Implementing RIP for IPv6

First Published: July 14, 2014

This module describes how to configure Routing Information Protocol for IPv6. RIP is a distance-vector routing protocol that uses hop count as a routing metric. RIP is an Interior Gateway Protocol (IGP) most commonly used in smaller networks.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the “Feature Information for Implementing RIP for IPv6” section on page 45-19.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

- Restrictions for Implementing RIP for IPv6, page 45-1
- Information About Implementing RIP for IPv6, page 45-2
- How to Implement RIP for IPv6, page 45-2
- Verifying IPv6 RIP Configuration and Operation, page 45-13
- Configuration Examples for IPv6 RIP, page 45-15
- Additional References, page 45-17
- Feature Information for Implementing RIP for IPv6, page 45-19

Restrictions for Implementing RIP for IPv6

- On default-vrf, the maximum number of processes supported is 4.
- On VRF, only one process is supported.
- String length of RIPng process name can have only 20 characters long.
Information About Implementing RIP for IPv6

RIP for IPv6

IPv6 RIP functions the same and offers the same benefits as RIP in IPv4. RIP enhancements for IPv6, detailed in RFC 2080, include support for IPv6 addresses and prefixes, and the use of the all-RIP-devices multicast group address FF02::9 as the destination address for RIP update messages.

In the Cisco software implementation of IPv6 RIP, each IPv6 RIP process maintains a local routing table, referred to as a Routing Information Database (RIB). The IPv6 RIP RIB contains a set of best-cost IPv6 RIP routes learned from all its neighboring networking devices. If IPv6 RIP learns the same route from two different neighbors, but with different costs, it will store only the lowest cost route in the local RIB. The RIB also stores any expired routes that the RIP process is advertising to its neighbors running RIP. IPv6 RIP will try to insert every non-expired route from its local RIB into the master IPv6 RIB. If the same route has been learned from a different routing protocol with a better administrative distance than IPv6 RIP, the RIP route will not be added to the IPv6 RIB but the RIP route will still exist in the IPv6 RIP RIB.

IPv6: RIPng VRF-Aware Support

When not in Virtual Routing and Forwarding (VRF) mode, every IPv6 Routing Information Protocol (RIP)—also know as RIP Next Generation (RIPng)—process and the configuration associated with it, keeps all the routes in the same routing table. In other routing protocols, it is often required to keep the protocol-related routes stored in separate routing tables.

The IPv6: RIPng VRF-Aware Support feature enables isolation, modularity, and potential performance improvement by reducing the number of routes stored in a single routing table. It also allows a network administrator to create different RIP routing tables and share the same protocol configuration stored in a single RIP protocol configuration block.

Nonstop Forwarding for IPv6 RIP

Cisco nonstop forwarding (NSF) continues forwarding packets while routing protocols converge, therefore avoiding a route flapping over. When an RP failover occurs, the Forwarding Information Base (FIB) marks installed paths as stale by setting a new epoch. Subsequently, the routing protocols re-converge and populate the RIB and FIB. Once all NSF routing protocols converge, any stale routes held in the FIB are removed. A failsafe timer is required to delete stale routes, in case of routing protocol failure to repopulate the RIB and FIB.

RIP registers as an IPv6 NSF client. Doing so has the benefit of using RIP routes installed in the Cisco Express Forwarding table until RIP has converged on the standby.

How to Implement RIP for IPv6

This section describes the following topics:

- Enabling IPv6 RIP, page 45-3
- Customizing IPv6 RIP, page 45-4
Enabling IPv6 RIP

Prerequisites

Before configuring the device to run IPv6 RIP, globally enable IPv6 using the `ipv6 unicast-routing` command in global configuration mode, and enable IPv6 on any interfaces on which IPv6 RIP is to be enabled.

If you want to set or change a global value, follow steps 1 and 2, and then use the optional `ipv6 router rip` command in global configuration mode.

SUMMARY STEPS

1. enable
2. configure terminal
3. ipv6 unicast-routing
4. interface type number
5. ipv6 rip name enable
How to Implement RIP for IPv6

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>Enter your password if prompted.</td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Step 3 ipv6 unicast-routing</td>
<td>Enables the forwarding of IPv6 unicast datagrams.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Step 4 interface type number</td>
<td>Specifies the interface type and number, and enters interface configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Step 5 ipv6 rip name enable</td>
<td>Enables the specified IPv6 RIP routing process on an interface.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
</tbody>
</table>

Customizing IPv6 RIP

SUMMARY STEPS

1. enable
2. configure terminal
3. ipv6 router rip word
4. maximum-paths number-paths
5. exit
6. interface type number
7. ipv6 rip name default-information {only | originate} [metric metric-value]
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
- Enter your password if prompted. |
| **Example:** | Device> enable |
| **Step 2** configure terminal | Enters global configuration mode. |
| **Example:** | Device# configure terminal |
| **Step 3** ipv6 router rip word | Configures an IPv6 RIP routing process and enters router configuration mode for the IPv6 RIP routing process.  
- Use the `word` argument to identify a specific IPv6 RIP routing process.  
Use the `no ipv6 router rip process-name` command to disable the IPv6 RIP routing process. |
| **Example:** | Device(config)# ipv6 router rip process1 |
| **Step 4** maximum-paths number-paths | (Optional) Defines the maximum number of equal-cost routes that IPv6 RIP can support.  
- The `number-paths` argument is an integer from 1 to 64.  
The default for RIP is four paths. |
| **Example:** | Device(config-router)# maximum-paths 1 |
| **Step 5** exit | Exits interface configuration mode and enters global configuration mode. |
| **Example:** | Device(config-if)# exit |
| **Step 6** interface type number | Specifies the interface type and number, and enters interface configuration mode. |
| **Example:** | Device(config)# interface gigabitethernet 0/0/0 |
| **Step 7** ipv6 rip name default-information {only | originate} [metric metric-value] | (Optional) Originates the IPv6 default route (::/0) into the specified RIP routing process updates sent out of the specified interface.  
**Note**  
To avoid routing loops after the IPv6 default route (::/0) is originated out of any interface, the routing process ignores all default routes received on any interface.  
- Specifying the only keyword originates the default route (::/0) but suppresses all other routes in the updates sent on this interface.  
- Specifying the originate keyword originates the default route (::/0) in addition to all other routes in the updates sent on this interface. |
| **Example:** | Device(config-if)# ipv6 rip process1 default-information originate |
Configuring IPv6: RIPng VRF-Aware Support

SUMMARY STEPS

1. enable
2. configure terminal
3. ipv6 unicast-routing
4. vrf definition vrf-name
5. address-family ipv6
6. exit
7. exit
8. ipv6 rip vrf-mode enable
9. ipv6 router rip rip-process-name
10. exit
11. interface type number
12. vrf forwarding vrf-name
13. ipv6 enable
14. ipv6 rip rip-process-name enable
15. end
16. debug ipv6 rip vrf vrf-name
17. show ipv6 rip vrf vrf-name next-hops
18. show ipv6 rip vrf vrf-name database
## DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable                  | Enables privileged EXEC mode.  
  - Enter your password if prompted. |
| **Example:** Device> enable        |         |
| **Step 2** configure terminal      | Enters global configuration mode. |
| **Example:** Device# configure terminal |     |
| **Step 3** ipv6 unicast-routing    | Enables the forwarding of IPv6 unicast datagrams. |
| **Example:** Device (config)# ipv6 unicast-routing | |
| **Step 4** vrf definition vrf-name | Configures a virtual routing and forwarding (VRF) routing table instance and enters VRF configuration mode. |
| **Example:** Device(config)# vrf definition vrf1 | |
| **Step 5** address-family ipv6     | Enters VRF address family configuration mode and enables IPv6 address prefixes. |
| **Example:** Device(config-vrf)# address-family ipv6 | |
| **Step 6** exit                   | Exits VRF address family configuration mode and returns to VRF configuration mode. |
| **Example:** Device(config-vrf-af)# exit | |
| **Step 7** exit                   | Exits VRF configuration mode and returns to global configuration mode. |
| **Example:** Device(config-vrf)# exit | |
| **Step 8** ipv6 rip vrf-mode enable | Enables VRF support for IPv6 RIP routing and enters RTR entry configuration mode. |
| **Example:** Device (config)# ipv6 rip vrf-mode enable | |
| **Step 9** ipv6 router rip rip-process-name | Creates an IPv6 Routing Information Protocol (RIP) routing process instance. |
| **Example:** Device (config)# ipv6 router rip myrip | |
| **Step 10** exit                  | Exits RTR entry configuration mode and returns to global configuration mode. |
| **Example:** Device (config-rtr)# exit | |
| **Step 11** interface type number | Specifies the interface type and number and enters interface configuration mode. |
| **Example:** Device (config)# interface Ethernet 0/0 | |
## How to Implement RIP for IPv6

### Step 12
- `vrf forwarding vrf-name`
  - Example: `Device(config-if)# vrf forwarding vrf1`
  - Binds the interface to the specified VRF routing instance table and removes all the Layer 3 interface configuration that is available when the command is entered.

### Step 13
- `ipv6 enable`
  - Example: `Device(config-if)# ipv6 enable`
  - Enables IPv6 on the interface.

### Step 14
- `ipv6 rip rip-process-name enable`
  - Example: `Device(config-if)# ipv6 rip myrip enable`
  - Enables an IPv6 RIP routing process on the interface.

### Step 15
- `end`
  - Example: `Device(config-if)# end`
  - Exits interface configuration mode and returns to privileged EXEC mode.

### Step 16
- `debug ipv6 rip vrf vrf-name`
  - Example: `Device# debug ipv6 rip vrf vrf1`
  - Displays debugging information related to VRF support for the specified IPv6 RIP VRF routing table instance.

### Step 17
- `show ipv6 rip vrf vrf-name next-hops`
  - Example: `Device# show ipv6 rip vrf vrf1 next-hops`
  - Displays the next hops in the specified VRF RIPng routing table.

### Step 18
- `show ipv6 rip vrf vrf-name database`
  - Example: `Device# show ipv6 rip vrf vrf1 database`
  - Displays the associated RIP local routing information base (RIB).

## Redistributing Routes into an IPv6 RIP Routing Process

The maximum metric that RIP can advertise is 16, and a metric of 16 denotes a route that is unreachable. Therefore, if you are redistributing routes with metrics greater than or equal to 16, then by default RIP will advertise them as unreachable. These routes will not be used by neighboring routers. The user must configure a redistribution metric of less than 15 for these routes.

### Note
You must to advertise a route with metric of 15 or less. A RIP router always adds an interface cost—the default is 1—onto the metric of a received route. If you advertise a route with metric 15, your neighbor will add 1 to it, making a metric of 16. Because a metric of 16 is unreachable, your neighbor will not install the route in the routing table.

If no metric is specified, then the current metric of the route is used. To find the current metric of the route, enter the `show ipv6 route` command.

## SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface type number`
Implementing RIP for IPv6

4. `ipv6 rip name enable`

5. `redistribute protocol [process-id] [level-1 | level-1-2 | level-2] [metric metric-value] [metric-type {internal | external}] [route-map map-name]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><strong>enable</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>Device&gt; enable</code></td>
</tr>
<tr>
<td></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td><strong>configure terminal</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>Device# configure terminal</code></td>
</tr>
<tr>
<td></td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td><strong>interface type number</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>Device(config)# interface gigabitethernet 0/0/0</code></td>
</tr>
<tr>
<td></td>
<td>Specifies the interface type and number, and enters interface configuration mode.</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><strong>ipv6 rip name enable</strong></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>Device(config-if)# ipv6 rip process1 enable</code></td>
</tr>
<tr>
<td></td>
<td>Enables the specified IPv6 RIP routing process on an interface.</td>
</tr>
<tr>
<td><strong>Step 5</strong></td>
<td>**redistribute protocol [process-id] [level-1</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td><code>Device(config-router)# redistribute bgp 65001 route-map bgp-to-rip</code></td>
</tr>
<tr>
<td></td>
<td>Redistributes the specified routes into the IPv6 RIP routing process.</td>
</tr>
<tr>
<td></td>
<td>• The protocol argument can be one of the following keywords: bgp, connected, isis, rip, or static.</td>
</tr>
<tr>
<td></td>
<td>• The rip keyword and process-id argument specify an IPv6 RIP routing process.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> The connected keyword refers to routes that are established automatically by assigning IPv6 addresses to an interface.</td>
</tr>
</tbody>
</table>

**Configuring Route Tags for IPv6 RIP Routes**

When performing route redistribution, you can associate a numeric tag with a route. The tag is advertised with the route by RIP and will be installed along with the route in neighboring router’s routing table.

If you redistribute a tagged route (for example, a route in the IPv6 routing table that already has a tag) into RIP, then RIP will automatically advertise the tag with the route. If you use a redistribution route map to specify a tag, then RIP will use the route map tag in preference to the routing table tag.

**SUMMARY STEPS**

1. enable
2. configure terminal
How to Implement RIP for IPv6

3. route-map map-tag [permit | deny] [sequence-number]
4. match ipv6 address {prefix-list prefix-list-name | access-list-name}
5. set tag tag-value

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td>Step 2 configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device# configure terminal</td>
<td></td>
</tr>
<tr>
<td>Step 3 route-map map-tag [permit</td>
<td>deny] [sequence-number]</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# route-map bgp-to-rip permit 10</td>
<td></td>
</tr>
<tr>
<td>Step 4 match ipv6 address {prefix-list prefix-list-name</td>
<td>access-list-name</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-route-map)# match ipv6 address prefix-list bgp-to-rip-flt</td>
<td></td>
</tr>
<tr>
<td>Step 5 set tag tag-value</td>
<td>Sets the tag value to associate with the redistributed routes.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-route-map)# set tag 4</td>
<td></td>
</tr>
</tbody>
</table>

Filtering IPv6 RIP Routing Updates

Route filtering using distribute lists provides control over the routes RIP receives and advertises. This control may be exercised globally or per interface.

Filtering is controlled by distribute lists. Input distribute lists control route reception, and input filtering is applied to advertisements received from neighbors. Only those routes that pass input filtering will be inserted in the RIP local routing table and become candidates for insertion into the IPv6 routing table.

Output distribute lists control route advertisement; Output filtering is applied to route advertisements sent to neighbors. Only those routes passing output filtering will be advertised.

Global distribute lists (which are distribute lists that do not apply to a specified interface) apply to all interfaces. If a distribute list specifies an interface, then that distribute list applies only to that interface.

An interface distribute list always takes precedence. For example, for a route received at an interface, with the interface filter set to deny, and the global filter set to permit, the route is blocked, the interface filter is passed, the global filter is blocked, and the route is passed.
IPv6 prefix lists are used to specify certain prefixes or a range of prefixes that must be matched before a permit or deny statement can be applied. Two operand keywords can be used to designate a range of prefix lengths to be matched. A prefix length of less than, or equal to, a value is configured with the \texttt{le} keyword. A prefix length greater than, or equal to, a value is specified using the \texttt{ge} keyword. The \texttt{ge} and \texttt{le} keywords can be used to specify the range of the prefix length to be matched in more detail than the usual \texttt{ipv6-prefix/prefix-length} argument. For a candidate prefix to match against a prefix list entry three conditions can exist:

- The candidate prefix must match the specified prefix list and prefix length entry.
- The value of the optional \texttt{le} keyword specifies the range of allowed prefix lengths from the prefix-length argument up to, and including, the value of the \texttt{le} keyword.
- The value of the optional \texttt{ge} keyword specifies the range of allowed prefix lengths from the value of the \texttt{ge} keyword up to, and including, 128.

\textbf{Note}  
Note that the first condition must match before the other conditions take effect.

An exact match is assumed when the \texttt{ge} or \texttt{le} keywords are not specified. If only one keyword operand is specified then the condition for that keyword is applied, and the other condition is not applied. The prefix-length value must be less than the \texttt{ge} value. The \texttt{ge} value must be less than, or equal to, the \texttt{le} value. The \texttt{le} value must be less than or equal to 128.

**SUMMARY STEPS**

1. \texttt{enable}
2. \texttt{configure terminal}
3. \texttt{ipv6 prefix list prefix-list-name seq seq-number} \{\texttt{deny ipv6-prefix/prefix-length} | \texttt{description text}\} \texttt{[ge ge-value] [le le-value]}
4. \texttt{ipv6 prefix list prefix-list-name seq seq-number} \{\texttt{deny ipv6-prefix/prefix-length} | \texttt{description text}\} \texttt{[ge ge-value] [le le-value]}
5. Repeat Steps 3 and 4 as many times as necessary to build the prefix list.
6. \texttt{ipv6 router rip name}
7. \texttt{distribute-list prefix-list prefix-list-name in | out} \{\texttt{interface-type interface-number}\}

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong> Device# configure terminal</td>
<td></td>
</tr>
</tbody>
</table>
## How to Implement RIP for IPv6

### Step 3
```
ipv6 prefix list prefix-list-name seq
    seq-number] (deny ipv6-prefix/prefix-length | description text) [ge ge-value] [le le-value]
```
Example:
```
Device(config)# ipv6 prefix-list abc permit 2001:DB8::/16
```
Creates an entry in the IPv6 prefix list.

### Step 4
```
ipv6 prefix list prefix-list-name seq
    seq-number] (deny ipv6-prefix/prefix-length | description text) [ge ge-value] [le le-value]
```
Example:
```
Device(config)# ipv6 prefix-list abc deny ::/0
```
Creates an entry in the IPv6 prefix list.

### Step 5
Repeat Steps 3 and 4 as many times as necessary to build the prefix list.

### Step 6
```
ipv6 router rip name
```
Example:
```
Device(config)# ipv6 router rip process1
```
Configures an IPv6 RIP routing process.

### Step 7
```
distribute-list prefix-list prefix-list-name in | out) [interface-type interface-number]
```
Example:
```
Device(config-rtr-rip)# distribute-list prefix-list process1 in gigabitethernet 0/0/0
```
Applies a prefix list to IPv6 RIP routing updates that are received or sent on an interface.

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 prefix list</td>
<td>Creates an entry in the IPv6 prefix list.</td>
</tr>
<tr>
<td>seq prefix-list-name seq-number</td>
<td></td>
</tr>
<tr>
<td>(deny ipv6-prefix/prefix-length</td>
<td></td>
</tr>
<tr>
<td>description text) [ge ge-value] [le le-value]</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# ipv6 prefix-list abc permit 2001:DB8::/16</td>
<td></td>
</tr>
<tr>
<td>ipv6 prefix list</td>
<td>Creates an entry in the IPv6 prefix list.</td>
</tr>
<tr>
<td>prefix-list-name seq-number</td>
<td></td>
</tr>
<tr>
<td>(deny ipv6-prefix/prefix-length</td>
<td></td>
</tr>
<tr>
<td>description text) [ge ge-value] [le le-value]</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# ipv6 prefix-list abc deny ::/0</td>
<td></td>
</tr>
<tr>
<td>Repeat Steps 3 and 4 as many times as necessary to build the prefix list.</td>
<td></td>
</tr>
<tr>
<td>ipv6 router rip name</td>
<td>Configures an IPv6 RIP routing process.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config)# ipv6 router rip process1</td>
<td></td>
</tr>
<tr>
<td>distribute-list prefix-list prefix-list-name in</td>
<td>Applies a prefix list to IPv6 RIP routing updates that are received or sent on an interface.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>out) [interface-type interface-number]</td>
<td></td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Device(config-rtr-rip)# distribute-list prefix-list process1 in gigabitethernet 0/0/0</td>
<td></td>
</tr>
</tbody>
</table>
Verifying IPv6 RIP Configuration and Operation

SUMMARY STEPS

1. `show ipv6 rip [name][database] next-hops`
2. `show ipv6 route [ipv6-address|ipv6-prefix/prefix-length|protocol | interface-type interface-number]`
3. `enable`
4. `debug ipv6 rip [interface-type interface-number]`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** `show ipv6 rip [name][database] next-hops` | (Optional) Displays information about current IPv6 RIP processes.  
- In this example, IPv6 RIP process database information is displayed for the specified IPv6 RIP process. |
| **Example:**  
Device> show ipv6 rip process1 database  
| **Step 2** `show ipv6 route [ipv6-address|ipv6-prefix/prefix-length|protocol | interface-type interface-number]` | (Optional) Displays the current contents of the IPv6 routing table.  
- In this example, only IPv6 RIP routes are displayed. |
| **Example:**  
Device> show ipv6 route rip  
| **Step 3** `enable` | Enables privileged EXEC mode.  
- Enter your password if prompted. |
| **Example:**  
Device> enable  
| **Step 4** `debug ipv6 rip [interface-type interface-number]` | (Optional) Displays debugging messages for IPv6 RIP routing transactions. |
| **Example:**  
Device# debug ipv6 rip 

Examples

Sample Output for the `show ipv6 rip` command

In the following example, output information about all current IPv6 RIP processes is displayed using the `show ipv6 rip` command:

```
Device> show ipv6 rip  
RIP process "process1", port 521, multicast-group FF02::9, pid 62  
  Administrative distance is 120. Maximum paths is 1  
  Updates every 5 seconds, expire after 15  
  Holddown lasts 10 seconds, garbage collect after 30  
  Split horizon is on; poison reverse is off  
  Default routes are generated  
  Periodic updates 223, trigger updates 1  
  Interfaces:  
    Gigabitethernet0/0/0
```
Chapter 45      Implementing RIP for IPv6

Verifying IPv6 RIP Configuration and Operation

Redistribution:
  Distributing protocol bgp 65001 route-map bgp-to-rip

In the following example, output information about a specified IPv6 RIP process database is displayed using the `show ipv6 rip` command with the name argument and the `database` keyword. In the following output for the IPv6 RIP process named process1, timer information is displayed, and route 2001::DB8::16/64 has a route tag set:

Device> show ipv6 rip process1 database
RIP process 'process1', local RIB
  2001:DB8::/64, metric 2
    Gigabitethernet0/0/0/FE80::A8BB:CCFF:FE00:B00, expires in 13 secs
  2001:DB8::/16, metric 2 tag 4, installed
    Gigabitethernet0/0/0/FE80::A8BB:CCFF:FE00:B00, expires in 13 secs
  2001:DB8::/2::/16, metric 2 tag 4, installed
    Gigabitethernet0/0/0/FE80::A8BB:CCFF:FE00:B00, expires in 13 secs
  ::/0, metric 2, installed
    Gigabitethernet0/0/0/FE80::A8BB:CCFF:FE00:B00, expires in 13 secs

In the following example, output information for a specified IPv6 RIP process is displayed using the `show ipv6 rip` user EXEC command with the name argument and the `next-hops` keyword:

Device> show ipv6 rip process1 next-hops
RIP process 'process1', Next Hops
  FE80::A8BB:CCFF:FE00:A00/Gigabitethernet0/0/0 [4 paths]

Sample Output for the show ipv6 route command

The current metric of the route can be found by entering the `show ipv6 route` command. In the following example, output information for all IPv6 RIP routes is displayed using the `show ipv6 route` command with the `rip` protocol keyword:

Device> show ipv6 route rip
IPv6 Routing Table - 17 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
      U - Per-user Static route
     I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
     O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
R  2001:DB8::/32 [120/2]
    via FE80::A8BB:CCFF:FE00:A00, gigabitethernet0/0/0
R  2001:DB8::/32 [120/2]
    via FE80::A8BB:CCFF:FE00:A00, gigabitethernet0/0/0
R  2001:DB8::/32 [120/2]
    via FE80::A8BB:CCFF:FE00:A00, gigabitethernet0/0/0

Sample Output for the debug ipv6 rip command

In the following example, debugging messages for IPv6 RIP routing transactions are displayed using the `debug ipv6 rip` command:

IN VRF-enabled mode
Device# debug ipv6 rip
Device# *Jul 20 22:11:13.877: RIPng ['vrf5']: a message has been received.
*Jul 20 22:11:13.877: RIPng [G10/11, "vrf5"]: response received from
  FE80::26B6:57FF:FEA4:964C for process '1'.
*Jul 20 22:11:13.877:      dst=FF02::9
Configuration Examples for IPv6 RIP

Configuring IPv6 RIP

In the following example, the IPv6 RIP process named process1 is enabled on the router and on Gigabit Ethernet interface 0/0/0. The IPv6 default route (::/0) is advertised in addition to all other routes in router updates sent on Gigabit Ethernet interface 0/0/0. Additionally, BGP routes are redistributed into the RIP process named process1 according to a route map where routes that match a prefix list are also tagged. The number of parallel paths is set to one to allow the route tagging, and the IPv6 RIP timers are adjusted. A prefix list named eth0/0-in-flt filters inbound routing updates on Gigabit Ethernet interface 0/0/0.

```
ipv6 router rip process1
maximum-paths 1
redistribute bgp 65001 route-map bgp-to-rip
distribute-list prefix-list eth0/0-in-flt in Gigabitethernet0/0/0
```
interface Gigabitethernet0/0/0
ipv6 address 2001:DB8::/64 eui-64
ipv6 rip process1 enable
ipv6 rip process1 default-information originate

ipv6 prefix-list bgp-to-rip-flt seq 10 deny 2001:DB8:3::/16 le 128
ipv6 prefix-list bgp-to-rip-flt seq 20 permit 2001:DB8:1::8 le 128

ipv6 prefix-list eth0/0-in-flt seq 10 deny ::/0
ipv6 prefix-list eth0/0-in-flt seq 15 permit ::/0 le 128

route-map bgp-to-rip permit 10
match ipv6 address prefix-list bgp-to-rip-flt
set tag 4

Configuring IPv6: RIPng VRF-Aware Support

Device> enable
Device# configure terminal
Device(config)# ipv6 unicast-routing
Device(config)# vrf definition vrf1
Device(config-vrf)# address-family ipv6
Device(config-vrf-af)# exit
Device(config-vrf)# exit
Device(config)# ipv6 rip vrf-mode enable
Device(config)# ipv6 router rip myrip
Device(config-rtz)# exit
Device(config)# interface Ethernet 0/0
Device(config-if)# vrf forwarding vrf1
Device(config-if)# ipv6 enable
Device(config-if)# ipv6 rip myrip enable
Device(config-if)# end
Additional References

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>RIP commands: complete command syntax, command mode, defaults, usage guidelines, and examples</td>
<td><em>Cisco IOS IP Routing Protocols Command Reference</em></td>
</tr>
<tr>
<td>IPv6 commands: complete command syntax, command mode, defaults, usage guidelines, and examples</td>
<td><em>Cisco IOS IPv6 Command Reference</em></td>
</tr>
</tbody>
</table>

Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.</td>
<td>—</td>
</tr>
</tbody>
</table>

MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>• None</td>
<td>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator available at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2080</td>
<td><em>RIPng for IPv6</em></td>
</tr>
</tbody>
</table>
## Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Feature Information for Implementing RIP for IPv6

Table 45-1 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 45-1 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| RIP for IPv6  | Cisco IOS XE 3.13.0S | RIP enhancements for IPv6 include support for IPv6 addresses and prefixes, and the use of the all-RIP-routers multicast group address FF02::9 as the destination address for RIP update messages. IPv6: RIPng VRF-Aware Support—When not virtual routing and forwarding (VRF) aware, IPv6 Routing Information Protocol (RIP), also known as RIP Next Generation (RIPng), works only with routes that are available in the default global routing table. When operating in VRF mode, RIPng, creates a separate routing table for each VRF instance. The IPv6: RIPng VRF-Aware Support feature enables the availability of separate routing tables for every provider edge-customer edge (PE-CE) scenario, thus allowing improved route protection, modularity, and a potential reduction in the size of the routing table. The following sections provide information about this feature:  
• IPv6: RIPng VRF-Aware Support, page 45-2  
• Nonstop Forwarding for IPv6 RIP, page 45-2  
• Redistributing Routes into an IPv6 RIP Routing Process, page 45-8  
The following commands were introduced or modified: debug ipv6 rip, ipv6 rip default-information, ipv6 rip enable, ipv6 router rip, ipv6 unicast-routing, maximum paths, distribute-list prefix-list (IPv6 RIP), ipv6 prefix-list, show ipv6 rip, timers (IPv6 RIP), ipv6 rip enable, match ipv6 address, redistribute, route-map, set tag, show ipv6 route, clear ipv6 rip, debug ipv6 rip ,ipv6 rip vrf-mode enable, and show ipv6 rip |
Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2014 Cisco Systems, Inc. All rights reserved.