

Configuring RIP

This chapter describes how to configure RIP. For a complete description of the RIP commands that appear in this chapter, refer to the “RIP Commands” chapter of the *Network Protocols Command Reference, Part 1*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

The Routing Information Protocol (RIP) is a relatively old, but still commonly used, interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a classical distance-vector routing protocol. RIP is documented in RFC 1058.

RIP uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The Cisco IOS software sends routing information updates every 30 seconds; this process is termed *advertising*. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by the nonupdating router as being unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the nonupdating router.

The metric that RIP uses to rate the value of different routes is *hop count*. The hop count is the number of routers that can be traversed in a route. A directly connected network has a metric of zero; an unreachable network has a metric of 16. This small range of metrics makes RIP an unsuitable routing protocol for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The network 0.0.0.0 does not exist; RIP treats 0.0.0.0 as a network to implement the default routing feature. The Cisco IOS software will advertise the default network if a default was learned by RIP, or if the router has a gateway of last resort and RIP is configured with a default metric.

RIP sends updates to the interfaces in the specified networks. If an interface’s network is not specified, it will not be advertised in any RIP update.

Cisco’s implementation of RIP Version 2 supports plain text and MD5 authentication, route summarization, classless interdomain routing (CIDR), and variable-length subnet masks (VLSMs).

For protocol-independent features, which also apply to RIP, see the chapter “Configuring IP Routing Protocol-Independent Features” in this document.

RIP Configuration Task List

To configure RIP, complete the tasks in the following sections. You must enable RIP. The remaining tasks are optional.

- Enable RIP
- Allow Unicast Updates for RIP

- Apply Offsets to Routing Metrics
- Adjust Timers
- Specify a RIP Version
- Enable RIP Authentication
- Disable Route Summarization
- Run IGRP and RIP Concurrently
- Disable the Validation of Source IP Addresses
- Enable or Disable Split Horizon
- Configure Interpacket Delay

For information about the following topics, see the “Configuring IP Routing Protocol-Independent Features” chapter:

- Filtering RIP information
- Key management (available in RIP Version 2)
- VLSM

Enable RIP

To enable RIP, perform the following tasks, starting in global configuration mode:

Task	Command
Step 1 Enable a RIP routing process, which places you in router configuration mode.	router rip
Step 2 Associate a network with a RIP routing process.	network <i>network-number</i>

Allow Unicast Updates for RIP

Because RIP is normally a broadcast protocol, in order for RIP routing updates to reach nonbroadcast networks, you must configure the Cisco IOS software to permit this exchange of routing information. To do so, perform the following task in router configuration mode:

Task	Command
Define a neighboring router with which to exchange routing information.	neighbor <i>ip-address</i>

To control the set of interfaces with which you want to exchange routing updates, you can disable the sending of routing updates on specified interfaces by configuring the **passive-interface** command. See the discussion on filtering in the “Filter Routing Information” section in the “Configuring IP Routing Protocol-Independent Features” chapter.

Apply Offsets to Routing Metrics

An offset list is the mechanism for increasing incoming and outgoing metrics to routes learned via RIP. This is done to provide a local mechanism for increasing the value of routing metrics. Optionally, you can limit the offset list with either an access list or an interface. To increase the value of routing metrics, perform the following task in router configuration mode:

Task	Command
Apply an offset to routing metrics.	offset-list [<i>access-list-number</i> <i>name</i>] { in out } <i>offset</i> [<i>type number</i>]

Adjust Timers

Routing protocols use several timers that determine such variables as the frequency of routing updates, the length of time before a route becomes invalid, and other parameters. You can adjust these timers to tune routing protocol performance to better suit your internetwork needs. You can make the following timer adjustments:

- The rate (time in seconds between updates) at which routing updates are sent
- The interval of time (in seconds) after which a route is declared invalid
- The interval (in seconds) during which routing information regarding better paths is suppressed
- The amount of time (in seconds) that must pass before a route is removed from the routing table
- The amount of time for which routing updates will be postponed

It also is possible to tune the IP routing support in the software to enable faster convergence of the various IP routing algorithms, and, hence, quicker fallback to redundant routers. The total effect is to minimize disruptions to end users of the network in situations where quick recovery is essential.

To adjust the timers, perform the following task in router configuration mode:

Task	Command
Adjust routing protocol timers.	timers basic <i>update invalid holddown flush</i> [<i>sleeptime</i>]

Specify a RIP Version

Cisco's implementation of RIP Version 2 supports authentication, key management, route summarization, classless interdomain routing (CIDR), and variable-length subnet masks (VLSMs). Key management and VLSM are described in the chapter "Configuring IP Routing Protocol-Independent Features."

By default, the software receives RIP Version 1 and Version 2 packets, but sends only Version 1 packets. You can configure the software to receive and send only Version 1 packets. Alternatively, you can configure the software to receive and send only Version 2 packets. To do so, perform the following task in router configuration mode:

Task	Command
Configure the software to receive and send only RIP Version 1 or only RIP Version 2 packets.	version { 1 2 }

The preceding task controls the default behavior of RIP. You can override that behavior by configuring a particular interface to behave differently. To control which RIP version an interface sends, perform one of the following tasks in interface configuration mode:

Task	Command
Configure an interface to send only RIP Version 1 packets.	ip rip send version 1
Configure an interface to send only RIP Version 2 packets.	ip rip send version 2
Configure an interface to send RIP Version 1 and Version 2 packets.	ip rip send version 1 2

Similarly, to control how packets received from an interface are processed, perform one of the following tasks in interface configuration mode:

Task	Command
Configure an interface to accept only RIP Version 1 packets.	ip rip receive version 1
Configure an interface to accept only RIP Version 2 packets.	ip rip receive version 2
Configure an interface to accept either RIP Version 1 or 2 packets.	ip rip receive version 1 2

Enable RIP 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 key chain determines the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed on that interface, not even the default authentication. Therefore, you must also perform the tasks in the section “Manage Authentication Keys” in the “Configuring IP Routing Protocol-Independent Features” chapter.

We support two modes of authentication on an interface for which RIP authentication is enabled: plain text authentication and MD5 authentication. The default authentication in every RIP Version 2 packet is plain text authentication.

Note Do not use plain text authentication in RIP packets for security purposes, because the unencrypted authentication key is sent in every RIP Version 2 packet. Use plain text authentication when security is not an issue, for example, to ensure that misconfigured hosts do not participate in routing.

To configure RIP authentication, perform the following tasks in interface configuration mode:

Task	Command
Step 1 Enable RIP authentication.	ip rip authentication key-chain <i>name-of-chain</i>
Step 2 Configure the interface to use MD5 digest authentication (or let it default to plain text authentication).	ip rip authentication mode { text md5 }

Task	Command
Step 3 Perform the authentication key management tasks.	See the section “Manage Authentication Keys” in the “Configuring IP Routing Protocol-Independent Features” chapter.

See the “Key Management Examples” section of the “Configuring IP Routing Protocol-Independent Features” chapter for key management examples.

Disable Route Summarization

RIP Version 2 supports automatic route summarization by default. The software summarizes subprefixes to the classful network boundary when crossing classful network boundaries.

If you have disconnected subnets, disable automatic route summarization to advertise the subnets. When route summarization is disabled, the software transmits subnet and host routing information across classful network boundaries. To disable automatic summarization, perform the following task in router configuration mode:

Task	Command
Disable automatic summarization.	no auto-summary

Run IGRP and RIP Concurrently

It is possible to run IGRP and RIP concurrently. The IGRP information will override the RIP information by default because of IGRP’s administrative distance.

However, running IGRP and RIP concurrently does not work well when the network topology changes. Because IGRP and RIP have different update timers, and because they require different amounts of time to propagate routing updates, one part of the network will end up believing IGRP routes and another part will end up believing RIP routes. This will result in routing loops. Even though these loops do not exist for very long, the time to live (TTL) will quickly reach zero, and ICMP will send a “TTL exceeded” message. This message will cause most applications to stop attempting network connections.

Disable the Validation of Source IP Addresses

By default, the software validates the source IP address of incoming RIP routing updates. If that source address is not valid, the software discards the routing update.

You might want to disable this feature if you have a router that is “off network” and you want to receive its updates. However, disabling this feature is not recommended under normal circumstances. To disable the default function that validates the source IP addresses of incoming routing updates, perform the following task in router configuration mode:

Task	Command
Disable the validation of the source IP address of incoming RIP routing updates.	no validate-update-source

Enable or Disable Split Horizon

Normally, routers that are connected to broadcast-type IP networks and that use distance-vector routing protocols employ the *split horizon* mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router out of any interface from which that information originated. This behavior usually optimizes communications among multiple routers, particularly when links are broken. However, with nonbroadcast networks (such as Frame Relay and SMDS), situations can arise for which this behavior is less than ideal. For these situations, you might want to disable split horizon. This applies to IGRP and RIP.

If an interface is configured with secondary IP addresses and split horizon is enabled, updates might not be sourced by every secondary address. One routing update is sourced per network number unless split horizon is disabled.

To enable or disable split horizon, perform the following tasks in interface configuration mode:

Task	Command
Enable split horizon.	ip split-horizon
Disable split horizon.	no ip split-horizon

Split horizon for Frame Relay and SMDS encapsulation is disabled by default. Split horizon is not disabled by default for interfaces using any of the X.25 encapsulations. For all other encapsulations, split horizon is enabled by default.

See the “Split Horizon Examples” section at the end of this chapter for examples of using split horizon.

Note In general, changing the state of the default is not recommended unless you are certain that your application requires making a change in order to advertise routes properly. Remember: If split horizon is disabled on a serial interface (and that interface is attached to a packet-switched network), you *must* disable split horizon for all routers in any relevant multicast groups on that network.

Configure Interpacket Delay

By default, the software adds no delay between packets in a multiple-packet RIP update being sent. If you have a high-end router sending to a low-speed router, you might want to add such interpacket delay to RIP updates, in the range of 8 to 50 milliseconds. To do so, perform the following task in router configuration mode:

Task	Command
Add interpacket delay for RIP updates sent.	output-delay <i>delay</i>

RIP Configuration Examples

This section contains the following RIP configuration examples:

- Split Horizon Examples

Split Horizon Examples

Two examples of configuring split horizon are provided.

Example 1

The following sample configuration illustrates a simple example of disabling split horizon on a serial link. In this example, the serial link is connected to an X.25 network.

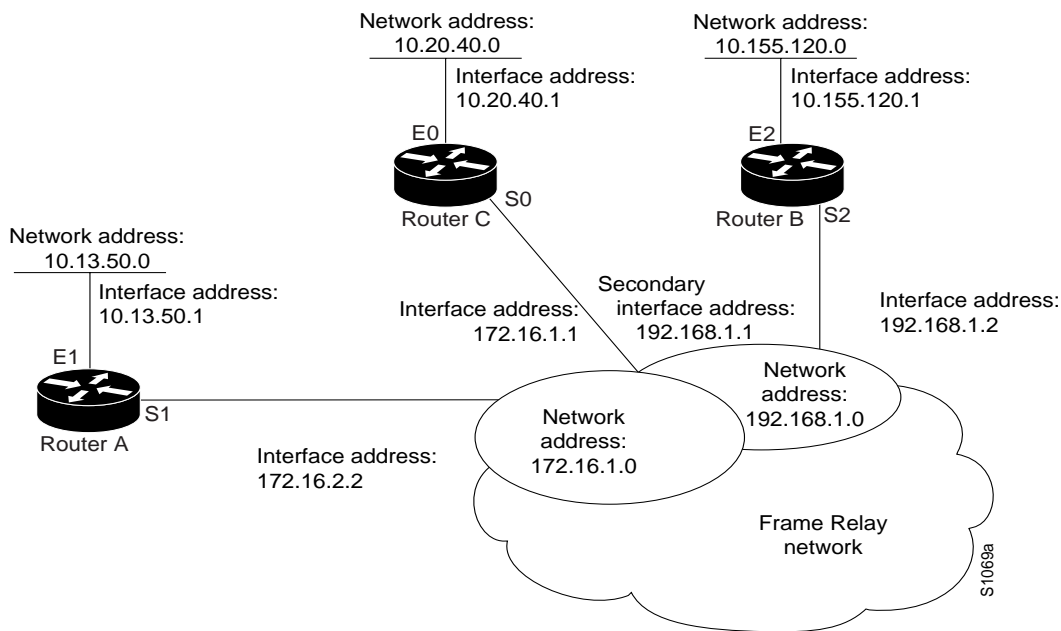
```
interface serial 0
encapsulation x25
no ip split-horizon
```

Example 2

In the next example, Figure 16 illustrates a typical situation in which the **no ip split-horizon** interface configuration command would be useful. This figure depicts two IP subnets that are both accessible via a serial interface on Router C (connected to Frame Relay network). In this example, the serial interface on Router C accommodates one of the subnets via the assignment of a secondary IP address.

The Ethernet interfaces for Router A, Router B, and Router C (connected to IP networks 12.13.50.0, 10.20.40.0, and 20.155.120.0) all have split horizon *enabled* by default, while the serial interfaces connected to networks 128.125.1.0 and 131.108.1.0 all have split horizon *disabled* by default. The partial interface configuration specifications for each router that follow Figure 16 illustrate that the **ip split-horizon** command is *not* explicitly configured under normal conditions for any of the interfaces.

Figure 16 Disabled Split Horizon Example for Frame Relay Network



In this example, split horizon must be disabled in order for network 128.125.0.0 to be advertised into network 131.108.0.0, and vice versa. These subnets overlap at Router C, interface S0. If split horizon were enabled on serial interface S0, it would not advertise a route back into the Frame Relay network for either of these networks.

Configuration for Router A

```
interface ethernet 1
 ip address 12.13.50.1
!
interface serial 1
 ip address 128.125.1.2
 encapsulation frame-relay
 no ip split-horizon
```

Configuration for Router B

```
interface ethernet 2
 ip address 20.155.120.1
!
interface serial 2
 ip address 131.108.1.2
 encapsulation frame-relay
 no ip split-horizon
```

Configuration for Router C

```
interface ethernet 0
 ip address 10.20.40.1
!
interface serial 0
 ip address 128.124.1.1
 ip address 131.108.1.1 secondary
 encapsulation frame-relay
```