Configuring Routing Information Protocol

Last Updated: July 20, 2011

Routing Information Protocol (RIP) is a commonly used routing protocol in small to medium TCP/IP networks. It is a stable protocol that uses a distance-vector algorithm to calculate routes.

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 Table at the end of this document.

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

Prerequisites for Configuring RIP

Before configuring RIP, the **ip routing** command is configured. For more information about configuring the **ip routing** command, see the *Cisco IOS IP Routing: RIP Command Reference*.

Restrictions for Configuring RIP

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.

Information About Configuring RIP

RIP Overview

Routing Information Protocol uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. Cisco IOS software sends routing information updates every 30 seconds, which 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.

A router that is running RIP can receive a default network via an update from another router that is running RIP, or the router can source (generate) the default network itself with RIP. In both cases, the default network is advertised through RIP to other RIP neighbors.

The Cisco implementation of RIP Version 2 supports plain text and Message Digest 5 (MD5) authentication, route summarization, classless interdomain routing (CIDR), and variable-length subnet masks (VLSMs).

RIP Routing Updates

RIP sends routing-update messages at regular intervals and when the network topology changes. When a router receives a RIP routing update that includes changes to an entry, the router updates its routing table to reflect the new route. The metric value for the path is increased by 1, and the sender is indicated as the next hop. RIP routers maintain only the best route (the route with the lowest metric value) to a destination. After updating its routing table, the router immediately begins transmitting RIP routing updates to inform other

network routers of the change. These updates are sent independently of the regularly scheduled updates that RIP routers send.

RIP Routing Metric

RIP uses a single routing metric (hop count) to measure the distance between the source and a destination network. Each hop in a path from source to destination is assigned a hop count value, which is typically 1. When a router receives a routing update that contains a new or changed destination network entry, the router adds 1 to the metric value indicated in the update and enters the network in the routing table. The IP address of the sender is used as the next hop. If the network of an interface network is not specified, it will not be advertised in any RIP update.

RIP Version 2 and Enabling Authentication

The Cisco implementation of RIP Version 2 supports authentication, key management, route summarization, CIDR, and VLSMs. For more information about managing authentication keys see the "Managing Authentication Keys" section of the "Configuring IP Routing Protocol-Independent Feature" module.

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 override the default behavior, you can configure which RIP version an interface sends. Similarly, you can also control how packets received from an interface are processed.

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 "Managing Authentication Keys" in the "Configuring IP Routing Protocol-Independent Features" module.

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.

Exchange of Routing Information

RIP is normally a broadcast protocol, and 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 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** router configuration command. See the discussion on filtering in the "Filter Routing Information" section in the "Configuring IP Routing Protocol-Independent Features" module.

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

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

In addition, an address family can have explicitly specified timers that apply to that address-family (or VRF) only. The timers basic command must be specified for an address family or the system defaults for the timers basic command are used regardless of what is configured for RIP routing. The VRF does not inherit the timer values from the base RIP configuration. The VRF will always use the system default timers unless explicitly changed using the timers basic command.

See the "Address Family Timers Example" section at the end of this chapter for examples of adjusting timers for an address family (VRF).

RIP Route Summarization

Summarizing routes in RIP Version 2 improves scalability and efficiency in large networks. Summarizing IP addresses means that there is no entry for child routes (routes that are created for any combination of the individual IP addresses contained within a summary address) in the RIP routing table, reducing the size of the table and allowing the router to handle more routes.

Summary IP address functions more efficiently than multiple individually advertised IP routes for the following reasons:

- The summarized routes in the RIP database are processed first.
- Any associated child routes that are included in a summarized route are skipped as RIP looks through the routing database, reducing the processing time required. Cisco routers can summarize routes in two ways:
- Automatically, by summarizing subprefixes to the classful network boundary when crossing classful network boundaries (automatic summary).

Note Automatic summary is enabled by default.

- As specifically configured, advertising a summarized local IP address pool on the specified interface (on a network access server) so that the address pool can be provided to dialup clients.

When RIP determines that a summary address is required in the RIP database, a summary entry is created in the RIP routing database. As long as there are child routes for a summary address, the address remains in the routing database. When the last child route is removed, the summary entry also is removed from the

database. This method of handling database entries reduces the number of entries in the database because each child route is not listed in an entry, and the aggregate entry itself is removed when there are no longer any valid child routes for it.

RIP Version 2 route summarization requires that the lowest metric of the "best route" of an aggregated entry, or the lowest metric of all current child routes, be advertised. The best metric for aggregated summarized routes is calculated at route initialization or when there are metric modifications of specific routes at advertisement time, and not at the time the aggregated routes are advertised.

The **ip summary-address rip router**configuration command causes the router to summarize a given set of routes learned via RIP Version 2 or redistributed into RIP Version 2. Host routes are especially applicable for summarization.

See the "Route Summarization Example, page 26" section at the end of this chapter for examples of using split horizon.

You can verify which routes are summarized for an interface using the **show ip protocols** EXEC command. You can check summary address entries in the RIP database. These entries will appear in the database only if relevant child routes are being summarized. To display summary address entries in the RIP routing database entries if there are relevant routes being summarized based upon a summary address, use the **show ip rip database** command in EXEC mode. When the last child route for a summary address becomes invalid, the summary address is also removed from the routing table.

Split Horizon Mechanism

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 Switched Multimegabit Digital System [SMDS]), situations can arise for which this behavior is less than ideal. For these situations, you may want to disable split horizon with RIP.

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

Interpacket Delay for RIP Updates

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.

RIP Optimization over WAN Circuits

Routers are used on connection-oriented networks to allow potential connectivity to many remote destinations. Circuits on the WAN are established on demand and are relinquished when the traffic subsides. Depending on the application, the connection between any two sites for user data could be short and relatively infrequent.

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.

Neighbor Router Authentication

You can prevent your router from receiving fraudulent route updates by configuring neighbor router authentication. When configured, neighbor authentication occurs whenever routing updates are exchanged between neighbor routers. This authentication ensures that a router receives reliable routing information from a trusted source.

Without neighbor authentication, unauthorized or deliberately malicious routing updates could compromise the security of your network traffic. A security compromise could occur if an unfriendly party diverts or analyzes your network traffic. For example, an unauthorized router could send a fictitious routing update to convince your router to send traffic to an incorrect destination. This diverted traffic could be analyzed to learn confidential information about your organization or merely used to disrupt your organization’s ability to effectively communicate using the network. Neighbor authentication prevents any such fraudulent route updates from being received by your router.

When neighbor authentication has been configured on a router, the router authenticates the source of each routing update packet that it receives. This is accomplished by the exchange of an authenticating key (sometimes referred to as a password) that is known to both the sending and the receiving router.

There are two types of neighbor authentication used: plain text authentication and Message Digest Algorithm Version 5 (MD5) authentication. Both forms work in the same way, with the exception that MD5 sends a "message digest" instead of the authenticating key itself. The message digest is created using the key and a message, but the key itself is not sent, preventing it from being read while it is being transmitted. Plain text authentication sends the authenticating key itself over the wire.

Note Note that plain text authentication is not recommended for use as part of your security strategy. Its primary use is to avoid accidental changes to the routing infrastructure. Using MD5 authentication, however, is a recommended security practice.

In plain text authentication, each participating neighbor router must share an authenticating key. This key is specified at each router during configuration. Multiple keys can be specified with some protocols; each key must then be identified by a key number.

In general, when a routing update is sent, the following authentication sequence occurs:

1. A router sends a routing update with a key and the corresponding key number to the neighbor router. In protocols that can have only one key, the key number is always zero. The receiving (neighbor) router checks the received key against the same key stored in its own memory.
2. If the two keys match, the receiving router accepts the routing update packet. If the two keys do not match, the routing update packet is rejected.

MD5 authentication works similarly to plain text authentication, except that the key is never sent over the wire. Instead, the router uses the MD5 algorithm to produce a "message digest" of the key (also called a "hash"). The message digest is then sent instead of the key itself. This ensures that nobody can eavesdrop on the line and learn keys during transmission.

Another form of neighbor router authentication is to configure key management using key chains. When you configure a key chain, you specify a series of keys with lifetimes, and the Cisco IOS software rotates through each of these keys. This decreases the likelihood that keys will be compromised. To find complete configuration information for key chains, refer to the "Managing Authentication Keys" section in the

Configuring IP Routing Protocol-Independent Features module of the Cisco IOS IP Routing: Protocol-Independent Configuration Guide.

IP-RIP Delay Start

The IP-RIP Delay Start feature is used on Cisco routers to delay the initiation of RIPv2 neighbor sessions until the network connectivity between the neighbor routers is fully operational, thereby ensuring that the sequence number of the first MD5 packet that the router sends to the non-Cisco neighbor router is 0. The default behavior for a router configured to establish RIPv2 neighbor sessions with a neighbor router using MD5 authentication is to start sending MD5 packets when the physical interface is up.

The IP-RIP Delay Start feature is often used when a Cisco router is configured to establish a RIPv2 neighbor relationship using MD5 authentication with a non-Cisco device over a Frame Relay network. When RIPv2 neighbors are connected over Frame Relay, it is possible for the serial interface connected to the Frame Relay network to be up while the underlying Frame Relay circuits are not yet ready to transmit and receive data. When a serial interface is up and the Frame Relay circuits are not yet operational, any MD5 packets that the router attempts to transmit over the serial interface are dropped. When MD5 packets are dropped because the Frame Relay circuits over which the packets need to be transmitted are not yet operational, the sequence number of the first MD5 packet received by the neighbor router after the Frame Relay circuits become active will be greater than 0. Some non-Cisco routers will not allow an MD5-authenticated RIPv2 neighbor session to start when the sequence number of the first MD5 packet received from the other router is greater than 0.

The differences in vendor implementations of MD5 authentication for RIPv2 are probably a result of the ambiguity of the relevant RFC (RFC 2082) with regards to packet loss. RFC 2082 suggests that routers should be ready to accept either a sequence number of 0 or a sequence number higher than the last sequence number received. For more information about MD5 message reception for RIPv2, see section 3.2.2 of RFC 2082 at the following url: http://www.ietf.org/rfc/rfc2082.txt .

Offset-list

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.

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.

How to Configure RIP

Enabling RIP and Configuring RIP Parameters

Perform the steps in this section to enable RIP and to configure RIP parameters.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router rip**
4. **network** *ip-address*
5. **neighbor** *ip-address*
6. **offset-list** [*access-list-number* | access-list-*name*] {**in** | **out**} *offset*[*interface-type interface-number*]
7. **timers basic** *update invalid holddown flush [sleeptime]*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**router rip** **Example:** `Router(config)# router rip`	Enables a RIP routing process and enters router configuration mode.
Step 4	**network** *ip-address* **Example:** `Router(config-router)# network 10.1.1.0`	Associates a network with a RIP routing process.
Step 5	**neighbor** *ip-address* **Example:** `Router(config-router)# neighbor 1.1.1.2`	Defines a neighboring router with which to exchange routing information.
Step 6	**offset-list** [*access-list-number* | access-list-*name*] {**in** | **out**} *offset*[*interface-type interface-number*] **Example:** `Router(config-router)# offset-list 98 in 1 Ethernet 1/0`	(Optional) Applies an offset to routing metrics.
Step 7	**timers basic** *update invalid holddown flush [sleeptime]* **Example:** `Router(config-router)# timers basic 1 2 3 4`	(Optional) Adjusts routing protocol timers.
Step 8	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode and returns to privileged EXEC mode.

Specifying a RIP Version and Enabling Authentication

Perform this task to specify a RIP version and enable authentication.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router rip**
4. **version** {**1** | **2**}
5. **exit**
6. **interface type number**
7. **ip rip send version [1] [2]**
8. **ip rip receive version [1] [2]**
9. **ip rip authentication key-chain** *name-of-chain*
10. **ip rip authentication mode** {**text** | **md5**}
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**router rip** **Example:** `Router(config)# router rip`	Enters router configuration mode.
Step 4	**version** {**1**	**2**} **Example:** `Router(config-router)# version 1`	Configures an interface to send only RIP Version 1 packets.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Router(config-router)# exit`	Exits the router configuration mode and enters the global configuration mode.
Step 6	**interface type number** **Example:** `Router(config)# interface Ethernet 3/0`	Enters interface configuration mode.
Step 7	**ip rip send version [1] [2]** **Example:** `Router(config-if)# ip rip send version 1`	Configures an interface to send only RIP Version 1 packets.
Step 8	**ip rip receive version [1] [2]** **Example:** `Router(config-if)# ip rip receive version 1`	Configures an interface to accept only RIP Version 1 packets.
Step 9	**ip rip authentication key-chain** *name-of-chain* **Example:** `Router(config-if)# ip rip authentication key-chain chainname`	Enables RIP authentication.
Step 10	**ip rip authentication mode** {**text** | **md5**} **Example:** `Router(config-if)# ip rip authentication mode md5`	Configures the interface to use MD5 digest authentication (or let it default to plain text authentication).
Step 11	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Summarizing RIP Routes

RIP Version 2 supports automatic route summarization by default. The software summarizes subprefixes to the classful network boundary when classful network boundaries are crossed. If you have disconnected subnets, disable automatic route summarization to advertise the subnets. When route summarization is disabled, the software sends subnet and host routing information across classful network boundaries. To disable automatic summarization, use the **no auto-summary** command in router configuration mode.

Note Supernet advertisement (advertising any network prefix less than its classful major network) is not allowed in RIP route summarization, other than advertising a supernet learned in the routing tables. Supernets learned on any interface that is subject to configuration are still learned. For example, the following summarization is invalid: (invalid supernet summarization)

```
Router(config)# interface Ethernet 1
Router(config-if)# ip summary-address rip 10.0.0.0 252.0.0.0
.
.
>
```

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip summary-address rip** *ip-address network-mask*
5. **exit**
6. **router rip**
7. **no auto-summary**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet 3/0`	Enters the interface configuration mode.
Step 4	**ip summary-address rip** *ip-address network-mask* **Example:** `Router(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0`	Specifies the IP address and network mask that identify the routes to be summarized.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits the interface configuration mode.
Step 6	**router rip** **Example:** `Router(config)# router rip`	Enters the router configuration mode.
Step 7	**no auto-summary** **Example:** `Router(config-router)# no auto-summary`	Used in router configuration mode, disables automatic summarization.
Step 8	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode and returns to privileged EXEC mode.

Enabling or Disabling Split Horizon

To enable or disable split horizon, use the following commands in interface configuration mode, as needed.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip split-horizon**
5. **no ip split-horizon**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet 3/0`	Enters interface configuration mode.
Step 4	**ip split-horizon** **Example:** `Router(config-if)# ip split-horizon`	Enables split horizon.
Step 5	**no ip split-horizon** **Example:** `Router(config-if)# no ip split-horizon`	Disables split horizon.

	Command or Action	Purpose
Step 6	**end** **Example:** `Router(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Disabling the Validation of Source IP Addresses

Perform this task to disable the default function that validates the source IP addresses of incoming routing updates.

Note 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.

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 that 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.

Note Summarized network will not be advertised when split horizon is enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip split-horizon**
5. **exit**
6. **router rip**
7. **no validate-update-source**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet 3/0`	Enters interface configuration mode.
Step 4	**ip split-horizon** **Example:** `Router(config-if)# ip split-horizon`	Enables split horizon.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 6	**router rip** **Example:** `Router(config)# router rip`	Enters router configuration mode.
Step 7	**no validate-update-source** **Example:** `Router(config-router)# no validate-update-source`	Disables the validation of the source IP address of incoming RIP routing updates.

	Command or Action	Purpose
Step 8	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode and returns to privileged EXEC mode.

Configuring Interpacket Delay

Perform this to configure interpacket delay.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **exit**
5. **router rip**
6. **output-delay** *milliseconds*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface Ethernet 3/0`	Enters interface configuration mode.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 5	**router rip** **Example:** `Router(config)# router rip`	Enters router configuration mode.
Step 6	**output-delay** *milliseconds* **Example:** `Router(config-router)# output-delay 8`	Configures interpacket delay for outbound RIP updates.
Step 7	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode and returns to privileged EXEC mode.

Optimizing RIP over WAN

There are two problems when RIP is not optimized:

- Periodic broadcasting by RIP generally prevents WAN circuits from being closed.
- Even on fixed, point-to-point links, the overhead of periodic RIP transmissions could seriously interrupt normal data transfer because of the quantity of information that passes through the line every 30 seconds.

To overcome these limitations, triggered extensions to RIP cause RIP to send information on the WAN only when there has been an update to the routing database. Periodic update packets are suppressed over the interface on which this feature is enabled. RIP routing traffic is reduced on point-to-point, serial interfaces. Therefore, you can save money on an on-demand circuit for which you are charged for usage. Triggered extensions to RIP partially support RFC 2091, *Triggered Extensions to RIP to Support Demand Circuits* .

Perform the following task to enable triggered extensions to RIP and to display the contents of the RIP private database.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface serial** *controller-number*
4. **ip rip triggered**
5. **end**
6. **show ip rip database** [*prefix mask*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface serial** *controller-number* **Example:** `Router(config)# interface serial3/0`	Configures a serial interface.
Step 4	**ip rip triggered** **Example:** `Router(config-if)# ip rip triggered`	Enables triggered extensions to RIP.
Step 5	**end** **Example:** `Router(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show ip rip database** [*prefix mask*] **Example:** `Router# show ip rip database`	Displays the contents of the RIP private database.

Configuring IP-RIP Delay Start for Routers Connected by a Frame Relay Network

The tasks in this section explain how to configure a router to use the IP-RIP Delay Start feature on a Frame Relay interface.

Timesaver Cisco routers allow an MD5-authenticated RIPv2 neighbor session to start when the sequence number of the first MD5 packet received from the other router is greater than 0. If you are using only Cisco routers in your network, you do not need to use the IP-RIP Delay Start feature.

Prerequisites

Your router must be running Cisco IOS Release 12.4(12) or a later release.

Note The IP-RIP Delay Start feature is supported over other interface types such as Fast Ethernet and Gigabit Ethernet. If your Cisco router cannot establish RIPv2 neighbor sessions using MD5 authentication with a non-Cisco device, the IP-RIP Delay Start feature might resolve the problem.

Restrictions

The IP-RIP Delay Start feature is required only when your Cisco router is configured to establish a RIPv2 neighbor relationship with a non-Cisco device and you want to use MD5 neighbor authentication.

Configuring RIPv2

This required task configures RIPv2 on the router.

This task provides instructions for only one of the many possible permutations for configuring RIPv2 on your router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router rip**
4. **network** *ip-network*
5. **version** {**1** | **2**}
6. [**no**] **auto-summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**router rip** **Example:** `Router(config)# router rip`	Enables a RIP routing process, which places you in router configuration mode.
Step 4	**network** *ip-network* **Example:** `Router(config-router)# network 192.168.0.0`	Associates a network with a RIP routing process.
Step 5	**version** {**1**	**2**} **Example:** `Router (config-router)# version 2`	Configures the software to receive and send only RIP Version 1 or only RIP Version 2 packets.

Command or Action	Purpose
Step 6 [**no**] **auto-summary** **Example:** `Router(config-router)# no auto-summary`	Disables or restores the default behavior of automatic summarization of subnet routes into network-level routes.

Configuring Frame Relay on a Serial Subinterface

This required task configures a serial subinterface for Frame Relay.

Note This task provides instructions for only one of the many possible permutations for configuring Frame Relay on a subinterface. For more information about and instructions for configuring Frame Relay, see the Configuring Frame Relay part of the *Cisco IOS Wide-Area Networking Configuration Guide* .

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no ip address**
5. **encapsulation frame-relay [mfr number | ietf]**
6. **frame-relay lmi-type** {**cisco** | **ansi** | **q933a**}
7. **exit**
8. **interface** *type number/subinterface-number* {**point-to-point** | **multipoint**}
9. **frame-relay interface-dlci** *dlci* [**ietf** | **cisco**]

DETAILED STEPS

Command or Action	Purpose
Step 1 **enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2 **configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *type number* **Example:** `Router(config)# interface serial3/0`	Specifies an interface and enters interface configuration mode.
Step 4	**no ip address** **Example:** `Router(config-if)# no ip address`	Removes a previously configured IP address from the interface.
Step 5	**encapsulation frame-relay [mfr number | ietf]** **Example:** `Router(config-if)# encapsulation frame-relay ietf`	Specifies the type of Frame Relay encapsulation for the interface.
Step 6	**frame-relay lmi-type** {**cisco** | **ansi** | **q933a**} **Example:** `Router(config-if)# frame-relay lmi-type ansi`	Specifies the type of Frame Relay local management interface (LMI) for the interface.
Step 7	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 8	**interface** *type number/subinterface-number* {**point-to-point** | **multipoint**} **Example:** `Router(config)# interface serial3/0.1 point-to-point`	Specifies a subinterface and the connection type for the subinterface and enters subinterface configuration mode.
Step 9	**frame-relay interface-dlci** *dlci* [**ietf** | **cisco**] **Example:** `Router(config-subif)# frame-relay interface-dlci 100 ietf`	Assigns a data-link connection identifier (DLCI) to a Frame Relay subinterface.

Configuring IP with MD5 Authentication for RIPv2 and IP-RIP Delay on a Frame Relay Subinterface

This required task configures IP, MD5 authentication for RIPv2 and the IP-RIP Delay Start feature on a Frame Relay subinterface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **key chain** *name-of-chain*
4. **key** *number*
5. **key-string** *string*
6. **exit**
7. **exit**
8. **interface** *type number/subinterface-number*
9. **no cdp enable**
10. **ip address** *ip-address subnet-mask*
11. **ip rip authentication mode** {**text** | **md5**}
12. **ip rip authentication key-chain** *name-of-chain*
13. **ip rip initial-delay** *delay*
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**key chain** *name-of-chain* **Example:** `Router(config)# key chain rip-md5`	Specifies the name of a key chain and enters key chain configuration mode.

	Command or Action	Purpose
Step 4	**key** *number* **Example:** `Router(config-keychain)# key 123456`	Specifies the key identifier and enters key chain key configuration mode. Range: 0 to 2147483647.
Step 5	**key-string** *string* **Example:** `Router(config-keychain-key)# key-string abcde`	Configures the key string.
Step 6	**exit** **Example:** `Router(config-keychain-key)# exit`	Exits key chain key configuration mode.
Step 7	**exit** **Example:** `Router(config-keychain)# exit`	Exits key chain configuration mode.
Step 8	**interface** *type number/subinterface-number* **Example:** `Router(config)# interface serial3/0.1`	Specifies a subinterface and enters subinterface configuration mode. **Note** The connection type keyword is not required for this step in this task because the connection type for this subinterface was specified in the previous task.
Step 9	**no cdp enable** **Example:** `Router(config-subif)# no cdp enable`	Disables Cisco Discovery Protocol (CDP) options on the interface. **Note** CDP is not supported by non-Cisco devices; and the IP-RIP Delay Start feature is required only when you are connecting to a non-Cisco router. Therefore, you should disable CDP on any interfaces on which you want to configure the IP-RIP Delay Start feature.
Step 10	**ip address** *ip-address subnet-mask* **Example:** `Router (config-subif)# ip address 172.16.10.1 255.255.255.0`	Configures an IP address for the Frame Relay subinterface.

	Command or Action	Purpose
Step 11	**ip rip authentication mode** {**text** | **md5**} **Example:** `Router(config-subif)# ip rip authentication mode md5`	Specifies the mode for RIPv2 authentication.
Step 12	**ip rip authentication key-chain** *name-of-chain* **Example:** `Router (config-subif)# ip rip authentication key-chain rip-md5`	Specifies a previously configured key chain for RIPv2 MD5 authentication.
Step 13	**ip rip initial-delay** *delay* **Example:** `Router(config-subif)# ip rip initial-delay 45`	Configures the IP-RIP Delay Start feature on the interface. The router will delay sending the first MD5 authentication packet to the RIPv2 neighbor for the number of seconds specified by the *delay* argument. Range: 0 to 1800.
Step 14	**end** **Example:** `Router(config-subif)# end`	Exits the sub-interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for RIP

Route Summarization Example

The following example shows how the **ip summary-address rip**router configuration command can be used to configure summarization on an interface. In this example, the subnets 10.1.3.0/25, 10.1.3.128/25, 10.2.1.0/24, 10.2.2.0/24, 10.1.2.0/24 and 10.1.1.0/24 can be summarized as shown below while sending the updates over an interface.

```
Router(config)#interface GigabitEthernet 0/2
Router(config-if)#ip summary-address rip 10.1.0.0 255.255.0.0
Router(config-if)#ip summary-address rip 10.2.0.0 255.255.0.0
Router(config-if)#ip summary-address rip 10.3.0.0 255.255.0.0
```

Split Horizon Examples

Two examples of configuring split horizon are provided.

Example 1

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

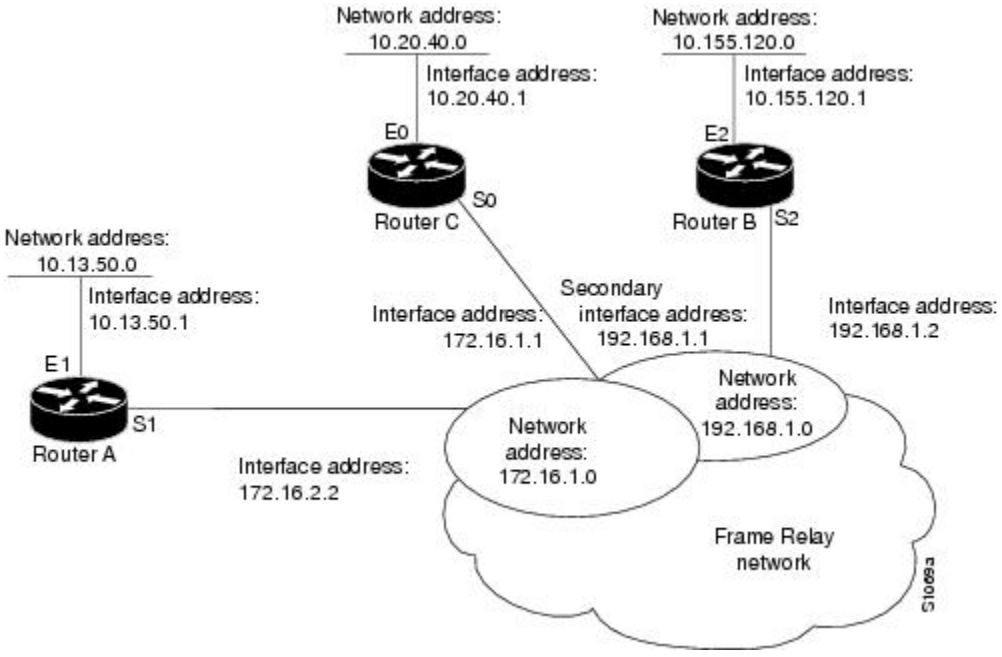
```
Router(config)# interface Serial 0
Router(config-if)# encapsulation x25

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

Example 2

In the next example, the figure below 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 a 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 10.13.50.0, 10.155.120.0, and 10.20.40.0, respectively all have split horizon enabled by default, while the serial interfaces connected to networks 172.16.1.0 and 192.168.1.0 all have split horizon disabled with the **no ip split-horizon** command.The figure below shows the topology and interfaces.

Figure 1



In this example, split horizon is disabled on all serial interfaces. Split horizon must be disabled on Router C in order for network 172.16.0.0 to be advertised into network 192.168.0.0 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 10.13.50.1
!
interface serial 1
 ip address 172.16.2.2
 encapsulation frame-relay
 no ip split-horizon
```

Configuration for Router B

```
interface ethernet 2
 ip address 10.155.120.1
!
interface serial 2
 ip address 192.168.1.2
 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 172.16.1.1
 ip address 192.168.1.1 secondary
 encapsulation frame-relay
 no ip split-horizon
```

Address Family Timers Example

The following example shows how to adjust individual address family timers. Note that the address family "notusingtimers" will use the system defaults of 30, 180, 180, and 240 even though timer values of 5, 10, 15, and 20 are used under the general RIP configuration. Address family timers are not inherited from the general RIP configuration.

```
Router(config)# router rip
Router(config-router)# version 2
Router(config-router)# timers basic 5 10 15 20
Router(config-router)# redistribute connected
Router(config-router)# network 5.0.0.0
Router(config-router)# default-metric 10
Router(config-router)# no auto-summary
Router(config-router)#
Router(config-router)# address-family ipv4 vrf abc
Router(config-router-af)# timers basic 10 20 20 20
Router(config-router-af)# redistribute connected
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# default-metric 5
Router(config-router-af)# no auto-summary
Router(config-router-af)# version 2
Router(config-router-af)# exit-address-family
Router(config-router)#
Router(config-router)# address-family ipv4 vrf xyz
Router(config-router-af)# timers basic 20 40 60 80
Router(config-router-af)# redistribute connected
Router(config-router-af)# network 20.0.0.0
Router(config-router-af)# default-metric 2
Router(config-router-af)# no auto-summary
Router(config-router-af)# version 2
Router(config-router-af)# exit-address-family
Router(config-router)#
Router(config-router)# address-family ipv4 vrf notusingtimers
```

```
Router(config-router-af)# redistribute connected
Router(config-router-af)# network 20.0.0.0
Router(config-router-af)# default-metric 2
Router(config-router-af)# no auto-summary
Router(config-router-af)# version 2
Router(config-router-af)# exit-address-family
Router(config-router)#
```

IP-RIP Delay Start on a Frame Relay Interface Examples

This excerpt from a router configuration file contains the minimum commands required to configure the IP-RIP Delay Start feature on your router.

```
!
key chain rip-md5
 key 123456
  key-string abcde
!
router rip
 version 2
 network 172.16.0.0
 no auto-summary
!
interface Serial3/0
 no ip address
 encapsulation frame-relay ietf
 frame-relay lmi-type ansi
!
interface Serial3/0.1 point-to-point
 ip address 172.16.10.1 255.255.255.0
 ip rip initial-delay 45
 ip rip authentication mode md5
 ip rip authentication key-chain rip-md5
 frame-relay interface-dlci 100
```

Additional References

The following sections provide references related to configuring Routing Information Protocol.

Related Documents

Related Topic	Document Title
Protocol-independent features, filtering RIP information, key management (available in RIP Version 2), and VLSM	*Configuring IP Routing Protocol-Independent Features*
RIP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	*Cisco IOS IP Routing: RIP Command Reference*
Configuring Frame Relay	*Cisco IOS Wide-Area Networking Configuration Guide*

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
No new or modified MIBS are supported and support for existing MIBs has not been modified.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 1058	*Routing Information Protocol*
RFC 2082	RIP-2 MD5 Authentication
RFC 2091	*Triggered Extensions to RIP to Support Demand Circuits*
RFC 2453	RIP version 2

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring RIP

The following table provides release information about the feature or features described in this module. This table 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.

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

Table 1 Feature Information for Configuring Routing Information Protocol

Feature Name	Releases	Feature Information
IP-RIP Delay Start	12.4(12) 15.0(1)M 12.2(33)SRE	The IP-RIP Delay Start feature is used on Cisco routers to delay the initiation of RIPv2 neighbor sessions until the network connectivity between the neighbor routers is fully operational, thereby ensuring that the sequence number of the first MD5 packet that the router sends to the non-Cisco neighbor router is 0. The default behavior for a router configured to establish RIPv2 neighbor sessions with a neighbor router using MD5 authentication is to start sending MD5 packets when the physical interface is up. The following commands were introduced or modified: **ip rip initial-delay**.
IP Summary Address for RIPv2	12.0(7)T 12.1(3)T 12.1(14) 12.2(2)T 12.2(27)SBB 15.0(1)M 12.2(33)SRE 15.0S	The IP Summary Adddress for RIPv2 feature introduced the ability to summarize routes. Summarizing routes in RIP Version 2 improves scalability and efficiency in large networks. Summarizing IP addresses means that there is no entry for child routes (routes that are created for any combination of the individual IP addresses contained within a summary address) in the RIP routing table, reducing the size of the table and allowing the router to handle more routes. The following commands were introduced or modified by this feature: **ip summary-address rip**.

Feature Name	Releases	Feature Information
Routing Information Protocol	12.2(27)SBB 15.0(1)M 12.2(33)SRE 15.0S	Routing Information Protocol (RIP) is a commonly used routing protocol in small to medium TCP/IP networks. It is a stable protocol that uses a distance-vector algorithm to calculate routes.
Triggered RIP	12.0(1)T 15.0(1)M 12.2(33)SRE 15.0S	Triggered RIP was introduced to overcome constant RIP updates over expensive circuit-based WAN links. Triggered extensions to RIP cause RIP to send information on the WAN only when there has been an update to the routing database. Periodic update packets are suppressed over the interface on which this feature is enabled. RIP routing traffic is reduced on point-to-point, serial interfaces. The following commands were introduced or modified: **ip rip triggered**, **show ip rip database**.

Glossary

address family --A group of network protocols that share a common format of network address. Address families are defined by RFC 1700.

IS-IS --Intermediate System-to-Intermediate System. OSI link-state hierarchical routing protocol based on DECnet Phase V routing, where routers exchange routing information based on a single metric, to determine network topology.

RIP --Routing Information Protocol.RIP is a dynamic routing protocol used in local and wide area networks.

VRF --VPN routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a PE router.

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.