

Routing Information Protocol

- Feature history for Routing Information Protocol, on page 1
- Information About RIP, on page 1
- How RIP Works, on page 2
- Split horizon, on page 2
- Default RIP Configuration, on page 2
- Configuring Basic RIP Parameters, on page 3
- Configuration Examples for Routing Information Protocol, on page 11

Feature history for Routing Information Protocol

This table provides release and related information for the features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature Name and Description	Supported Platform
Cisco IOS XE 26.2.1ea	Routing Information Protocol: Routing Information Protocol (RIP) feature support has been introduced.	Cisco C9550 Series Smart Switches
Cisco IOS XE 17.18.1	Routing Information Protocol: The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small and homogeneous networks.	Cisco C9350 Series Smart Switches Cisco C9610 Series Smart Switches

Information About RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) designed for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. RIP is documented in RFC 1058.

How RIP Works

Summary

Using RIP, the switch sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If no update is received after 240 seconds, the router removes all routing table entries for the non-updating router.

Workflow

RIP uses hop count to rate the value of different routes. The hop count represents the number of routers that can be traversed in a route. A directly connected network has a hopcount of zero. A network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable 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 0.0.0.0 network does not exist; RIP treats it as a network to implement the default routing feature. The switch advertises the default network if RIP learned a default, 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 specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

RIP for IPv6

Routing Information Protocol (RIP) for IPv6 is a distance-vector protocol that uses hop count as a routing metric. It includes support for IPv6 addresses and prefixes, and uses the all-RIP-routers multicast group address FF02::9 as the destination address for RIP update messages.

For information about configuring RIP for IPv6, see the "Configuring RIP for IPv6" section. For more information about RIP for IPv6, see the "Implementing RIP for IPv6" chapter in the Cisco IOS IPv6 Configuration Library on Cisco.com.

Split horizon

A split horizon is a loop-prevention technique in distance-vector routing protocols that

- prevents a router from advertising a route back on the interface from which it learned the route,
- reduces the possibility of routing loops among routers, and
- optimizes network communication, especially when network links are broken.

Split horizon operates on routers connected to broadcast-type IP networks. By blocking route advertisements on the interface where the information originated, it enhances convergence speed and network reliability.

Default RIP Configuration

Feature	Default Setting

Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP triggered update	Disabled.
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.
Timers basic	• Update: 30 seconds. • Invalid: 180 seconds. • Hold-down: 180 seconds. • Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP Version 1 and 2 packets; sends Version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters. On the switch, RIP configuration commands are ignored until you configure the network number.

Procedure

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

	Command or Action	Purpose
Step 3	**ip routing** **Example:** `Device(config)# ip routing`	Enables IP routing. (Required only if IP routing is disabled.)
Step 4	**router rip** **Example:** `Device(config)# router rip`	Enables a RIP routing process, and enter router configuration mode.
Step 5	**network** *network number* **Example:** `Device(config-router)# network 12.0.0.0`	Associates a network with a RIP routing process. You can specify multiple **network** commands. RIP routing updates are sent and received through interfaces only on these networks. **Note** You must configure a network number for the RIP commands to take effect.
Step 6	**neighbor** *ip-address* **Example:** `Device(config-router)# neighbor 10.2.5.1`	(Optional) Defines a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 7	**offset-list** {*access-list number* | *name*} {**in** | **out**} *offset* [*type number*] **Example:** `Device(config-router)# offset-list 10 in 10`	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
Step 8	**timers basic** *update invalid holddown flush* **Example:** `Device(config-router)# timers basic 45 360 400 300`	(Optional) Adjusts routing protocol timers. The valid ranges for all timers are up to 2147483 seconds. • *update*—The time between sending routing updates. The default is 30 seconds. • *invalid*—The timer after which a route is declared invalid. The default is 180 seconds. • *holddown*—The time before a route is removed from the routing table. The default is 180 seconds.

	Command or Action	Purpose
		• *flush*—The amount of time for which routing updates are postponed. The default is 240 seconds.
Step 9	**version** {**1** | **2**} **Example:** `Device(config-router)# version 2`	(Optional) Configures the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands **ip rip** {**send** | **receive**} **version 1** | **2** | **1 2**} to control what versions are used for sending and receiving on interfaces.
Step 10	**no auto-summary** **Example:** `Device(config-router)# no auto-summary`	(Optional) Disables automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 11	**output-delay** *delay* **Example:** `Device(config-router)# output-delay 8`	(Optional) Adds interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	**end** **Example:** `Device(config-router)# end`	Returns to privileged EXEC mode.
Step 13	**show ip protocols** **Example:** `Device# show ip protocols`	Verifies your entries.
Step 14	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.
Step 15	**show ip rip database** **Example:** `Device# show ip rip database`	(Optional) Displays RIP routing database entries.

	Command or Action	Purpose
Step 16	**show ip route rip** **Example:** `Device# show ip route rip`	(Optional) Display contents of the RIP routing table.

Configuring 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 specifies the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default.

The switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 4	**ip rip authentication key-chain** *name-of-chain* **Example:** `Device(config-if)# ip rip authentication key-chain trees`	Enables RIP authentication.
Step 5	**ip rip authentication mode** {**text** | **md5**} **Example:** `Device(config-if)# ip rip authentication mode md5`	Configures the interface to use plain text authentication (the default) or MD5 digest authentication.

	Command or Action	Purpose
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring RIP for IPv6

Before you begin

Before configuring the switch to run IPv6 RIP, you must enable routing by using the **ip routing** command in global configuration mode, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** command in global configuration mode, and enable IPv6 on any Layer 3 interfaces on which IPv6 RIP is to be enabled.

For more information about configuring RIP routing for IPv6, see the "Implementing RIP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com,

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router rip** *name* **Example:** `Device(config)# ipv6 router rip cisco`	Configures an IPv6 RIP routing process, and enters router configuration mode for the process.

	Command or Action	Purpose
Step 4	**maximum-paths** *number-paths* **Example:** `Device(config-rtr)# maximum-paths 6`	(Optional) Define the maximum number of equal-cost routes that IPv6 RIP can support. The range is from 1 to 32, and the default is 16 routes.
Step 5	**exit** **Example:** `Device(config-rtr)# exit`	Returns to global configuration mode.
Step 6	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 7	**ipv6 enable** **Example:** `Device(config-if)# ipv6 enable`	Enables the interface for IPv6 processing.
Step 8	**ipv6 rip** *name* **enable** **Example:** `Device(config-if)# ipv6 rip cisco enable`	Enables the specified IPv6 RIP routing process on the interface.
Step 9	**ipv6 rip** *name* **default-information** {**only** | **originate**} **Example:** `Device(config-if)# ipv6 rip cisco default-information only`	(Optional) Originates the IPv6 default route (::/0) into the RIP routing process updates sent from the specified interface. **Note** To avoid routing loops after the IPv6 default route (::/0) is originated from any interface, the routing process ignores all default routes received on any interface. • **only**—Select to originate the default route, but suppress all other routes in the updates sent on this interface. • **originate**—Select to originate the default route in addition to all other routes in the updates sent on this interface.
Step 10	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 11	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Summary Entries



Note If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	**ip address** *ip-address subnet-mask* **Example:** `Device(config-if)# ip address 10.1.1.10 255.255.255.0`	Configures the IP address and IP subnet.
Step 5	**ip summary-address rip** *ip-network mask* **Example:** `Device(config-if)# ip summary-address rip 10.1.0.0 255.255.0.0`	Configures the IP address to be summarized and the IP network mask.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show ip interface** *interface-id* **Example:**	Verifies your entries.

	Command or Action	Purpose
	`Device# show ip interface gigabitethernet 1/0/1`	
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature can optimize communication among multiple routers, especially when links are broken.

Note In general, we do not recommend disabling split horizon unless you are certain that your application requires it to properly advertise routes.

If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Device(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.

	Command or Action	Purpose
Step 4	**ip address** *ip-address subnet-mask* **Example:** `Device(config-if)# ip address 10.1.1.10 255.255.255.0`	Configures the IP address and IP subnet.
Step 5	**no ip split-horizon** **Example:** `Device(config-if)# no ip split-horizon`	Disables split horizon on the interface.
Step 6	**end** **Example:** `Device(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show ip interface** *interface-id* **Example:** `Device# show ip interface gigabitethernet 1/0/1`	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuration Examples for Routing Information Protocol

The following sections provide configuration examples for RIP.

Example: Summary Addresses and Split Horizon

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet port 2, and 10.0.0.0 is not advertised. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport interface configuration** command before entering the **ip address interface configuration** command.

Note If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip interface configuration** command) are advertised.

```
Device(config-router)# interface gigabitethernet1/0/2
Device(config-if)# ip address 10.1.5.1 255.255.255.0
Device(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Device(config-if)# no ip split-horizon
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# network 10.0.0.0
Device(config-router)# end
```

Example: Configuring RIP for IPv6

This example shows how to enable the RIP routing process *cisco* with a maximum of eight equal-cost routes and to enable it on an interface:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 router rip cisco
Device(config-rtr)# maximum-paths 8
Device(config)# exit
Device(config)# interface gigabitethernet2/0/11
Device(config-if)# ipv6 rip cisco enable
```