



Check Point
SOFTWARE TECHNOLOGIES LTD.

17 July 2021

QUANTUM SPARK 1500, 1600, AND 1800 APPLIANCE SERIES

R80.20.30

Dynamic Routing CLI Guide

[Classification: Protected]



INFINITY VISION



QUANTUM



CLOUDGUARD



HARMONY

Check Point Copyright Notice

© 2021 Check Point Software Technologies Ltd.

All rights reserved. This product and related documentation are protected by copyright and distributed under licensing restricting their use, copying, distribution, and decompilation. No part of this product or related documentation may be reproduced in any form or by any means without prior written authorization of Check Point. While every precaution has been taken in the preparation of this book, Check Point assumes no responsibility for errors or omissions. This publication and features described herein are subject to change without notice.

RESTRICTED RIGHTS LEGEND:

Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c) (1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and FAR 52.227-19.

TRADEMARKS:

Refer to the [Copyright page](#) for a list of our trademarks.

Refer to the [Third Party copyright notices](#) for a list of relevant copyrights and third-party licenses.

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.



Certifications

For third party independent certification of Check Point products, see the [Check Point Certifications page](#).



Check Point R80.20.30

For more about this release, see the R80.20.30 [home page](#).



Latest Version of this Document in English

Open the latest version of this [document in a Web browser](#).

Download the latest version of this [document in PDF format](#).



Feedback

Check Point is engaged in a continuous effort to improve its documentation.

[Please help us by sending your comments.](#)

Revision History

Date	Description
17 July 2021	First release of this document

Table of Contents

Introduction to Advanced Routing	7
BGP (IPv4)	8
Support for BGP-4++	9
BGP Sessions (Internal and External)	10
Preventing Private AS Numbers from Propagating	10
BGP Route Refresh	11
BGP Path Attributes	12
BGP Multi-Exit Discriminator	14
BGP Interactions with IGP	15
Inbound BGP Route Filters	16
Redistributing Routes to BGP	17
BGP Communities	18
BGP Route Reflection	19
BGP Confederations	20
EBGP Multihop Support	21
BGP Route Dampening	22
Configuring BGP in Gaia Clish	23
Configuring Initial BGP Settings	24
Configuring Internal BGP	26
Configuring External BGP	31
Configuring BGP Peers	32
Configuring BGP Confederation	36
Configuring BGP Route Reflection	37
Configuring BGP Route Dampening	39
Configuring BGP Communities	41
Monitoring BGP	42
IGMP	43
Configuring IGMP in Gaia Clish	44
Configuring Interfaces for IGMP	45
Monitoring IGMP	47
RIP	48
RIP 1	49
RIP 2	50

Configuring RIP in Gaia Clish	51
Configuring RIP Global Settings	52
Configuring Interfaces for RIP	53
Monitoring RIP	55
OSPF v2 (IPv4)	56
Types of OSPF Areas	57
Area Border Routers	58
High Availability Support for OSPF	59
Configuring OSPF in Gaia Clish	60
Configuring Initial OSPF Settings	61
Configuring OSPF Global Settings	63
Configuring Interfaces for OSPF	64
Configuring OSPF Areas	68
Configuring OSPF Virtual Links	70
Monitoring OSPF	72
Route Aggregation	77
Configuring Route Aggregation in Gaia Clish	78
Route Maps	80
Configuring a Routemap in Gaia Clish	80
Configuring a Routemap	81
Configuring a Routemap	81
Configuring Actions for a Routemap	82
Configuring the Criteria that must be Matched for the Routemap to Take Effect	85
Viewing Routemaps	88
Routemap Protocol Commands	89
Supported Route Map Statements by Protocol	90
RIP	90
OSPFv2	91
Route Map Examples	92
Redistributing Static, Interface, or Aggregate Routes	94
PIM	95
Introduction	95
PIM Dense Mode (DM)	95
PIM Sparse Mode (SM)	95
PIM Source-Specific Multicast (SSM) Mode	96

Configuring PIM in Gaia Clish	97
Configuring PIM	98
Debugging PIM	105

Introduction to Advanced Routing

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

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

BGP (IPv4)

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 (IGP) and common metrics to route packets within an AS; it uses an exterior routing protocol to route packets to other ASs.



Note - This implementation supports BGP version 4, with Multiprotocol Extensions.

BGP sends update messages that consist of network number-AS path pairs. The AS path contains the string of ASs 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 IGP, which build their own reliability on top of a datagram service.

As a path-vector routing protocol, BGP limits the distribution of router reachability information to its peer or neighbor routers.

You can run BGP over a route-based VPN by enabling BGP on a virtual tunnel interface (VTI).

Support for BGP-4++

Gaia Embedded implements BGP-4++ to support multiprotocol extensions.

You must use an IPv4 address for the router ID (BGP identifier). After the BGP session is completed, you can advertise and withdraw prefixes by sending normal "UPDATE" messages that include one or both of the new multiprotocol attributes "MP_REACH_NLRI" (advertises reachability of routes) and "MP_UNREACH_NLRI" (withdraws routes).

The new attributes are backward compatible. If two routers have a BGP session and only one supports the multiprotocol attributes, they can still exchange unicast IPv4 routes.

On each peer you configure the type of routes (capability) that should be exchanged between peers. Choose this selection:

- IPv4 unicast (the default)

To establish peering, the routers must share a capability.



Note - Do not use the route redistribution and inbound filter pages of the Portal to configure routing policies for BGP-4++. Use the route map commands in the CLI.

BGP Sessions (Internal and External)

BGP supports these session types between neighbors:

- **Internal** (sometimes referred to as IBGP or iBGP) - Runs between routers in the same autonomous system.
- **External** (sometimes referred to as EBGP or eBGP) - Runs between routers in different autonomous systems.

When you send routes to an external peer, the local AS number is prepended to the AS path. Routes received from an internal neighbor have the same AS path that the route had when it was received from an external peer.

BGP sessions might include a single metric (Multi-Exit Discriminator or MED) in the path attributes. Smaller values 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 depends on the type of internal protocol processing.

For BGP implementation, external peers are directly attached to a shared subnet and advertise next hops that are host addresses on the subnet. If you enable the multihop option in the BGP peer template during configuration, this constraint is relaxed.

Internal groups determine the immediate next hops for routes. The next hop received with a route from a peer is used as a forwarding address and to look up an immediate next hop in IGP routes. Internal groups support distant peers, but need to know 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 needed to run 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 to conserve globally unique AS numbers. When an ISP does this, 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 is usually included. To avoid this, you can configure Gaia Embedded to remove the private AS number from BGP update messages to external peers.

To configure Gaia Embedded to remove private AS numbers from BGP updates, enable the Remove Private AS option on the configuration page for an external peer.

If you enable this option, private AS numbers are removed from BGP updates based on these rules:

- If the "AS_PATH" includes both public and private AS numbers, the private AS numbers are not removed.
- If the "AS_PATH" contains the AS number of the destination peer, private AS numbers are not removed.
- If the "AS_PATH" includes confederations and all the AS numbers in the "AS_PATH" are private, all the private AS numbers are removed.

BGP Route Refresh

Gaia Embedded supports the ability to dynamically request BGP route updates from peers and to respond to requests for BGP route updates.

For example, if you change the inbound routing policy, you can request that a peer readvertise its previously advertised routes so that the routes can be checked against the new policy.

This feature is often referred to as a soft reset because it provides the ability to refresh routes received from a peer without tearing down the established session.

To configure BGP route updates in the:

- Gaia Clish- 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
```

- Gaia Portal - Click the appropriate buttons in the **Edit Peer** page, in the section **Advanced Settings > Route Refresh**.

These options work only with peers that support the same capabilities. Gaia Embedded systems can also peer with systems that do not support these options.

BGP Path Attributes

A path attribute is a list of AS numbers that a route has traversed to reach a destination. BGP uses path attributes to provide more information about each route and to help prevent routing loops in an arbitrary topology. You can also use path attributes to determine administrative preferences.

BGP collapses routes with similar path attributes into a single update for advertisement. Routes that are received in a single update are readadvertised 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.

Path attributes:

Attribute	Description
AS_PATH	Identifies the autonomous systems through which routing information carried in an "UPDATE" message passed. Components of this list can be of type "AS_SET" or "AS_SEQUENCE".
NEXT_HOP	Defines the IP address of the border router that should be used as the next hop to the destinations listed in the "UPDATE" message.
MULTI_EXIT_DISC	Discriminates among multiple exit or entry points to the same neighboring autonomous system. Used only on external links.
LOCAL_PREF	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.
ATOMIC_AGGREGATE	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.

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 ASs 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 gateway (that support the IGP only) and border gateway (that support both the IGP and BGP). All interior gateway receive information about external routes from one or more of the border gateway 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 gateway 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, the `routerd` daemon does not.



Note - To define a redistribution policy, use routemaps (see ["Route Maps" on page 80](#)).

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 can 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.

This table shows some special community attributes that a BGP speaker can apply:

Community Attribute	Description
NO_EXPORT (0xFFFFFFFF01)	Not advertised outside a BGP confederation boundary. A standalone autonomous system that is not part of a confederation should be considered a confederation itself.
NO_ADVERTISE (0xFFFFFFFF02)	Not advertised to other BGP peers.
NO_EXPORT_SUBCONFED (0xFFFFFFFF03)	Not advertised to external BGP peers. This includes peers in other members' autonomous systems inside a BGP confederation.

For more about communities, see [RFC 1997](#) and [RFC 1998](#).

BGP 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.



Best Practice - Do not use multiple redundant reflectors, as they increase 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).

BGP 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).

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.

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, and so on) have set up route suppression.



Note - BGP route dampening is supported only for EBGp. It is not supported for IBGP.

Configuring BGP in Gaia Clish

- To see the available "set" commands for BGP, enter in Gaia Clish:

```
set bgp [Space] [Tab]
```

- To see the available "show" commands for BGP, enter in Gaia Clish:

```
show bgp [Space] [Tab]
```

Configuring Initial BGP Settings



During the initial configuration:

1. Configure the Router ID:

```
set router-id {default | <IP Address>}
```

You can use this command to change the current Router ID.


Parameters

Parameter	Description
default	Selects the highest interface address when BGP is enabled.
<IP Address>	<p>The Router ID uniquely identifies the router in the BGP autonomous system.</p> <p> Best Practice - We recommend you configure the Router ID explicitly.</p> <p>This prevents the Router ID from changing if the interface used for the Router ID goes down. Use an IP address on a loopback interface that is not the loopback address (127.0.0.1).</p> <p> Note - In a cluster, you must select a Router ID and make sure that it is the same on all cluster members.</p> <p>Range: Dotted-quad. 9[0-255].[0-255].[0-255]. Do not use 0.0.0.0 Default: The IP address of one of the local interfaces.</p>

2. Configure the Autonomous System number:

```
set as {<AS Number> | off}
```

Parameters

Parameter	Description
<AS Number>	<p>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.</p> <p> Caution - When you change the autonomous system number, all current peer sessions are reset and all BGP routes are deleted.</p> <p>Range: 1 - 4294967295 Default: none</p>

Parameter	Description
off	Disables the configured local autonomous system number.

Configuring Internal BGP


Syntax:

```
set bgp internal
  description "Text"
  graceful-restart-helper {off | on}
  graceful-restart-helper-stalepath-time seconds
  interface [{all | <Name of Interface>}] {off | on}
  local-address <IP Address> {off | on}
  med {<0-65535> | default}
  nexthop-self {off | on}
  outdelay {<0-65535> | off}
  {off | on}
  protocol [{all | <BGP Internal Protocol>}] {off | on}
  route-refresh {off | on}
```

```
set bgp internal peer <Peer IP Address>
  accept-routes {all | none}
  authtype none
  holdtime {<6-65535> | default}
  ignore-first-ashop {off | on}
  keepalive {<2-21845> | default}
  log-state-transitions {off | on}
  log-warnings {off | on}
  no-aggregator id {off | on}
  passive-tcp {off | on}
  peer_type <Peer Type> {off | on}
  send-keepalives {off | on}
  send-route-refresh {request | route-update} unicast
  throttle-count {<0-65535> | off}
  trace <BGP Trace Option> {off | on}
  weight {<0-65535> | off}
```

Parameters:

Parameter	Description
description "Text"	You can enter a brief text description of the group.
graceful-restart-helper {off on}	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.

Parameter	Description
<pre>interface [all <Name of Interface>] {off on}</pre>	<p>Specifies whether to enable the specified internal peer group on all interfaces or a specific interface.</p>
<pre>local-address <IP Address> {off on}</pre>	<p>The address used on the local end of the TCP connection with the peer.</p> <p>For external peers that do not have multihop enabled, 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.</p> <p>A session with an external peer is opened only when an interface with a local address through which the peer or gateway address is directly reachable is operating.</p> <p>For other types of peers, a peer session is maintained when any interface with the specified local address is operating.</p> <p>In either case, incoming connections are recognized as matching a configured peer only if they are addressed to the configured local address.</p> <p>Default: off</p> <p> Note - If running BGP in a cluster you must not configure the local address</p>
<pre>med {<0- 65535> default}</pre>	<p>Specifies the MED value.</p>
<pre>nexthop-self {off on}</pre>	<p>Specifies for this router to send one of its own IP addresses as the BGP next hop.</p> <p>Default: off</p>
<pre>{off on}</pre>	<p>Specifies whether to enable or disable an internal BGP group.</p>
<pre>outdelay {<0- 65535> off}</pre>	<p>Specifies the amount of time in seconds that a route must be present in the routing database before it is redistributed to BGP.</p> <p>The configured value applies to all peers configured in this group.</p> <p>This feature dampens route fluctuation.</p> <p>Zero (0) means that this feature is disabled.</p> <p>Default: 0</p>
<pre>peer <Peer IP Address> accept-routes all</pre>	<p>Specifies an inbound BGP policy route if one is not already configured.</p> <ul style="list-style-type: none"> ■ all Accept routes and installing them with an invalid preference. Depending on the local inbound route policy, these routes are then made active or inactive. ■ none Delete routes learned from a peer. This option saves memory overhead when many routes are rejected because no inbound policy exists.
<pre>peer <Peer IP Address> aggregator id {off on}</pre>	<p>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</p> <p>Default: off</p>

Parameter	Description
<pre>peer <Peer IP Address> authtype none</pre>	<p>Specifies not to use an authentication scheme between peers. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.</p>
<pre>peer <Peer IP Address> holdtime {<6-65535> default}</pre>	<p>Specifies the BGP holdtime interval, in seconds, when negotiating a connection with the specified peer.</p> <p>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.</p> <p>Range: 6-65535 seconds Default: 180 seconds</p>
<pre>peer <Peer IP Address> ignore-first-ashop {off on}</pre>	<p>Specifies to ignore the first autonomous system number in the autonomous system path for routes learned from the corresponding peer.</p> <p>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.</p>
<pre>peer <Peer IP Address> keepalive {<2-21845> default}</pre>	<p>The keepalive option is an alternative way to specify a holdtime value in seconds when negotiating a connection with the specified peer.</p> <p>You can use the keepalive interval instead of the holdtime interval.</p> <p>You can also use both interval, but the holdtime value must be 3 times the keepalive interval value.</p> <p>Range: 2-21845 seconds Default: 60 seconds</p>
<pre>peer <Peer IP Address> log-state-transitions {off on}</pre>	<p>Specifies for the router to log a message whenever a peer enters or leave the established state.</p>
<pre>peer <Peer IP Address> log-warnings {off on}</pre>	<p>Specifies for the router to log a message whenever a warning scenario is encountered in the codepath.</p>
<pre>peer <Peer IP Address> passive-tcp {off on}</pre>	<p>Specifies for the router to wait for the specified peer to issue an open message. No TCP connections are initiated by the router.</p> <p>Default:off</p>

Parameter	Description
<pre>peer <Peer IP Address> peer_type <Peer Type> {off on}</pre>	<p>Specifies an internal peer address and peer type.</p> <p>Peer types:</p> <ul style="list-style-type: none"> ■ reflector-client Specifies 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 readvertises, or reflects, routes it receives from one of its clients to all other IBGP peers, including the other peers in that client's group. ■ no-client-reflector Specifies 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. ■ none Specifies not to use route reflection.
<pre>peer <Peer IP Address> send- keepalives {off on}</pre>	<p>Specifies for this router always to send keepalive messages even when an update message is sufficient.</p> <p>This option allows interoperability with routers that do not strictly adhere to protocol specifications regarding update.</p>
<pre>peer <Peer IP Address> throttle- count {<0- 65535> off}</pre>	<p>Specifies the number of BGP updates to send at one time.</p> <p>The throttle count option limits the number of BGP updates when there are many BGP peers.</p> <p>The value "off" disables the throttle count option.</p>
<pre>peer <Peer IP Address> trace <BGP Trace Option> {off on}</pre>	<p>Specifies tracing options for your BGP implementation.</p> <p>Log messages are saved in the <code>var/log/routed/</code> directory.</p> <p>Enter the following words to set each trace option:</p> <ul style="list-style-type: none"> ■ all - to trace all the message types ■ general - to trace message related to Route and Normal ■ keepalive - to trace all the keepalive messages to this peer ■ normal - to trace normal protocol occurrences (abnormal protocol occurrences are always traced) ■ open - to trace all the BGP open messages to this peer ■ packets - to trace all BGP packets to this peer ■ policy - to trace application of the protocol and user-specified policy to routes being imported and exported. ■ route - to trace routing table changes for routes installed by this peer ■ state - to trace state machine transitions in the protocol ■ update - to trace all the BGP update messages to this peer

Parameter	Description
<pre><Peer IP Address> weight {<0- 65535> off}</pre>	<p>Specifies the weight associated with the specified peer.</p> <p>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.</p> <p>A negative weight prevents a route from becoming active, which prevents it from being installed in the forwarding table or exported to other protocols.</p> <p>This eliminates the need to break and reestablish a session upon reconfiguration if import route policy is changed.</p> <p>The value "off" disables the weight associated with the specified peer.</p>
<pre>protocol {all <BGP Internal Protocol>} {off on}</pre>	<p>Specifies whether to enable all internal routing protocols on the specified internal peer group or specific internal protocols.</p> <p>You can enter one of these internal protocols:</p> <ul style="list-style-type: none"> ■ direct ■ ospf ■ ospfase ■ rip ■ static
<pre>route-refresh {off on}</pre>	<p>Re-learns routes previously sent by the BGP peer or refreshes the routing table of the peer.</p> <p>The peer responds to the message with the current routing table.</p> <p>Similarly, if a peer sends a route refresh request the current routing table is re-sent.</p> <p>A user can also trigger a route update without having to wait for a route refresh request from the peer.</p>
<pre>send-route- refresh {request route-update} unicast</pre>	<p>Specifies that the router dynamically request BGP route updates from peers or respond to requests for BGP route updates.</p>

Configuring External BGP

Syntax:

```
set bgp external remote-as <AS Number>
    {off | on}
    aspath-prepend-count <1-25 | default>
    description text
    local-address <IP Address> {off | on}
    outdelay <0-65535>
    outdelay off
```

Parameters:

Parameter	Description
<AS Number> {off on}	Specifies the autonomous system number of the external peer group. Enter an integer between 1 and 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. Range: 1-25 Default: 1
description text	You can enter a brief text description of the group.
local-address <IP Address> {off on}	Specifies the address used on the local end of the TCP connection with the peer group. The local address must be on an interface that is shared with the peer or with the peer's gateway when the gateway parameter is used.
outdelay {<0-65535> off}	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. Range: 0-65535 Default: 0

Configuring BGP Peers

Gaia Embedded supports IPv4 addresses for BGP peers.

Syntax:

```
set bgp external remote-as <AS Number> peer <IP Address>
    accept-med {off | on}
    accept-routes {all | none}
    authtype none
    capability {default | ipv4-unicast}
    graceful-restart-helper {off | on}
    graceful-restart-helper-stalepath-time <Seconds>
    holdtime <6-65535 | default>
    ignore-first-ashop {off | on}
    keepalive <2-21845 | default>
    log-state-transitions {off | on}
    log-warnings {off | on}
    med-out <0-4294967294 | default>
    multihop {off | on}
    no-aggregator-id {off | on}
    {off | on}
    passive-tcp {off | on}
    removeprivateas {off | on}
    route-refresh {off | on}
    send-keepalives {off | on}
    send-route-refresh {request | route-update} {ipv4 | ipv6 | All}
unicast
    suppress-default-originate {off | on}
    throttle-count <0-65535 | off>
    trace <BGP Trace Option> {off | on}
    ttl <1-255 | default>
```

Parameters:

Parameter	Description
accept-med {off on}	Specifies that MED be accepted from the specified peer address. If you do not enable this option, the MED is stripped from the advertisement before the update is added to the routing table.
accept-routes <all none>	Specifies an inbound BGP policy route if one is not already configured. <ul style="list-style-type: none"> ■ all Accept routes and installing them with an invalid preference. Depending on the local inbound route policy, these routes are then made active or inactive. ■ none Delete routes learned from a peer. This option saves memory overhead when many routes are rejected because no inbound policy exists.

Parameter	Description
<code>authtype none</code>	Specifies not to use an authentication scheme between peers. Using an authentication scheme guarantees that routing information is accepted only from trusted peers.
<code>capability {default ipv4-unicast}</code>	Specifies capabilities setting. Only IPv4 unicast is supported.
<code>graceful-restart-helper-stalepath-time <Seconds></code>	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.
<code>graceful-restart-helper {off on}</code>	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.
<code>holdtime <6-65535 default></code>	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. Range: 6-65535 seconds Default: 180 seconds
<code>ignore-first-ashop {off on}</code>	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.
<code>keepalive <2-21945 default></code>	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. Range: 2-21945 seconds Default: 60 seconds
<code>log-state-transitions {off on}</code>	Specifies for the router to log a message whenever a peer enters or leave the established state.
<code>log-warnings {off on}</code>	Specifies for the router to log a message whenever a warning scenario is encountered in the codepath.

Parameter	Description
<code>med-out <0-4294967294 default></code>	<p>Specifies the multi-exit discriminator (MED) metric used as the primary metric on all routes sent to the specified peer address.</p> <p>This metric overrides the default metric on any metric specified by the redistribute policy.</p> <p>External peers uses MED values to decide which of the available entry points into an autonomous system is preferred.</p> <p>A lower MED value is preferred over a higher MED value.</p> <p>Range: 0-4294967294</p> <p>Default: 4294967294</p>
<code>multihop {off on}</code>	<p>Enables multihop connections with external BGP peers more than one hop away.</p> <p>By default, external BGP peers are expected to be directly connected. This option can also be used for external load-balancing.</p>
<code>no-aggregator-id {off on}</code>	<p>Specifies the router's aggregate attribute as zero (rather than the router ID value).</p> <p>This option prevents different routers in an AS from creating aggregate routes with different AS paths.</p>
<code>{off on}</code>	Disables (off) or enables (on) the configuration of the specified peer.
<code>passive-tcp {off on}</code>	<p>Specifies for the router to wait for the specified peer to issue an open message.</p> <p>No TCP connections are initiated by the router.</p>
<code>peer <IP Address></code>	Specifies the peer for the group.
<code>remote-as <AS Number></code>	Specifies the Autonomous System.
<code>removeprivateas {off on}</code>	Specifies that private AS numbers be removed from BGP update messages to external peers.
<code>route-refresh {off on}</code>	<p>Re-learns routes previously sent by the BGP peer or refreshes the routing table of the peer.</p> <p>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.</p> <p>A user can also trigger a route update without having to wait for a route refresh request from the peer.</p>
<code>send-keepalives {off on}</code>	<p>Specifies for this router always to send keepalive messages even when an update message is sufficient.</p> <p>This option allows interoperability with routers that do not strictly adhere to protocol specifications regarding updates.</p>
<code>send-route-refresh {request route-update} {ipv4 ipv6 All} unicast</code>	Specifies that the router dynamically request BGP route updates from peers or respond to requests for BGP route updates.

Parameter	Description
<code>suppress-default-originate {off on}</code>	Specifies NOT to generate a default route when the peer receives a valid update from its peer.
<code>throttle-count <0-65535 off></code>	Specifies number of BGP updates to send at one time. This option limits the number of BGP updates when there are many BGP peers. The value "off" disables the throttle count option.
<code>trace <BGP Trace Option> {off on}</code>	Specifies tracing options for your BGP implementation. Log messages are saved in the <code>var/log/routed/</code> directory. Enter the following words to set each trace option: <ul style="list-style-type: none"> ■ <code>all</code> - to trace all the message types ■ <code>general</code> - to trace message related to Route and Normal ■ <code>keepalive</code> - to trace all the keepalive messages to this peer ■ <code>normal</code> - to trace normal protocol occurrences (abnormal protocol occurrences are always traced) ■ <code>open</code> - to trace all the BGP open messages to this peer ■ <code>packets</code> - to trace all BGP packets to this peer ■ <code>policy</code> - to trace application of the protocol and user-specified policy to routes being imported and exported. ■ <code>route</code> - to trace routing table changes for routes installed by this peer ■ <code>state</code> - to trace state machine transitions in the protocol ■ <code>update</code> - to trace all the BGP update messages to this peer
<code>ttl <1-255 default></code>	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 you enabled the "multihop" option. Range: 1-255 Default: 64

Configuring BGP Confederation

Syntax:

```
set bgp
    confederation identifier {<AS Number> | off}
    confederation aspath-loops-permitted {<1-10> | default}
    routing-domain identifier {<AS Number> | off}
    routing-domain aspath-loops-permitted {<1-10> | default}
    synchronization {off | on}
```

Parameters:

Parameter	Description
confederation identifier {<AS Number> off}	<p>Configures or disables the identifier for the entire confederation. This identifier is used as the autonomous system number in external BGP sessions.</p> <p>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.</p>
confederation aspath-loops-permitted {<1-10> default}	<p>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.</p> <p>Range: 1-10 Default: 1</p>
routing-domain identifier {<AS Number> off}	<p>Configures or disables the routing domain identifier (RDI) for this router.</p> <p>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.</p>
routing-domain aspath-loops-permitted {<1-10> default}	<p>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.</p> <p>Range: 1-10 Default: 1</p>
synchronization {off on}	<p>Disables (off) or enables (on) IGP synchronization.</p> <p>Enable this option to force internal and confederation BGP peers to check for a matching route from IGP protocol before installing a BGP learned route.</p>

Configuring BGP Route Reflection

You can configure route reflection as an alternative to BGP confederations.

Route reflection supports both internal and external BGP routing groups.

Syntax:

```
set bgp
  internal peer <IP Address>
    peer-type {no-client-reflector | none | reflector-client}
  cluster-id {<IP Address> | off}
  default-med {<0-65535> | off}
  default-route-gateway {<IP Address> | off}
```

Parameters:

Parameter	Description
<code>internal peer <IP Address></code>	Specifies the peer router.
<code>peer-type {no-client-reflector none reflector-client}</code>	<p>Specifies if this is a Route Reflector client.</p> <ul style="list-style-type: none"> ■ <code>no-client-reflector</code> Peer is a 'non-client' Route Reflector. Enter this option 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. ■ <code>none</code> Peer is not a Route Reflector. Enter this option, if you do not want to specify route reflection. ■ <code>reflector-client</code> Peer is a Route Reflector. Enter this option 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 readvertises, or reflects, routes it receives from one of its clients to all other iBGP peers, including the other peers in that client's group.
<code>cluster-id {<IP Address> off}</code>	<p>Configures or disables the cluster ID used for route reflection.</p> <p>The cluster ID default is that of the Router ID.</p> <p>Override the default if the cluster has more than one route reflector.</p>
<code>default-med {<0-65535> off}</code>	Configures or disables the Multi-Exit Discriminator (MED) metric used to advertise routes through BGP.

Parameter	Description
<code>default-route-gateway</code> <code>{<IP Address> off}</code>	<p>Configures or disables the default BGP route.</p> <p>This route has a higher rank than any configured default static route for this router.</p> <p>If you do not want a BGP peer considered for generating the default route, use this command:</p> <pre>peer <IP Address> suppress-default-originate on</pre>

Configuring BGP Route Dampening

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.

Syntax:

```
set bgp dampening
    {off | on}
    suppress-above {<2-32> | default}
    reuse-below {<1-32> | default}
    max-flat {<3-64> | default}
    reachable-decay {<1-900> | default}
    unreachable-decay {<1-2700> | default}
    keep-history {<2-5400> | default}
```



Note - BGP route dampening is only supported for External BGP (EBGP).

Parameters:

Parameter	Description
{off on}	Specifies whether to enable or disable BGP route dampening.
suppress-above <2-32>	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 is reachable during the period that it is suppressed.
suppress-above default	Specifies an instability metric value for suppressing routes of 3.
reuse-below metric <1-32>	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.
reuse-below metric default	Specifies an instability metric value for announcing previously suppressed routes of 2.
max-flap <3- 64>	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.
max-flat default	Specifies the upper limit of the instability metric as 16.

Parameter	Description
<code>reachable-decay <1-900></code>	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.
<code>reachable-decay default</code>	Specifies a value of 300.
<code>unreachable-decay <1-2700></code>	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.
<code>unreachable-decay default</code>	Specifies a value of 900
<code>keep-history <2-5400></code>	Specifies the period for which route flapping history is maintained for a given route.
<code>keep-history default</code>	Specifies a value of 1800.

Configuring BGP Communities

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 {off | on}
```

Monitoring BGP

Syntax:

```
show bgp
  groups
  memory
  errors
  paths
  stats
  peer <IP Address> {advertise | detailed | received}
  peers {advertise | detailed | established | received}
  summary
```

IGMP

Internet Group Management Protocol (IGMP) allows hosts on multi-access 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 IGMPv1 host membership query message to specify a maximum response time. The leave group message allows a host to report when its membership in a multicast group terminates. Then, the IGMP querier router can send a group-directed query with a very small maximum response time to probe for any remaining active group members. This accelerated leave extension can reduce the time required to expire a group and prune the multicast distribution tree from minutes, down to several seconds.

The unicast traceroute program allows the tracing of a path from one device to another, using mechanisms that already exist in IP. Unfortunately, you cannot apply such mechanisms to IP multicast packets. The key mechanism for unicast traceroute is the ICMP TTL exceeded message that is specifically precluded as a response to multicast packets. The traceroute facility implemented within routed conforms to the traceroute facility for IP multicast draft specification.

Gaia Embedded supports IGMP version 1, v2, and v3. Version 2 runs by default.

Gaia Embedded supports these IGMP features:

- Multicast traceroute
- Ability to configure protocol timers
- Support for interfaces with secondary addresses

You can configure the these 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 in Gaia Clish

Use the IGMP commands to configure parameters for the internet group management protocol.

- To see the available "set" commands for IGMP, enter in Gaia Clish:

```
set igmp[Space][Tab]
```

- To see the available "show" commands for IGMP, enter in Gaia Clish:

```
show igmp[Space][Tab]
```

Configuring Interfaces for IGMP

Syntax:

```
set igmp interface <Name of Interface>
    last-member-query-interval {<1-25> | default}
    local-group address {off | on}
    loss-robustness {<1-255> | default}
    query-interval {<1-3600> | default}
    query-response-interval {<1-25> | default}
    router-alert {off | on}
    static-group address {off | on}
    version <1 | 2 | 3>
```



Note - IGMP version 2 runs by default.



Important - In a cluster, run commands on each cluster member. The configuration of each cluster member must be identical

Parameters:

Parameter	Description
interface <Name of Interface>	Specifies the interface, on which IGMP should be configured.
last-member-query-interval {<1-25> default}	<p>The maximal response time (in seconds) inserted into IGMP group-specific queries.</p> <p>You can use the last member query interval to tune the "leave latency." A smaller value results in a reduction in the time to detect the loss of the last member of a multicast group.</p> <p>This value must always be less than the query interval.</p> <p>Range: 1-25</p> <p>Default: 1</p>
local-group address {off on}	<p>A multicast group address. A local group provides a mechanism to simulate the presence of local receivers for specific groups.</p> <p>When a multicast group is added to an interface, IGMP sends a membership report on the interface.</p> <p>Range: off, on</p> <p>Default: off</p>
loss-robustness {<1-255> default}	<p>Lets you tune the expected packet loss on a subnet.</p> <p>If you expect the subnet to be highly lossy, then you can increase the "loss robustness" value.</p> <p>IGMP protocol operation is robust to (lossrobustness - 1) packet loss.</p> <p>Range: 1-255</p> <p>Default: 2</p>

Parameter	Description
<code>query-interval {<1-3600> default}</code>	<p>The interval (in seconds) between IGMP general queries which the querier router sends.</p> <p>You can use this parameter to tune the IGMP messaging overhead and has a secondary effect on the timeout of idle IP multicast groups.</p> <p>Range: 1-3600 Default: 125</p>
<code>query-response-interval {<1-25> default}</code>	<p>The maximal response time (in seconds) inserted into the periodic IGMP general queries.</p> <p>You can use the query response interval to tune the burstiness of IGMP messages.</p> <p>A larger value spreads the host IGMP reports over a larger interval, which reduces burstiness.</p> <p>This value must always be less than the value of the "query-interval".</p> <p>Range: 1-25 Default: 10.</p>
<code>router-alert {off on}</code>	<p>Lets you disable the insertion of IP router alert in all IGMP messages sent on the interface.</p> <p>This can be useful with broken IP implementations that may discard the packet because of the use of this option.</p> <p>Range: off, on Default: off</p>
<code>static-group address {off on}</code>	<p>A multicast group address. A static group provides a mechanism to simulate the presence of local receivers on an interface.</p> <p>When a static group is configured on an interface that also runs a parent multicast protocol (such as PIM), IGMP informs the parent of the presence of a local receiver.</p> <p>In contrast to regular IGMP, no membership reports are sent on the corresponding interface.</p> <p>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.</p> <p>Range: off, on Default: off</p>
<code>version <1 2 3></code>	<p>IGMP version 2 is compatible with IGMP version 1. IGMP version 3 is compatible with IGMP versions 2 and 1.</p> <div>  <p>Best Practice - Use IGMP version 1 only on networks that include multicast routers that are not upgraded to IGMP versions 2 or 3.</p> </div>

Monitoring IGMP

Syntax:

```
show igmp
  group <Name of Logical Interface>
  groups [interface <Name of Logical Interface>] [{local | static}]
  if-stat <Name of Interface>
  if-stats
  interface <Name of Interface>
  interfaces
  stats
  stats receive
  stats transmit
  stats error
  summary
```

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 IGP, 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 maximal number of hops in a RIP network is 15 and the protocol treats anything equal to or greater than 16 as unreachable.

RIP 1

Background

The original specification of RIP, defined in [RFC 1058](#), was published in 1988.

When starting up, and every 30 seconds thereafter, a router with RIPv1 implementation broadcasts to 255.255.255.255 a request message through every RIPv1 enabled interface. Neighbouring routers receiving the request message respond with a RIPv1 segment, containing their routing table. The requesting router updates its own routing table, with the reachable IP network address, hop count and next hop, that is the router interface IP address from which the RIPv1 response was sent. As the requesting router receives updates from different neighbouring routers it will only update the reachable networks in its routing table, if it receives information about a reachable network it has not yet in its routing table or information that a network it has in its routing table is reachable with a lower hop count. Therefore, a RIPv1 router will in most cases only have one entry for a reachable network, the one with the lowest hop count. If a router receives information from two different neighbouring router that the same network is reachable with the same hop count but via two different routes, the network will be entered into the routing table two times with different next hop routers. The RIPv1 enabled router will then perform what is known as equal-cost load balancing for IP packets.

RIPv1 enabled routers not only request the routing tables of other routers every 30 seconds, they also listen to incoming requests from neighbouring routers and send their own routing table in turn. RIPv1 routing tables are therefore updated every 25 to 35 seconds. The RIPv1 protocol adds a small random time variable to the update time, to avoid routing tables synchronizing across a LAN. It was thought, as a result of random initialization, the routing updates would spread out in time, but this was not true in practice.

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.

RIP 2

Background

Due to the deficiencies of the original RIP specification, RIP version 2 (RIPv2) was developed in 1993, published as [RFC 1723](#) in 1994, and declared Internet Standard 56 in 1998.

It included the ability to carry subnet information, thus supporting Classless Inter-Domain Routing (CIDR). To maintain backward compatibility, the hop count limit of 15 remained. RIPv2 has facilities to fully interoperate with the earlier specification if all Must Be Zero protocol fields in the RIPv1 messages are properly specified. In addition, a compatibility switch feature allows fine-grained interoperability adjustments.

In an effort to avoid unnecessary load on hosts that do not participate in routing, RIPv2 multicasts the entire routing table to all adjacent routers at the address 224.0.0.9, as opposed to RIPv1 which uses broadcast. Unicast addressing is still allowed for special applications.

Route tags were also added in RIP version 2. This functionality allows a distinction between routes learned from the RIP protocol and routes learned from other protocols.

Network Mask

The RIPv1 protocol assumes that all sub-networks 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.

RIPv2 adds the ability to specify explicitly the network mask for each network in a packet.

Authentication

RIPv2 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.

Configuring RIP in Gaia Clish

Use these commands to configure and view RIP properties for specific interfaces.

- To see the available "set" commands for RIP, enter in Gaia Clish:

```
set rip[Space][Tab]
```

- To see the available "show" commands for RIP, enter in Gaia Clish:

```
show rip[Space][Tab]
```



Note - Gaia Embedded does not have CLI commands for route filtering and route redistribution. You must configure inbound routing policies and redistribution of routes through the Portal. 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 Portal. 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 Portal to configure route redistribution into RIP.





Best Practice - Do **not** configure RIP 1 and RIP 2 together.

Configuring RIP Global Settings

Syntax:

```
set rip
  auto-summary {off | on}
  expire-interval {<1-65535> | default}
  update-interval {<1-65535> | default}
```

Parameters:

Parameter	Description
auto-summary {off on}	<p>Automatically aggregates and redistributes non-classful RIP Version 1 into RIP. This applies only to RIP Version 1. If the Auto summarization field option is unchecked, you must do the aggregation and redistribution manually by using route aggregation and route redistribution.</p> <p> Note - Take care when you set this parameter, as RIP has no protocol mechanism to detect misconfiguration.</p> <p>Default: on</p>
expire-interval {<1-65535> default}	<p>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.</p> <p>Range: 1-65535 seconds Default: 180 seconds</p>
update-interval {<1-65535> default}	<p>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.</p> <p> Note - Take care when you set this parameter, as RIP has no protocol mechanism to detect misconfiguration.</p> <p>Range: 1-65535 seconds Default: 30 seconds</p>


Configuring Interfaces for RIP

Syntax:

```
set rip interface <Name of Interface>
  accept-updates {off | on}
  authtype
    md5 secret <Secret> [cisco-compatibility {off | on}]
    none
    simple <Password>
  metric {<0-16> | default}
  {off | on}
  send-updates {off | on}
  transport {multicast | broadcast}
  version <1 | 2> on
```

Parameters:

Parameter	Description
accept-updates {off on}	Whether RIP packets from other routers using the interface are accepted or ignored. Ignoring an update may result in suboptimal routing. Default: off
authtype md5 secret <Secret>	Implement an authentication scheme that uses an MD5 algorithm for the interface to accept routing information from neighboring routers. Enter the password.
cisco-compatibility {off on}	Disables (off) or enables (on) interoperability with Cisco routers running RIP MD5 authentication. By default, RIP MD5 is set to conform to the Check Point standard, and not for Cisco compatibility. Default: off
authtype none	There is no authentication scheme for the interface to accept routing information from neighboring routers. This option applies to RIP version 2 only. In general, routers on a given link must agree on the authentication configuration in order to form neighbor adjacencies. This is used to guarantee that routing information is accepted only from trusted routers.
authtype simple <Password>	Implement a simple authentication scheme for the interface to accept routing information from neighboring routers. Enter the password string: from 1 to 16 characters, alphanumeric characters only. This option applies to RIP version 2 only.
interface <Name of Interface>	Specifies the interface, in which to configure RIP.

Parameter	Description
<code>metric {<0-16> default}</code>	<p>Specifies the RIP metric to be added to routes that are sent using the specified interface(s). This is used to make other routers prefer other sources of RIP routes over this router. Range: 0-16 Default: 0</p>
<code>{off on}</code>	<p>Disables (<code>off</code>) or enables (<code>on</code>) RIP on the specified interface. Default: off</p>
<code>send-updates {off on}</code>	<p>Whether RIP packets should be sent via the interface. This causes the interface to be a passive RIP listener.</p>
<code>transport {multicast broadcast}</code>	<p>The transport mechanism. Selecting Multicast specifies that RIP version 2 packets should be multicast on this interface. Default: multicast</p> <div>  <p>Note - When you use RIP 2, always configure "multicast".</p> </div>
<code>version <1 2> on</code>	<p>Specifies the version of RIP to run. If you specify version 2, the default is to send full version 2 packets on the RIP multicast address. Default: 1</p>

Monitoring RIP

Syntax:

```
show rip
  errors
  interfaces
  interface <Name of Interface>
  neighbors
  packets
  summary
```

OSPF v2 (IPv4)

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.

You can run OSPF over a route-based VPN by enabling OSPF on a virtual tunnel interface (VTI).

Gaia Embedded supports OSPFv2, which supports IPv4 addressing.

Types of OSPF 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 these types:

Area Type	Description
Normal Area	Allows all LSAs to pass through. The backbone is always a normal area.
Stub Area	Does 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.

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.

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.



Best Practice - 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).



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.

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 these types of routes:

Route Type	Description
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. Incorrect configuration prevents adjacencies from forming between neighbors, and routing black holes or loops can form.

High Availability Support for OSPF

Gaia Embedded supports the OSPF protocol in a cluster of appliances.

In this configuration, the cluster becomes a Virtual Router, which is seen by neighboring routers as a single router that has an IP address that is the same as the virtual IP address of the cluster. Each member of the cluster runs the OSPF task, but only the member which is designated as primary or master actively participates in the network and exchanges routing information with neighbor routers. When a failover occurs, the standby member of the cluster becomes the master and its OSPF task becomes the active participant in protocol exchanges with neighbor routers.

Gaia Embedded also supports the OSPF protocol over VPN tunnels which terminate in the ClusterXL cluster.

ClusterXL

Gaia Embedded ClusterXL advertises the Virtual IP address of the ClusterXL Virtual Router. The OSPF routes database of the master is synchronized across all members of the cluster. The OSPF task of each standby member obtains routing state and information from the master and installs the routes in the kernel as the master does. On a failover, one of the standby members becomes the new master and then continues where the old master failed. During the time that the new master resynchronizes routes database with the neighbor routers, traffic forwarding continues using the old kernel routes until OSPF routes are fully synchronized and pushed into the kernel.

Configuring OSPF in Gaia Clish

- To see the available "set" commands for OSPF, enter in Gaia Clish:

```
set ospf [Space] [Tab]
```

- To see the available "show" commands for OSPF, enter in Gaia Clish:

```
show ospf [Space] [Tab]
```



Note - Gaia Embedded does not have CLI commands for route filtering and redistribution. You must configure inbound routing policies and redistribution of routes through the Portal. 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 Portal. For example if OSPF uses route maps for inbound filtering, anything configured on the Portal page for inbound route filters for OSPF is ignored. You can still use the Portal to configure route redistribution into OSPF.

Configuring Initial OSPF Settings


During the initial configuration:

1. Configure the Router ID:

```
set router-id {default | <IP Address>}
```

You can use this command to change the current Router ID.


Parameters

Parameter	Description
default	Selects the highest interface address when OSPF is enabled.
<IP Address>	<p>The Router ID uniquely identifies the router in the OSPF autonomous system.</p> <p> Best Practice - We recommend you configure the Router ID explicitly.</p> <p>This prevents the Router ID from changing if the interface used for the Router ID goes down. Use an IP address on a loopback interface that is not the loopback address (127.0.0.1).</p> <p>Notes:</p> <ul style="list-style-type: none"> ■ Do not use 0.0.0.0 as the Router ID. ■ In a cluster, you must select a router ID and make sure that it is the same on all cluster members. <p>Range: Dotted-quad. 9[0-255].[0-255].[0-255]. Do not use 0.0.0.0</p> <p>Default: The IP address of one of the local interfaces.</p>

2. Configure the OSPF Area:

```
set ospf area <OSPF Area ID>
```

Parameters

Parameter	Description
<OSPF Area ID>	<p>Specifies an OSPF Area ID.</p> <p> Best Practice - Enter the area ID as a dotted quad. The area ID 0.0.0.0 is reserved for the backbone.</p> <p>Range:</p> <ul style="list-style-type: none">■ <code>backbone</code> By default, the backbone area is enabled. You can disable the backbone area if the system does not have interfaces on the backbone area.■ integer between 1 and 4294967295■ dotted quad form (example: 0.0.0.1 for area id 1) <p>Default: none</p>

Configuring OSPF Global Settings

Syntax:

```
set ospf
    default-ase-cost <1-677215>
    default-ase-type {1 | 2}
    graceful-restart-helper {off | on}
    rfc1583-compatibility {off | on}
    spf-delay {<1-60> | default}
    spf-holdtime {<1-60> | default}
```

Parameters:

Parameter	Description
default-ase-cost <1-677215>	Specifies the cost assigned to routes from other protocols that are redistributed into OSPF as autonomous systems external. If the route has a cost already specified, that cost takes precedent. 1
default-ase-type {1 2}	Specifies the type assigned to routes from other protocols that are redistributed into OSPF as autonomous systems external. If the route has a type already specified, that type takes precedent. 1
graceful-restart-helper {off on}	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.
rfc1583-compatibility {off on}	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.
spf-delay {<1-60> default}	Specifies the time, in seconds, to wait before recalculating the OSPF routing table after a change in the topology. Range: 1-60 seconds Default: 2 seconds
spf-holdtime {<1-60> default}	Specifies the minimum time, in seconds, between recalculations of the OSPF routing table. Range: 1-60 seconds Default: 5 seconds

Configuring Interfaces for OSPF

Syntax:

```

set ospf
  area {backbone | <OSPF Area ID>}
    range <IPv4 Address>/<Subnet Mask>
      {off | on}
    restrict {off | on}
  stub-network <IPv4 Address>/<Subnet Mask>
    {off | on}
  stub-network-cost <1-677722>

set ospf interface <Name of Interface>
  authtype
    md5 key authorization key <ID> [secret md5 <Secret>]
    none
    simple <Password>
  area {backbone | <OSPF Area ID>} {off | on}
  cost <1-65535>
  dead-interval {<1-65535> | default}
  hello-interval {<1-65535> | default}
  passive {off | on}
  priority <0-255>
  retransmit-interval {<1-65535> | default}

```

Parameters:

Parameter	Description
<pre> area {backbone <OSPF Area ID>} range <IPv4 Address>/<Subnet Mask> {off on} </pre>	<p>Specifies the OSPF area to which the specified interface range belongs.</p> <p>Select an area from the areas already configured.</p> <p>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.</p> <p>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.</p> <p>Be careful when configuring an address range that covers part of a prefix that is not contained within an area.</p> <p>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.</p>

Parameter	Description
<pre>area {backbone <OSPF Area ID>} range <IPv4 Address>/<Subnet Mask> restrict {off on}</pre>	<p>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.</p> <p>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.</p> <p>Be careful when configuring an address range that covers part of a prefix that is not contained within an area.</p> <p>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.</p>
<pre>stub-network <IPv4 Address>/<Subnet Mask> {off on}</pre>	<p>Specifies a stub network to which the specified interface range belongs.</p> <p>Configure a stub network to advertise reachability to prefixes that are not running OSPF.</p> <p>The advertised prefix appears as an OSPF internal route and is filtered at area borders with the OSPF area ranges.</p> <p>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.</p> <p>Use a mask length of 32 to configure the stub host.</p> <p>The local address of a point-to-point interface can activate the advertised prefix and mask.</p> <p>To advertise reachability to such an address, enter an IP address for the prefix and a non-zero cost for the prefix.</p>
<pre>stub-network <IPv4 Address>/<Subnet Mask> stub-network-cost <1- 677722></pre>	<p>Configure a stub network to advertise reachability to prefixes that are not running OSPF.</p> <p>The advertised prefix appears as an OSPF internal route and is filtered at area borders with the OSPF area ranges.</p> <p>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.</p> <p>Use a mask length of 32 to configure the stub host.</p> <p>The local address of a point-to-point interface can activate the advertised prefix and mask.</p> <p>To advertise reachability to such an address, enter an IP address for the prefix and a non-zero cost for the prefix.</p>
<pre>interface <Name of Interface> area {backbone <OSPF Area ID>} {off on}</pre>	<p>Specifies the OSPF area to which the specified interface belongs.</p>

Parameter	Description
<pre> authtype md5 key authorization key <ID> secret md5 <Secret> </pre>	<p>Specifies to use MD5 authorization.</p> <p>Enter at least one key ID and its corresponding MD5 secret.</p> <p>If you configure multiple key IDs, the largest key ID is used for authenticating outgoing packets.</p> <p>All keys can be used to authenticate incoming packets.</p> <p>Generally, routers on a given link must agree on the authentication configuration to form peer adjacencies.</p> <p>Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.</p>
<pre> authtype none </pre>	<p>Specifies not to use an authentication scheme for the specified interface.</p>
<pre> authtype simple <Password> </pre>	<p>Specifies to use simple authentication for the specified interface.</p> <p>Enter an ASCII string that is 8 characters long.</p> <p>Generally, routers on a given link must agree on the authentication configuration to form peer adjacencies.</p> <p>Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.</p>
<pre> cost <1-65535> </pre>	<p>Specifies the weight of the given path in a route.</p> <p>The higher the cost, the less preferred the link.</p> <p>To use one interface over another for routing paths, assign one a higher cost.</p>
<pre> dead-interval {<1-65535> default} </pre>	<p>Specifies the number of seconds after which a router stops receiving hello packets that it declares the peer down.</p> <p>Generally, you should set this value at 4 times the value of the hello interval.</p> <p>Do not set the value at 0.</p> <p>For a given link, this value must be the same on all routers or adjacencies do not form.</p> <p>Default: 40 seconds</p>
<pre> hello-interval {<1-65535> default} </pre>	<p>Specifies the interval, in seconds, between hello packets that the router sends on the specified interface.</p> <p>For a given link, this value must be the same on all routers or adjacencies do not form.</p> <p>Default: 10 seconds</p>
<pre> passive {off on} </pre>	<p>Enabling this option puts the specified interface into passive mode; that is, hello packets are not sent from the interface.</p> <p>Putting an interface into passive mode means that no adjacencies are formed on the link.</p> <p>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.</p> <p>Default: off</p>

Parameter	Description
<code>priority <0-255></code>	<p>Specifies the priority for becoming the designated router (DR) on the specified link.</p> <p>When two routers attached to a network attempt to become a designated router, the one with the highest priority wins.</p> <p>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.</p> <p>A router with a priority of 0 is not eligible to become the DR.</p>
<code>retransmit-interval {<1-65535> default}</code>	<p>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.</p> <p>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.</p> <p>Default: 5 seconds</p>

Configuring 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 {off | on}

set ospf area <OSPF Area>
  {off | on}
  stub {off | on}
  stub default-cost <1-677215>
  stub summary {off | on}
  nssa {off | on}
  nssa default-cost <1-677215>
  nssa default-metric-type <1-2>
  nssa import-summary-routes {off | on}
  nssa translator-role {always | candidate}
  nssa translator-stability-interval <1-65535>
  nssa redistribution {off | on}
  nssa range <Range of IP Addresses> [restrict] {off | on}
```

Parameters:

Parameter	Description
backbone {off on}	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.
{off on}	Specifies the area ID for a new OSPF area. The area ID 0.0.0.0 is reserved for the backbone.  Best Practice - We recommend that you enter the area ID as a dotted quad, but you can use any integer as the area ID.
stub {off on}	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 {off on}	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.

Parameter	Description
<code>nssa {off on}</code>	<p>Specifies the area ID for an NSSA.</p> <div>  Note - The backbone area cannot be an NSSA area. </div>
<code>nssa default-cost <1-677215></code>	Specifies the cost associated with the default route to the NSSA.
<code>nssa default-metric-type <1-2></code>	<p>Specifies the type of metric.</p> <ul style="list-style-type: none"> 1 - Route is internal and its metric can be used directly by OSPF for comparison. 2 - Route is external and its metric cannot be used for comparison directly.
<code>nssa import-summary-routes {off on}</code>	Specifies if summary routes (summary link advertisements) are imported into the NSSA.
<code>nssa translator-role {always candidate}</code>	<p>Specifies whether this NSSA border router will unconditionally translate Type-7 LSAs into Type-5 LSAs.</p> <ul style="list-style-type: none"> always - Type-7 LSAs are translated into Type-5 LSAs regardless of the translator state of other NSSA border routers. candidate - This router participates in the translator election to determine if it will perform the translations duties.
<code>nssa translator-stability-interval <1-65535></code>	<p>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.</p> <p>Default: 40 seconds.</p>
<code>nssa redistribution {off on}</code>	Specifies if both Type-5 and Type-7 LSAs or only Type-7 LSAs will be originated by this NSSA border router.
<code>nssa range <Range of IP Addresses> [restrict] {off on}</code>	<p>Specify the range of addresses to reduce the number of Type-5 LSAs for the NSSA border router.</p> <p>To prevent a specific prefix from being advertised, use the restrict argument.</p>

Configuring 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.

Syntax:

```
set ospf area backbone virtual-link <IP Address> transit-area <OSPF Area>
  authtype
    md5 key <Authorization Key>
      secret <Secret>
      off
    none
    simple <Password>
  dead interval {<1-4294967295> | default}
  hello-interval {<1-65535> | default}|
  {off | on}
  retransmit-interval {<1-4294967295> | default}
```

Parameters:

Parameter	Description
virtual-link <IP Address>	<p>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.</p> <p>Configure the OSPF area you are using as the transit area before you configure the virtual link.</p> <p>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.</p> <p>Traffic between the endpoints of the virtual link flow through this area.</p> <p>The virtual link IP address functions as the router ID of the remote endpoint of the virtual link.</p>
authtype md5 key <Authorization Key> secret <Secret>	<p>Specifies to use MD5 authorization. Enter at least one key ID and its corresponding MD5 secret.</p> <p>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.</p> <p>Generally, routers on a given link must agree on the authentication configuration to form neighbor adjacencies.</p> <p>Use an authentication scheme to guarantee that routing information is accepted only from trusted peers.</p> <p>The value "off" disables the configured MD5 key.</p>
authtype none	<p>Specifies not to use an authentication scheme for the specified interface.</p>

Parameter	Description
<pre> authtype simple <Password> </pre>	<p>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.</p>
<pre> dead-interval {<1-4294967295> default} </pre>	<p>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. Default: 10 seconds</p>
<pre> hello-interval {<1-65535> default} </pre>	<p>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. Default: 10 seconds</p>
<pre> {off on} </pre>	<p>Disables (<code>off</code>) and enables (<code>on</code>) the configured OSPF virtual link.</p>
<pre> retransmit-interval {<1-4294967295> default} </pre>	<p>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. Default: 5 seconds</p>

Monitoring OSPF

Syntax:

```
show ospf
  border-routers
  errors
    dd
    hello
    ip
    lsack
    lsr
    lsu
    protocol
  events
  database
  database areas
    area <OSPF Area>
    asbr-summary-lsa
    checksum
    database-summary
    detailed
    external-lsa
    network-lsa
    nssa-external-lsa [detailed]
    router-lsa
    summary-lsa
    type <1 | 2 | 3 | 4 | 5 | 7> [detailed]
  interfaces
    stats
    detailed
  interface <Name of Interface>
    stats
    detailed
  neighbors
  neighbor <IP Address>
  packets
  summary
```

Parameters:

Parameter	Description
neighbors	Shows: <ul style="list-style-type: none"> ■ IP addresses of neighboring interfaces ■ Priority and status of neighboring interfaces ■ Number of errors logged for each interface

Parameter	Description
<code>neighbor <IP Address></code>	Shows for the specified IP address: <ul style="list-style-type: none"> ■ Priority ■ Status ■ Number of errors logged
<code>interfaces</code>	Shows for all configured logical interfaces: <ul style="list-style-type: none"> ■ Names of interfaces ■ IP addresses of interfaces ■ Area, to which each interface is assigned ■ Status of interfaces ■ IP address of the designated router ■ IP address of the backup designated router
<code>interfaces stats</code>	Shows for each OSPF interface: <ul style="list-style-type: none"> ■ Number of each type of error message logged ■ Number of link state advertisements sent
<code>interfaces detailed</code>	Shows detailed information about each OSPF interface, including: <ul style="list-style-type: none"> ■ Authentication type configured (if any) ■ Router IDs and IP addresses of the designated router ■ Router IDs and IP addresses of the backup designated router ■ Timer intervals configured for Hello, Wait, Dead, and Retransmit packets ■ Number of neighbors
<code>interface <Name of Interface></code>	Shows for the specified interface: <ul style="list-style-type: none"> ■ IP address of the interface ■ Area ID ■ Status ■ Number of errors logged ■ IP addresses of the designated router and backup designated router
<code>interface <Name of Interface> stats</code>	Shows for the specified interface: <ul style="list-style-type: none"> ■ Number of each type of error message logged ■ Number of link-state advertisements sent

Parameter	Description
<code>interface <Name of Interface></code> <code>detailed</code>	Shows detailed information about the specified interface, including: <ul style="list-style-type: none"> ■ Authentication type configured (if any) ■ Router IDs and IP addresses of the designated router ■ Router IDs and IP addresses of the backup designated router ■ Timer intervals configured for Hello, Wait, Dead, and Retransmit packets ■ Number of neighbors
<code>packets</code>	Shows the number of each type of packet sent, including: <ul style="list-style-type: none"> ■ Hello packets ■ Link-state update packets ■ Link-state acknowledgment packets ■ Link-state request packets
<code>errors</code>	Shows the number of each type of error message sent, including: <ul style="list-style-type: none"> ■ Hello protocol errors ■ Database description errors ■ Protocol errors ■ Link-state acknowledgment errors ■ Link-state request errors ■ Link-state update errors ■ IP errors
<code>errors dd</code>	Shows the number of each type of database errors.
<code>errors hello</code>	Shows the number of each type of Hello errors.
<code>errors ip</code>	Shows the number of each type of IP errors.
<code>errors lsack</code>	Shows the number of each type of Link-state acknowledgment errors.
<code>errors lsu</code>	Shows the number of each type of Link-state update errors.
<code>errors lsr</code>	Shows the number of each type of Link-state request errors.
<code>errors protocol</code>	Shows the number of each type of Protocol errors.
<code>border-routers</code>	Shows: <ul style="list-style-type: none"> ■ IP address of each area border router ■ OSPF area of each border router ■ Cost associated with each IP address

Parameter	Description
<code>database</code>	<p>Shows these statistics for each OSPF area:</p> <ul style="list-style-type: none"> ■ Router-link state ■ Network-link state <p>Shows for each OSPF interface:</p> <ul style="list-style-type: none"> ■ Checksum ■ Sequence number ■ Link count
<code>database areas</code>	<p>Shows these statistics for each OSPF area:</p> <ul style="list-style-type: none"> ■ Router-link state ■ Network-link state ■ AS-border-router link state ■ AS-external link state ■ Summary-link state <p>Shows for each OSPF interface:</p> <ul style="list-style-type: none"> ■ Checksum ■ Sequence number ■ Link count
<code>database area <OSPF Area></code>	<p>Shows these statistics for the specified OSPF area:</p> <ul style="list-style-type: none"> ■ Router-link state ■ Network-link state ■ AS-border-router-link state ■ AS- external-link state ■ Summary-link state statistics <p>Shows for each IP address configured within the specified OSPF area:</p> <ul style="list-style-type: none"> ■ Checksum ■ Sequence number ■ Link count
<code>database asbr-summary</code>	<p>Shows a summary of AS-border-router link state statistics for each OSPF area. For OSPFv2 only.</p>
<code>database external</code>	<p>Shows AS-external-link state statistics for each OSPF area.</p>
<code>database database-summary</code>	<p>Shows these statistics:</p> <ul style="list-style-type: none"> ■ Summary of router-link-state ■ Network-link state ■ Summary-link-state ■ AS-border-router-link state

Parameter	Description
<code>database network</code>	Shows network-link-state statistics for each OSPF interface, including: <ul style="list-style-type: none"> ■ Advertised router ■ Sequence number ■ Checksum For OSPFv2 only.
<code>database nssa-external-lsa [detailed]</code>	Shows type 7 LSAs (NSSA). For OSPF v2 only.
<code>database router-lsa</code>	Shows these the router-link-state statistics for each OSPF interface, including: <ul style="list-style-type: none"> ■ Advertised router ■ Sequence number ■ Checksum ■ Link count For OSPFv2 only.
<code>database summary-lsa</code>	Shows a summary of link-state statistics for each OSPF area. For OSPFv2 only.
<code>database type <1 2 3 4 5 7> [detailed]</code>	Shows link-state statistics associated with the specified LSA number: <ul style="list-style-type: none"> ■ 1 - Router-link-state statistics. ■ 2 - Network-link-state statistics. ■ 3 - Summary-link-state statistics. ■ 4 - AS-border-router-link-state statistics. ■ 5 - AS-external-link-state statistics. ■ 7 - NSSA. For OSPF v2 only.
<code>events</code>	Shows: <ul style="list-style-type: none"> ■ 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 ■ Link state advertisement messages

Route Aggregation

Route aggregation is used to combine a set of more specific routes into a single more general route.

This reduces the number of routes advertised by a given protocol.

Example:

- A router has many stub interface routes subnetted from a Class C network.
- A router runs RIPv2 on another interface.
- In this case, these interface routes can be combined into a single aggregate route (for example, the Class C network).

This single aggregate route can be redistributed into RIPv2, instead of the large list of individual routes.



Important - Be careful when aggregating if there are gaps in the route that is aggregated.

The interface that originates the aggregate routes does not use them to forward packets. Only the router that receives the routes uses them.

A router that receives a packet that does not match one of the component routes, should respond with an ICMP "Network Unreachable" message.

This prevents packets for unknown component routes from following a default route into another network.

In this situation, they might be continually forwarded back to the border router until their TTL expires.

To create an aggregate route, first specify the network address and subnet mask, followed by the set of contributing routes.

Define the contributing routes by specifying a source, such as a routing protocol or a static route, followed by a route filter, which is either a prefix or the keyword "all IPv4 routes".

An aggregate route can have many contributing routes. However, at least one of the routes must be already present to generate the aggregate.

Configuring Route Aggregation in Gaia Clish

Create aggregate routes using these Gaia Clish commands:

```
set aggregate <IP Prefix>
  contributing protocol <Protocol> contributing-route
  all {off | on}
  <IP Prefix>
  exact on
  off
  on
  refines on
  off
  contributing protocol <Protocol> off
  rank {default | <0-255>}
  weight default
  aspath-truncate {off | on}
```

Parameters

Parameter	Description
contributing protocol <Protocol> contributing-route {all <IP Prefix>} {off on}	The IP address and mask length of the new aggregate route and the contributing protocol or interface route. To specify a protocol, enter <code>direct</code> , <code>static</code> , <code>ospf2</code> , <code>ospf2ase</code> , <code>bgp</code> , <code>rip</code> , <code>igrp</code> , or <code>aggregate</code> . 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.
<IP Prefix> exact on	The contributing route is limited to the specified IP address and mask length only. You cannot enable both <code>exact on</code> and <code>refines on</code> at the same time. If you enable "refines on" when "exact on" is disabled, "exact on" is automatically disabled.
<IP Prefix> refines on	The contributing route is based on addresses with a greater value than the specified mask length of the specified IP address. You cannot enable both "exact on" and "refines on" at the same time. If you enable "refines on" when "exact on" is disabled, "exact on" is automatically disabled.

Parameter	Description
<code>rank default</code>	<p>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.</p> <p>Each routing protocol has a different default rank value. Aggregate routes have a default rank of 130.</p>
<code>rank <0-255></code>	<p>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.</p> <p>Each routing protocol has a different default rank value. Each routing protocol has a different default rank value. Aggregate routes have a default rank of 130.</p>
<code>weight default</code>	<p>A value that breaks a tie if select routes going to the same destination have the same rank value.</p> <p>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.</p> <ul style="list-style-type: none"> ■ Range: 0-65535. ■ Default: 0
<code>weight <0-65535></code>	<p>A value that breaks a tie if select routes going to the same destination have the same rank value.</p> <p>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.</p> <ul style="list-style-type: none"> ■ Default: 0
<code>aspath-truncate {off on}</code>	<p>Specifies that the autonomous system (AS) path be truncated to the longest common AS path.</p> <p>The default behavior is to build an AS path that consists of sets and sequences of all contributing AS paths.</p> <ul style="list-style-type: none"> ■ Default: off

Route Maps

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 in Gaia Clish.

Route maps support RIP, BGP, and OSPFv2.

You can also use the Route Redistribution and Inbound Route Filters features that you configure in Gaia Clish or Gaia Portal.

Route map for import policy corresponds to Inbound Route Filters; route map for export policy corresponds to Route Redistribution.

Protocols can use:

- Route maps for route redistribution
- Other settings for inbound route filtering

However, if one or more route maps are assigned to a protocol (for import or export), any other configuration for route redistribution or inbound route filters is ignored.



Note - Route maps offer more configuration options than the configuration of route redistribution and inbound route filters. They are not functionally equivalent.

Configuring a Routemap in Gaia Clish

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 other Route Redistribution and Inbound Route Filters settings that you configure in Gaia Clish or Gaia Portal.

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.

Configuring a Routemap

Configuring a Routemap

Syntax:

```
set routemap <Name of Routemap> id {<1-65535> | default}
    {off | on}
    allow
    inactive
    restrict
```

Parameters:

Parameter	Description
routemap <Name of Routemap>	The name of the routemap.
id {<1-65535> default}	The ID of the routemap. You can enter the keyword "default" or the default value "10".
{off on}	Control the routemap: <ul style="list-style-type: none"> ■ off - To delete a routemap. ■ on - To create 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.

Configuring Actions for a Routemap

Syntax:



Note - Some statements affect only a particular protocol (see ["Supported Route Map Statements by Protocol" on page 90](#)). The same parameter cannot appear both as a match and action statement in a routemap. These include Community, Metric, and Nexthop.

```
set routemap <Name of Routemap> id {<1-65535> | default} action
aspath-prepend-count <1-25>
community {append | replace | delete} [{off | on}]
    <1-65535> as <1-65535> [{off | on}]
    no-export [{off | on}]
    no-advertise [{off | on}]
    no-export-subconfed [{off | on}]
    none [{off | on}]
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 <Name of Attribute>
ospfautomatictag <0-4095>
ospfmanualtag <1-4294967295>
riptag <1-65535>
```

Parameters:

Parameter	Description
routemap <Name of Routemap>	Specifies the name of the routemap.
id {<1-65535> default}	Specifies the ID of the routemap. You can enter the keyword <code>default</code> or the default value 10.
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. Applies only to BGP.
community {append replace delete} [{off on}]	Operate on a BGP community string. A community string can be formed using multiple community action statements. You can specify keywords <code>append</code> , <code>replace</code> , or <code>delete</code> for the kind of operation to be performed using the community string. The default operation is <code>append</code> . Applies only to BGP.

Parameter	Description
<code>community <1-65535> as <1-65535> [{off on}]</code>	Specifies a BGP community value.
<code>community no-export [{off on}]</code>	Routes received that carry a communities attribute containing this value must not be advertised outside a BGP confederation boundary (a stand-alone autonomous system that is not part of a confederation should be considered a confederation itself)
<code>community no-advertise [{off on}]</code>	Routes received that carry a communities attribute containing this value must not be advertised to other BGP peers.
<code>community no-export-subconfed [{off on}]</code>	All routes received carrying a communities attribute containing this value MUST NOT be advertised to external BGP peers (this includes peers inside a BGP confederation that belong to the autonomous systems of other members).
<code>community none [{off on}]</code>	In action statement, this statement makes sense only if used with replace. This deletes all communities associated with a route so that the route has no communities associated with it. Using it with append or delete would be a no-operation. The CLI returns an error if you turn "none" on and other community values already defined or if "none" is defined and you add some other community value.
<code>localpref <1-65535></code>	Set the local preference for BGP route. Applies only to BGP.
<code>metric [{add subtract}] <1-16></code>	Add to or subtract from the metric value. Applies only to RIPv1.
<code>metric igp [{add subtract} <1-4294967295>]</code>	Set metric to IGP metric value or add to or subtract from the IGP metric value. Applies only to RIPv1.
<code>metric value <1-4294967295></code>	Set the metric value. Protocol metrics: <ul style="list-style-type: none"> ■ For RIP - <i>metric</i> ■ For OSPF - <i>cost</i> ■ For BGP - <i>MED</i>
<code>nexthop ip IPv4 Address></code>	Set IPv4 Nexthop Address. Applies only to BGP.
<code>precedence <1-65535></code>	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.

Parameter	Description
<code>preference <1-65535></code>	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. Applies only to BGP.
<code>route-type <type-1 type-2></code>	Type of OSPF external route. The metric type of AS External route is set to the specified value. Applies only to routes redistributed to OSPF.
<code>remove Name of Attribute</code>	Remove the specified action from the routemap. For community, it removes all community statements. Accepted values are: <ul style="list-style-type: none"> ■ aspath-regex ■ community ■ ifaddress ■ interface ■ metric ■ neighbor ■ network ■ nexthop ■ protocol ■ route-type
<code>ospfautomatictag <0-4095></code>	Configures the automatic tag for OSPF external routes that match the Route Map. This action only applies when you export non-OSPF routes into OSPF with the "export-routemap" command. See RFC 1403 for more information on OSPF tags.
<code>ospfmanualtag <1-4294967295></code>	Configures the manual tag for OSPF external routes that match this Route Map ID. This action only applies when you export non-OSPF routes into OSPF with the "export-routemap" command. See RFC 1403 for more information on OSPF tags.
<code>riptag <1-65535></code>	Creates a RIP tag for external routes that match the given Route Map ID. This action only applies to export of non-RIP routes into RIP with the "export-routemap" command. See RFC 2453 for more information on RIP tags.

Configuring the Criteria that must be Matched for the Routemap to Take Effect

Syntax:



Note - Some statements affect only a particular protocol (see ["Supported Route Map Statements by Protocol" on page 90](#)). The same parameter cannot appear both as a match and action statement in a routemap. These include Community, Metric, and Nexthop.

```
set routemap <Name of Routemap> id <1-65535> match
  as <1-65535> [{off | on}]
  aspath-regex [{"<Regular Expression>" | empty}] origin {any | igp |
incomplete}
  community
    <1-65535> as <1-65535> [{off | on}]
    exact [{off | on}]
    no-export [{off | on}]
    no-advertise [{off | on}]
    no-export-subconfed [{off | on}]
    none [{off | on}]
  ifaddress <IPv4 Address> [{off | on}]
  interface <Name of Interface> [{off | on}]
  metric value <1-4294967295>
  neighbor <Neighbor IPv4 Address> [{off | on}]
  network <Network IPv4 Address>/<Masklength>
    {all | exact | off | refines}
    between <Masklength_1> and <Masklength_2>
  nexthop <IPv4 Address>
    [{off | on}]
  protocol {ospf2 | ospf2ase | bgp | rip | static | direct | aggregate}
  route-type {inter-area | intra-area | type-1 | type-2} [{off | on}]
  remove <Name of Match Condition>
```

Parameters:

Parameter	Description
as <1-65535> [{off on}]	Match the specified autonomous system number with the AS number of a BGP peer. Applies only to BGP.
aspath-regex [{"<Regular Expression>" empty}] origin {any igp incomplete}	Match the specified ASpath regular expression. Note - You must enter the regular expression in quotation marks. Use the empty keyword to match a null ASpath. Applies only to BGP.
community <1-65535> as <1-65535> [{off on}]	Specify the BGP community value.

Parameter	Description
<code>community exact [{off on}]</code>	Specify that the communities present in the route must exactly match all the communities in the routemap. In absence of the exact clause, the route can have other community values associated with it in addition to the ones contained in the routemap. You can have multiple community statements in a route map to form a community string.
<code>community no-export [{off on}]</code>	All routes received that carry a communities attribute containing this value must not be advertised outside a BGP confederation boundary (a stand-alone AS that is not part of a confederation should be considered a confederation itself).
<code>community no-advertise [{off on}]</code>	All routes received carrying a communities attribute containing this value must not be advertised to other BGP peers.
<code>community no-export-subconfed [{off on}]</code>	All routes received carrying a communities attribute containing this value must not be advertised to external BGP peers (this includes peers in other members autonomous systems inside a BGP confederation).
<code>community none [{off on}]</code>	Matches an empty community string, i.e. a route which does not have any communities associated with it. The CLI returns an error if you turn "none" on and other community values already defined, or if "none" is defined and you add some other community value.
<code>ifaddress <IPv4 Address> [{off on}]</code>	Match the specified interface address. There can be multiple if address statements.
<code>interface <Name of Interface> [{off on}]</code>	Match the route if the nexthop lies on the specified interface name. There can be multiple interface statements.
<code>metric value <1-4294967295></code>	Match the specified metric value.
<code>neighbor <Neighbor IPv4 Address> [{off on}]</code>	Match the neighbors IP address. There can be multiple neighbor statements. Applies only to BGP and RIPv1.

Parameter	Description
<pre>network <Network IPv4 Address>/<Masklength> {all exact off refines}</pre>	<p>Match the network as specified:</p> <ul style="list-style-type: none"> ■ all Match all networks belonging to this prefix and masklength. ■ exact Match prefix exactly. ■ off Delete the network match statement. ■ refines Match networks with more specific mask lengths only. Matches only subnets. ■ between <Masklength_1> and <Masklength_2> Specify a range of masklengths to be accepted for the specified prefix. <p>There can be multiple network match statements in a route map.</p>
<pre>nexthop <IPv4 Address></pre>	Match the specified nexthop address.
<pre>protocol {ospf2 ospf2ase bgp rip static direct aggregate}</pre>	Match the specified protocol. Use this for route redistribution.
<pre>route-type {inter-area intra-area type-1 type-2} [{off on}]</pre>	<p>As a match statement in routemap for export policy, it can be used by any protocol to redistribute OSPF routes.</p> <p>If you specify "inter-area" or "intra-area", then set the protocol match condition to "ospf2".</p> <p>If you specify "type-1" or "type-2", then set the protocol match condition to "ospf2ase".</p> <p>While exporting OSPF ASE routes to other protocol, if metric match condition is set but the "route-type" match condition is not set, it tries to match the metric value for both "type-1" and "type-2" routes.</p> <p>There can be multiple route-type match statements.</p>
<pre>remove <Name of Match Condition></pre>	<p>Remove the specified match condition from the routemap.</p> <p>For match conditions which can have multiple match statements (such as network, neighbor), this argument removes all of them.</p>

Viewing Routemaps

```
show routemaps
```

```
show routemap <<Name of Routemap>> {all | <1-65535>}
```


Routemap Protocol Commands

To assign routemaps to protocols:

The preference value specifies which order the protocol will use each routemap.

```
set {ospf | rip}
    export-routemap <Name of Routemap> preference <1-65535> on
    import-routemap <Name of Routemap> preference <1-65535> on
```

To turn a routemap off:

```
set {ospf | rip}
    export-routemap <Name of Routemap> off
    import-routemap <Name of Routemap> off
```

To view routemaps assigned to protocols:

```
show {ospf | rip} routemap
```

To set BGP routemaps for export and import policies:

```
set bgp external remote-as <1-65535> export-routemap <Name of Routemap>
    off
    preference <1-65535> [family inet] on

set bgp external remote-as <1-65535> import-routemap <Name of Routemap>
    off
    preference <1-65535> [family inet] on

set bgp internal export-routemap <Name of Routemap>
    off
    preference <1-65535> [family inet] on

set bgp internal import-routemap <Name of Routemap>
    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 Portal.

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 written in the `/var/log/messages` file 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

Category	Supported Statements
Import Match conditions	<ul style="list-style-type: none"> ■ Neighbor ■ Network ■ Interface ■ Ifaddress ■ Metric ■ Neighbor ■ Nexthop
Import Actions	<ul style="list-style-type: none"> ■ Precedence ■ Metric Add/Subtract
Export Match conditions when exporting from RIP	<ul style="list-style-type: none"> ■ 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

Category	Supported Statements
Import Match conditions	Network (Route Prefix)
Import Actions	Precedence
Export Match conditions when other protocols redistribute OSPF routes	<ul style="list-style-type: none"> ■ Network ■ Interface ■ Ifaddress ■ Metric ■ Route-type ■ Nexthop
Export Match conditions when OSPF redistributes routes from other protocols	According to the conditions supported by that protocol
Export Actions when redistributing to AS External	<ul style="list-style-type: none"> ■ Metric ■ Route-type

Route Map Examples

Example 1

Requirements:

1. Redistribute interface route for eth3 into OSPF.
2. Set the OSPF route-type to AS type-2 with cost 20.

Syntax:

```
set routemap direct-to-ospf id 10 on
set routemap direct-to-ospf id 10 match interface eth3
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

Requirements:

1. Do not accept routes from RIP neighbor 192.0.2.3.
2. Accept routes from neighbor 192.0.2.4 as is.
3. For all other routes increment the metric by 2.

Syntax:

```
set routemap rip-in id 10 onset 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

Requirements:

1. Redistribute all static routes into BGP AS group 400.
2. Set the MED value to 100.
3. Prepend the AS number to the aspath 4 times.
4. If the route belongs to the prefix 192.0.2.0/8, do not redistribute.
5. Send all BGP routes whose aspath matches the regular expression (100 200+) and set the MED value to 200.

Syntax:

```
set routemap static-to-bgp id 10 onset 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 that belong to the same protocol.

Redistributing Static, Interface, or Aggregate Routes

This table shows these supported match conditions when redistributing different routes into BGP, OSPFv2, or RIP:

Route Type	Supported Match Conditions
static	<ul style="list-style-type: none"> ■ Network Prefix ■ Nexthop ■ Interface ■ Ifaddress ■ Protocol (proto = static)
interface and direct	<ul style="list-style-type: none"> ■ Network Prefix ■ Interface ■ Ifaddress ■ Protocol (proto = direct)
aggregate	<ul style="list-style-type: none"> ■ Network Prefix ■ Protocol (proto = aggregate)

PIM

Introduction

Protocol-Independent Multicast (PIM) can forward multicast packets with a unicast protocol.

PIM efficiently routes multicast traffic for groups that span wide area (and inter-domain) networks.

It works with all existing unicast routing protocols.

PIM in Gaia Embedded supports these modes:

- Dense Mode (PIM DM)
- Sparse Mode (PIM SM)
- Source-Specific Multicast Mode(PIM SSM)



Important - The implementation does not support enabling both Dense Mode and Sparse Mode, or either mode of PIM and DVMRP on the same appliance.



Note - You can run PIM over a route-based VPN by enabling PIM on an unnumbered Virtual Tunnel Interface (VTI).

PIM Dense Mode (DM)

This mode is most useful when:

- Senders and receivers are in close proximity to one another.
- There are few senders and many receivers.
- The volume of multicast traffic is high.
- The stream of multicast traffic is constant.

PIM Sparse Mode (SM)

This mode is most useful when:

- There are few receivers in a group.
- Senders and receivers are separated by WAN links.
- The type of traffic is intermittent.

PIM Source-Specific Multicast (SSM) Mode

This mode is most useful when:

- Most multicast traffic is from well-known sources.
- It is desirable to avoid the overhead of shared tree and rendezvous point processing associated with sparse mode.

SSM is a version of PIM Sparse Mode. It is used in conjunction with IGMP v3 to request or block multicast traffic from specific sources. For example, when a host requests traffic for a multicast group from a specific source, SSM sends PIM join/prune messages towards the source.

The multicast group range 232.0.0.0/8 is reserved for SSM. When SSM is enabled, Sparse Mode accepts only IGMP v3 reports for groups that fall within this range. Sparse Mode ignores IGMP v1 and IGMP v2 reports in this range.

In addition, only shortest-path-tree (SPT) join/prune messages for these groups are accepted from neighboring routers. All other multicast groups are processed as in native Sparse Mode.

SSM does not need a Rendezvous Point (RP). The presence of an RP for any of the SSM groups does not have any influence on the processing of join/prune messages.

Configuring PIM in Gaia Clish

- To see the available "set" commands for PIM, enter in Gaia Clish:

```
set pim[Space] [Tab]
```

- To see the available "show" commands for PIM, enter in Gaia Clish:

```
show pim[Space] [Tab]
```

Configuring PIM

Syntax:


```

set pim
  assert-interval {<1-3600> | default}
  assert-limit {<10-10000> | default}
  assert-rank protocol <Protocol> rank {<0-255> | default}
  bootstrap-candidate
    local-address <IP Address>
    {off | on}
    priority {<0-255> | default}
  candidate-rp
    advertise-interval {<1-3600> | default}
    local-address <IP Address>
    multicast group <IPv4 Address>/<Subnet Mask> {off | on}
    {off | on}
    priority {<0-255> | default}
  cisco compatibility {off | on}
  data-interval {<11-3600> | default}
  hello-interval {<1-21845> | default}
  ha-mode {off | on}
  interface <Name of Interface>
    {off | on}
    local-address <IP Address>
    dr-priority {<0-4294967295> } default}
  jp-delay-interval {<1-3600> | default}
  jp-interval {<1-3600> | default}
  mode {dense | sparse | ssm}
  nat-mode {off | on}
  register-suppress-interval {<60-3600> | default}
  spt-threshold <options>
  state-refresh {off | on}
  state-refresh-interval <0 - 255>
  state-refresh-ttl <1 - 255>
  static-rp
    {off | on}
    rp-address <IP Address>
      multicast-group <IPv4 Address>/<Subnet Mask> {off | on}
      {off | on}


```


Parameters:


Parameter	Description
assert-interval {<1-3600> default}	<p>Configures the assert interval.</p> <p>If an assert battle on an upstream interface results in the selection of a PIM neighbor other than the unicast reverse-path-forwarding (RPF) neighbor towards the source of the data traffic (for which the assert battle was generated) as the designated forwarder on that interface, then the winner is used as the upstream neighbor for all subsequent join/prune messages. This change is timed-out after expiry of the assert interval.</p> <p>Range: 1-3600 seconds Default: 180 seconds</p>
assert-limit {<10-10000> default}	<p>Reserved for future use.</p> <p>Configures the PIM assert limit.</p> <p>Range: 10-10000 Default: 10</p>
assert-rank protocol <Protocol>	<p>Configures the protocol, for which to configure the assert rank:</p> <ul style="list-style-type: none"> ■ <code>bgp</code> - Routes learned via the BGP protocol ■ <code>direct</code> - Routes directly connected to a network interface ■ <code>igrp</code> - Routes learned via the IGRP protocol ■ <code>kernel</code> - Kernel routes ■ <code>ospf</code> - Routes learned via the OSPF protocol ■ <code>ospfase</code> - External routes learned via OSPF ■ <code>rip</code> - Routes learned via the RIP protocol ■ <code>static</code> - Static routes
bootstrap-candidate local-address {<IPv4 address> default}	<p>Configures the Bootstrap Candidate Local Address used for the C-BSR state machine and the bootstrap messages.</p> <p>Important:</p>  <ul style="list-style-type: none"> ■ On a single Security Gateway, this address can be that of the PIM interfaces or an address configured on the loopback interface. If an address from the loopback interface is used, do not select an address in the 127/8 address range. ■ On a ClusterXL Cluster Member, you must configure the Cluster Virtual IP address configured on this PIM interface. <p>Range: Address of PIM interface, or a non 127.0.0.0/8 loopback address. Default: The IP address of one of the interfaces on which PIM is enabled. The default does not apply on Cluster Members.</p>
bootstrap-candidate {off on}	<p>Disables (<code>off</code>) or enables (<code>on</code>) the Bootstrap Candidate.</p> <p>Range: off, on Default: off</p>

Parameter	Description
bootstrap-candidate priority {<0-255> default}	<p>Configures the priority advertised in C-BSR messages.</p> <p>The candidate bootstrap router with the highest priority value is selected as the bootstrap router for the domain.</p> <p>The C-RP with the lowest priority has the highest preference.</p> <p>The highest priority value is 0.</p> <p>Range: 0-255</p> <p>Default: 0</p>
candidate-rp advertise-interval {<1-3600> default}	Configures the candidate Rendezvous Point (RP) Advertisement Interval.
candidate-rp local-address {<IPv4 address> default}	<p>Configures the Candidate Rendezvous Point router address used for the C-RP state machine and in the C-RP-Advertisements sent to the elected bootstrap router.</p> <p>Important:</p>  <ul style="list-style-type: none"> On a single Security Gateway, this address can be that of the PIM interfaces or an address configured on the loopback interface. If an address from the loopback interface is used, do not select an address in the 127/8 address range. In ClusterXL Cluster Member or VRRP Cluster Member, you must configure the Cluster Virtual IP address configured on this PIM interface. <p>Range: Address of PIM interface or a non 127.0.0.0/8 loopback address.</p> <p>Default: Selects the IP address of one of the interfaces on which PIM is enabled. The default does not apply on Cluster Members.</p>
candidate-rp multicast-group <IPv4 address>/<Subnet mask> {off on}	<p>Configure the Multicast Group, for which this router is designated as the candidate rendezvous point.</p> <ul style="list-style-type: none"> <IPv4 address> The multicast IP address of the group(s) in CIDR notation, for which this rendezvous point is responsible. Range: Dotted-quad ([224-239].[0-255].[0-255].[0-255]) Default: 224.0.0.0/4 <Subnet mask> Mask length. Range: 1-32 Default: None
candidate-rp {off on}	<p>Disables (off) or enables (on) Gaia as a candidate rendezvous point router.</p> <p>Range: off, on</p> <p>Default: off</p>

Parameter	Description
<pre>candidate-rp priority {<0-255> default}</pre>	<p>Configures the priority of this C-RP.</p> <p>All PIM routers select the same RP for a multicast group address from the list of C-RPs received in the bootstrap messages from the elected BSR.</p> <p>The lower the Local Preference of the C-RP, the higher the priority.</p> <p>Range: 0-255</p> <p>Default: 0</p>
<pre>cisco compatibility {off on}</pre>	<p>Configures the Cisco Compatibility for Register Checksums.</p> <p>Does not support the PIM Dense Mode.</p> <p>Range: off, on</p> <p>Default: off</p>
<pre>data-interval {<11-3600> default}</pre>	<p>Configures the life-time of a new PIM forwarding entry.</p> <p>Subsequently, the life of the entry is extended in different ways based on the location of this router in the network.</p> <p>For example, in some cases the receipt of PIM control messages (periodic join/prune messages) extends the life of the entry and in others the presence of local senders of multicast traffic prevents the deletion of the entry.</p> <p>Range: 11-3600 seconds</p> <p>Default: 210 seconds</p>
<pre>hello-interval {<1-21845> default}</pre>	<p>Configures the interval between PIM Hello messages that are sent on a multicast-capable interface.</p> <p>Hello messages are addressed to the All-PIM-Routers multicast group (224.0.0.13), so that PIM routers may discover neighbors on a multi-access network.</p> <p>Range: 1-21845 seconds</p> <p>Default: 30 seconds</p>
<pre>jp-delay-interval {<1-3600> default}</pre>	<p>Configures the maximal interval from the time when the unicast Reverse Path Forwarding (RPF) neighbor (towards a source or the RP) changes, and a triggered Join/Prune message is sent.</p> <p>Range: 1-3600 seconds</p> <p>Default: 5 seconds</p>
<pre>jp-interval {<1- 3600> default}</pre>	<p>Configures the interval between sending Join/Prune messages.</p> <p>Range: 1-3600 seconds</p> <p>Default: 60 seconds</p>

Parameter	Description
<code>ha-mode {off on}</code>	<p>Disables (<code>off</code>) or enables (<code>on</code>) the High-Availability mode in PIM Sparse-Mode.</p> <p>Enable the High-Availability (HA) mode when two routers are configured to back each other up to forward multicast traffic and PIM Sparse-Mode is implemented.</p> <p>When this option is enabled, all PIM-enabled interfaces are available only if each interface is up and has a valid address assigned.</p> <p>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.</p> <p>The HA mode feature applies only to Sparse-Mode PIM.</p> <p>The HA mode feature does not affect the functioning of Dense-Mode PIM.</p> <p>Range: off, on Default: off</p>
<code>interface <Name of Interface></code>	Specifies the interface, on which to enable PIM.
<code>interface <Name of Interface> local-address <IP Address></code>	<p>Specifies the local IP address used in all advertisements sent on the interface.</p> <p>This option is useful when multiple IP addresses are configured on the interface.</p> <p>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.</p> <p>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.</p> <div>  <p>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.</p> </div>
<code>interface <Name of Interface> {off on}</code>	Disables (<code>off</code>) or enables (<code>on</code>) PIM on the specified interface.

Parameter	Description
<pre>interface <Name of Interface> dr- priority {<0- 4294967295> default}</pre>	<p>Configures the Designated Router priority advertised in the PIM Hello messages that are sent on the interface.</p> <p>This is used for DR selection on a LAN.</p> <p>The router with the highest priority is selected as 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 configured, the DR election is based on the IP address.</p> <p>Note - To make sure that a PIM neighbor supports DR Priority:</p>  <ol style="list-style-type: none"> Run this command in Gaia Clish on the Security Gateway: <div> <pre>show pim neighbor <IP Address of Neighbor></pre> </div> For neighbors that advertise a DR selection priority value, this message shows in the summary: <div> <pre>DRPriorityCapable Yes</pre> </div> <p>Range: 0-4294967295 Default: 1</p>
<pre>mode {dense sparse ssm}</pre>	<p>Configures the PIM mode:</p> <ul style="list-style-type: none"> ■ dense - Dense Mode ■ sparse - Sparse Mode ■ ssm - Source-Specific Multicast
<pre>nat-mode {off on}</pre>	<p>Disables (off) or enables (on) PIM NAT mode to translate the IP addresses in a PIM protocol message to the relevant IP address(es) for the interface, on which the message is being sent.</p> <p>Range: off, on Default: off</p>
<pre>register-suppress- interval {<60- 3600> default}</pre>	<p>Configures the mean interval between receipt of a register-stop and the time when registers can be sent again.</p> <p>A lower value means more frequent register bursts at the rendezvous point. A higher value means a longer join latency for new receivers.</p> <p>Range: 60-3600 seconds Default: 60</p>
<pre>spt-threshold <options></pre>	<p>This command is deprecated and not supported.</p>
<pre>state-refresh {off on}</pre>	<p>Disables (off) or enables (on) the use of 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.</p>
<pre>state-refresh- interval {<1-255> default}</pre>	<p>For Dense Mode, configures the interval at which state refresh messages are sent for multicast traffic originated by directly-connected sources.</p> <p>Range: 1-255 seconds Default: 60 seconds</p>

Parameter	Description
<pre>state-refresh-ttl {<1-255> default}</pre>	<p>For Dense Mode, configures the time-to-live (TTL) placed in the state refresh messages originated for multicast traffic from directly-connected sources. You can use this value to limit the forwarding of state refresh messages in the network.</p> <p>In the absence of user configuration, it is derived from the multicast data.</p> <p>Range: 1-255 Default: None</p>
<pre>static-rp {off on}</pre>	<p>Disables (<code>off</code>) or enables (<code>on</code>) the router as a Static Rendezvous Point.</p>
<pre>static-rp rp- address <IPv4 address></pre>	<p>Configures the Static Rendezvous Point IP address.</p> <p>If an associated multicast group and prefix is not configured, the Static Rendezvous Point (RP) is considered to be responsible for all multicast groups (224.0.0.0/4).</p> <p>This needs to be consistent with the RP information at other routers in a multicast domain irrespective of the RP-dissemination mechanism (bootstrap or autoRP) used.</p> <p> Note - The static RP overrides the RP information received from other RP-dissemination mechanisms, such as bootstrap routers.</p> <p>Range: Any IP address Default: None</p>
<pre>static-rp rp- address <IPv4 address> multicast-group <IPv4 address>/<Subnet mask> {off on}</pre>	<p>Configures the Multicast Group, for which this router is designated as the static rendezvous point.</p> <ul style="list-style-type: none"> ■ <code><IPv4 address></code> The multicast IP address of the group(s) in CIDR notation, for which this rendezvous point is responsible. Range: Dotted-quad ([224-239].[0-255].[0-255].[0-255]) Default: 224.0.0.0/4 ■ <code><Subnet mask></code> Mask length. Range: 1-32 Default: None
<pre>static-rp rp- address <IPv4 address> {off on}</pre>	<p>Disables (<code>off</code>) or enables (<code>on</code>) the specified static Rendezvous Point.</p> <p>Range: off, on Default: off</p>

Debugging PIM

Use these commands to debug PIM:

Command	Shows
<code>show pim interface</code>	Which interfaces are running PIM, their status, and the mode they are running. This command also shows the interface and its DR priority and the number of PIM neighbors on the interface.
<code>show pim neighbors</code>	The IP address of each PIM neighbor and the interface on which the neighbor is present. This command also shows 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.
<code>show pim statistics</code>	The number of different types of PIM packets received and transmitted and any associated errors.
<code>show mfc cache</code>	Multicast source and group forwarding state by prefix.
<code>show mfc interfaces</code>	Shows multicast source and group forwarding state by interface.

Use these commands to debug Sparse-Mode PIM:

Command	Shows
<code>show pim bootstrap</code>	The IP address and state of the Bootstrap router.
<code>show pim candidate-rp</code>	The state of the Candidate Rendezvous Point state machine.
<code>show pim joins</code>	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 " <code>show mfc cache</code> " command.
<code>show pim rps</code>	The active Rendezvous Point (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.
<code>show pim group-rp-mapping <group-address></code>	The RP selected for a particular group based on information from the active RP-set.

Command	Shows
<code>show pim sparse-mode statistics</code>	Error statistics for: <ul style="list-style-type: none">■ Multicast Forwarding Cache (MFC)■ Bootstrap Router (BSR) messages;■ Candidate Rendezvous Point (CRP) advertisements■ Internet Group Management Protocol (IGMP)