

RIP Commands

Note All commands applicable for the Cisco NCS 5500 Series Router are also supported on the Cisco NCS 540 Series Router that is introduced from Cisco IOS XR Release 6.3.2. References to earlier releases in Command History tables apply to only the Cisco NCS 5500 Series Router.

Note

- Starting with Cisco IOS XR Release 6.6.25, all commands applicable for the Cisco NCS 5500 Series Router are also supported on the Cisco NCS 560 Series Routers.
- Starting with Cisco IOS XR Release 6.3.2, all commands applicable for the Cisco NCS 5500 Series Router are also supported on the Cisco NCS 540 Series Router.
- References to releases before Cisco IOS XR Release 6.3.2 apply to only the Cisco NCS 5500 Series Router.
- Cisco IOS XR Software Release 7.0.1 specific updates are not applicable for the following variants of Cisco NCS 540 Series Routers:
  - N540-28Z4C-SYS-A
  - N540-28Z4C-SYS-D
  - N540X-16Z4G8Q2C-A
  - N540X-16Z4G8Q2C-D
  - N540X-16Z8Q2C-D
  - N540-12Z20G-SYS-A
  - N540-12Z20G-SYS-D
  - N540X-12Z16G-SYS-A
  - N540X-12Z16G-SYS-D

This module describes the commands used to display and clear information in the Routing Information Base (RIB).

For detailed information about RIB concepts, configuration tasks, and examples, see the Implementing RIB on Cisco NCS 5500 Series Routers module in *Routing Configuration Guide for Cisco NCS 5500 Series Routers*.

- authentication keychain mode (RIP), on page 3
- clear rip , on page 5
- interface (RIP), on page 7
- redistribute (RIP), on page 11
- router rip, on page 14
- show protocols (RIP), on page 20
- show rip, on page 21

authentication keychain mode (RIP)

To enable an authentication keychain mechanism on RIP interfaces, use the **authentication keychain mode** command in interface configuration mode. To disable authentication keychain configuration on RIP interfaces, use the **no** form of this command.

authentication keychain *name* **mode** { **md5** | **text** }
no authentication keychain

Syntax Description

name	Specifies the name of the keychain configured using the keychain command. **Note** All keychains need to be configured in Cisco IOS XR keychain database using the keychain configuration commands described in *Implementing Keychain Management* module of *System Security Configuration Guide for Cisco NCS 5500 Series RoutersSystem Security Configuration Guide for Cisco NCS 540 Series RoutersSystem Security Configuration Guide*
md5	Specifies that the authentication keychain mode is keyed message digest (md5).
text	Specifies that the authentication keychain mode is clear text.

Command Default

Keychain authentication is disabled.

Command Modes

Interface configuration

Command History

Usage Guidelines

All keychains need to be configured in Cisco IOS XR keychain database using the keychain configuration commands described in *Implementing Keychain Management* module of *System Security Configuration Guide for Cisco NCS 5500 Series RoutersSystem Security Configuration Guide for Cisco NCS 540 Series RoutersSystem Security Configuration Guide*

This example shows how to configure an authentication keychain in md5 mode on a RIP VRF interface:

```
Router# configure
Router(config)# router rip
Router(config-rip)# vrf vrf_rip_auth
Router(config-rip-vrf)# interface POS 0/6/0/0
Router(config-rip-vrf-if)# authentication keychain key1 mode md5
```

This example shows how to configure an authentication keychain in clear text mode on a RIP interface:

```
Router# configure
Router(config)# router rip
Router(config-rip)# interface POS 0/6/0/1
Router(config-rip-if)# authentication keychain key2 mode text
```

Related Commands

Command	Description
router rip	Configures a routing process and enters RIP router configuration mode.
key chain (key chain)	Creates or modifies a keychain. Refer *System Security Command Reference for Cisco NCS 5500 Series Routers and Cisco NCS 540 and NCS 560 Series Routers* for complete command reference information.

clear rip

To clear VRF and interface-related information for Routing Information Protocol (RIP) such as database entries and statistics, use the **clear rip** command in EXEC configuration mode.

clear rip [**vrf** { *name* | **all** }] [**database** | **out-of-memory** | **statistics**] [**interface** *type number*]

Syntax Description

vrf {*name*|**all**}	(Optional) Clears RIP-associated VRF information. You can specify a VRF, or clear all VRF information.
database	(Optional) Clears database entries from the RIP topology table.
out-of-memory	(Optional) Clears RIP out-of-memory state information.
statistics	(Optional) Clears RIP statistics information.
interface *type number*	(Optional) Clears RIP interface information.

Command Default

No default behavior or values

Command Modes

EXEC configuration

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

RIP VRFs and interfaces that are forcibly deactivated by the software because of a severe memory state are not activated again until the out-of-memory state is cleared by using the **clear rip** , **clear rip interface** , or **clear rip out-of-memory** command. Use the **clear rip out-of-memory** command to clear the out-of-memory state completely and allow the RIP process to force the VRF or interface to shut down.

When the router begins to run out of memory, the RIP process can transition through different memory states defined as Normal, Minor, Severe, and Critical:

- In Normal state, RIP VRFs and interfaces function normally.
- In Minor state, RIP VRFs and interfaces that are currently active are allowed to remain active. VRFs and interfaces that are not currently active are not allowed to become active until the RIP process transitions to Normal state.
- In Severe state, a few VRFs and interfaces are forcibly brought down periodically until the RIP process transitions to another state.
- In Critical state, the RIP process is forcibly shut down.

Examples

The following example shows how to clear all database, interface, and VRF entries in RIP:

```
Router# clear rip vrf all
```

The following example shows how to clear only database entries from the topology table for the GigabitEthernet 0/1/0/0 interface:

```
Router# clear rip database interface GigabitEthernet 0/1/0/0
```

interface (RIP)

To define the interfaces on which the Routing Information Protocol (RIP) runs and to associate a RIP interface to a VRF, use the **interface** command in RIP router configuration or RIP VRF configuration mode. To disable RIP routing for interfaces, use the **no** form of this command.

interface *type number* [**broadcast-for-v2** | **metric-zero-accept** | **passive-interface** | **poison-reverse** | { **receive** | **send** } **version** *RIP-version* | **route-policy** *name* [*parameter-value*] { **in** | **out** } | **site-of-origin** *ID* | **split-horizon disable**]
no interface *type number* [**broadcast-for-v2** | **metric-zero-accept** | **passive-interface** | **poison-reverse** | { **receive** | **send** } **version** | **route-policy** *name* [*parameter-value*] { **in** | **out** } | **site-of-origin** | **split-horizon disable**]

Syntax Description

type number	(Optional) Interface type and number. For more information, use the question mark (**?**) online help function. The specified interface is associated with the RIP process.
broadcast-for-v2	(Optional) Specifies that RIPv2 messages are sent to the broadcast address on the specified interface.
metric-zero-accept	(Optional) Specifies that RIP accept routing entries from RIP updates with a metric set to zero (0) on the specified interface.
passive-interface	(Optional) Specifies that RIP routing updates are suppressed on the specified interface.
poison-reverse	(Optional) Enables poison reverse processing of RIP router updates on the specified interface.
{**receive** | **send**} **version** *RIP-version*	(Optional) Configures the RIP interface to accept and send RIP v1 or RIP v2 packets on the specified interface. For the *RIP-version* argument, enter **1** for RIP v1, **2** for RIP v2, and **1 2** for both versions.
route-policy *name* [*(parameter-value)*] {**in**|**out**}	(Optional) Applies a routing policy with specific parameter values on a RIP interface, on inbound and outbound routes.

site-of-origin *ID*	(Optional) Configures the Site of Origin (SoO) filtering on a RIP interface. You can enter the *ID* in one of these formats: *as-number : number* *ip-address : number* AS number (ASN) range - 1 to 65535 for a 2-byte ASN, 1 to 4294967295 for a 4-byte ASN in asplain format, and 1.0 to 65535.6553 for a 4-byte ASN in asdot format. Network number range - 0 to 4294967295 for a 2-byte ASN, and 0 to 65535 for a 4-byte ASN. A colon separates the ASN and network number, or the IP address and network number.
split-horizon disable	(Optional) Disables split horizon function for the RIP process.

Command Default

When you do not specify this command in configuration mode, RIP routing for interfaces is not enabled.

Command Modes

RIP Router configuration

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

Use the **interface** command to associate a specific interface with a RIP process. The interface remains associated with the process even when the IPv4 address of the interface changes. This command places the router in RIP interface configuration mode or RIP VRF interface configuration mode, from which you can configure interface-specific settings. Commands configured under this mode are automatically bound to that interface. Pointers:

- **metric-zero-accept** - After this command is configured on routing entries from RIP updates, RIP accepts these routes and then sets the metric to one (1).
- **passive-interface** - While RIP stops sending routing updates to the multicast (or broadcast) address on a passive interface, RIP continues to receive and process routing updates from its neighbors on that interface.
- **poison-reverse** - Route poisoning prevents routing loops by communicating to other routers that a route is no longer reachable, effectively removing these routes from other router's routing tables. The system default, split horizon, provides that routes learned through RIP are not advertised from the interface over which they were learned.

  The **poison-reverse** command enables poison reverse processing of RIP router updates. A router that receives route poisoning information sends the poisoning information back to the sending router, a process called poison reverse. This process ensures that all routers on the same interface have received the poisoned route information.

  If both poison reverse and split horizon functions are configured, then simple split horizon behavior (suppression of routes from the interface over which they were learned) is replaced by poison reverse behavior. If split horizon is disabled, the poison reverse configuration is ignored.

receive version and **send version** - Overrides the default behavior of RIP. This command applies only to the interface being configured.

route-policy - The policy can be used to filter routes or modify route attributes. If a route policy is configured on the interface and a VRF, the interface route policy is applied.

site-of-origin - A RIP process must be capable of retrieving the SoO attribute on routes redistributed from BGP when required to support complex topologies that include MPLS VPN links between sites with backdoor links. Use this command to set an SoO BGP extended community attribute that is used to identify routes that have originated from a site so that the readvertisement of that prefix back to the source site can be prevented. The SoO extended community uniquely identifies the site from which a provider edge (PE) router has learned a route.

split-horizon disable - If split horizon is disabled, the poison reverse configuration is ignored. We recommend that you do not change the default state of split horizon unless you are certain that your application requires the change to properly advertise routes.

Examples

The following example shows how to enter interface configuration mode for a RIP process, and send RIP Version 2 messages to the broadcast address on the GigabitEthernet interface 0/1/0/0:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# broadcast-for-v2\
Router(config-rip-if)# commit
```

The following example shows how to set the RIP interface to accept metric zero on routing entries:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# metro-zero-accept
Router(config-rip-if)# commit
```

The following example shows that GigabitEthernet interface 0/1/0/0 stops multicasting (or broadcasting) RIP updates while continuing to receive RIP updates normally. GigabitEthernet interface 0/1/0/3 sends and receives updates normally. Also RIP updates are unicast to neighbor 172.168.1.2 over the appropriate interface:

```
Router(config)# router rip
Router(config-rip)# neighbor 172.16.1.2
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# passive-interface
Router(config-rip-if)# commit
Router(config-rip-if)# exit
Router(config-rip)# interface GigabitEthernet 0/1/0/3
Router(config-rip-if)# exit
```

The following example shows how to enable poison reverse processing for an interface running RIP:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# poison-reverse
Router(config-rip-if)# commit
```

The following example shows how to configure an interface to accept both RIP Version 1 and 2 packets:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
```

```
Router(config-rip-if)# receive version 1 2
Router(config-rip-if)# commit
```

The following example shows how to filter routing updates received on an interface:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# route-policy updpol-1 in
Router(config-rip-if)# commit
```

The following example shows how to configure SoO filtering on a RIP interface:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# site-of-origin 10.0.0.1:20
Router(config-rip-if)# commit
```

The following example shows how to disable split horizon on a Packet-over-SONET/SDH link:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# split-horizon disable
Router(config-rip-if)# commit
```

redistribute (RIP)

To redistribute routes from another routing domain into Routing Information Protocol (RIP), use the **redistribute** command in the appropriate configuration mode. To remove the **redistribute** command from the configuration file and restore the system to its default condition in which the software does not redistribute routes, use the **no** form of this command.

Border Gateway Protocol (BGP)
redistribute bgp *process-id* [**route-policy** *name*] [**external** | **internal** | **local**]
no redistribute bgp *process-id*

Connected Interface Routes
redistribute connected [**route-policy** *name*]
no redistribute connected

Enhanced Interior Gateway Routing Protocol (EIGRP)
redistribute eigrp *process-id* [**route-policy** *name*]
no redistribute eigrp *process-id*

Intermediate System-to-Intermediate System (ISIS)
redistribute isis *process-id* [**route-policy** *name*] [**level-1** | **level-1-2** | **level-2**]
no redistribute isis *process-id*

Open Shortest Path First (OSPF)
redistribute ospf *process-id* [**route-policy** *name*] [**match** { **external** [**1** | **2**] | **internal** | **nssa-external** [**1** | **2**] }]
no redistribute ospf *process-id*

IP Static Routes
redistribute static [**route-policy** *name*]
no redistribute static

Syntax Description

bgp	Distributes routes from the BGP protocol.

process-id	• For the **bgp** keyword:
	  • Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.
	  • Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.
	  • Range for 4-byte Autonomous system numbers (ASNs) is asdot format is 1.0 to 65535.65535.
	• For the **eigrp** keyword, an EIGRP instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.
	• For the **isis** keyword, an IS-IS instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.
	• For the **ospf** keyword, an OSPF instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.

external	(Optional) Specifies BGP external routes only.
internal	(Optional) Specifies BGP internal routes only.
local	(Optional) Specifies BGP local routes only.
route-policy *name*	(Optional) Specifies the identifier of a configured policy. A policy is used to filter the importation of routes from this source routing protocol to RIP.
level-1	(Optional) Redistributes Level 1 IS-IS routes into other routing protocols independently.
level-1-2	(Optional) Distributes both Level 1 and Level 2 IS-IS routes into other routing protocols.
level-2	(Optional) Distributes Level 2 IS-IS routes into other routing protocols independently.
[**match** { **external** [**1** | **2**] | **internal** | **nssa-external** [**1** | **2**]]} [**route-policy** *name*]	(Optional) Specifies the criteria by which OSPF routes are redistributed into other routing domains. It can be one or more of the following: • **internal** —Routes that are internal to a specific autonomous system (intra- and inter-area OSPF routes). • **external** [**1** | **2**]—Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 external routes. • **nssa-external** [**1** | **2**]—Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 not-so-stubby area (NSSA) external routes. For the external and nssa-external options, if a type is not specified, then both Type 1 and Type 2 are assumed. If no match is specified, the default is no filtering.
static	Redistributes IP static routes.

Command Default Route redistribution is disabled.

Command Modes Router configuration

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

Note When you are redistributing routes (into RIP) using both command keywords for setting or matching of attributes and a route policy, the routes are run through the route policy first, followed by the keyword matching and setting.

Redistributed routing information may be filtered by the **route-policy** *name* keyword and argument. This filtering ensures that only those routes intended by the administrator are redistributed by RIP.

The RIP metric used for redistributed routes is determined by the route policy. If a route policy is not configured or the route policy does not set the RIP metric, the metric is determined based on the redistributed protocol.

In all other cases (BGP, IS-IS, OSPF, EIGRP, connected, static), the metric set by the **default-metric** command is used. If a valid metric cannot be determined, then redistribution does not happen.

For information about routing policies, see the *Routing Policy Commands on* module of the *Routing Command Reference*.

Examples The following example shows how to cause BGP routes to be redistributed into a RIP process:

```
RP/0/RP0/CPU0:router(config)# router rip
RP/0/RP0/CPU0:router(config-rip)# redistribute bgp 100
RP/0/RP0/CPU0:router(config-rip)# commit
```

Related Commands

Command	Description
router rip	Configures the RIP routing process.

router rip

To configure a routing process and enter router configuration mode for a Routing Information Protocol (RIP) process, use the **router rip** command in mode. To turn off the RIP routing process, use the **no** form of this command.

router rip [**vrf** *name*] [**auto-summary** | **broadcast-for-v2** | **default-information originate** [**route-policy** *name* [**parameter-value**]] | **route-policy** *name* [**parameter-value**] { **in** | **out** } | { **default-metric** | **maximum-paths** | **output-delay** } *value* | **distance** *value* [*IP-address*] | **neighbor** *IP-address* | **nsf** | **timers basic** *timer-values* | **validate-update-source disable**]
no router rip [**vrf** *name*] [**auto-summary** | **broadcast-for-v2** | **default-information originate** | **route-policy** *name* [**parameter-value**] { **in** | **out** } | **default-metric** [*value*] | **maximum-paths** | **output-delay** *value* | **distance** *value* [*IP-address*] | **neighbor** *IP-address* | **nsf** | **timers basic** | **validate-update-source disable**]

Syntax Description

auto-summary	(Optional) Enables the automatic summarization of subnet routes into network-level routes. By default, this option is disabled.
broadcast-for-v2	(Optional) Sends RIPv2 messages to the broadcast address. By default, RIPv2 output packets are not broadcasted.
default-information originate [**route-policy** *name* [*parameter-value*]]	(Optional) Generates a default route into the RIP process. The route policy indicates the criteria for the default route. By default, this option is disabled.
route-policy *name* [*(parameter-value)*] {**in**|**out**}	(Optional) Applies a routing policy with specific parameter values on inbound and outbound routes.
default-metric *value*	(Optional) Set default metric values for routes redistributed from other protocols into RIP. By default, metrics are not set.
maximum-paths *value*	(Optional) Configures the maximum number of equal cost parallel routes that RIP installs into the routing table.
output-delay *value*	(Optional) Updates the interpacket delay duration for outgoing RIP updates. For *value*, enter a delay value of 8 to 50 (milliseconds') duration. The delay is applied between consecutive packets in a multiple-packet RIP update. By default, delay duration is not enabled.
distance *value* [*IP-address*]	(Optional) Defines the administrative distance assigned to routes discovered by RIP. By default, an administrative distance value of 120 is configured.

neighbor *IP-address*	(Optional) Defines a neighboring router with which to exchange RIP information. For *IP address*, enter the IP address of the peer router with which routing information is exchanged. By default, no RIP neighbor or peer is configured.
nsf	(Optional) Configures nonstop forwarding (NSF) on RIP routes after a RIP process shutdown or restart. By default, this option is disabled.
timers basic *timer-values*	(Optional) Adjusts RIP network timers. For the *timer-values* argument, you can use these 4 timer values: • *update* - This is the rate, in seconds, at which updates are sent, and the fundamental timing parameter of the routing protocol. Range is 5 to 50000, and default value is 30. • *invalid* - This is the interval, in seconds, after which a route is declared invalid. It should be at least three times the value of the *update* value. A route becomes invalid when there is an absence of updates that refresh the route. The route then enters the holddown state. The route is marked inaccessible and is advertised as unreachable. Range is 15 to 200000, and default value is 180. • *holddown* - This is the interval, in seconds, during which routing information regarding better paths is suppressed. It should be at least three times the value of the *update* argument. A route enters into a holddown state when an update packet is received that indicates that the route is unreachable. The route is marked inaccessible and is advertised as unreachable. When holddown expires, routes advertised by other sources are accepted, and the route is no longer inaccessible. Range is 15 to 200000 and default value is 180. • *flush* - This is the time, in seconds, that must pass before the route is removed from the routing table; the interval specified should be greater than the value of the *invalid* argument. If it is less than the invalid timer value, the proper holddown interval cannot elapse, which results in a new route being accepted before the holddown interval expires. Range is 16 to 250000 and default value is 240.
validate-update-source disable	(Optional) Stops the Cisco IOS XR software from validating the source IP address of incoming RIP routing updates. By default, the source IP address of incoming RIP updates is validated.

Command Default

No router process is defined.

Command Modes

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

Use the **router rip** command to enable RIP protocol parameters. This command places the router in RIP configuration mode or RIP VRF configuration mode (for the **router rip vrf** *name* command form), where you can configure RIP-specific settings. Pointers:

Note The **interface** and **redistribute** parameters within the **router rip** command are documented as separate commands.

- **auto-summary** - Use this option to turn on route summarization. Route summarization reduces the amount of routing information in the routing tables. Disable automatic summarization if you must perform routing between disconnected subnets. When automatic summarization is off, subnets are advertised. Automatic summarization is disabled by default.
- **broadcast-for-v2** - Use this option to broadcast RIPv2 broadcast updates to hosts that do not listen to multicasts. RIPv2 requests and responses are sent to the IP broadcast address 255.255.255.255 instead of the IP multicast address 244.0.0.9.
- **route-policy** - Use this command to specify a routing policy for an inbound or outbound route. The policy can be used to filter routes or modify route attributes. If a route policy is configured both on the interface and on the VRF, the interface route policy is applied.
- **default-metric** - Use this option with the **redistribute** command to cause RIP to use the same metric value for all redistributed routes. A default metric helps solve the problem of redistributing routes with incompatible metrics by providing a reasonable substitute and enables redistribution to proceed. If you want to set different metrics for other redistributed protocols, use the **route-policy** option in the **redistribute** command.

  The RIP metric used for redistributed routes is determined by the route policy. If a route policy is not configured or the route policy does not set the RIP metric, the metric is determined based on the redistributed protocol. For VPNv4 routes redistributed by BGP, the RIP metric set at the remote PE router is used, if valid.

  In all other cases (BGP, IS-IS, OSPF, EIGRP, connected, static), the metric set by the **default-metric** command is used. If a valid metric cannot be determined, then redistribution does not happen.
- **output-delay** - Use this command if traffic is sent to a low-speed router that might not be able to receive high speed updates. Configuring this command helps prevent the routing table from losing information.
- **distance** - Use this command to change the preference of RIP routes over other protocol routes. When administrative distance and redistribution features are used together, routing behavior may be affected for routes accepted from and advertised to RIP neighbors.

  Numerically, an administrative distance is an integer from 0 to 255. In general, the higher the value, the lower the trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. The order in which you enter distance commands can affect the assigned administrative distances in unexpected ways.

This table lists default administrative distances.

Table 1: Default Administrative Distances of Routing

Routing Protocol	Administrative Distance Value
Connected interface	0
Static route out an interface	0
Static route to next-hop	1
EIGRP Summary Route	5
External BGP	20
Internal EIGRP	90
OSPF	110
IS-IS	115
RIP Versions 1 and 2	120
External EIGRP	170
Internal BGP	200
Unknown	255

- **neighbor** - Use this option to permit the point-to-point (nonbroadcast) exchange of routing information. When the **neighbor** command is used in combination with **passive-interface** in RIP router configuration mode, routing information can be exchanged between a subset of routers and access servers on a LAN.

  You can specify multiple neighbors or peers.

- **nsf** - When you use this command, NSF lifetime is automatically set to two times the update time (with a minimum value of 60 seconds). The RIP process must reconverge within this time. If the convergence exceeds the NSF lifetime, routes are purged from the Routing Information Base (RIB) and NSF may fail.

- **timers basic** - The basic timing parameters for RIP are adjustable. Because RIP is running a distributed, asynchronous routing algorithm, these timers must be the same for all routers in the network.

  Use the **show rip** command to view the current and default timer values.

- **validate-update-source disable** - When the **validate-update-source disable** command is used, validation is not performed.

  By default, the software ensures that the source IP address of incoming routing updates is on the same IP network as one of the addresses defined for the receiving interface. For unnumbered IP interfaces (interfaces configured as IP unnumbered), no checking is performed.

Examples

The following example shows how to turn on RIP auto-summarization:

```
Router(config)# router rip
Router(config-rip)# auto-summary
Router(config-rip)# commit
```

The following example shows how to send RIP v2 output messages to a broadcast address for all RIP interfaces:

```
Router(config)# router rip
Router(config-rip)# broadcast-for-v2
Router(config-rip)# commit
```

The following example shows how to originate a default route in RIP updates based on the result of running the route policy on the routing table:

```
Router(config)# router rip
Router(config-rip)# default-information originate route-policy policy1
Router(config-rip)# commit
```

The following example shows how to filter routing updates received on an interface:

```
Router(config)# router rip
Router(config-rip)# interface GigabitEthernet 0/1/0/0
Router(config-rip-if)# route-policy updpol-1 in
Router(config-rip-if)# commit
```

The following example shows how a router in autonomous system 109 uses both the RIP and the Open Shortest Path First (OSPF) routing protocols. The example advertises OSPF-derived routes using RIP and assigns the OSPF-derived routes a RIP metric of 10:

```
Router(config)# router rip
Router(config-rip)# vrf vpn-1
Router(config-rip-vrf)# default-metric 10
Router(config-rip-vrf)# redistribute ospf 109
Router(config-rip-vrf)# commit
```

The following example shows how to allow a maximum of 16 equal cost paths to a destination:

```
Router(config)# router rip
Router(config-rip)# maximum-paths 16
Router(config-rip)# commit
```

The following example shows how to set the administrative distance for a particular prefix:

```
Router(config)# router rip
Router(config-rip)# distance 85 192.168.10.0/24
Router(config-rip)# commit
```

The following example shows how to configure RIP NSF:

```
Router(config)# router rip
Router(config-rip)# vrf vpn-1
Router(config-rip-vrf)# nsf
Router(config-rip-vrf)# commit
```

The following example shows how to permit the sending of RIP updates to specific neighbors. One copy of the update is generated per neighbor:

```
Router(config)# router rip
Router(config-rip)# vrf vpn-1
```

```
Router(config-rip-vrf)# neighbor 172.16.1.2
Router(config-rip-vrf)# commit
```

The following example shows how to set updates to be broadcast every 5 seconds. If a router is not heard from in 15 seconds, the route is declared unusable. Further information is suppressed for an additional 15 seconds. At the end of the flush period, the route is flushed from the routing table.

```
Router(config)# router rip
Router(config-rip)# timers basic 5 15 15 30
Router(config-rip)# commit
```

The following example shows how to disable source validation:

```
Router(config)# router rip
Router(config-rip)# validate-update-source disable
Router(config-rip)# commit
```

show protocols (RIP)

To display information about the Routing Information Protocol (RIP) process configuration, use the **show protocols** command in mode.

show protocols [**ipv4** | **afi-all**] [**all** *protocol*] [**default-context**] [**private**]

Syntax Description

ipv4	(Optional) Specifies an IPv4 address family.
afi-all	(Optional) Specifies all address families.
all	(Optional) Specifies all protocols for a given address family.
protocol	(Optional) Specifies a routing protocol. • For the IPv4 address family, the options are **eigrp** , **bgp** , **isis** , **ospf** , and **rip** .
default-context	(Optional) Displays default context information. This keyword is available when the **eigrp** or **rip** protocol is specified.
private	(Optional) Displays private EIGRP data. This keyword is available when the **eigrp** protocol is specified.

Command Modes

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

Use the **show protocols** command to get information about the protocols running on the router and to quickly determine which protocols are active. The command summarizes the important characteristics of the running protocol, and command output varies depending on the specific protocol selected.

For RIP, the command output lists the instance number, default AS context, router ID, default networks, distance, maximum paths, and so on.

Examples

The following is sample output from the **show protocols rip** command:

```
Router# show protocols rip
Routing Protocol RIP
  2 VRFs (including default) configured, 2 active
  25 routes, 16 paths have been allocated
  Current OOM state is "Normal"
  UDP socket descriptor is 37
    VRF             Active   If-config If-active Routes     Paths      Updates
    default         Active   3         3         11         7              30s
```

show rip

To display configuration and status of Routing Information Protocol (RIP), use the **show rip** command in mode.

show rip [**vrf** { **name** | **all** }] [**database** [*IP-address*] | **interface** [*type number*] | **statistics**]

Syntax Description

vrf { *name* | **all** }	(Optional) Specifies a specific VRF instance or all VRF instances associated with the RIP process.
database [*IP-address*]	(Optional) Displays RIP topology table information.
interface [*type number*]	(Optional) Displays RIP interface information for all interfaces, or a specific interface.
statistics	(Optional) Displays RIP statistics information.

Command Default

No default behavior or values

Command Modes

Command History

Release	Modification
Release 7.4.1	This command was introduced.

Usage Guidelines

Pointers on **show rip** command's options:

- **database** *IP-address* - Summary address entries appear in the database only if relevant child routes are summarized. When the last child route for a summary address becomes invalid, the summary address is also removed from the routing table.

Examples

The following example shows sample output from the **show rip** command form:

```
Router# show rip

RIP config:
Active?:                    Yes
Added to socket?:           Yes
Out-of-memory state:        Normal
Version:                    2
Default metric:             Not set
Maximum paths:              4
Auto summarize?:            No
Broadcast for V2?:          No
Packet source validation?:  Yes
NSF:                        Disabled
Timers: Update:             30 seconds (25 seconds until next update)
        Invalid:            180 seconds
        Holddown:           180 seconds
        Flush:              240 seconds
```

This table describes the significant fields shown in the display.

Table 2: show rip Field Descriptions

Field	Description
Active?	Active state setting.
Out-of-memory state	Out-of-memory state for RIP can be one of the following: Normal, Minor, Severe, or Critical.
Version	Version number is 2.
Default metric	Default metric value, if configured. Otherwise Not set.
Maximum paths	Number of maximum paths allowed per RIP route.
Auto summarize?	Auto-summarize state setting.
Broadcast for V2?	RIP Version 2 broadcast setting.
Packet source validation?	Validation setting for the source IP address of incoming routing updates to RIP.
Timers	RIP network timer settings.

The following is sample output from the **show rip database** command form:

```
Router# show rip database

  Routes held in RIP's topology database:
      10.0.0.0/24
          [0]    directly connected, GigabitEthernet0/6/0/0
      10.0.0.0/8    auto-summary
      12.0.0.0/24
          [5] distance: 20    redistributed
      12.0.0.0/8    auto-summary
      50.50.0.0/24
          [1] via 10.0.0.20, next-hop 10.0.0.20, Uptime: 1s, GigabitEthernet0/6/0/0
      50.50.1.0/24 (inactive)
          [1] via 10.0.0.20, next-hop 10.0.0.20, Uptime: 1s, GigabitEthernet0/6/0/0
      50.0.0.0/8    auto-summary
      90.90.0.0/24
          [5] distance: 20    redistributed
      90.90.1.0/24
          [5] distance: 20    redistributed
```

This table describes the significant fields shown in the display.

Table 3: show rip database Field Descriptions

Field	Description
10.0.0.0/24 [0] directly connected, GigabitEthernet0/6/0/0	Prefix and prefix length for a RIP connected route. 10.0.0.0/24 is directly connected to GigabitEthernet 0/6/0/0. The [0] represents the metric.
12.0.0.0/24 [5] distance: 20 redistributed	12.0.0.0/24 is a redistributed route. The metric is 5, and the distance is 20.

Field	Description
50.50.0.0/24 [1] via 10.0.0.20, next-hop 10.0.0.20, Uptime: 1s, GigabitEthernet0/6/0/0	The destination route 50.50.0.0/24 is learned through RIP, and the source 10.0.0.20 advertised it from GigabitEthernet 0/6/0/0. The route was last updated one second ago.
50.50.1.0/24 (inactive) [1] via 10.0.0.20, next hop 10.0.0.20, Uptime: 1s, GigabitEthernet0/6/0/0	The destination route 50.50.1.0/24 is not active in the routing table.

This example is sample output from the **show rip interface** command form:

```
Router# show rip interface

  GigabitEthernet0_6_0_0
  Rip enabled?:               Yes
  Out-of-memory state:        Normal
  Broadcast for V2:           No
  Accept Metric 0?:           No
  Send versions:              2
  Receive versions:           2
  Interface state:            Up
  IP address:                 10.0.0.12/24
  Metric Cost:                0
  Split horizon:              Enabled
  Poison Reverse:             Disabled
  Joined multicast group?:    Yes

  GigabitEthernet0_6_0_2
  Out-of-memory state:        Normal
  Rip enabled?:               Yes
  Broadcast for V2:           No
  Accept Metric 0?:           No
  Send versions:              2
  Receive versions:           2
  Interface state:            Up
  IP address:                 12.0.0.12/24
  Metric Cost:                0
  Split horizon:              Enabled
  Poison Reverse:             Disabled
  Joined multicast group?:    Yes

  RIP peers attached to this interface:
      12.0.0.13
          uptime: 3    version: 2
          packets discarded: 0    routes discarded: 402
```

This table describes the significant fields shown in the display.

Table 4: show rip interface Field Descriptions

Field	Description
Out-of-memory state	Specifies the current out-of-memory state on the interface.
Accept Metric 0?	Specifies whether this interface accepts routing entries from RIP updates with a metric set to zero (0).

Field	Description
Send versions: Receive versions:	Specifies the RIP version used to send and receive packets on the interface.
Joined multicast group?:	Specifies whether the interface has joined the RIP multicast group 224.0.0.9.
RIP peers attached to this interface	Lists the RIP neighbors on this interface, and corresponding details.

Authentication Keychain Configuration for RIP Interface on Default VRF

These examples are output of the **show rip interface** command to display authentication keychain configuration for RIP interface on default VRF.

When an existing keychain with MD5 cryptographic algorithm was configured on the RIP interface:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No
 Authentication mode: MD5  Key chain: <key-chain-name>
 Current active send  key id: <send key id>
 Current active receive key id: <recv key id>
 Packets received: <num-rx-packets>
 Authenticated packets received: <num-auth-rx-packets>
 Packets dropped due to wrong keychain config: <num-rx-wrong-auth-cfg-pkts>
 Packets received without authentication data: <num-rx-auth-missing-pkt>
 Packets received with invalid authentication: <num-rx-invalid-auth-pkt>
```

When the keychain configured on the RIP interface does not exists or does not have any active keys:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No

 Authentication mode: MD5  Key chain: <key-chain-name>
 No active key found  in keychain database.
 Packets received: <num-rx-packets>
```

```
 Authenticated packets received: <num-auth-rx-packets>
 Packets dropped due to wrong keychain config: <num-rx-wrong-auth-cfg-pkts>
 Packets received without authentication data: <num-rx-auth-missing-pkt>
 Packets received with invalid authentication: <num-rx-invalid-auth-pkt>
```

When an active key exists in the keychain configured on the RIP interface, but not configured with MD5 cryptographic algorithm:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No

 Authentication mode: MD5  Key chain: <key-chain-name>
 Key(s) not configured with MD5 cryptographic algorithm.
 Packets received: <num-rx-packets>
 Authenticated packets received: <num-auth-rx-packets>
 Packets dropped due to wrong keychain config: <num-rx-wrong-auth-cfg-pkts>
 Packets received without authentication data: <num-rx-auth-missing-pkt>
 Packets received with invalid authentication: <num-rx-invalid-auth-pkt>
```

When no authentication keychain was configured on the RIP interface:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
 LPTS filter set?:           No

Authentication mode is not set.
 Packets received: <num-rx-packets>
```

Authentication keychain Configuration for RIP Interface on Non-default VRF

These examples are output of the **show rip vrf interface** command to display authentication keychain configuration for RIP interface on a non- default VRF.

When an existing keychain with MD5 cryptographic algorithm has been configured on the RIP interface:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
```

```
Accept Metric 0?:           No
Send versions:                2
Receive versions:           2
Interface state:             Unknown State
IP address:                   0.0.0.0/0
Metric Cost:                0
Split horizon:                Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No

 Authentication mode: MD5  Key chain: <key-chain-name>
 Packets received: <num-rx-packets>
 Authenticated packets received: <num-auth-rx-packets>
 Packets dropped due to wrong keychain config: <num-rx-wrong-auth-cfg-pkts>
 Packets received without authentication data: <num-rx-auth-missing-pkt>
 Packets received with invalid authentication: <num-rx-invalid-auth-pkt>
```

When the keychain configured on the RIP interface does not exist or does not have any active keys:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No

 Authentication mode: MD5  Key chain: <key-chain-name>
 No active key found  in keychain database.
 Packets received: <num-rx-packets>
 Authenticated packets received: <num-auth-rx-packets>
 Packets dropped due to wrong keychain config: <num-rx-wrong-auth-cfg-pkts>
 Packets received without authentication data: <num-rx-auth-missing-pkt>
 Packets received with invalid authentication: <num-rx-invalid-auth-pkt>
```

When an active key exists in the keychain configured on the RIP interface, but not configured with MD5 cryptographic algorithm:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No

 Authentication mode: MD5  Key chain: <key-chain-name>
```

```
 Key(s) not configured with MD5 cryptographic algorithm.
 Packets received: <num-rx-packets>
 Authenticated packets received: <num-auth-rx-packets>
 Packets dropped due to wrong keychain config: <num-rx-wrong-auth-cfg-pkts>
 Packets received without authentication data: <num-rx-auth-missing-pkt>
 Packets received with invalid authentication: <num-rx-invalid-auth-pkt>
```

When no authentication keychain has been configured on the RIP interface:

```
GigabitEthernet0/3/0/3 (Forward Reference)
Rip enabled?:               No
Out-of-memory state:        Normal
Broadcast for V2:           No
Accept Metric 0?:           No
Send versions:              2
Receive versions:           2
Interface state:            Unknown State
IP address:                 0.0.0.0/0
Metric Cost:                0
Split horizon:              Enabled
Poison Reverse:             Disabled
Socket set options:
Joined multicast group?:    No
LPTS filter set?:           No

 Authentication mode is not set.
 Packets received: <num-rx-packets>
```

The following is sample output from the **show rip statistics** command form:

```
Router# show rip statistics

  RIP statistics:
      Total messages sent:        5597
      Message send failures:      0
      Regular updates sent:       5566
      Queries responsed to:       0
      RIB updates:                6
      Total packets received:     5743
      Discarded packets:          0
      Discarded routes:           0
      Number of routes allocated: 18
      Number of paths allocated:  14

      Route malloc failures:      0
      Path malloc failures:       0
```

This table describes the significant fields shown in the display.

Table 5: show rip statistics Field Descriptions

Field	Description
RIB updates	Number of route addition and deletion messages sent to RIB.
Number of routes allocated Number of paths allocated	Number of routes or paths allocated for the RIP internal topology database.
Route malloc failures Path malloc failures	Number of route and path failures during route allocation.