: 802.3 and FDDI.

When an interface configured for BOOTP Relay receives a boot request, it forwards the request to all the servers in its server list. It does this after waiting a specified length of time to see if a local server answers the boot request. If a primary IP is specified, it stamps the request with that address, otherwise it stamps the request with the lowest numeric IP address specified for the interface.

### Configuring DHCP Relay - CLI (bootp)

**Description**

Use this group of commands to set and view parameters for the bootstrap protocol.

**Syntax**

**Set Commands**

- `set bootp interface VALUE off`
- `set bootp interface VALUE primary VALUE wait-time VALUE on`
- `set bootp interface VALUE relay-to VALUE off`
- `set bootp interface VALUE relay-to VALUE on`
- `set bootp network VALUE off`
- `set bootp network VALUE primary VALUE wait-time VALUE on`
- `set bootp network VALUE relay-to VALUE off`
- `set bootp network VALUE relay-to VALUE on`

**Show Commands**

- `show bootp interface VALUE`
- `show bootp interfaces`
- `show bootp network VALUE`
- `show bootp networks`
- `show bootp stats`
- `show bootp stats receive`
- `show bootp stats reply`
- `show bootp stats request`
**BOOTP Interfaces**

Use this group of commands to configure BOOTP properties for specific interfaces.

```plaintext
set bootp interface if_name
    primary ip_address wait-time <0-65535> on
    relay-to ip_address <on | off>
    off
```

**Arguments**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>primary ip_address wait-time &lt;0-65535&gt; on</td>
<td>Specifies the ip_address to stamp as the gateway address on all BOOTP requests. The wait-time value specifies the minimum amount of time, in seconds, to wait before forwarding a bootp request. Each client-generated bootp request includes the elapsed time since the client began the booting process. The bootp relay does not forward the request until the indicated elapsed time at least equals the specified wait time. This delay provides an opportunity for a local configuration server to reply before attempting to relay to a remote server.</td>
</tr>
<tr>
<td>relay-to ip_address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>off</td>
<td>Disables BOOTP on the specified interface.</td>
</tr>
</tbody>
</table>

**BOOTP Show Commands**

Use this group of commands to monitor and troubleshoot BOOTP implementation.

```plaintext
show bootp
    interfaces
    interface if_name
    stats
    stats receive
    stats request
    stats reply
```
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 implements BGP-4++ to support multiprotocol extensions and exchange IPv6 prefixes as described in RFCs 2545, 2858, and 3392.

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 even though they cannot exchange IPv6 routes.

On each peer you configure the type of routes (capability) that should be exchanged between peers. Choose from the following selections:
- IPv4 unicast (the default)
- IPv6 unicast
- IPv4 unicast and IPv6 unicast

For peering to be established, the routers must share a capability.

If your system is exchanging IPv4 routes over IPv6 or vice versa, use route map commands to set nexthop to match the family of the routes being exchanged. If they do not match, the routes will not be active.

**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 to remove the private AS number from BGP update messages to external peers.

To configure Gaia 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 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 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>AS_PATH</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 AS_SEQUENCES.</td>
</tr>
<tr>
<td>NEXT_HOP</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>MULTI_EXIT_DISC</td>
<td>Discriminates among multiple exit or entry points to the same neighboring autonomous system. Used only on external links.</td>
</tr>
<tr>
<td>LOCAL_PREF</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>ATOMIC_AGGREGATE</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 Advanced Routing > Route Distribution in the WebUI, or 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 (0xFFFFF01)</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 (0xFFFFF02)</td>
<td>Not advertised to other BGP peers.</td>
</tr>
<tr>
<td>NO_EXPORT_SUBCONFED (0xFFFFF03)</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.

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.

Note - TCP MD5 authentication is not available for BGP session over IPv6.

Configuring BGP - CLI (bgp)

Syntax

Show Commands:

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
        auth-type md5 secret VALUE
        auth-type none
        capability default
        capability ipv4-unicast VALUE
        capability ipv6-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 request ipv6 unicast
        send-route-refresh route-update all unicast
        send-route-refresh route-update ipv4 unicast
        send-route-refresh route-update ipv6 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
capability ipv6-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
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 request ipv6 unicast
send-route-refresh route-update all unicast
send-route-refresh route-update ipv4 unicast
send-route-refresh route-update ipv6 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
```
### BGP

<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</td>
</tr>
<tr>
<td></td>
<td>router ID is used by the BGP and OSPF protocols. We recommend setting the</td>
</tr>
<tr>
<td></td>
<td>router ID rather than relying on the default setting. This prevents the</td>
</tr>
<tr>
<td></td>
<td>router ID from changing if the interface used for the router ID goes down.</td>
</tr>
<tr>
<td></td>
<td>Use an address on a loopback interface that is not the loopback address</td>
</tr>
<tr>
<td></td>
<td>(127.0.0.1).</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> - In a cluster, you must select a router ID and make sure that it</td>
</tr>
<tr>
<td></td>
<td>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</td>
</tr>
<tr>
<td></td>
<td>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>

Use the following group of commands to set and view parameters for BGP.

```
set as as_number
set as off
```

### 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>
virtual-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.

Note: If you run BGP in a cluster, you must not configure the local address. Set local-address to off.

virtual-address <on | off> Specifies for this router to use the VRRP virtual IP address as the local endpoint for TCP connections. You must also configure a local address to enable this option. See the command above. You can configure this option only on a VRRP master.

Note:

• You must use Monitored Circuit mode when configuring virtual IP support for BGP or any other dynamic routing protocol.

• In a cluster configuration, do not use virtual address. Set virtual-address to off.

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 supports both IPv4 and IPv6 addresses for BGP peers.

A BGP IPv6 address can be either link local or global scoped. If a link local address is used for peering, the outgoing interface must also be configured.
set bgp external 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-ashop <on | off>
send-keepalives <on | off>
send-route-refresh [request | route-update] [ipv4 | ipv6 | 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 | default>
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 | ipv6-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

**keepalive <2-21945 | default>**

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

**ignore-first-ashop <on | off>**

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.

**send-keepalives <on | off>**

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.

**send-route-refresh [request | route-update] [ipv4 | ipv6 | All] [unicast]**

Specifies that the router dynamically request BGP route updates from peers or respond to requests for BGP route updates.

**route-refresh <on | off>**

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.

**accept-routes <all | none>**

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.

**passive-tcp <on | off>**

Specifies for the router to wait for the specified peer to issue an open message. No tcp connections are initiated by the router.

**removeprivateas <on | off>**

Specifies that private AS numbers be removed from BGP update messages to external peers.
authype none

Specifies not to use an authentication scheme between peers. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.

none

authype md5 secret secret

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.

throttle-count

<0-65535 | off>

Specifies number of BGP updates to send at one time. This option limits the number of BGP updates when there are many BGP peers. Off disables the throttle count option.

ttl <1-255 | default>

Specifies the value of the TTL (time to live) parameter, the number of hops over which the external BGP multihop session is established. Configure this value only if the multihop option is enabled.

64

suppress-default-originate <on | off>

Specifies NOT to generate a default route when the peer receives a valid update from its peer.

log-state-transitions

<on | off>

Specifies for the router to log a message whenever a peer enters or leave the established state.

log-warnings

<on | off>

Specifies for the router to log a message whenever a warning scenario is encountered in the codepath.

trace bgp_traceoption

<on | off>

Specifies tracing options for your BGP implementation. Log messages are saved in the var/log/isprd directory. Enter the following words to set each trace option:

- packets—Trace all BGP packets to this peer.
- open—Trace all BGP open messages to this peer.
- update—Trace all BGP update messages to this peer.
- keepalive—Trace all keepalive messages to this peer.
- all—Trace all message types.
- general—Trace message related to Route and Normal.
- route—Trace routing table changes for routes installed by this peer.
- normal—Trace normal protocol occurrences. Abnormal protocol occurrences are always traced.
- state—Trace state machine transitions in the protocol.
- policy—Trace application of the protocol and user-specified policy to routes being imported and exported.

capability <default | ipv4-unicast | ipv6-unicast>

Specifies capabilities setting. Default is IPv4 unicast.
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.

graceful-restart-helper-stalepath-time seconds
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.

**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><code>confederation identifier as_number</code></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><code>confederation identifier off</code></td>
<td>Disables the confederation identifier.</td>
</tr>
<tr>
<td><code>confederation aspath-loops-permitted &lt;1-10&gt;</code></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><code>confederation aspath-loops-permitted default</code></td>
<td>Specifies a value of 1.</td>
</tr>
<tr>
<td><code>routing-domain identifier as_number</code></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><code>routing-domain identifier off</code></td>
<td>Disables the routing-domain identifier.</td>
</tr>
<tr>
<td><code>routing-domain aspath-loops-permitted &lt;1-10&gt;</code></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><code>routing-domain aspath-loops-permitted default</code></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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>internal peer &lt;peer ip_address&gt; peer-type reflector-client</td>
<td>The peer router &lt;peer ip_address&gt; is a reflector client of the local router.</td>
</tr>
<tr>
<td>internal peer &lt;peer ip_address&gt; peer-type none</td>
<td>The peer router &lt;peer ip_address&gt; is not a reflector client of the local router. This is the default.</td>
</tr>
<tr>
<td>internal peer &lt;peer ip_address&gt; peer-type no-client-reflector</td>
<td>An advanced option.</td>
</tr>
<tr>
<td>cluster-id ip_address</td>
<td>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</td>
</tr>
<tr>
<td>cluster-id off</td>
<td>Disable the cluster ID.</td>
</tr>
<tr>
<td>default-med &lt;0-65535&gt;</td>
<td>The multi-exit discriminator (MED) metric used to advertise routes through BGP.</td>
</tr>
<tr>
<td>default-med off</td>
<td>Disable the specified MED metric.</td>
</tr>
<tr>
<td>default-route-gateway ip_address</td>
<td>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 &lt;ip_address&gt; suppress-default-originate on command.</td>
</tr>
<tr>
<td>default-route-gateway off</td>
<td>Disables the configured default BGP route.</td>
</tr>
</tbody>
</table>

**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>
    virtual-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
    peer_type <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>
    authtype none
    authtype 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></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></td>
<td><strong>Default:</strong> 0</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></td>
<td><strong>Default:</strong> off</td>
</tr>
<tr>
<td>local-address ip_address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If running BGP in a cluster you must not configure the local address. <strong>Default:</strong> Off</td>
</tr>
</tbody>
</table>
| virtual-address <on | off>                | Note:  
- Use this option only in a lab environment with the Firewall turned off.  
- Do not use in a cluster configuration.  
Specifies for this router to use the VRRP virtual IP address as the local endpoint for TCP connections. You must also configure a local address to enable this option. See the command above. You can configure this option only on a VRRP master.  
**Default:** off.                                                                 |
| interface [all | if_name] <on | off>                | Specifies whether to enable the specified internal peer group on all interfaces or a specific interface.                                                                                                              |
| protocol [all bgp_internal_protocol] <on | off>                | Specifies whether to enable all internal routing protocols on the specified internal peer group or specific internal protocols. You can enter the following specific internal protocols: direct, rip, static, ospf, and ospfase. |

Gaia Advanced Routing Administration Guide R75.40 | 29
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>peer ip_address</td>
<td>Specifies an internal peer address and peer type. Enter reflector-client to specify that the local router acts as a route reflector for the group of peers named. That is, the local router is the route reflection server, and the named peers are route reflection clients. Normally, the routing daemon re-advertises, or reflects, routes it receives from one of its clients to all other IBGP peers, including the other peers in that client's group. Enter no-client-reflector to specify that a reflection client's routes are reflected only to internal BGP peers in other groups. Clients in the group are assumed to be direct IBGP peers of each other. Enter none if you do not want to specify route reflection.</td>
</tr>
<tr>
<td>peer_type &lt;on</td>
<td>off&gt;</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 re-establish a session upon reconfiguration if import route policy is changed.</td>
</tr>
<tr>
<td>peer ip_address weight off</td>
<td></td>
</tr>
<tr>
<td>peer ip_address aggregator id &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>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 holdtime &lt;6-65535&gt;</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>peer ip_address holdtime default</td>
<td>Specifies a holdtime of 180 seconds.</td>
</tr>
<tr>
<td>peer ip_address keepalive &lt;2-21845&gt;</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>peer ip_address keepalive default</td>
<td>Specifies a keepalive interval of 60 seconds.</td>
</tr>
<tr>
<td>peer ip_address ignore-first-ashop &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>peer ip_address 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 accept-routes all</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>peer ip_address accept-routes none</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>peer ip_address passive-tcp &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>peer ip_address 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 authtype md5 secret secret</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. <strong>Note</strong> - TCP MD5 authentication for BGP is not supported on 64-bit Gaia or on a 32/64-bit Gaia VSX cluster.</td>
</tr>
<tr>
<td>peer ip_address throttle-count &lt;0-65535&gt;</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>
</tbody>
</table>
### BGP Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>peer ip_address throttle count off</td>
<td>Enables the throttle count option.</td>
</tr>
<tr>
<td>peer ip_address log-state-transitions &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>peer ip_address log-warnings &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>peer ip_address trace bgp_traceoption &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>graceful-restart-helper &lt;on</td>
<td>off&gt;</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 so that they can be 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>off&gt;</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>off&gt;</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 Chapter

Configuring IGMP - CLI (igmp) 35

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 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</td>
</tr>
<tr>
<td>&lt;1-25&gt;</td>
<td>queries. The last member query interval may be used to tune the &quot;leave</td>
</tr>
<tr>
<td></td>
<td>latency.&quot; A smaller value results in a reduction in the time to detect the</td>
</tr>
<tr>
<td>last-member-query-interval</td>
<td>loss of the last member of a multicast group. This value must always be</td>
</tr>
<tr>
<td>default</td>
<td>be less than the query interval.</td>
</tr>
<tr>
<td>loss-robustness &lt;1-255&gt;</td>
<td>Allows tuning for the expected packet loss on a subnet. If the subnet is</td>
</tr>
<tr>
<td></td>
<td>expected to be highly lossy, then the &quot;loss robustness&quot; value may be</td>
</tr>
<tr>
<td></td>
<td>increased. IGMP protocol operation is robust to (lossrobustness - 1) packet</td>
</tr>
<tr>
<td></td>
<td>loss</td>
</tr>
<tr>
<td></td>
<td>• Default: 2</td>
</tr>
<tr>
<td>loss-robustness default</td>
<td>A value of 2.</td>
</tr>
<tr>
<td>query-interval &lt;1-3600&gt;</td>
<td>The interval (in seconds) between IGMP general queries sent by the querier</td>
</tr>
<tr>
<td></td>
<td>router. This parameter can be used to tune the IGMP messaging overhead and</td>
</tr>
<tr>
<td></td>
<td>has a secondary effect on the timeout of idle IP multicast groups.</td>
</tr>
<tr>
<td></td>
<td>• Default: 125</td>
</tr>
<tr>
<td>query-interval default</td>
<td>A value of 125.</td>
</tr>
<tr>
<td>query-response-interval</td>
<td>The maximum response time (in seconds) inserted into the periodic IGMP</td>
</tr>
<tr>
<td>&lt;1-25&gt;</td>
<td>general queries. The query response interval may be used to tune the</td>
</tr>
<tr>
<td></td>
<td>burstiness of IGMP messages; a larger value spreads the host IGMP reports</td>
</tr>
<tr>
<td></td>
<td>over a larger interval, reducing burstiness. This value must always be</td>
</tr>
<tr>
<td></td>
<td>be less than the query interval.</td>
</tr>
<tr>
<td></td>
<td>• Default: 10.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>default</td>
<td></td>
</tr>
<tr>
<td>router-alert &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td>Allows the &quot;disable insertion of IP router alert&quot; option in all IGMP messages sent on the interface. This can be useful in interoperating with broken IP implementations that may discard the packet due to the use of this option.</td>
</tr>
<tr>
<td></td>
<td>• Default: off</td>
</tr>
<tr>
<td>local-group address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td>A multicast group address. A local group provides a mechanism to simulate the presence of local receivers for specific groups. When a multicast group is added to an interface, IGMP sends a membership report on the interface.</td>
</tr>
<tr>
<td>static-group address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td>• A multicast group address. A static group provides a mechanism to simulate the presence of local receivers on an interface. When a static group is configured on an interface that is also running a parent multicast protocol (such as PIM) IGMP informs the parent of the presence of a local receiver. In contrast to regular IGMP, no membership reports are sent on the corresponding interface.</td>
</tr>
<tr>
<td></td>
<td>If the same multicast group is configured as both a local and a static group, local group takes precedence, that is, membership reports are sent out on the interface.</td>
</tr>
<tr>
<td>version &lt;1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>IGMP version 2 is compatible with IGMP version 1, and version 3 is compatible with versions 2 and 1. Check Point recommends that you use version 1 only on networks that include multicast routers that are not upgraded to IGMP versions 2 or 3.</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

IP Broadcast Helper

In This Chapter

Configuring IP Broadcast Helper - CLI (iphelper) 37

IP Broadcast Helper is a form of static addressing that uses directed broadcasts to forward local and all-nets broadcasts to desired destinations within the internetwork. IP Broadcast Helper allows the relaying of broadcast UDP packets on a LAN as unicasts to one or more remote servers. This is useful when you relocate servers or hosts from their original common segment and still want the service to be available.

Note - For more information, see RFC1542 section 4.

Configuring IP Broadcast Helper - CLI (iphelper)

Description

Use the following group of commands to set and view parameters for IP Broadcast Helper.

Syntax

Set Commands:

set iphelper forward-nonlocal off
set iphelper forward-nonlocal on
set iphelper interface VALUE off
set iphelper interface VALUE udp-port VALUE off
set iphelper interface VALUE udp-port VALUE relay-to VALUE off
set iphelper interface VALUE udp-port VALUE relay-to VALUE on

Show Commands:

show iphelper services
show iphelper stats

IP Broadcast Helper Forwarding

Use the following commands to control whether to forward packets that are not locally originated by a source directly on the receiving interface.

set iphelper forward-nonlocal <on | off>

IP Broadcast Helper Interfaces

Use the following commands configure IP Broadcast Helper properties for specific interfaces.
set iphelper interface if_name off
  udp-port <1-65535> off
  udp-port <1-65535> relay-to ip_address <on | off>

### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>forward-nonlocal &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>interface &lt;if_name&gt; off</td>
<td>Specifies to disable the interface configured for <code>iphelper</code>.</td>
</tr>
<tr>
<td>udp-port &lt;1-65535&gt; off</td>
<td>Specifies to disable the UDP services configured for this interface.</td>
</tr>
<tr>
<td>udp-port &lt;1-65535&gt; relay-to ip_address &lt;on</td>
<td>off&gt;</td>
</tr>
</tbody>
</table>

### IP Broadcast Helper Show Commands

Use these commands to monitor and troubleshoot your IP Broadcast Helper implementation.

- `show iphelper services`
- `show iphelper stats`
Chapter 6

RIP

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.

**Virtual IP Address Support for VRRP**
Gaia supports the advertising of the virtual IP address of the VRRP Virtual Router. You can configure RIP to advertise the virtual IP address rather than the actual IP address of the interface. If you enable this option, RIP runs only on the master of the Virtual Router; on a failover, RIP stops running on the old master and then starts running on the new master. A traffic break might occur during the time it takes both the VRRP and RIP protocols to learn the routes again. The larger the network, the more time it would take RIP to synchronize its database and install routes again.

**Note**
Gaia also provides support for BGP, OSPF, and PIM, both Sparse-Mode and Dense-Mode, to advertise the virtual IP address of the VRRP Virtual Router.

You must use Monitored Circuit mode when configuring virtual IP support for any dynamic routing protocol, including RIP.

**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

set rip update-interval 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 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>  
virtual address <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>Default: 1</td>
<td></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 &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>Default: off</td>
<td></td>
</tr>
<tr>
<td>send-updates &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>transport &lt;multicast</td>
<td>broadcast&gt;</td>
</tr>
<tr>
<td>Selecting Multicast specifies that RIP version 2 packets should be multicast on this interface. This is the default.</td>
<td></td>
</tr>
<tr>
<td>Note - When you use RIP 2, always select multicast. We recommend that you do not operate RIP 1 and RIP 2 together.</td>
<td></td>
</tr>
<tr>
<td>Selecting Broadcast specifies that RIP version 1 packets that are compatible with version 2 should be broadcast on this interface.</td>
<td></td>
</tr>
<tr>
<td>authtype none</td>
<td>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.</td>
</tr>
<tr>
<td>authtype simple password</td>
<td>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.</td>
</tr>
<tr>
<td>authtype md5 secret secret</td>
<td>Implement an authentication scheme that uses an MD5 algorithm for the interface to accept routing information from neighboring routers. Enter the password.</td>
</tr>
</tbody>
</table>
### RIP Global Commands

Use these commands to configure RIP properties that apply to all interfaces configured for RIP.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| set rip            | **auto-summary <on | off>**  
|                    | **update-interval <1-65535>**  
|                    | **update-interval default**  
|                    | **expire-interval <1-65535>**  
|                    | **expire-interval default** |

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>auto-summary &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> - Take care when you set this parameter, as RIP has no protocol mechanism to detect misconfiguration.</td>
</tr>
<tr>
<td></td>
<td>Default: on</td>
</tr>
</tbody>
</table>
| 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.                               |
|                   | Default: on                                                                              |
| 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. |
|                   | 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 7

OSPF

In This Chapter

- Types of Areas 45
- Area Border Routers 46
- High Availability Support for OSPF 46
- Configuring OSPF - CLI (ospf) 47

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 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 supports the OSPF protocol in clusters configured either via VRRP or 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 also supports the OSPF protocol over VPN tunnels which terminate in the VRRP or ClusterXL cluster.

VRRP

Gaia supports the advertising of the virtual IP address of the VRRP Virtual Router. You can configure OSPF to advertise the virtual IP address rather than the actual IP address of the interface.

If you enable this option, OSPF runs only on the master of the Virtual Router; on a failover, OSPF stops being active on the old master and then starts becoming active on the new master. Because the OSPF routes database of the master is not synchronized across all members of the cluster, a traffic break may occur during the time it takes the VRRP to become active and the OSPF protocol to learn routes again. The larger the network, the more time it takes OSPF to synchronize its database and install routes again.

Note - You must use monitored-circuit VRRP, not VRRP v2, when configuring virtual IP support for OSPF or any other dynamic routing protocol.
ClusterXL
Gaia 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 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, anything configured on 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. Use these commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set router-id</td>
<td>Selects the highest interface address when OSPF is enabled.</td>
</tr>
<tr>
<td>default</td>
<td></td>
</tr>
<tr>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>router-id default</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). 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.</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:
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.

The time, in seconds, to wait before recalculating the OSPF routing table after a change in the topology.

Specifies an SPF delay time of 2 seconds.

Specifies the minimum time, in seconds, between recalculations of the OSPF routing table.

Specifies an SPF holdtime of 5 seconds.

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 precedence.

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 precedence.

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.

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>
  hello-interval default
  dead-interval <1-65535>
  dead-interval default
  retransmit-interval <1-65535>
  retransmit-interval default
  cost <1-65535>
  priority <0-255>
  passive <on | off>
  virtual-address <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 virtual-address <on | off>

Enables OSPF on the virtual IP address associated with this interface. This option functions only if this router is a VRRP master. You must also configure VRRP to accept connections to VRRP IPs. See “General VRRP Commands” for more information.

Default: off

Note - You must use Monitored Circuit mode when configuring virtual IP support for OSPF or any other dynamic routing protocol.

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.

**ip_address transit-area ospf_area 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.

**ip_address transit-area ospf_area hello-interval default**
Specifies an interval of 10 seconds.

**ip_address transit-area ospf_area dead-interval <1-4294967295>**
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.

**ip_address transit-area ospf_area dead-interval default**
Specifies a value of 40 seconds.

**ip_address transit-area ospf_area retransmit-interval <1-4294967295>**
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 network.

**ip_address transit-area ospf_area retransmit-interval default**
Specifies a value of 5 seconds.

**ip_address transit-area ospf_area authtype none**
Specifies not to use an authentication scheme for the specified interface.
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.

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.

**OSPF Areas**

Use the following commands to configure OSPF areas, including the backbone and stub areas.

For OSPFv2, use the following commands.

```
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>
```

For OSPFv3 use the following commands. NSSA is not available for OSPFv3.

```
set ipv6 ospf3 area backbone <on | off>
set ipv6 ospf3 area ospf_area
  <on| off>
  stub <on | off>
  stub default-cost <1-677215>
  stub summary <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.
**OSPF**

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.

interface ifname <on | off>
  Displays the use of the VRRP virtual link-local address as the source of its control packets

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.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface if_name</td>
<td>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.</td>
</tr>
<tr>
<td>interface if_name stats</td>
<td>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.</td>
</tr>
<tr>
<td>interface if_name detailed</td>
<td>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.</td>
</tr>
<tr>
<td>packets</td>
<td>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.</td>
</tr>
<tr>
<td>errors</td>
<td>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.</td>
</tr>
<tr>
<td>errors dd</td>
<td>Displays the number of each type of database-description error messages only.</td>
</tr>
<tr>
<td>errors hello</td>
<td>Displays the number of each type of hello-error message only.</td>
</tr>
<tr>
<td>errors ip</td>
<td>Displays the number of each type of IP-errors message only.</td>
</tr>
<tr>
<td>errors lsack</td>
<td>Displays the number of each type of link-state acknowledgment error message only.</td>
</tr>
<tr>
<td>errors lsu</td>
<td>Displays the number of each type of link-state update error message only.</td>
</tr>
<tr>
<td>errors lsr</td>
<td>Displays the number of each type of link-state request error messages only.</td>
</tr>
<tr>
<td>errors protocol</td>
<td>Displays the number of each type of protocol error message only.</td>
</tr>
<tr>
<td>border-routers</td>
<td>Displays the IP address of each area border router, the OSPF area of each border router, and the cost associated with each IP address.</td>
</tr>
<tr>
<td>database</td>
<td>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.</td>
</tr>
<tr>
<td>database areas</td>
<td>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.</td>
</tr>
<tr>
<td>database area ospf_area</td>
<td>Displays router-link state, 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.</td>
</tr>
<tr>
<td>database asbr-summary</td>
<td>Displays a summary of AS-border-router link state statistics for each OSPF area. For OSPFv2 only.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>database external</strong></td>
<td>Displays AS-external-link state statistics for each OSPF area.</td>
</tr>
<tr>
<td><strong>database database</strong></td>
<td>Displays a summary of router-link-state, network-link state, summary-link-state, and AS-border-router-link state statistics.</td>
</tr>
<tr>
<td><strong>database database-summary</strong></td>
<td>Displays a summary of router-link-state, network-link state, summary-link-state, and AS-border-router-link state statistics.</td>
</tr>
<tr>
<td><strong>database network</strong></td>
<td>Displays network-link-state statistics, including the advertised router, sequence number, and checksum of each OSPF interface. For OSPFv2 only.</td>
</tr>
<tr>
<td><strong>database nssa-external-lsa</strong> [detailed]</td>
<td>Displays type 7 LSAs (NSSA). This argument applies only to OSPF v2.</td>
</tr>
<tr>
<td><strong>database router-lsa</strong></td>
<td>Displays router-link-state statistics, including the advertised router, sequence number, checksum, and link count, of each OSPF interface. For OSPFv2 only.</td>
</tr>
<tr>
<td><strong>database summary-lsa</strong></td>
<td>Displays a summary of link-state statistics for each OSPF area. For OSPFv2 only.</td>
</tr>
<tr>
<td>**database type &lt;1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>• 1—router-link-state statistics.</td>
</tr>
<tr>
<td></td>
<td>• 2—network-link-state statistics.</td>
</tr>
<tr>
<td></td>
<td>• 3—summary-link-state statistics.</td>
</tr>
<tr>
<td></td>
<td>• 4—AS-border-router-link-state statistics.</td>
</tr>
<tr>
<td></td>
<td>• 5—AS-external-link-state statistics.</td>
</tr>
<tr>
<td></td>
<td>• 7—NSSA. This option applies only to OSPF v2.</td>
</tr>
<tr>
<td><strong>events</strong></td>
<td>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.</td>
</tr>
</tbody>
</table>
Chapter 8

Route Aggregation

In This Chapter

Configuring Route Aggregation - CLI (aggregate) 58

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>contributing-route &lt;all</td>
<td>ip_prefix &lt;on</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>&lt;ip_prefix&gt; refines on</code></td>
<td>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 <code>exact on</code> and <code>refines on</code> at the same time. If you enable <code>refines on</code> when <code>exact on</code> is enabled, <code>exact on</code> is automatically disabled.</td>
</tr>
<tr>
<td><code>rank default</code></td>
<td>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.</td>
</tr>
<tr>
<td><code>rank &lt;0-255&gt;</code></td>
<td>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.</td>
</tr>
<tr>
<td><code>weight default</code></td>
<td>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.</td>
</tr>
<tr>
<td><code>weight &lt;0-65535&gt;</code></td>
<td>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.</td>
</tr>
<tr>
<td>`aspath-truncate &lt;on</td>
<td>off&gt;`</td>
</tr>
</tbody>
</table>

- **Range**: 0-65535.
- **Default**: 0
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 10

Router Discovery

In This Chapter

How Router Discovery Works 61
Configuring Router Discovery - CLI (rdisc) 61

The ICMP Router Discovery protocol is an IETF standard protocol that allows hosts running an ICMP router discovery client to learn dynamically about the presence of a viable default router on a LAN. It is intended to be used instead of having hosts wiretap routing protocols such as RIP. It is used in place of, or in addition to, statically configured default routes in hosts.

Note - Only the server portion of the Router Discovery Protocol is supported.

Gaia implements only the ICMP router discovery server portion, which means that a Check Point router can advertise itself as a candidate default router, but it will not adopt a default router using the router discovery protocol.

The ICMP Router Discovery Service provides a mechanism for hosts attached to a multicast or broadcast network to discover the IP addresses of their neighboring routers. This section describes how you can configure a router to advertise its addresses by using ICMP Router Discovery.

How Router Discovery Works

The router discovery server runs on routers and announces their existence to hosts. It does this by periodically multicasting or broadcasting a router advertisement to each interface on which it is enabled. These advertisements contain a list of all the router addresses on a given interface and their preference for use as a default router.

Initially, these router advertisements occur every few seconds. They then fall back to every few minutes. In addition, a host can send a router solicitation, to which the router responds with a unicast router advertisement. However, if a multicast or broadcast advertisement is due in a moment, the router does not respond with a unicast advertisement.

Each router advertisement contains an advertisement lifetime field indicating the length of time that the advertised addresses are valid. This lifetime is configured such that another router advertisement is sent before the lifetime expires. A lifetime of zero (0) indicates that one or more addresses are no longer valid.

On systems that support IP multicasting, the router advertisements are sent by default to the all-hosts multicast address 224.0.0.1. However, you can specify the use of broadcast. All IP addresses configured on the physical interface are included in the router advertisement when:

- Router advertisements are sent to the all-hosts multicast address, or
- An interface is configured for the limited-broadcast address 255.255.255.255.

When the router advertisements are sent to a net or subnet broadcast, only the address associated with that net or subnet is included.

Configuring Router Discovery - CLI (rdisc)

Description  Use this group of commands to set and view parameters for the ICMP router discovery protocol.
**Syntax**

Set Commands:

```
set rdisc interface VALUE
adv-lifetime VALUE
advertise VALUE off
advertise VALUE on
advertise VALUE preference VALUE
max-adv-interval VALUE
min-adv-interval VALUE
off
on
```

Show commands:

```
show rdisc

show rdisc
   interface VALUE
   interfaces
   stats
   summary
```

---

**ICMP Router Discovery Interfaces**

Use the following commands to configure router discovery properties for specific interfaces.

```
set rdisc interface if_name
   <on | off>
   min-adv-interval <3-1800>
   min-adv-interval default
   max-adv-interval <4-1800>
   max-adv-interval default
   adv-lifetime integer
   adv-lifetime default
   advertise ip_address <on | off>
   advertise ip_address preference ineligible
   advertise ip_address preference integer
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>min-adv-interval</td>
<td>Specifies the minimum time (in seconds) allowed between sending unsolicited broadcast or multicast ICMP router advertisements on the interface.</td>
</tr>
<tr>
<td>&lt;3-1800&gt;</td>
<td>Specifies a value of 450 seconds.</td>
</tr>
<tr>
<td>min-adv-interval</td>
<td>Specifies the maximum time (in seconds) allowed between sending unsolicited broadcast or multicast ICMP router advertisements on the interface.</td>
</tr>
<tr>
<td>default</td>
<td>Specifies a value of 600 seconds.</td>
</tr>
<tr>
<td>max-adv-interval</td>
<td>Specifies the time (in seconds) placed in the lifetime field of router advertisement packets sent from the interface. Enter an integer value between the configured value for the maximum advertisement interval and 9000.</td>
</tr>
<tr>
<td>&lt;4-1800&gt;</td>
<td>Specifies a value of 600 seconds.</td>
</tr>
</tbody>
</table>
adv-lifetime default: Specifies a value of 1800 or 3 times the configured maximum advertisement interval.

advertise ip_address <on | off>: Specifies whether to advertise the specified IP address that is associated with the interface should be advertised in router advertisement packets.

advertise ip_address preference ineligible: Specifies not to use the specified IP address as a default router.

advertise ip_address preference integer: Specifies the preferability of the specified IP address as a default router address relative to other router addresses on the same subnet.

**ICMP Router Discovery Show Commands**

Use the following commands to monitor and troubleshoot your ICMP router discovery implementation.

```
show rdisc
  interfaces
  interface if_name
  stats
  summary
```
Route Maps

In This Chapter
Configuring Route Map - CLI (routemap) 64

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 route map:

```
set routemap rm_name id <1-65535> <off|on>
   allow
   inactive
   restrict
```
### Route Maps

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>routemap <em>rm_name</em></td>
<td>The name of the routemap.</td>
</tr>
<tr>
<td>id <code>&lt;1-65535&gt;</code></td>
<td>The ID of the routemap. You can enter the keyword <code>default</code> or the default value <code>10</code>.</td>
</tr>
</tbody>
</table>
| `<off|on>`            | - on to create a routemap,  
- off to delete a routemap.                                                                                                                |
| allow                | Allow routes that match the routemap.                                                                                                     |
| inactive             | Temporarily disable a routemap. To activate the routemap, use the allow or restrict arguments.                                            |
| restrict             | Routes that match the routemap are not allowed.                                                                                           |

#### To specify actions for a routemap:

> **Note** - Some statements affect only a particular protocol (["Supported Route Map Statements by Protocol" on page 70](#)).

The same parameter cannot appear both as a match and action statement in a routemap. These include Community, Metric, and Nexthop.

```markdown
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| 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  
|                                   | ospfautomaticitag tag  
|                                   | ospfmanualtag tag  
|                                   | riptag tag                                                                                                                                 |
```

#### Parameter  | Description
---           | ---
<p>| routemap <em>rm_name</em> | Specifies the name of the routemap. |
| id <em>id_number</em>    | Specifies the ID of the routemap. You can enter the keyword <code>default</code> or the default value <code>10</code>. |
| aspath-prepend-count | 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 ASPATH before sending out an update. BGP only. |
| community <code>&lt;append | replace | delete&gt; [on|off]</code> | Operate on a BGP community string. A community string can be formed using multiple community action statements. You can specify keywords append, replace, or delete for the kind of operation to be performed using the community string. The default operation is append. BGP only. |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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 [&lt;add</td>
<td>subtract&gt;] &lt;1-16&gt;</td>
</tr>
<tr>
<td>metric igp [&lt;add</td>
<td>subtract&gt; &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. Note: The ipv6 address should not be a link-local address.</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>
**Parameter**  
**Description**  

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| remove action_name | Remove the specified action from the routemap. For community, it removes all community statements. Allowed values for action_name are:  
  - aspath-regex  
  - community  
  - ifaddress  
  - interface  
  - metric  
  - neighbor  
  - network  
  - nexthop  
  - protocol  
  - route-type |
| ospfautomatictag tag | Creates an automatic OSPF route tag. |
| ospfmanualtag tag | Creates a manual OSPF route tag. |
| riptag tag | Creates a RIP route tag. |

**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 70).

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
```

**Parameter**  
**Description**  

<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>
</tbody>
</table>
| aspath-regex ["<regular-expression" | empty] origin <any | igp | incomplete] | Match the specified aspath regular expression. For BGP only.  
**Note**: Enter the regular expression in quotation marks. Use the empty keyword to match a null ASPath. |
<p>| community &lt;1-65535&gt; as &lt;1-65535&gt; [on|off] | Specify the BGP community value. |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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. |
<p>| nexthop IPv4_addr [on | off] | Match the specified nexthop address. |
| protocol &lt;ospf2 | ospf2ase | bgp | rip | static | direct | aggregate&gt; | Match the specified protocol. Use this for route redistribution. |</p>
<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>
<tr>
<td>remove match_condition_name</td>
<td>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.</td>
</tr>
</tbody>
</table>

**Show Routemap Commands**

```plaintext```
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.

```plaintext```
set <ospf | rip > export-routemap rm_name preference VALUE on
import-routemap rm_name preference VALUE on
```

To turn a routemap off:

```plaintext```
set <ospf | rip > export-routemap rm_name off
import-routemap rm_name off
```

To view routemaps assigned to protocols:

```plaintext```
show <ospf | rip> routemap
```

To set BGP routemaps for export and import policies:

```plaintext```
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
show bgp routemap
```

**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). <strong>Note</strong> - In a cluster, you must select a router ID and make sure that it is the same on all cluster members. <strong>Range:</strong> Dotted-quad([0-255].[0-255].[0-255].[0-255]). Do not use 0.0.0.0 <strong>Default:</strong> 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 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.
Policy Based Routing

Note - This feature is available in a clean installation of R75.45 and higher. It is not available when upgrading to R75.40 Gaia.

You can control traffic forwarding in great detail using policy based routing (PBR). When you use PBR, you create routing tables of static routes (Action Tables), and direct traffic to the appropriate tables using Policy Rules.

Policy Rules
The Policy Rules specify what traffic is matched. You can route traffic by matching on one or more of the following:

- Security Gateway interface.
- Source IPv4 address and subnet mask.
- Destination IPv4 address and subnet mask.

The Policy Rules also specify the action to take if the traffic is matched:

- Prohibit: Send a Prohibit message to the sending host.
- Unreachable: Send an Unreachable message to the sending host.
- Table: Do the actions defined in an Action Table

You can define many Policy Rules. Traffic is matched to all the rules, one rule at a time, according to the priority that is configured for the rule.

Action Tables
The Action Tables define the static routes, that is, where the traffic is sent. You define the destination of the route and the next hop gateway to that destination.

Static Route Priorities
PBR static routes have priority over static routes in the routing table. When a packet arrives at the Gaia Security Gateway, the packet is checked for a match to a PBR static route. If it matches, the packet is forwarded according to the priority of the PBR static route. If the packet does not match a PBR static route, the packet is forwarded according to the priority of the static routes in the routing table.

Configuring Policy Based Routing - WebUI
The workflow for configuring Policy Based Routing (PBR) is

1. In the Gaia WebUI, go to the Advanced Routing > Policy Based Routing page.
2. Configure one or more Action Tables. The Action Tables define the static routes, that is, where the traffic is sent. You define the destination of the route and the next hop gateway to that destination.
3. Configure Policy Rules. Define the traffic to match and the action to take if the traffic is matched. One of the possible Actions is to forward traffic to the static routes defined in an Action Table.

To Add an Action Table:

Note - For the meaning of the parameters, see Action Table Parameters (on page 74).

1. In the Action Tables section of the Policy Based Routing page, click Add.
2. In the Add Policy Table with Static Route window, define the Table Name.
3. Define the route to the destination. Choose one of:
   - Default Route.
   - Destination IPv4 address and Subnet mask.
4. Select the Next Hop Type. One of:
- Normal
- Blackhole
- Reject

5. Add one or more Gateways to a normal destination. Click **Add Gateway**. You can select:
   - **IP Address**. Define the Gateway **Priority**.
   - **Network interfaces**. Define the Gateway **Priority**.

6. Click **Save**.

### To Add a Policy Rule:

*Note* - For the meaning of the parameters, see **Policy Rule Parameters** (on page 75).

1. In the **Policy Rules** section of the **Policy Based Routing** page, click **Add**.
2. In the **Add Policy Rule** window, configure the **Priority** of the rule.
3. Configure the **Action** to take on the traffic that is matched. Choose one of:
   - **Prohibit**
   - **Unreachable**
   - **Table**. Select the Action Table.
4. Configure the traffic to match. Choose one or more of the following:
   - **Interface**
   - **Source** IPv4 address and **subnet mask**.
   - **Destination** IPv4 address and **subnet mask**.
5. Click **Save**.

### To Delete a Policy Rule:

1. In the **Policy Rules** section of the **Policy Based Routing** page, select a rule.
2. Click **Delete**.

### To Delete an Action Table:

1. In the **Action Tables** section of the **Policy Based Routing** page, select a static route table.
2. Click **Delete**.

### Action Table Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Name</td>
<td>The name of the table.</td>
</tr>
<tr>
<td>Table ID</td>
<td>A numerical ID for the table. Assigned by the system.</td>
</tr>
<tr>
<td>Default route</td>
<td>The default static route in the system routing table.</td>
</tr>
<tr>
<td>Destination</td>
<td>The destination of the route.</td>
</tr>
<tr>
<td>Subnet mask</td>
<td>Subnet mask for the destination of the route.</td>
</tr>
<tr>
<td>Next Hop Type</td>
<td>Choose one of:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Normal</strong>: Accept and forward packets.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Reject</strong>: Drop packets and send <em>unreachable</em> messages.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Black Hole</strong>: Drop packets but don't send <em>unreachable</em> messages.</td>
</tr>
<tr>
<td>Gateway IP address</td>
<td>Next hop gateway IPv4 address.</td>
</tr>
<tr>
<td>Gateway Interface</td>
<td>Security Gateway interface that leads to the next hop gateway.</td>
</tr>
</tbody>
</table>
Policy Based Routing

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Name</td>
<td>The name of the table.</td>
</tr>
<tr>
<td>Gateway Priority</td>
<td>The preference of the particular route.</td>
</tr>
<tr>
<td></td>
<td>• Range: 1-8</td>
</tr>
</tbody>
</table>

Policy Rule Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>You can define many Policy Rules. Traffic is matched to all the rules, one</td>
</tr>
<tr>
<td></td>
<td>rule at a time, according to the priority that is configured for the rule.</td>
</tr>
<tr>
<td>Action</td>
<td>The action to take if the traffic is matched</td>
</tr>
<tr>
<td>Prohibit</td>
<td>Send a Prohibit message to the sending host.</td>
</tr>
<tr>
<td>Unreachable</td>
<td>Send an Unreachable message to the sending host.</td>
</tr>
<tr>
<td>Table</td>
<td>Do the actions defined in an Action Table.</td>
</tr>
<tr>
<td>Match</td>
<td></td>
</tr>
<tr>
<td>Interface</td>
<td>Match by: Interface through which the packets enter the Security Gateway</td>
</tr>
<tr>
<td></td>
<td>from the source host.</td>
</tr>
<tr>
<td>Source, subnet mask</td>
<td>Match by: Source IPv4 address and subnet mask.</td>
</tr>
<tr>
<td>Destination, Subnet mask</td>
<td>Match by: Destination IPv4 address and subnet mask.</td>
</tr>
</tbody>
</table>

Configuring Policy Based Routing - CLI

Create routing tables of static routes (Action Tables) and direct traffic to the appropriate tables by using Policy Rules.

To configure Action Tables:

<table>
<thead>
<tr>
<th>Description</th>
<th>The Action Tables define the static routes, that is, where the traffic is sent. You define the destination of the route and the next hop gateway to that destination.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syntax</td>
<td>set pbr table VALUE static-route VALUE nexthop blackhole nexthop reject nexthop gateway address VALUE &lt;on</td>
</tr>
</tbody>
</table>

Parameters
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>table VALUE</td>
<td>The name of the table.</td>
</tr>
<tr>
<td>static-route VALUE</td>
<td>Choose one of:</td>
</tr>
<tr>
<td></td>
<td>• The default static route in the system routing table. VALUE is default.</td>
</tr>
<tr>
<td></td>
<td>• The destination of the route. VALUE is IPv4 address /mask length. For example 192.0.2.0/24</td>
</tr>
<tr>
<td>nexthop blackhole</td>
<td>Drop packets but don't send unreachable messages.</td>
</tr>
<tr>
<td>nexthop reject</td>
<td>Drop packets and send unreachable messages.</td>
</tr>
<tr>
<td>gateway address VALUE</td>
<td>Accept and forward packets to the Next hop gateway IPv4 address.</td>
</tr>
<tr>
<td>gateway logical VALUE</td>
<td>Accept and forward packets to the Security Gateway interface that leads to the next hop gateway.</td>
</tr>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
</tbody>
</table>

**Example**

```
set pbr table PBRtable2 static-route 192.0.2.0/24 nexthop
gateway logical logical eth0 on
```

**To configure Policy Rules**

**Description**

Define the traffic to match and the action to take if the traffic is matched. One of the possible Actions is to forward traffic to the static routes defined in an Action Table.

**Syntax**

```
set pbr rule priority VALUE
    action prohibit
    action unreachable
    action table VALUE
    match from VALUE to VALUE interface VALUE <on | off>
```

**Parameters**
Policy Based Routing

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>priority VALUE</td>
<td>You can define many Policy Rules. Traffic is matched to all the rules, one</td>
</tr>
<tr>
<td></td>
<td>rule at a time, according to the priority that is configured for the rule.</td>
</tr>
<tr>
<td></td>
<td>VALUE is a number. The highest priority is 1.</td>
</tr>
<tr>
<td>action prohibit</td>
<td>Send a Prohibit message to the sending host.</td>
</tr>
<tr>
<td>action unreachable</td>
<td>Send an Unreachable message to the sending host.</td>
</tr>
<tr>
<td>action table VALUE</td>
<td>Do the actions defined in an Action Table. You must specify one or more</td>
</tr>
<tr>
<td></td>
<td>match parameters.</td>
</tr>
<tr>
<td>from VALUE</td>
<td>Match by: Source IPv4 address and subnet mask.</td>
</tr>
<tr>
<td>to VALUE</td>
<td>Match by: Destination IPv4 address and subnet mask.</td>
</tr>
<tr>
<td>interface VALUE</td>
<td>Match by: Interface through which the packets enter the Security Gateway</td>
</tr>
<tr>
<td></td>
<td>from the source host.</td>
</tr>
<tr>
<td>&lt;on</td>
<td>off&gt;</td>
</tr>
</tbody>
</table>

Example

```
corp_fw> set pbr rule priority 3 match interface eth2
corp_fw> set pbr rule priority 3 action table PBRtable2
```

Monitoring Policy Based Routing

To monitor Policy Based Routing - WebUI

1. In the Gaia WebUI, go to the Advanced Routing > Policy Based Routing page.
2. Click the Monitoring tab.

To monitor Policy Based Routing - CLI

Run these commands:

```
show pbr tables
show pbr rules
show pbr summary
```
In This Chapter

Configuring PIM - CLI (pim) 79

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

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>
  virtual-address <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.
**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>
    assert-limit default
    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.
virtual-address <on | off>
Specifies to enable VRRP virtual IP address on the specified PIM interface. This option lets you configure either a PIM Sparse-Mode or PIM Dense-Mode interface to advertise the VRRP virtual IP address if the router transitions to become VRRP master after a failover. When you enable virtual IP support for VRRP on a PIM interface, it establishes the neighbor relationship using the virtual IP if the router is a VRRP master. The master in the VRRP pair sends hello messages that include the virtual IP as the source address and processes PIM control messages from routers that neighbor the VRRP pair.

Note - You must use Monitored Circuit mode when configuring virtual IP support for any dynamic routing protocol, including PIM, either sparse-mode or dense-mode.

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.

Note - Beginning with Gaia 3.8, you can configure PIM to advertise the virtual VRRP IP address on a interface with PIM enabled. You do not need to enable HA mode if you configure the interface to advertise the virtual VRRP IP address. Default value: off

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.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bootstrap-candidate</td>
<td>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.</td>
</tr>
<tr>
<td>priority &lt;0-255&gt;</td>
<td>Specifies a value of 0.</td>
</tr>
<tr>
<td>candidate-rp &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>candidate-rp</td>
<td>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.</td>
</tr>
<tr>
<td>local-address</td>
<td></td>
</tr>
<tr>
<td>ip_address</td>
<td></td>
</tr>
<tr>
<td>candidate-rp priority</td>
<td>Specifies the priority of the candidate rendezvous point included in the corresponding multicast group address. The higher the priority, the lower the value.</td>
</tr>
<tr>
<td>&lt;0-255&gt;</td>
<td>Specifies a value of 0.</td>
</tr>
<tr>
<td>static-rp off</td>
<td>Disables the static rendezvous point option.</td>
</tr>
<tr>
<td>static-rp rp-address</td>
<td>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).</td>
</tr>
<tr>
<td>ip_address &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>candidate-rp</td>
<td>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.</td>
</tr>
<tr>
<td>multicast-group</td>
<td></td>
</tr>
<tr>
<td>mcast_ip_prefix &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>register-suppress-interval</td>
<td>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.</td>
</tr>
<tr>
<td>&lt;60-3600&gt;</td>
<td>Specifies a value of 60.</td>
</tr>
<tr>
<td>candidate-rp advertise-interval</td>
<td>Specifies the interval between which candidate-rendezvous point routers send candidate-rendezvous point advertisements.</td>
</tr>
<tr>
<td>&lt;1-3600&gt;</td>
<td>Specifies a value of 60.</td>
</tr>
<tr>
<td>candidate-rp advertise-interval default</td>
<td></td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>cisco-compatibility &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>spt-threshold multicast mcast_ip_prefix threshold &lt;0-1000000&gt;</td>
<td>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.</td>
</tr>
<tr>
<td>spt-threshold multicast mcast_ip_prefix threshold infinity &lt;on</td>
<td>off&gt;</td>
</tr>
<tr>
<td>hello interval &lt;1-21845&gt;</td>
<td>Specifies the interval, in seconds, at which PIM hello messages are sent on the LAN.</td>
</tr>
<tr>
<td>hello interval default</td>
<td>Specifies a value of 30.</td>
</tr>
<tr>
<td>data-interval &lt;11-3600&gt;</td>
<td>Specifies the interval, in seconds, after which multicast (S,G) state for a silent source is deleted.</td>
</tr>
<tr>
<td>data-interval default</td>
<td>Specifies a value of 210.</td>
</tr>
<tr>
<td>assert-interval &lt;1-3600&gt;</td>
<td>Specifies the interval between the last time an assert is received and the assert is timed out.</td>
</tr>
<tr>
<td>assert-interval default</td>
<td>Specifies a value of 180.</td>
</tr>
<tr>
<td>assert-limit &lt;10-10000&gt;</td>
<td>Specifies the number of asserts to send per second.</td>
</tr>
<tr>
<td>assert-limit default</td>
<td>Specifies a value of 10.</td>
</tr>
<tr>
<td>assert-limit &lt;0&gt;</td>
<td>Disables the limit placed on the number of asserts that can be sent per second.</td>
</tr>
<tr>
<td>jp-interval &lt;1-3600&gt;</td>
<td>Specifies the interval, in seconds, between which join/prune messages are sent.</td>
</tr>
<tr>
<td>jp-interval default</td>
<td>Specifies a value of 60.</td>
</tr>
<tr>
<td>jp-delay-interval &lt;1-3600&gt;</td>
<td>Specifies maximum interval, in seconds, between the time when the RPF neighbor changes and a triggered Join/Prune message is sent.</td>
</tr>
<tr>
<td>jp-delay-interval default</td>
<td>Specifies a value of 5.</td>
</tr>
<tr>
<td>jp-suppress-interval &lt;2-3600&gt;</td>
<td>Specifies the mean interval between receiving a Join/Prune with a higher &quot;holdtime&quot; and allowing duplicate Join/Prunes to be sent again. Check Point recommends that you set the join/prune suppress interval 1.25 times that of the join/prune interval.</td>
</tr>
</tbody>
</table>
jp-suppress-interval
default

Specifies a value of 75.

assert-rank protocol
protocol name rank
<0-255>

Specifies the value assigned to a particular protocol in assert messages. This value is used to compare protocols to determine which router will forward multicast packets on a multi-access LAN. The value is included in assert messages when more than one router on a LAN is capable of forwarding multicast packets and one router detects the other routers’ duplicate packets. Use the following protocol names to set this option: ospf; kernel; igrp; rip; static; bgp; direct and ospfase. The values assigned to each protocol must match for each router on a multi-access LAN.

assert-rank protocol
protocol name rank
default

Specifies default assert-rank values for supported protocols that match other implementations. The direct default value is 0. The ospf default value is 10; the kernel default value is 40; the static route default value is 60; the IGRP default value is 80; the rip default value is 100; the bgp default value is 170.

nat-mode <on | off>

Translates the addresses in a PIM protocol message to the relevant address(es) for the interface on which the message is being sent.

state-refresh <on | off>

Directs dense mode to use state refresh messages to delay timing out prune state of multicast traffic that has no active receivers. This helps suppress the flood-and-prune cycle inherent to dense mode.

state-refresh-interval
<0 – 255>

Specifies the interval at which state refresh messages are sent for multicast traffic originated by directly-connected sources.

state-refresh-ttl <1 – 255>

Specifies the time-to-live (TTL) placed in the state refresh messages originated for multicast traffic from directly-connected sources. This value can be used to limit the forwarding of state refresh messages in the network.

---

**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:

```
show pim
   bootstrap
   candidate-rp
   joins
   rps
   parse-mode-stats
   group-rp-mapping <mcast_address>
```
# 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>
</tbody>
</table>
T
TCP MD5 Authentication • 15
Timer and Assert Rank Parameters for Dense
   Mode and Sparse Mode • 81
Types of Areas • 45

V
Virtual IP Address Support for VRRP • 40