Configuring RIP

6

This chapter describes how to configure the Routing Information Protocol (RIP) on your ERX system.

Overview

RIP is an interior gateway protocol (IGP) typically used in small, homogeneous networks. RIP uses distance-vector routing to route information through IP networks.

Distance-vector routing requires that each router simply inform its neighbors of its routing table. For each network path, the receiving router picks the neighbor advertising the lowest metric, then adds this entry into its routing table for readvertisement.

Any host that uses RIP is assumed to have interfaces to one or more networks. These networks are considered to be directly connected networks. RIP relies on access to certain information about each of these networks. The most important information is the network's metric.

RIP Metric

RIP uses the hop count as the metric (also known as cost) to compare the value of different routes. The hop count is the number of routers that data packets must traverse between RIP networks. Metrics range from 0 for a directly connected network to 16 for an unreachable network. This small range prevents RIP from being useful for large networks.

RIP Messages

RIP exchanges routing information via User Datagram Protocol (UDP) data packets. Each RIP router sends and receives datagrams on UDP port number 520, the RIP version 1/RIP version 2 port. All communications intended for another router's RIP process area are sent from the RIP port.

Every RIP message contains a RIP header that consists of a command and a version number. The system supports RIP version 1 and RIP version 2 extensions.

RIP employs the following message types:

- request – a request for the responding system to send all or part of its routing table.
- response – a message containing all or part of the sender's routing table. This message is sent in response to a request or is an unsolicited routing update generated by the sender.

The RIP request and response messages also contain a list of route entries. Each route entry contains the following:

- Address Entry Identifier – the type of address
- Destination IP address – the destination address of the message
- Cost to reach the destination – a value between 1 and 15, which specifies the current metric for reaching the destination

References

For more information about RIP, consult the following resources:

- RFC 1058 – Routing Information Protocol (June, 1998)
- RFC 2453 – RIP Version 2 (November, 1998)

Features

Some of the major RIP features supported by the system include:

RIP version 1	multicast addressing
RIP version 2	route summarization
route tags	split horizon
authentication	equal-cost multipath
subnet masks	remote neighbors
next hop	poison reverse

Route Tags

A route tag is a field in a RIP message that allows boundary routers in an autonomous system (AS) to exchange information about external routes. Route tags provide a method of separating internal RIP routes (routes within the RIP routing domain) from external RIP routes, which may have been imported from an EGP (exterior gateway protocol) or another IGP (interior gateway protocol).

Routers supporting protocols other than RIP should be configurable to allow the route tags to be configured for routes imported from different sources. For example, routes imported from BGP should be able to have their route tags set to the number of the ASs from which the routes were learned.

Authentication

RIP version 1 does not support authentication. If you are sending and receiving RIP version 2 packets, you can enable RIP authentication on an interface.

The system provides the Simple Authentication scheme for RIP-2. Since authentication is a per message function and only one 2-octet field is available in the RIP message header, authentication uses the space of an entire RIP message.

The first 20-byte entry in a RIP authentication message contains an address family identifier value of 0xffff and a route tag value of 2. If the 0xffff address family is present in the RIP message, the remaining 16 octets of the entry contain a plain text password. If the password is fewer than 16 octets, it must be left-justified and padded to the right with nulls (0x00).

Authentication is applied per RIP interface. You can specify either **text** or **MD5** authentication. Text authentication uses a simple password that

must be shared by the neighbors receiving updates or requests. If they do not have this password, the neighbors reject all updates or requests from the system. MD5 authentication uses a shared key to encrypt the RIP message. The neighbors must have the MD5 key to decrypt the message and encrypt a response.

Note: *Do not use text authentication when security is important, because the system sends the unencrypted password in every RIP packet it sends.*

Example 1 The following example shows how to use password authentication:

```
host1(config)#interface fastEthernet 0/0
host1(config-if)#ip rip send version 2
host1(config-if)#ip rip authentication mode text
host1(config-if)#ip rip authentication key ke6G72mV
```

Example 2 The following example shows how to use MD5 authentication:

```
host1(config)#interface fastEthernet 0/0
host1(config-if)#ip rip send version 2
host1(config-if)#ip rip authentication mode md5 8
host1(config-if)#ip rip authentication key sf43nBScE9
```

Subnet Masks

The Subnet Mask field of a RIP message contains the subnet mask that is applied to the IP address to set the nonhost portion of the address. If the subnet mask field in a RIP message contains a zero, then no subnet mask was included for the entry.

On an interface where a RIP-1 router may hear and operate on information in a RIP-2 routing entry, the following rules apply:

- Information internal to one network must never be advertised into another network.
- Information about a more specific subnet may not be advertised where RIP-1 routers would consider it a host route.
- Supernet routes (routes where a netmask is less specific than the natural network mask) must not be advertised where they could be misinterpreted by RIP-1 routers.

Next Hop

The Next Hop field in a RIP message contains the next IP address where a packet is sent. A value of zero in this field indicates that the next address the packet should be sent to is the system that originally sent the RIP message.

Multicasting

In order to reduce unnecessary load on hosts that are not listening to RIP-2 messages, an IP multicast address is used for periodic broadcast messages. The IP multicast address is 224.0.0.9.

Route Summaries

You can summarize routes reported by RIP to reduce the size of the routing table and the amount of traffic resulting from RIP updates. Configuring a RIP summary will cause that prefix to be advertised with the associated metric regardless of the presence of more-specific prefixes. Any more-specific prefixes will not be advertised when they are covered by the summary. You can choose the degree of summarization by using a prefix tree to specify the number of bits to report for routes matching a route map. Alternatively, you can explicitly specify routes for RIP to summarize.

Prefix Tree Example

The following example shows how to configure a 16-bit route summary:

1 Specify a route map for RIP in Router Configuration mode.

```
host1#configure t
Enter configuration commands, one per line. End with CNTL/Z.
host1(config)#router rip
host1(config-router)#route-map 1
host1(config-router)#exit
```

2 Define a route map associated with a prefix tree.

```
host1(config)#
host1(config)#route-map 1
host1(config-route-map)#match-set
host1(config-route-map)#match-set summary prefix-tree boston
host1(config-route-map)#exit
host1(config)#
```

3 Set the conditions for summarization in the prefix tree, including which routes are summarized and how many bits of the network addresses are preserved as the network prefix.

```
host1(config)#ip prefix-tree boston permit 2.1.0.0/16
```

This example summarizes routes for networks addressed by 2.1.*x*.*x*. The first 16 bits of the network address are preserved in the summary. For example, routes 2.1.3.0, 2.1.2.0, and 2.1.1.0 would all be summarized as 2.1.0.0.

Static Summary Example

You can use the **ip summary-address** command to specify routes that RIP will summarize.

ip summary-address

- Use to specify an IP address and network mask to identify which routes to summarize.
- You can optionally specify a metric associated with the summary address. The default metric is 1.
- Example

```
host1(config-router)#ip summary-address 4.4.0.0
  255.255.0.0 5
host1(config-router)#ip summary-address 4.3.0.0
  255.255.0.0 6
```

- Use the **no** version to stop summarization for the specified routes.

Split Horizon

Split horizon is a mechanism to aid in preventing routing loops when distance-vector routing protocols such as RIP are employed in broadcast networks. When split horizon is enabled, the router cannot advertise information about routes on an interface from which the information originates. Split horizon is enabled by default on the system.

You can disable split horizon and enable poison reverse routing updates that advertise routes originating on the interface, but for each of these routes the metric is set to infinity to explicitly advertise that these networks are not reachable.

Equal-Cost Multipath

RIP supports equal-cost multipath (ECMP) and installs into the routing table multiple entries for paths to the same destination. Each of these multiple paths to a given destination must have the same cost as the others, but a different next hop.

Before You Run RIP

At least one IP address must be configured on your system for RIP to run.

Configuration Tasks

To configure RIP:

1 Create a RIP process by enabling RIP.

```
host1(config)#router rip
```

2 (Optional) Configure the global RIP version. RIP version 1 is used by default.

```
host1(config-router)#version 2
```

3 (Optional) Do one of the following:

- Associate a network with a RIP routing process and optionally configure RIP for the network.

```
host1(config-router)#network 10.2.1.0 255.255.255.0
host1(config-if)#ip rip
host1(config-if)#ip rip receive version 1
host1(config-if)#ip rip send version 2
host1(config-if)#ip rip authentication mode text
host1(config-if)#ip rip authentication key klaatu42
```

- Associate the RIP routing process with an interface specified by an IP address or on an unnumbered interface and configure RIP for the interface).

```
host1(config-router)#address 10.2.1.1
host1(config-router)#address 10.2.1.1 receive version 1
host1(config-router)#address 10.2.1.1 send version 2
host1(config-router)#address 10.2.1.1 authentication mode
  text
host1(config-router)#address 10.2.1.1 authentication key
  31barada
```

Each configuration step is optional, and includes the following:

- (Optional) Specify a RIP receive version for an interface. By default, RIP interfaces on your system receive both RIP version 1 and RIP version 2.
- (Optional) Specify a RIP send version for an interface. By default, RIP interfaces on your system send only RIP version 1.
- (Optional) Specify an authentication mode and authentication password or key. This step is permitted only if both receive version and send version are set to RIP version 2.

4 (Optional) Enable RIP to advertise a default route.

```
host1(config-router)#default-information originate
```

5 (Optional) Specify a default metric for advertised routes.

```
host1(config-router)#default-metric 5
```

6 (Optional) Set the administrative distance for advertised routes.

```
host1(config-router)#distance 150
```

7 (Optional) Control the dynamic distribution of routes caused by changes to an associated route map.

```
host1(config-router)#disable-dynamic-redistribute
```

8 (Optional) Adjust RIP timers.

```
host1(config-router)#timers update 20
host1(config-router)#timers invalid 60
host1(config-router)#timers holddown 60
host1(config-router)#timers flush 90
```

9 (Optional) Specify maximum number of ECMP paths.

```
host1(config-router)#maximum-paths 2
```

10 (Optional) Summarize routes.

```
host1(config)#ip prefix-tree boston permit 10.10.2.0/24
host1(config-router)#route-map 4
host1(config-route-map)#match-set summary prefix-tree boston
```

11 (Optional) Redistribute routes from other protocols into RIP, or from RIP to other protocols.

```
host1(config-router)#redistribute rip 5
host1(config-router)#route-map 4
host1(config-router)#redistribute bgp 100 route-map 4
```

12 (Optional) Enable unicast communication with RIP neighbors.

```
host1(config-router)#neighbor 10.10.21.100
host1(config-router)#passive-interface atm atm 2/0.16
```

Relationship Between address and network Commands

If you use the **network** command to configure a RIP network, use the **ip rip** commands to configure the RIP attributes for that network. Do not use the **address** commands.

If you use the **address** command to configure a RIP network, use the **address** commands to configure the RIP attributes for that network. Do not use the **ip rip** commands.

Note: *The* ***network*** *and* ***ip rip*** *commands are maintained for industry compatibility. You can configure all your RIP interfaces with the* ***address*** *commands. You cannot configure unnumbered interfaces with the* ***network*** *and* ***ip rip*** *commands.*

address

- Use to configure RIP to run on the interface specified by the IP address or on an unnumbered interface. Use the **address** commands to configure RIP attributes on the network.
- Configures RIP with the default values: Send version is RIP version 1, receive version is RIP version 1 and version 2, authentication is not enabled.
- Example

```
host1(config-router)#address 10.2.1.1
```

- Use the **no** version to delete the RIP interface.

address authentication key

- Use to specify either the simple password for text authentication or the encryption/decryption key for MD5 authentication. The key is a string of up to 16 alphanumeric characters and can be mixed uppercase and lowercase.
- You can specify whether the key is entered in unencrypted or encrypted format. If you do not specify which, the string is assumed to be unencrypted.
- Example

```
host1(config-router)#address 10.2.1.1 authentication key
  ke6G72mV
```

- Use the **no** version to clear all authentication keys.

address authentication mode

- Use to specify the authentication mode.
- Specify **text** to send a simple text password to neighbors. If a neighbor does not have the same password, requests and updates from this system are rejected.
- Specify **md5** *keyID* to send an MD5 hash to neighbors. Neighbors must share the MD5 key to decrypt the message and encrypt the response.
- Example

```
host1(config-router)#address 10.2.1.1 authentication mode
  text
```

- Use the **no** version to remove authentication from all RIP interfaces.

address receive version

- Use to restrict the RIP version that the system can receive on an interface. The default is to receive both RIP version 1 and version 2.
- Example

```
host1(config-router)#address 10.2.1.1 receive version 1
```

- Use the **no** version to restore the default value, 1 2.

address send version

- Use to restrict the RIP version that the system can send on an interface. The default is to send only RIP version 1.
- Example

```
host1(config-router)#address 10.2.1.1 send version 2
```

- Use the **no** version to restore the default value, 1.

clear ip rip redistribution

- Use to clear all the routes that have previously been redistributed into RIP.
- Example

```
host1#clear ip rip redistribution
```

- There is no **no** version.

default-information originate

- Use to enable RIP to advertise a default route (0.0.0.0/0) if the default route exists in the IP routing table.
- If the default route does not exist, you must configure it using the **ip route** command, or specify the **always** keyword. The **always** keyword causes RIP to always advertise the default route, and creates it if it is not present in the IP routing table.
- Example

```
host1(config-router)#default-information originate
```

- Use the **no** version to disable advertisement of the default route.

default-metric

- Use to configure RIP to apply this metric when advertising routes on all subsequently created interfaces.
- Configuring a default metric lowers the priority of the routes.
- Use a metric from 1 to 16.
- Example

```
host1(config-router)#default-metric 5
```

- Use the **no** version to restore the default value, 0.

disable

- Use to disable RIP processing.
- Example

```
host1(config-router)#disable
```

- Use the **no** version to enable RIP processing.

disable-dynamic-redistribute

- Use to halt the dynamic redistribution of routes that are initiated by changes to a route map.
- Dynamic redistribution is enabled by default.
- Example

```
host1(config-router)#disable-dynamic-redistribute
```

- Use the **no** version to reenable dynamic redistribution.

distance

- Use to set the administrative distances for routes. The default is 120.
- Example

```
host1(config-router)#distance 150
```

- Use the **no** version to restore the default value, 120.

distribute-list

- Use to apply a specific access list to incoming or outgoing RIP route updates.
- An IP access list acts as a filter. Refer to the **access list** command in the *ERX Command Reference Guide* for more information.
- Example

```
host1(config-router)#distribute-list 5 incoming
```

- Use the **no** version to stop application of the distribute list.

ip prefix-tree

- Use to create a prefix tree to match routes to be summarized by a route map; specifies a tree entry—a deny or permit clause for a network address.
- The prefix tree name can be up to 32 characters long.
- Example

```
host1(config)#ip prefix-tree boston42 permit 10.10.2.0/24
```

- Use the **no** version to remove the specified prefix tree or the specified tree entry.

ip rip

- Use to configure RIP on the network interface specified with the **network** command.
- Configures RIP with the default values: Send version is RIP version 1, receive version is RIP version 1 and version 2, authentication is not enabled.
- Example

```
host1(config-if)#ip rip
```

- Use the **no** version to delete the RIP interface.

ip rip authentication key

- Use to specify either the simple password for text authentication or the encryption/decryption key for MD5 authentication. The key is a string of up to 16 alphanumeric characters and can be mixed uppercase and lowercase.
- You can specify whether the key is entered in unencrypted or encrypted format. If you do not specify which, the string is assumed to be unencrypted.
- Example

```
host1(config-if)#ip rip authentication key ke6G72mV
```

- Use the **no** version to clear all authentication keys.

ip rip authentication mode

- Use to specify the authentication mode.
- Specify **text** to send a simple text password to neighbors. If a neighbor does not have the same password, requests and updates from this system are rejected.
- Specify **md5** *keyID* to send an MD5 hash to neighbors. Neighbors must share the MD5 key to decrypt the message and encrypt the response.
- Example

```
host1(config-if)#ip rip authentication mode text
```

- Use the **no** version to remove authentication from all RIP interfaces.

ip rip receive version

- Use to restrict the RIP version that the system can receive on an interface. The default is both RIP version 1 and version 2.
- Example

```
host1(config-if)#ip rip receive version 1
```

- Use the **no** version to restore the default value, 1 2.

ip rip send version

- Use to restrict the RIP version that the system can send on an interface. The default is RIP version 1.
- Example

```
host1(config-if)#ip rip send version 2
```

- Use the **no** version to restore the default value, 1.

ip split-horizon

- Use to configure the split horizon feature and poison reverse features for the interface. Enabled by default, split horizon prevents the RIP router from advertising routes from the originating interface.
- Poison reverse routing updates are disabled by default; when enabled, they set the metric for routes originating on the interface to infinity, thus explicitly advertising that the network is not reachable. This helps to prevent routing loops.

- In most configurations, you will want to accept the default condition.
- Example

```
host1(config-if)#no ip split-horizon
```

- Use the **no** version to disable split horizon and enable poison reverse routing updates.

match-set summary prefix-tree

- Use to specify a prefix tree that summarizes routes for a particular route map.
- Use the **ip prefix-tree** command to set the conditions of the prefix tree, including which routes to summarize and how many bits of the network address to preserve.
- Example

```
host1(config-route-map)#match-set summary prefix-tree boston
```

- Use the **no** version to disable the use of the prefix tree by the route map.

maximum-paths

- Use to control the maximum number of parallel routes that RIP can support.
- RIP installs multiple equal-cost paths to a given destination only if each has a different next hop.
- The maximum number of routes can range from 1–16.
- Example

```
host1(config-router)#maximum-paths 2
```

- Use the **no** version to restore the default value, 4.

neighbor

- Use to specify a RIP neighbor to which the router sends unicast messages.
- You must also use the **passive-interface** command to specify the interface as passive, thereby restricting the interface to unicast RIP messages.
- Example

```
host1(config-router)#neighbor 10.10.21.100
```

- Use the **no** version to remove the neighbor.

network

- Use to associate a network with a RIP routing process. Use the **ip rip** commands to configure RIP attributes on the network.
- You supply a network mask to the new address so that RIP runs on that specific network.
- If you do not specify an interface's network, the network is not advertised in any RIP updates.
- You can specify either the standard subnet mask or the inverse subnet mask.
- Example 1

```
host1(config-router)#network 10.2.1.0 255.255.255.0
```

- Example 2

```
host1(config-router)#network 10.2.1.0 0.0.0.255
```

- Use the **no** version to disable RIP on the specified interface.

passive-interface

- Use to disable the transmission of multicast RIP messages on the interface.
- RIP messages are unicast to a RIP neighbor on the interface if the interface is present in the IP routing table as the next-hop interface to the configured neighbor.
- Example

```
host1(config-router)#passive-interface atm atm 2/0.16
```

- Use the **no** version to reenable the transmission of RIP multicast messages on the specified interface.

redistribute

- Use to redistribute information from a routing domain other than RIP into the RIP domain.
- Example 1

```
host1(config)#router rip 5
host1(config-router)#redistribute bgp 100 route-map 4
```

- Specify the source protocol from which routes are being redistributed. It can be one of the following keywords: **bgp**, **isis**, **ospf**, **static** [**ip**], and **connected**. Use the **static** keyword to redistribute IP static routes; optionally add the **ip** keyword when redistributing into IS-IS. The keyword **connected** refers to routes that are established automatically by virtue of having enabled IP on an interface. For routing protocols such as OSPF and IS-IS, these routes will be redistributed as external to the AS.
- Use the **route-map** keyword to interrogate the route map to filter the importation of routes from the source routing protocol to the current routing protocol. If you do not specify the route-map option, all routes are redistributed. If you specify the route-map option, but no route map tags are listed, no routes will be imported.
- Use to redistribute routes from RIP into other non-RIP routing domains.
- Example 2

```
host1(config)#router bgp 100
host1(config-router)#redistribute rip 5
```

- Use the **no** version to disable redistribution.

route-map

- Use to specify a route map for RIP.
- Example

```
host1(config)#router rip
host1(config-router)#route-map 4
```

- Use the **no** version to delete the route map. If you do not specify an interface, it removes the global route map if it exists.

router rip

- Use to enable RIP routing protocol and specify a RIP process for IP, or to access Router Configuration mode.
- Specify only one RIP process per router.
- Example

```
host1(config)#router rip
```

- Use the **no** version to delete the RIP process and removes the configuration from your system.

timers

- Use to configure RIP timers.
- The system supports the following RIP timers:
  - update – the interval in seconds at which routing updates are sent. The default is 30 seconds.
  - invalid – the interval in seconds after which a route is declared invalid (null). Set this value to at least three times the update value. The default is 180 seconds.
  - holddown – the interval in seconds during which routing information about better paths is suppressed. Set this value to at least three times the update value. The default is 120 seconds.
  - flush – the interval in seconds that must pass before a route is removed from the routing table. Set this value greater than the invalid value. The default is 300 seconds.
- Examples

```
host1(config-router)#timers update 20
host1(config-router)#timers invalid 60
host1(config-router)#timers holddown 60
host1(config-router)#timers flush 90
```

- Use the **no** version to restore the default values, 30 180 120 300.

version

- Use to specify the global RIP version. The default is RIP version 1.
- To change the RIP version on a specific interface, use the **ip rip receive version** and the **ip rip send version** commands, or the **address receive version** and **address send version** commands.
- Example

```
host1(config-router)#version 2
```

- Use the **no** version to revert to the default value, 1.

Using RIP Routes for Multicast RPF Checks

You can use the **ip route-type** command to specify whether RIP routes are available for only unicast forwarding protocols or only multicast reverse path forwarding (RPF) checks. Routes available for unicast forwarding appear in the unicast view of the routing table, whereas routes available for multicast RPF checks appear in the multicast view of the routing table.

ip route-type

- Use to specify whether RIP routes are available only for unicast forwarding, only for multicast reverse path forwarding checks, or for both.
- Use the **show ip route** command to view the routes available for unicast forwarding.
- Use the **show ip rpf-routes** command to view the routes available for multicast reverse path forwarding checks.
- By default, RIP routes are available for both unicast forwarding and multicast reverse path forwarding checks.
- Example

```
host1(config)#router rip

host1(config-router)#ip route-type unicast
```

- Use the **no** version to restore the default value, both.

Remote Neighbors

You can create RIP remote neighbors to enable the router to establish neighbor adjacencies through unidirectional interfaces, such as MPLS tunnels, rather than the standard practice of using the same interface for receipt and transmission of RIP packets. The remote neighbor can be more than one hop away through intermediate routes that are not running RIP. RIP uses the interface associated with the best route to the remote neighbor to reach the neighbor. A best route to the neighbor must exist in the IP routing table.

You must explicitly configure remote neighbors on the RIP routers to specify the remote neighbor with which the router will form an adjacency and the source IP address the router will use for RIP packets destined to its peer remote neighbor.

In order to form an adjacency with its remote neighbor, the system sends all RIP packets to the remote neighbor as unicast packets with the destination IP address equal to the source IP address of the remote neighbor. The loopback interface associated with the source IP address for the remote neighbor acts as a logical RIP interface for the neighbor.

To prevent routing loops, you can disable split horizon and enable poison reverse routing updates.

The **remote-neighbor** command to specify the remote neighbors is mandatory. Configuration of all other remote-neighbor attributes is optional.

authentication key

- Use to specify the password for text authentication and the key for MD5 authentication for RIP remote-neighbor interface.
- This command is supported only in RIP version 2. Authentication is disabled by default.
- Example

```
host1(config-router-rn)#authentication key 0 jun27ior
```

- Use the **no** version to clear the key for the remote-neighbor interface.

authentication mode

- Use to specify the authentication mode for the remote neighbor interface.
- Specify **text** to send a simple text password to remote neighbors. If a remote neighbor does not have the same password, requests and updates from this system are rejected.
- Specify **md5** *keyID* to send an MD5 hash to remote neighbors. Remote neighbors must share the MD5 key to decrypt the message and encrypt the response.
- This command is supported only in RIP version 2. Authentication is disabled by default.
- Example

```
host1(config-router-rn)#authentication mode text
```

- Use the **no** version to remove authentication from the RIP remote-neighbor interface.

distribute-list

- Use to apply a specific access list to either incoming or outgoing RIP route updates on the RIP remote-neighbor interface.
- An IP access list acts as a filter. Refer to the **access list** command in the *ERX Command Reference Guide* for more information.
- Example

```
host1(config)#distribute-list 5 in
```

- Use the **no** version to stop application of the distribute list.

exit-remote-neighbor

- Use to exit from the Remote Neighbor Configuration mode and return to Router Configuration mode.
- Example

```
host1(config-router-rn)#exit-remote-neighbor
```

- There is no **no** version.

receive version

- Use to restrict the RIP version that the system can receive on a RIP remote-neighbor interface. The default is to receive both RIP version 1 and version 2.
- The **off** keyword overrides any other specified option; for example, configuring both **1** and **off** or both **2** and **off** has the same result as configuring only **off**.
- Example

```
host1(config-router-rn)#receive version 1
```

- Use the **no** version to restore the default value, 1 2.

remote-neighbor

- Use to configure a RIP remote neighbor.
- Example

```
host1(config-router)#remote-neighbor 10.25.100.14
```

- Use the **no** version to remove the remote neighbor and any attributes configured for the remote neighbor.

send version

- Use to restrict the RIP version that the system can send on an interface. The default is to send only RIP version 1.
- Example

```
host1(config-router-rn)#send version 1
```

- Use the **no** version to restore the default value, 1.

split-horizon

- Use to configure the split horizon and poison reverse features for RIP remote neighbors.
- Split horizon is enabled by default; poison reverse routing updates are disabled by default.
- Poison reverse routing updates set the metric for routes originating on the interface to infinity, thus explicitly advertising that the network is not reachable. This helps to prevent routing loops.
- Example

```
host1(config-router-rn)#no split-horizon
```

- Use the **no** version to disable the split horizon and enable poison reverse routing updates.

time-to-live

- Use to configure a hop count by setting the value of the time-to-live field used by packets sent to a RIP remote neighbor.
- Example

```
host1(config-router-rn)#time-to-live 12
```

- Use the **no** version to restore the default value, 16.

update-source

- Use to specify the RIP interface whose local address is used as the source address for the RIP connection to a remote neighbor.
- The source address assigned to a remote neighbor must be unique. If you configure a RIP router to form neighbor adjacencies with two RIP remote neighbors, then the RIP router must have two unique local source IP addresses one for each of its remote neighbors.
- Example

```
host1(config-router-rn)#update-source atm 2/0.17
```

- Use the **no** version to delete the source address from the connection to the remote neighbor.

Monitoring RIP

Two sets of commands enable you to monitor RIP operation on your system: the **debug** and the **show** commands. Both sets of commands provide information about your system's RIP state and configuration.

The task you are performing with each of these monitoring commands is basically the same for each command; that is, you are requesting information. The results of this request may vary. For instance, the **debug** commands provide information about problems with the network or the system, whereas the **show** commands provide information about the actual state and configuration of your system.

debug Commands

The **debug** commands provide information about the following RIP items:

- General events, such as creating a RIP process or removing RIP from an interface
- Routing events, such as when two RIP routers exchange routes

debug ip rip

- Use to display information on selected RIP events. This command has many keywords that allow you to specify a variety of RIP events.
- You can set the level of severity for the events you want displayed; specify the desired descriptive term or a corresponding number (0–7).
- You can set the verbosity of the messages you want displayed: low, medium, high.
- Example

```
host1#debug ip rip events
```

- Use the **no** version to cancel the display of any information on the designated variable.

undebug ip rip

- Use to cancel the display of information on a selected event.
- The same RIP variables can be designated as in the **debug ip rip** command.
- Example

```
host1#undebug ip rip events
```

- There is no **no** version.

show Commands

Use the **show** commands to monitor the following types of RIP information:

- Configuration
- IP-related
- Global counters
- Counters for a specified network
- Statistics

You can set a statistics baseline for RIP interfaces using the **baseline ip rip** command.

You can specify a VRF instance for the **show ip rip** commands. You can use the output filtering feature of the **show** command to include or exclude lines of output based on a text string you specify. Refer to *ERX System Basics Configuration Guide, Chapter 2, Command Line Interface*, for details.

baseline ip rip

- Sets a statistics baseline for RIP interfaces.
- The system implements the baseline by reading and storing the statistics at the time the baseline is set and then subtracting this baseline whenever baseline-relative statistics are retrieved.
- Use the optional **delta** keyword with the **show ip rip statistics** command to specify that baselined statistics are to be shown.
- Example

```
host1#baseline ip rip
```

- There is no **no** version.

show ip rip

- Use to display RIP information.
- Specify **vrf** *vrfName* to limit the display to a specific VRF.
- Use the **ifconfig** keyword to display address and interface configuration information instead of the default operational data.
- Field descriptions
  - Router Administrative State – displays the RIP state. Enable means the system is allowed to send and receive updates. Disable means that RIP might be configured but it is NOT allowed to run yet.
  - System version RIP1 – RIP versions allowed for sending and receiving RIP updates. The system version is currently set to RIP1, which sends RIP version 1 but will receive version 1 or 2. If it is set to RIP2, it will send version 2 and receive version 2 only. The default is configured for RIP1.
  - Incoming filters – access list applied to incoming route updates
  - Outgoing filters – access list applied to outgoing route updates
  - Global route map – route map that specifies all RIP interfaces on the system
  - Default metric – value for redistributed routes. The default is 1. This global value is superseded by metrics applied to a RIP interface.
  - Distance – value added to RIP routes added to the IP routing table. The default is 120.
  - Number of route changes – number of times the router has been told to route changes by its peers
  - Number of route queries – number of times the router has received route requests from other routers
  - Update interval – current setting of the update timer (in seconds)
  - Invalid interval – current setting of the invalid timer (in seconds)
  - Hold down time – current setting of the hold-down timer (in seconds)
  - Flush interval – current setting of the flush timer (in seconds)
  - Network – IP address of a network on which RIP is running
  - Netmask – network mask applied to the network address
  - Address/status/interface – values listed for each network that is running RIP
  - Send version – version of RIP used for sending updates
  - Receive version – version of RIP accepted in received updates

- Authentication mode – password or MD5 authentication, or none
- Default metric – metric value applied to the RIP interface. The default is 1.
- Outgoing access-list – name of the access list applied to outgoing routes
- Incoming access-list – name of the access list applied to incoming routes
- Outgoing route-map – name of the route map applied to outgoing routes
- Received bad packets – number of bad packets received
- Received bad routes – number of bad routes received
- Triggered updates sent – number of triggered updates sent; triggered updates are sent before the entire RIP routing table is sent; triggered by events such as adding a new RIP route or redistribution
- Received updates – number of updates received

• Example

```
host1#show ip rip
Routing Information Protocol
 Router Administrative State = enable
 System version RIP1: send = 1, receive = 1 or 2
 No filter is applied to outgoing route update for all
  interfaces
 No filter is applied to incoming route update for all
  interfaces
 No global route map
 Default metric = 1
 Distance = 120
 Number of route changes = 0
 Number of route queries = 0
 Update interval = 30 (secs)
 Invalid interval = 180 (secs)
 Hold down time = 120 (secs)
 Flush interval = 300 (secs)
 Network         netmask
 10.2.1.0        255.255.255.0
10.2.1.32, Rip is up, fastEthernet0/0
  Send version = 1
  Receive version = 1,2
  Authentication mode = none
  Default metric = 1
  Access-list applied to outgoing route = none
  Access-list applied to incoming route = none
  Route-map applied to outgoing route = none
  Received bad packet = 0
  Received bad routes = 0
  Triggered updates sent = 0
  Received updates = 0
170.100.100.2, Rip is up, fastEthernet0/0
  Send version = 1
  Receive version = 1,2
```

```
Authentication mode = none
Default metric = 1
Access-list applied to outgoing route = none
Access-list applied to incoming route = none
Route-map applied to outgoing route = none
Received bad packet = 0
Received bad routes = 0
Triggered updates sent = 0
Received updates = 0
```

show ip rip brief

- Use to display limited RIP information.
- Specify **vrf** *vrfName* to limit the display to a specific VRF.
- Field descriptions
  - IP Address – IP address of the interface where RIP is running
  - Tx – transmit version of RIP on this interface, which can override the system configuration
  - Rx – receive version of RIP on this interface
  - Auth – type of authentication, password (text) or MD5
  - Met – current value is the same as the system one (the default metric). Based on MIB 2 for RIP, the interface's route metric can be set individually.
  - AccList O/I – access list applied to outgoing/incoming RIP route updates
  - RtMap – identifier for the route map that specifies a summary of RIP routes
  - Status – status of RIP, either up or down
  - Intf – interface type on which RIP is running
- Example

```
host1#show ip rip brief
IP Address Tx  Rx  Auth  Met AccList O/I  RtMap Status Intf
10.2.1.32  1   1,2 none  1   no/no   no   up    fastEthernet0/0
10.10.1.2  1   1,2 none  1   no/no   no   up    serial5:1/1:1
```

show ip rip database

- Use to display the route entries in the RIP routing table.
- Specify **vrf** *vrfName* to limit the display to a specific VRF.
- Specify the **active** keyword to limit the display to active routes learned via RIP updates.
- Specify the **inactive** keyword to limit the display to routes that the system will discard in the immediate future.
- Field descriptions
  - Prefix – IP address prefix
  - Length – prefix length
  - ttl – (time to live) indicates how many seconds the specific route remains in the routing table. If an entry reaches 0, it is removed from the routing table.

- Met – metric that RIP uses to rate the value of different routes (hop count). The hop count is the number of routers that can be traversed in a route.
- Next Hop – next IP address where a packet is sent. A value of zero in this field indicates that the next address the packet should be sent to is the system that originally sent the RIP message.
- Intf – interface that the route has learned

- Example

```
host1#show ip rip database
Prefix/Length: ttl  Met:  Next Hop       Intf:
3.0.0.0/8      0    1     72.30.100.2    tm2/1.100
9.20.0.0/17    0    2     172.30.100.1   tm2/1.100
10.2.1.0/24    0    2     172.30.100.1   tm2/1.100
```

show ip rip network

- Use to display the networks associated with the RIP routing process.
- Specify **vrf** *vrfName* to limit the display to a specific VRF.
- Field descriptions
  - network – IP address of a network on which RIP protocol is running
  - netmask – network mask applied to the network address
- Example

```
host1#show ip rip network
Network         netmask
 10.2.1.0       255.255.255.0
 172.30.100.0   255.255.255.0
 172.30.200.0   255.255.255.0
```

show ip rip peer

- Use to display limited information about each RIP neighbor.
- Specify **vrf** *vrfName* to limit the display to a specific VRF.
- Field descriptions
  - Time since last update received – time in seconds since an update was received from this peer
  - Peer version – version of IS-IS running on the peer
  - Bad packets received – number of bad packets received from the peer
  - Bad routes received – number of bad routes received from the peer
- Example

```
host1#show ip rip peer
192.168.1.102
   Time since last update received = 24
   Peer version = 1
   Bad packet received = 0
   Bad routes received = 0
192.168.1.151
   Time since last update received = 24
```

```
   Peer version = 1
   Bad packet received = 0
   Bad routes received = 0
192.168.1.158
  Time since last update received = 15
   Peer version = 1
   Bad packet received = 0
   Bad routes received = 0
192.168.1.250
   Time since last update received = 7
   Peer version = 2
   Bad packet received = 0
   Bad routes received = 0
```

show ip rip statistics

- Use to display global and session statistics counters for RIP. If you specify an IP address, statistics for that interface are displayed in addition to the global RIP statistics.
- Specify **vrf** *vrfName* to limit the display to a specific VRF.
- Use the optional **delta** keyword to specify that baselined statistics are to be shown. You must use the **baseline ip rip** command to set a baseline.
- Field descriptions
  - Number of route changes – number of times the router has been told to route changes by its peers
  - Number of route queries – number of times the router has received route requests from other routers
  - Received bad packets – number of bad packets received from the peer
  - Received bad routes – number of bad routes received from the peer
  - Triggered updates sent – number of triggered updates sent; triggered updates are sent before the entire RIP routing table is sent; triggered by events such as adding a new RIP route or redistribution
  - Received updates – number of updates received
- Example

```
host1#show ip rip statistics
 Number of route changes = 23
 Number of route queries = 0
```

- Example

```
host1#show ip rip statistics 10.2.1.32
Number of route changes = 901
Number of route queries = 0

fastEthernet 0/0, 10.2.1.32
  Received bad packet = 0
  Received bad routes = 0
  Triggered updates sent = 2
  Received updates = 41
```

show ip rip summary-address

- Use to display the specified summary address or all summary addresses for RIP.
- Field descriptions
  - Summary Address – address summarizing RIP routes
  - Mask – network mask specified in the **ip summary-address** command to identify which routes to summarize
  - Metric – metric advertised with the summary RIP prefix
- Example

```
host1#show ip rip summary-address
Summary Address Mask            Metric
 4.3.0.0        255.255.0.0     3
 4.4.0.0        255.255.0.0     5
```