VYATTA, INC. | Vyatta System

RIPng

REFERENCE GUIDE

RIPng

Vyatta
Suite 200
1301 Shoreway Road
Belmont, CA 94002
vyatta.com
650 413 7200
1 888 VYATTA 1 (US and Canada)

COPYRIGHT

PROPRIETARY NOTICES

RELEASE DATE: October 2012

DOCUMENT REVISION. 6.5R1 v01

RELEASED WITH: 6.5R1

PART NO. A0-0239-10-0007

Contents

Quick List of Commands

Use this list to help you quickly locate commands.

List of Examples

Use this list to help you locate examples you'd like to look at or try.

Preface

This document describes the various deployment, installation, and upgrade options for Vyatta software.

This preface provides information about using this guide. The following topics are presented:

Intended Audience

This guide is intended for experienced system and network administrators. Depending on the functionality to be used, readers should have specific knowledge in the following areas:

- Networking and data communications
- TCP/IP protocols
- General router configuration
- Routing protocols
- Network administration
- Network security
- IP services

Organization of This Guide

This guide has the following aid to help you find the information you are looking for:

- Quick List of Commands

  Use this list to help you quickly locate commands.

- List of Examples

  Use this list to help you locate examples you'd like to try or look at.

This guide has the following chapters:

Chapter	Description	Page
Chapter 1: RIPng Configuration	This chapter describes how to configure the Routing Information Protocol next generation (RIPng) on the Vyatta System.	1
Chapter 2: Router-Level Configuration	This chapter describes commands for configuring RIPng at the router level.	8
Chapter 3: Route Redistribution	This chapter describes commands for redistributing routes from other routing protocols into RIPng.	33
Chapter 4: Route Filtering	This chapter describes commands for RIPng route filtering.	45

Document Conventions

This guide uses the following advisory paragraphs, as follows.

WARNING *Warnings alert you to situations that may pose a threat to personal safety.*

CAUTION *Cautions alert you to situations that might cause harm to your system or damage to equipment, or that may affect service.*

NOTE *Notes provide information you might need to avoid problems or configuration errors.*

This document uses the following typographic conventions.

`Monospace`	Examples, command-line output, and representations of configuration nodes.
`bold Monospace`	Your input: something you type at a command line.
bold	Commands, keywords, and file names, when mentioned inline. Objects in the user interface, such as tabs, buttons, screens, and panes.
italics	An argument or variable where you supply a value.
<key>	A key on your keyboard, such as <Enter>. Combinations of keys are joined by plus signs (“+”), as in <Ctrl>+c.
[key1 | key2]	Enumerated options for completing a syntax. An example is [enable | disable].
num1–numN	A inclusive range of numbers. An example is 1–65535, which means 1 through 65535, inclusive.
arg1..argN	A range of enumerated values. An example is eth0..eth3, which means eth0, eth1, eth2, or eth3.
arg[*arg*...] *arg*[,*arg*...]	A value that can optionally represent a list of elements (a space-separated list and a comma-separated list, respectively).

Vyatta Publications

Full product documentation is provided in the Vyatta technical library. To see what documentation is available for your release, see the *Guide to Vyatta Documentation*. This guide is posted with every release of Vyatta software and provides a great starting point for finding the information you need.

Additional information is available on www.vyatta.com and www.vyatta.org.

Chapter 1: RIPng Configuration

This chapter describes how to configure the Routing Information Protocol next generation (RIPng) on the Vyatta System.

The following topics are covered:

RIPng Overview

Routing Information Protocol next generation (RIPng) is a dynamic routing protocol suitable for small, homogenous IPv6 networks. It is classified as an interior gateway protocol (IGP) and employs the distance-vector routing algorithm. RIPng determines the best path by counting the hops to the destination. The maximum hop count is 15 (16 is considered an infinite distance), making RIPng less suitable for large networks. RIPng is an extemtion of RIP Version 2 for IPv6.

Supported Standards

The Vyatta implementation of RIPng complies with the following standards:

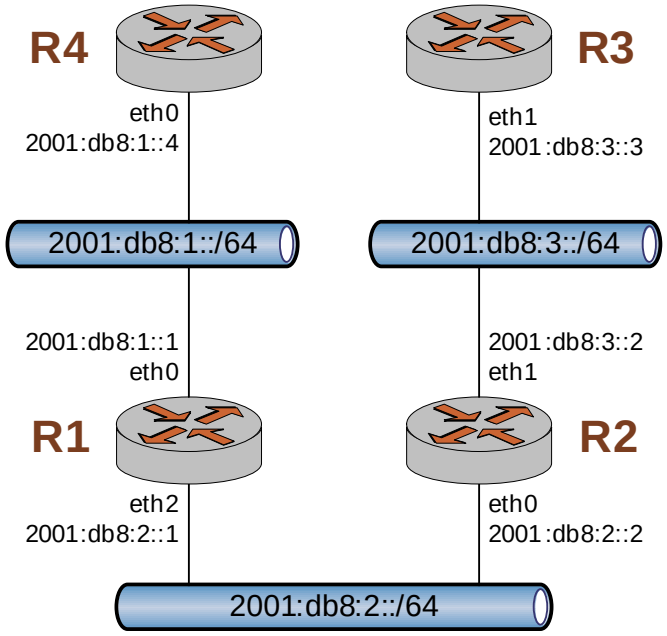
- RFC 2080: RIPng for IPv6
- RFC 2081: RIPng Protocol Applicability Statement

Configuring RIPng

This section presents the following topics:

This section presents a sample configuration for RIPng. The configuration example is based on the reference diagram in Figure 1-1. In this example we will show configuration of the nodes using dynamic IPv6 routing with RIPng to enable R3 and R4 to communicate via R1 and R2.

Figure 1-1 Dynamic IPv6 routing example - RIPng



Enable Forwarding on R1 and R2

For R1 to be able to pass data between interfaces eth0 and eth2, and for R2 to be able to pass data between interfaces eth0 and eth1, they must be configured to enable forwarding. To enable forwarding on R1, perform the following step in configuration mode.

Example 1-1 Enable forwarding on R1

Step	Command
Enable forwarding on R1.	`vyatta@R1# delete system ipv6 disable-forwarding`
Commit the change.	`vyatta@R1# commit`

To enable forwarding on R2, perform the following steps in configuration mode.

Example 1-2 Enable forwarding on R2

Step	Command
Enable forwarding on R2.	`vyatta@R2# delete system ipv6 disable-forwarding`
Commit the change.	`vyatta@R2# commit`

Enable RIPng on an Interface

In order to enable dynamic routing using RIPng, it must be enabled on the interfaces that are to use it. To enable RIPng on R1, perform the following steps in configuration mode.

Example 1-3 Enable RIPng on R1

Step	Command
Enable RIPng on eth0.	`vyatta@R1# set protocols ripng interface eth0`
Enable RIPng on eth2.	`vyatta@R1# set protocols ripng interface eth2`
Commit the change.	`vyatta@R1# commit`

Change to operational mode.	`vyatta@R1# exit`
	`exit`
	`vyatta@R1:~$`

Verify the status of RIPng.	`vyatta@R1:~$ show ipv6 ripng status`
	`Routing Protocol is "RIPng"`
	`  Sending updates every 30 seconds with +/-50%, next due in 4 seconds`
	`  Timeout after 180 seconds, garbage collect after 120 seconds`
	`  Outgoing update filter list for all interface is not set`
	`  Incoming update filter list for all interface is not set`
	`  Default redistribution metric is 1`
	`  Redistributing:`
	`  Default version control: send version 1, receive version 1`
	`    Interface        Send  Recv`
	`    eth0             1     1`
	`    eth2             1     1`
	`  Routing for Networks:`
	`    eth0`
	`    eth2`
	`  Routing Information Sources:`
	`    Gateway         BadPackets BadRoutes  Distance Last Update`
	`    fe80::20c:29ff:fed7:c4a4`
	`                          0          0        120      00:00:25`

Advertise Connected Networks

The **redistribute** command is then used to advertise the connected networks. To advertise connected networks on R1, perform the following steps in configuration mode.

Example 1-4 Advertise connected networks on R1

Step	Command
Advertise connected networks via ripng.	`vyatta@R1# set protocols ripng redistribute connected`
Commit the change.	`vyatta@R1# commit`

Verify the redistribution.	`Routing Protocol is "RIPng"`
	`  Sending updates every 30 seconds with +/-50%, next due in 29 seconds`
	`  Timeout after 180 seconds, garbage collect after 120 seconds`
	`  Outgoing update filter list for all interface is not set`
	`  Incoming update filter list for all interface is not set`
	`  Default redistribution metric is 1`
	`  Redistributing:  connected`
	`  Default version control: send version 1, receive version 1`
	`  Interface  Send  Recv`
	`  eth0  1  1`
	`  eth2  1  1`
	`  Routing for Networks:`
	`  eth0`
	`  eth2`
	`  Routing Information Sources:`
	`  Gateway  BadPackets BadRoutes  Distance Last Update`
	`  fe80::20c:29ff:fed7:c4a4`
	`  0  0  120  00:00:25`

Confirm Visibility of Remote Networks

After enabling RIPng on the other interfaces of R2, R3, and R4, and advertising connected networks on R2, we can check the routing table of R4 to verify that it has learned the network. To confirm visibility of remote networks on R4, perform the following step in operational mode.

Example 1-5 Confirm visibility of remote networks on R4

Step	Command
Trace the route from R2 to R4.	vyatta@R4:~$ show ipv6 route Codes: K - kernel route, C - connected, S - static, R - RIPng, O - OSPFv3, I - ISIS, B - BGP, * - FIB route. S>* ::/0 [1/0] via 2001:db8:1::1, eth0 C>* ::1/128 is directly connected, lo C>* 2001:db8:1::/64 is directly connected, eth0 R>* 2001:db8:2::/64 [120/2] via fe80::20c:29ff:fed6:816c, eth0, 00:43:00 R>* 2001:db8:3::/64 [120/3] via fe80::20c:29ff:fed6:816c, eth0, 00:00:03 C>* fe80::/64 is directly connected, eth0

The "R" in the first column indicates that two routes have been learned from RIPng. Since there is now a route for 2001:db8:3::/64 we should be able to ping R3. To confirm connectivity, perform the following steps in operational mode.

Example 1-6 Confirm connectivity between R4 and R3

Step	Command
Ping R3 from R4.	vyatta@R4:~$ ping 2001:db8:3::3 PING 2001:db8:3::3(2001:db8:3::3) 56 data bytes 64 bytes from 2001:db8:3::3: icmp_seq=1 ttl=62 time=5.98 ms 64 bytes from 2001:db8:3::3: icmp_seq=2 ttl=62 time=0.603 ms ^C --- 2001:db8:3::3 ping statistics --- 2 packets transmitted, 2 received, 0% packet loss, time 1011ms rtt min/avg/max/mdev = 0.603/3.294/5.986/2.692 ms

Example 1-6 Confirm connectivity between R4 and R3

Display the RIPng status.

```
vyatta@R4:~$ show ipv6 ripng
Codes: R - RIPng, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) -
redistribute,
      (i) - interface, (a/S) - aggregated/Suppressed

  Network     Next Hop                 Via    Metric Tag Time
C(i) 2001:db8:1::/64
                 ::                        self       1    0
R(n) 2001:db8:2::/64
                 fe80::20c:29ff:fed6:816c    eth0       2
0  02:56
R(n) 2001:db8:3::/64
                 fe80::20c:29ff:fed6:816c    eth0       3
0  02:56
```

Chapter 2: Router-Level Configuration

This chapter describes commands for configuring RIPng at the router level.

This chapter presents the following topics:

Router-Level Configuration Commands

This chapter contains the following commands.

Configuration Commands	
protocols ripng aggregate-address <ipv6net>	Specifies an aggregate RIPng route announcement.
protocols ripng default-information originate	Generates a default route into a RIPng routing domain.
protocols ripng default-metric <metric>	Sets the default metric for external routes redistributed into RIPng.
protocols ripng interface <ethx>	Enables the Routing Information Protocol next generation (RIPng) for an interface.
protocols ripng network <ipv6net>	Specifies a network for the RIPng.
protocols ripng passive-interface <ethx>	Suppresses RIPng routing updates on an interface.
protocols ripng route <ipv6net>	Specifies a RIPng static route.
protocols ripng timers garbage-collection <seconds>	Allows you to set timers for RIPng garbage collection.
protocols ripng timers timeout <seconds>	Allows you to set the interval for RIPng time-outs.
protocols ripng timers update <seconds>	Allows you to set the timer for RIPng routing table updates.
Operational Commands	
monitor protocol ripng disable events	Disables debug message generation related to RIPng events.
monitor protocol ripng disable packet	Disables debug message generation related to all RIPng packet types.
monitor protocol ripng disable rib	Disables debug message generation for the RIPng Routing Information Base (RIB.
monitor protocol ripng enable events	Enables debug message generation related to RIPng events.
monitor protocol ripng enable packet	Enables debug message generation related to all RIPng packet types.
monitor protocol ripng enable rib	Enables debug message generation for the RIPng Routing Information Base (RIB.
show ipv6 route ripng	Displays all IPv6 RIPng routes.

show monitoring protocols ripng	Displays RIPng protocol debugging flags.

monitor protocol ripng disable events

Disables debug message generation related to RIPng events.

Syntax

monitor protocol ripng disable events

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to disable generation of trace-level messages related to Routing Information Protocol next generation (RIPng) events.

monitor protocol ripng disable packet

Disables debug message generation related to all RIPng packet types.

Syntax

monitor protocol ripng disable packet [**recv** | **send**]

Command Mode

Operational mode.

Parameters

recv	Optional. Disables debugging on all received packets.
send	Optional. Disables debugging on all sent packets.

Default

None.

Usage Guidelines

Use this command to disable generation of trace-level messages related to all Routing Information Protocol next generation (RIPng) packet types.

monitor protocol ripng disable rib

Disables debug message generation for the RIPng Routing Information Base (RIB.

Syntax

monitor protocol ripng disable rib

Command Mode

Operational mode.

Parameters

None.

Default

Debug messages are disabled for actions related to the RIPng RIB.

Usage Guidelines

Use this command to disable generation of trace-level messages related to the RIPng RIB.

monitor protocol ripng enable events

Enables debug message generation related to RIPng events.

Syntax

monitor protocol ripng enable events

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to enable generation of trace-level messages related to Routing Information Protocol next generation (RIPng) events.

monitor protocol ripng enable packet

Enables debug message generation related to all RIPng packet types.

Syntax

monitor protocol ripng enable packet [**recv** | **send**]

Command Mode

Operational mode.

Parameters

recv	Optional. Provides debugging on all received packets.
send	Optional. Provides debugging on all sent packets.

Default

None.

Usage Guidelines

Use this command to enable generation of trace-level messages related to all Routing Information Protocol next generation (RIPng) packet types.

monitor protocol ripng enable rib

Enables debug message generation for the RIPng Routing Information Base (RIB.

Syntax

monitor protocol ripng enable rib

Command Mode

Operational mode.

Parameters

None.

Default

Debug messages are generated for actions related to the RIPng RIB.

Usage Guidelines

Use this command to enable generation of trace-level messages related to the Routing RIPng RIB.

show monitoring protocols ripng

Displays RIPng protocol debugging flags.

Syntax

show monitoring protocols ripng

Command Mode

Operational mode.

Parameters

None

Default

None.

Usage Guidelines

Use this command to see how debugging is set for RIPng.

protocols ripng aggregate-address <ipv6net>

Specifies an aggregate RIPng route announcement.

Syntax

set protocols ripng aggregate-address *ipv6net*

delete protocols ripng aggregate-address *ipv6net*

show protocols ripng aggregate-address [*ipv6net*]

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        aggregate-address ipv6net
    }
}
```

Parameters

ipv6net	Mandatory. The IPv6 network from which routes are to be aggregated. The format is *ipv6-address/prefix*.

Usage Guidelines

Use this command for IPv6 address aggregation.

Use the **set** form of this command to specify a contiguous block of IPv6 addresses to aggregate.

Use the **delete** form of this command to delete an aggregate address.

Use the **show** form of this command to view aggregate address configuration settings.

protocols ripng default-information originate

Generates a default route into a RIPng routing domain.

Syntax

set protocols ripng default-information originate

delete protocols ripng default-information originate

show protocols ripng default-information originate

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        default-information {
            originate
        }
    }
}
```

Parameters

None.

Default

By default, the system does not generate an external default route into the RIPng routing domain.

Usage Guidelines

Use the **set** form of this command to generate a default route into the RIPng routing domain.

Use the **delete** form of this command to restore the default behavior for default route generation into RIPng.

Use the **show** form of this command to display default route generation configuration.

protocols ripng default-metric <metric>

Sets the default metric for external routes redistributed into RIPng.

Syntax

set protocols ripng default-metric *metric*

delete protocols ripng default-metric

show protocols ripng default-metric

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        default-metric metric
    }
}
```

Parameters

metric	Mandatory. The metric that will be assigned to external routes imported into RIPng for redistribution. The range is 1-16. The default is 1.

Default

Routes being imported into RIPng are assigned a metric of 1.

Usage Guidelines

Use the **set** form of this command to set the metric for routes being redistributed into RIPng.

Use the **delete** form of this command to restore the default RIPng metric to default values.

Use the **show** form of this command to display the default metric for routes being redistributed into RIPng.

protocols ripng interface <ethx>

Enables the Routing Information Protocol next generation (RIPng) for an interface.

Syntax

set protocols ripng interface *ethx*

delete protocols ripng interface *ethx*

show protocols ripng interface *ethx*

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        interface ethx
    }
}
```

Parameters

ethx	Mandatory. Multi-node. The name of a configured Ethernet interface. You can enable RIPng on more than one interface by creating multiple **protocols ripng interface** configuration nodes.

Default

None.

Usage Guidelines

Use the **set** form of this command to enable RIPng on an interface. The interface must be enabled for RIPng before you can use it for RIPng routing.

Use the **delete** form of this command to disable RIPng on an interface.

Use the **show** form of this command to display RIPng interface configuration.

protocols ripng network <ipv6net>

Specifies a network for the RIPng.

Syntax

set protocols ripng network *ipv6net*

delete protocols ripng network *ipv6net*

show protocols ripng network

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        network ipv6net
    }
}
```

Parameters

ipv6net	Mandatory. Multi-node. The IPv6 network address of the RIPng network. You can identify more than one RIPng network by creating multiple **protocols ripng network** configuration nodes.

Default

None.

Usage Guidelines

Use this command to identify Routing Information Protocol next generation (RIPng) networks.

Use the **set** form of this command to specify a RIPng network.

Use the **delete** form of this command to remove a RIPng network.

Use the **show** form of this command to display RIPng network configuration.

protocols ripng passive-interface <ethx>

Suppresses RIPng routing updates on an interface.

Syntax

set protocols ripng passive-interface *ethx*

delete protocols ripng passive-interface *ethx*

show protocols ripng passive-interface

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        passive-interface ethx
    }
}
```

Parameters

ethx	Mandatory. Multi-node. The name of a configured Ethernet interface on which to suppress RIPng routing updates. You can suppress routing updates on more than one RIPng interface by creating multiple **protocols ripng passive-interface** configuration nodes.

Default

RIPng routing updates are not suppressed.

Usage Guidelines

Use the **set** form of this command to suppress RIPng routing updates on an interface

Use the **delete** form of this command to disable RIPng routing update suppression on an interface.

Use the **show** form of this command to display RIPng route suppression configuration for an interface.

protocols ripng route <ipv6net>

Specifies a RIPng static route.

Syntax

set protocols ripng route *ipv6net*

delete protocols ripng route *ipv6net*

show protocols ripng route

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        route ipv6net
    }
}
```

Parameters

ipv6net	Mandatory. The IPv6 network address defining the RIPng static route.

Default

None.

Usage Guidelines

Use this command for setting static routes in Routing Information Protocol next generation (RIPng).

Use the **set** form of this command to define a RIPng static route.

Use the **delete** form of this command to remove a RIPng static route.

Use the **show** form of this command to display RIPng static route configuration.

protocols ripng timers garbage-collection <seconds>

Allows you to set timers for RIPng garbage collection.

Syntax

set protocols ripng timers garbage-collection *seconds*

delete protocols ripng timers garbage-collection [*seconds*]

show protocols ripng timers garbage-collection

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        timers {
            garbage-collection seconds
        }
    }
}
```

Parameters

seconds	Mandatory. The timer interval in seconds. The range is 0 to 65535.

Default

The default is 120.

Usage Guidelines

Use the **set** form of this command to set the garbage collection timer. When the timer expires, the system will scan for stale RIPng resources and release them for use.

Use the **delete** form of this command to restore the default value for the RIPng garbage collection timer.

Use the **show** form of this command to display RIPng garbage collection timer configuration.

protocols ripng timers timeout <seconds>

Allows you to set the interval for RIPng time-outs.

Syntax

set protocols ripng timers timeout *seconds*

delete protocols ripng timers timeout [*seconds*]

show protocols ripng timers timeout

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        timers {
            timeout seconds
        }
    }
}
```

Parameters

seconds	Mandatory. The RIPng timeout interval, in seconds. The range is 0 to 65535. The default is 180.

Default

RIPng time-outs occur at 180 seconds.

Usage Guidelines

Use the **set** form of this command to set the value for RIPng time-outs.

Use the **delete** form of this command to restore the RIPng timeout interval to the default value.

Use the **show** form of this command to display RIPng timeout configuration.

protocols ripng timers update <seconds>

Allows you to set the timer for RIPng routing table updates.

Syntax

set protocols ripng timers update *seconds*

delete protocols ripng timers update [*seconds*]

show protocols ripng timers update

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        timers {
            update seconds
        }
    }
}
```

Parameters

seconds	Mandatory. The interval at which RIPng routing table updates will occur. The range is 0 is 65535. The default is 30.

Default

The RIPng routing table is updated every 30 seconds.

Usage Guidelines

Use the **set** form of this command to set the interval between RIPng routing table updates. The shorter this interval, the more accurate the routing information in the tables; however, the more protocol network traffic occurs.

Use the **delete** form of this command to restore the RIPng update timer to the default value.

Use the **show** form of this command to display the RIPng update time configuration.

show ipv6 route ripng

Displays all IPv6 RIPng routes.

Syntax

show ipv6 route ripng

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display RIPng routes contained in the Routing Information Base (RIB).

Examples

Example 2-1 shows all RIPng routes from the RIB.

Example 2-1 "show ip route ripng": Displaying RIPng routes

```
   vyatta@vyatta:~$ show ipv6 route ripng
Codes: K - kernel route, C - connected, S - static, R - RIPng, O - OSPFv3,
       I - ISIS, B - BGP, * - FIB route.

R>* 2001:db8:2::/64 [120/2] via fe80::20c:29ff:fed6:816c, eth0, 00:43:00
R>* 2001:db8:3::/64 [120/3] via fe80::20c:29ff:fed6:816c, eth0, 00:00:03
   vyatta@vyatta:~$
```

show ipv6 ripng

Displays information for the RIPng protocol.

Syntax

show ipv6 ripng [**status**]

Command Mode

Operational mode.

Parameters

status	Optional. Displays only RIPng protocol status information.

Default

Displays all RIPng protocol information.

Usage Guidelines

Use this command to see information about the Routing Information Protocol next generation (RIPng).

Examples

Example 2-2 lists RIPng information.

Example 2-2 "show ipv6 ripng": Displaying RIPng information

```
   vyatta@vyatta:~$ show ipv6 ripng
Codes: R - RIPng, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
      (n) - normal, (s) - static, (d) - default, (r) - redistribute,
      (i) - interface, (a/S) - aggregated/Suppressed

   Network      Next Hop                      Via     Metric Tag Time
C(i) 2001:db8:1::/64
                  ::                         self       1    0
R(n) 2001:db8:2::/64
                  fe80::20c:29ff:fed6:816c    eth0       2    0  02:56
R(n) 2001:db8:3::/64
```

```
                fe80::20c:29ff:fed6:816c    eth0       3    0  02:56
vyatta@vyatta:~$
```

Example 2-3 lists RIPng protocol status information.

Example 2-3 “show ipv6 ripng status”: Displaying RIPng protocol status information

```
 vyatta@vyatta:~$ show ipv6 ripng status
Routing Protocol is "RIPng"
Sending updates every 30 seconds with +/-50%, next due in 4 seconds
Timeout after 180 seconds, garbage collect after 120 seconds
Outgoing update filter list for all interface is not set
Incoming update filter list for all interface is not set
Default redistribution metric is 1
Redistributing:
Default version control: send version 1, receive version 1
  Interface        Send  Recv
  eth0             1     1
  eth2             1     1
Routing for Networks:
  eth0
  eth2
Routing Information Sources:
  Gateway          BadPackets BadRoutes  Distance Last Update
  fe80::20c:29ff:fed7:c4a4
                       0          0        120      00:00:25
 vyatta@vyatta:~$
```

Chapter 3: Route Redistribution

This chapter describes commands for redistributing routes from other routing protocols into RIPng.

This chapter presents the following topics:

Route Redistribution Commands

This chapter contains the following commands.

Configuration Commands	
protocols ripng redistribute bgp	Allows you to redistribute BGP routes into RIPng routing tables.
protocols ripng redistribute connected	Allows you to redistribute directly connected routes into RIPng routing tables.
protocols ripng redistribute kernel	Allows you to redistribute kernel routes into RIPng routing tables.
protocols ripng redistribute ospfv3	Allows you to redistribute OSPFv3 routes into RIPng routing tables.
protocols ripng redistribute static	Allows you to redistribute static routes into RIPng routing tables.
Operational Commands	
None	

protocols ripng redistribute bgp

Allows you to redistribute BGP routes into RIPng routing tables.

Syntax

set protocols ripng redistribute bgp [**metric** *metric* | **route-map** *map-name*]

delete protocols ripng redistribute bgp [**metric** | **route-map**]

show protocols ripng redistribute bgp [**metric** | **route-map**]

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        redistribute {
            bgp {
                metric metric
                route-map map-name
            }
        }
    }
}
```

Parameters

metric	The routing metric to be applied to BGP routes being imported into RIPng routing tables. The range is 1 to 16. The default is 1.
map-name	Optional. Applies the specified route map to BGP routes being imported into RIPng routing tables.

Default

BGP routes being redistributed into RIPng are assigned a routing metric of 1. By default, no route map is applied to redistributed BGP routes.

Usage Guidelines

Use the **set** form of this command to set the routing metric for BGP routes being redistributed into RIPng, or to specify a route map to be applied to redistributed BGP routes.

Use the **delete** form of this command to remove BGP route redistribution configuration.

Use the **show** form of this command to display BGP route redistribution configuration.

protocols ripng redistribute connected

Allows you to redistribute directly connected routes into RIPng routing tables.

Syntax

set protocols ripng redistribute connected [**metric** *metric* | **route-map** *map-name*]

delete protocols ripng redistribute connected [**metric** | **route-map**]

show protocols ripng redistribute connected [**metric** | **route-map**]

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        redistribute {
            connected {
                metric metric
                route-map map-name
            }
        }
    }
}
```

Parameters

metric	Optional. The routing metric to be applied to connected routes being imported into RIPng routing tables. The range is 1 to 16. The default is 1.
map-name	Optional. Applies the specified route map to connected routes being imported into RIPng routing tables.

Default

Connected routes being redistributed into RIPng are assigned a routing metric of 1. By default, no route map is applied to redistributed connected routes.

Usage Guidelines

Use the **set** form of this command to set the routing metric for connected routes being redistributed into RIPng, or to specify a route map to be applied to redistributed connected routes.

Use the **delete** form of this command to remove connected route redistribution configuration.

Use the **show** form of this command to display connected route redistribution configuration.

protocols ripng redistribute kernel

Allows you to redistribute kernel routes into RIPng routing tables.

Syntax

set protocols ripng redistribute kernel [**metric** *metric* | **route-map** *map-name*]

delete protocols ripng redistribute kernel [**metric** | **route-map**]

show protocols ripng redistribute kernel [**metric** | **route-map**]

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        redistribute {
            kernel {
                metric metric
                route-map map-name
            }
        }
    }
}
```

Parameters

metric	Optional. The routing metric to be applied to kernel routes being imported into RIPng routing tables. The range is 1 to 16. The default is 1.
map-name	Optional. Applies the specified route map to kernel routes being imported into RIPng routing tables.

Default

Kernel routes being redistributed into RIPng are assigned a routing metric of 1. By default, no route map is applied to redistributed kernel routes.

Usage Guidelines

Use the **set** form of this command to set the routing metric for kernel routes being redistributed into RIPng, or to specify a route map to be applied to redistributed kernel routes.

Use the **delete** form of this command to remove kernel route redistribution configuration.

Use the **show** form of this command to display kernel route redistribution configuration.

protocols ripng redistribute ospfv3

Allows you to redistribute OSPFv3 routes into RIPng routing tables.

Syntax

set protocols ripng redistribute ospfv3 [**metric** *metric* | **route-map** *map-name*]

delete protocols ripng redistribute ospfv3 [**metric** | **route-map**]

show protocols ripng redistribute ospfv3 [**metric** | **route-map**]

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        redistribute {
            ospfv3 {
                metric metric
                route-map map-name
            }
        }
    }
}
```

Parameters

metric	Optional. The routing metric to be applied to OSPFv3 routes being imported into RIPng routing tables. The range is 1 to 16. The default is 1.
map-name	Optional. Applies the specified route map to OSPFv3 routes being imported into RIPng routing tables.

Default

OSPFv3 routes being redistributed into RIPng are assigned a routing metric of 1. By default, no route map is applied to redistributed OSPFv3 routes.

Usage Guidelines

Use the **set** form of this command to set the routing metric for OSPFv3 routes being redistributed into RIPng, or to specify a route map to be applied to redistributed OSPFv3 routes.

Use the **delete** form of this command to remove OSPFv3 route redistribution configuration.

Use the **show** form of this command to display OSPFv3 route redistribution configuration.

protocols ripng redistribute static

Allows you to redistribute static routes into RIPng routing tables.

Syntax

set protocols ripng redistribute static [**metric** *metric* | **route-map** *map-name*]

delete protocols ripng redistribute static [**metric** | **route-map**]

show protocols ripng redistribute static [**metric** | **route-map**]

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        redistribute {
            static {
                metric metric
                route-map map-name
            }
        }
    }
}
```

Parameters

metric	Optional. The routing metric to be applied to static routes being imported into RIPng routing tables. The range is 1 to 16. The default is 1.
map-name	Optional. Applies the specified route map to static routes being imported into RIPng routing tables.

Default

Static routes being redistributed into RIPng are assigned a routing metric of 1. By default, no route map is applied to redistributed static routes.

Usage Guidelines

Use the **set** form of this command to set the routing metric for static routes being redistributed into RIPng, or to specify a route map to be applied to redistributed static routes.

Use the **delete** form of this command to remove static route redistribution configuration.

Use the **show** form of this command to display static route redistribution configuration.

Chapter 4: Route Filtering

This chapter describes commands for RIPng route filtering.

This chapter presents the following topics:

RIPng Route Filtering Commands

This chapter contains the following commands.

Configuration Commands	
protocols ripng distribute-list access-list	Applies an access list for filtering inbound or outbound RIPng packets.
protocols ripng distribute-list interface <ethx> access-list	Applies an access list to a specific interface for filtering inbound or outbound RIPng packets.
protocols ripng distribute-list interface <ethx> prefix-list	Applies a prefix list to a specific interface for filtering inbound or outbound RIPng packets.
protocols ripng distribute-list prefix-list	Applies a prefix list for filtering inbound or outbound RIPng packets.
Operational Commands	
None.	

protocols ripng distribute-list access-list

Applies an access list for filtering inbound or outbound RIPng packets.

Syntax

set protocols ripng distribute-list access-list {**in** *in-list* | **out** *out-list*}

delete protocols ripng distribute-list access-list {**in** | **out**}

show protocols ripng distribute-list access-list {**in** | **out**}

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        distribute-list {
            access-list {
                in in-list
                out out-list
            }
        }
    }
}
```

Parameters

in-list	The identifier of a defined access list. The access list will be applied to filter inbound RIPng packets.
out-list	The identifier of a defined access list. The access list will be applied to filter outbound RIPng packets.

Default

None.

Usage Guidelines

Use the **set** form of this command to apply an access list for filtering inbound or outbound RIPng packets.

Use the **delete** form of this command to remove access list packet filtering from RIPng packets.

Use the **show** form of this command to display RIPng access list filtering configuration.

protocols ripng distribute-list interface <ethx> access-list

Applies an access list to a specific interface for filtering inbound or outbound RIPng packets.

Syntax

set protocols ripng distribute-list interface *ethx* **access-list** {**in** *in-list* | **out** *out-list*}

delete protocols ripng distribute-list interface *ethx* **access-list** {**in** | **out**}

show protocols ripng distribute-list interface *ethx* **access-list** {**in** | **out**}

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        distribute-list {
            interface ethx
                access-list {
                    in in-list
                    out out-list
                }
            }
        }
    }
}
```

Parameters

ethx	Mandatory. Interface on which to filter packets.
in-list	The identifier of a defined access list. The access list will be applied to the specified interface to filter inbound RIPng packets.
out-list	The identifier of a defined access list. The access list will be applied to the specified interface to filter outbound RIPng packets.

Default

None.

Usage Guidelines

Use the **set** form of this command to apply an access list to a specific interface for filtering inbound or outbound RIPng packets.

Use the **delete** form of this command to remove RIPng access list packet filtering from an interface.

Use the **show** form of this command to display RIPng access list filtering configuration for an interface.

protocols ripng distribute-list interface <ethx> prefix-list

Applies a prefix list to a specific interface for filtering inbound or outbound RIPng packets.

Syntax

set protocols ripng distribute-list interface *ethx* **prefix-list** {**in** *in-list* | **out** *out-list*}

delete protocols ripng distribute-list interface *ethx* **prefix-list** {**in** | **out**}

show protocols ripng distribute-list interface *ethx* **prefix-list** {**in** | **out**}

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        distribute-list {
            interface ethx
                prefix-list {
                    in in-list
                    out out-list
                }
            }
        }
    }
}
```

Parameters

ethx	Mandatory. Interface on which to apply the access list filter.
in-list	The identifier of a defined prefix list. The prefix list will be applied to the specified interface to filter inbound RIPng packets.
out-list	The identifier of a defined prefix list. The prefix list will be applied to the specified interface to filter outbound RIPng packets.

Default

None.

Usage Guidelines

Use the **set** form of this command to apply a prefix list to a specific interface for filtering inbound or outbound RIPng packets.

Use the **delete** form of this command to remove RIPng prefix list packet filtering from an interface.

Use the **show** form of this command to display RIPng prefix list filtering configuration for an interface.

protocols ripng distribute-list prefix-list

Applies a prefix list for filtering inbound or outbound RIPng packets.

Syntax

set protocols ripng distribute-list prefix-list {**in** *in-list* | **out** *out-list*}

delete protocols ripng distribute-list prefix-list {**in** | **out**}

show protocols ripng distribute-list prefix-list {**in** | **out**}

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    ripng {
        distribute-list {
            prefix-list {
                in in-list
                out out-list
            }
        }
    }
}
```

Parameters

in-list	The identifier of a defined prefix list. The prefix list will be applied to filter inbound RIPng packets.
out-list	The identifier of a defined prefix list. The prefix list will be applied to filter outbound RIPng packets.

Default

None.

Usage Guidelines

Use the **set** form of this command to apply a prefix list for filtering inbound or outbound RIPng packets.

Use the **delete** form of this command to remove RIPng prefix list packet filtering.

Use the **show** form of this command to display RIPng prefix list filtering configuration.

Chapter 5: RIPng Interface Commands

This chapter describes commands for configuring RIPng on various interfaces.

This chapter presents the following topics:

RIPng Interface Commands

This chapter contains the following commands.

Configuration Commands	
interfaces <interface> ipv6 ripng	Enables RIPng on an interface.
interfaces <interface> ipv6 ripng split-horizon	Configures split-horizon in RIPng updates coming from this interface.
Operational Commands	
None.	

interfaces <interface> ipv6 ripng

Enables RIPng on an interface.

Syntax

set interfaces *interface* **ipv6 ripng**

delete interfaces *interface* **ipv6 ripng**

show interfaces *interface* **ipv6 ripng**

Command Mode

Configuration mode.

Configuration Statement

```
interfaces interface {
    ipv6 {
        ripng
    }
}
```

Parameters

interface	Mandatory. The type of interface. For detailed keywords and arguments that can be specified as interface types, see the table in the Usage Guidelines below.

Default

None.

Usage Guidelines

Use this command to enable Routing Information Protocol next generation (RIPng) on an interface.

The following table shows the syntax and parameters for supported interface types.

Interface Type	Syntax	Parameters
ADSL Bridged Ethernet	adsl *adslx* pvc *pvc-id* bridged-ethernet	*adslx* The name of a Bridged Ethernet- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.
ADSL Classical IPOA	adsl *adslx* pvc *pvc-id* classical-ipoa	*adslx* The name of a Classical IPoA- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.
ADSL PPPoA	adsl *adslx* pvc *pvc-id* pppoa *num*	*adslx* The name of a Classical IPoA- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically. *num* The PPPoA unit number. This number must be unique across all PPPoA interfaces. In addition, only one PPPoA instance can be configured on a PVC. PPPoA units range from 0 to 15 and the resulting interfaces are named pppoa0 to pppoa15.
ADSL PPPoE	adsl *adslx* pvc *pvc-id* pppoe *num*	*adslx* The name of a Classical IPoA- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically. *num* The name of a defined PPPoE unit. The range is 0 to 15.
Bonding	bonding *bondx*	*bondx* The identifier for the bonding interface. Supported values are **bond0** through **bond99**.
Bonding Vif	bonding *bondx* vif *vlan-id*	*bondx* The identifier for the bonding interface. Supported values are **bond0** through **bond99**. *vlan-id* The VLAN ID for the vif. The range is 0 to 4094.

Interface Type	Syntax	Parameters
Bridge	bridge *brx*	*brx* The name of a Bridge group. The range is **br0** through **br999**.
Ethernet	ethernet *ethx*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system.
Ethernet PPPoE	ethernet *ethx* pppoe *num*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system. *num* The name of a defined PPPoE unit. The range is 0 to 15.
Ethernet Vif	ethernet *ethx* vif *vlan-id*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system. *vlan-id* The VLAN ID for the vif. The range is 0 to 4094.
Ethernet Vif PPPoE	ethernet *ethx* vif *vlan-id* pppoe *num*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system. *vlan-id* The VLAN ID for the vif. The range is 0 to 4094. *num* The name of a defined PPPoE unit. The range is 0 to 15.
Loopback	loopback *lo*	*lo* The name of the loopback interface.
Multilink	multilink *mlx* vif *1*	*mlx* The identifier of the multilink bundle. You can create up to two multilink bundles. Supported values are **ml0** (“em ell zero”) through **ml23** (“em ell twenty-three”). *1* The identifier of the virtual interface. Currently, only one vif is supported for multilink interfaces, and the identifier must be 1. The vif must already have been defined.
OpenVPN	openvpn *vtunx*	*vtunx* The identifier for the OpenVPN interface. This may be **vtun0** to **vtun*x***, where ***x*** is a non-negative integer.
Pseudo-Ethernet	pseudo-ethernet *pethx*	*pethx* The name of a pseudo-Ethernet interface. The range is **peth0** through **peth999**.
Serial Cisco HDLC	serial *wanx* cisco-hdlc vif *1*	*wanx* The serial interface you are configuring: one of **wan0** through **wan23**. The interface must already have been defined. *1* The identifier of the virtual interface. Currently, only one vif is supported for Cisco HDLC interfaces, and the identifier must be 1. The vif must already have been defined.

Interface Type	Syntax	Parameters
Serial Frame Relay	serial *wanx* frame-relay vif *dlci*	*wanx* The serial interface you are configuring: one of **wan0** through **wan23**. The interface must already have been defined. *dlci* The identifier of the virtual interface. For Frame Relay interfaces, this is the DLCI number for the interface. the range is 16 to 991. The vif must already have been defined.
Serial PPP	serial *wanx* ppp vif *1*	*wanx* The serial interface you are configuring: one of **wan0** through **wan23**. The interface must already have been defined. *1* The identifier of the virtual interface. Currently, only one vif is supported for point-to-point interfaces, and the identifier must be 1. The vif must already have been defined.
Tunnel	tunnel *tunx*	*tunx* An identifier for the tunnel interface you are defining. This may be **tun0** to **tun*x***, where ***x*** is a non-negative integer.
Virtual Tunnel	vti *vtix*	*vtix* An identifier for the virtual tunnel interface you are defining. This may be **vti0** to **vti*x***, where ***x*** is a non-negative integer. **Note:** This interface does not support IPv6.
VRRP	interface *parent-if* vrrp vrrp-group *group* interface	*parent-if* The type and identifier of the parent interface; for example, **ethernet eth0** or **bonding bond0**. *group* The VRRP group identifier. The name of the VRRP interface is not specified. The system internally constructs the interface name from the parent interface identifier plus the VRRP group number—for example, **eth0v99**, **eth0.15v99**, **bond0v99**, ot **bond0.15v99**. Note that VRRP interfaces support the same feature set as the parent interface does.
Wireless	wireless *wlanx*	*wlanx* The identifier for the wireless interface you are using. This may be **wlan0** to **wlan999**.
Wireless Modem	wirelessmodem *wlmx*	*wlmx* The identifier for the wireless modem interface you are using. This may be **wlm0** to **wlm999**.

Use the **set** form of this command to enable RIPng on an interface.

Use the **delete** form of this command to remove all RIPng configuration and disable RIPng on the interface.

Use the **show** form of this command to display RIPng configuration.

interfaces <interface> ipv6 ripng split-horizon

Configures split-horizon in RIPng updates coming from this interface.

Syntax

set interfaces *interface* **ipv6 ripng split-horizon** [**disable** | **poison-reverse**]

delete interfaces *interface* **ipv6 ripng split-horizon** [**disable** | **poison-reverse**]

show interfaces *interface* **ipv6 ripng split-horizon**

Command Mode

Configuration mode.

Configuration Statement

```
interfaces interface {
    ipv6 {
        ripng {
            split-horizon {
                disable
                poison-reverse
            }
        }
    }
}
```

Parameters

interface	Mandatory. The type of interface. For detailed keywords and arguments that can be specified as interface types, see the table in the Usage Guidelines below.
disable	Disables split-horizon on the interface.
poison-reverse	Enables poison-reverse on the interface.

Default

Split-horizon is enabled.

Usage Guidelines

Use this command to disable split-horizon or enable split-horizon poison-reverse on an interface running RIPng.

Split-horizon is a stability feature that reduces the possibility of network loops, particularly in the case where links become disconnected. It stops an interface from including in its network updates any routes that it learned from that interface. Split horizon is effective at preventing loops between routers that are directly connected to one another, and speeds convergence when network conditions change and is the default setting in RIPng.

Poison reverse is a variation of split horizon. When an interface with poison reverse enabled detects that a link is down, it increases the metric for that route to 16, and propagates that information in its next update. Since 15 is the largest number of hops considered reachable on a RIPng network, increasing the metric to 16 renders the route unreachable as far as downstream RIPng routers are concerned. This is called "poisoning" the route. Poison reverse can be useful for propagating information about bad routes to routers that are downstream but not immediate neighbors, where split horizon is ineffective.

When this option is enabled, the router includes the route in announcements to the neighbor from which it was learned. When this option is disabled, the router omits the route in announcements to the neighbor from which it was learned.

The following table shows the syntax and parameters for supported interface types.

Interface Type	Syntax	Parameters
ADSL Bridged Ethernet	adsl *adslx* pvc *pvc-id* bridged-ethernet	*adslx* The name of a Bridged Ethernet- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.
ADSL Classical IPOA	adsl *adslx* pvc *pvc-id* classical-ipoa	*adslx* The name of a Classical IPoA- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically.

Interface Type	Syntax	Parameters
ADSL PPPoA	adsl *adslx* pvc *pvc-id* pppoa *num*	*adslx* The name of a Classical IPoA- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically. *num* The PPPoA unit number. This number must be unique across all PPPoA interfaces. In addition, only one PPPoA instance can be configured on a PVC. PPPoA units range from 0 to 15 and the resulting interfaces are named pppoa0 to pppoa15.
ADSL PPPoE	adsl *adslx* pvc *pvc-id* pppoe *num*	*adslx* The name of a Classical IPoA- encapsulated DSL interface. *pvc-id* The identifier for the PVC. It can either be the *vpi*/*vci* pair or the keyword **auto**, where *vpi* is a Virtual Path Index from 0 to 255, *vci* is a Virtual Circuit Index from from 0 to 65535, and **auto** directs the system to detect the Virtual Path Index and Virtual Circuit Index automatically. *num* The name of a defined PPPoE unit. The range is 0 to 15.
Bonding	bonding *bondx*	*bondx* The identifier for the bonding interface. Supported values are **bond0** through **bond99**.
Bonding Vif	bonding *bondx* vif *vlan-id*	*bondx* The identifier for the bonding interface. Supported values are **bond0** through **bond99**. *vlan-id* The VLAN ID for the vif. The range is 0 to 4094.
Bridge	bridge *brx*	*brx* The name of a Bridge group. The range is **br0** through **br999**.
Ethernet	ethernet *ethx*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system.
Ethernet PPPoE	ethernet *ethx* pppoe *num*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system. *num* The name of a defined PPPoE unit. The range is 0 to 15.
Ethernet Vif	ethernet *ethx* vif *vlan-id*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system. *vlan-id* The VLAN ID for the vif. The range is 0 to 4094.

Interface Type	Syntax	Parameters
Ethernet Vif PPPoE	ethernet *ethx* vif *vlan-id* pppoe *num*	*ethx* The name of an Ethernet interface. The range is **eth0** through **eth23**, depending on the physical interfaces available on your system. *vlan-id* The VLAN ID for the vif. The range is 0 to 4094. *num* The name of a defined PPPoE unit. The range is 0 to 15.
Loopback	loopback *lo*	*lo* The name of the loopback interface.
Multilink	multilink *mlx* vif *1*	*mlx* The identifier of the multilink bundle. You can create up to two multilink bundles. Supported values are **ml0** ("em ell zero") through **ml23** ("em ell twenty-three"). *1* The identifier of the virtual interface. Currently, only one vif is supported for multilink interfaces, and the identifier must be 1. The vif must already have been defined.
OpenVPN	openvpn *vtunx*	*vtunx* The identifier for the OpenVPN interface. This may be **vtun0** to **vtun*x***, where ***x*** is a non-negative integer.
Pseudo-Ethernet	pseudo-ethernet *pethx*	*pethx* The name of a pseudo-Ethernet interface. The range is **peth0** through **peth999**.
Serial Cisco HDLC	serial *wanx* cisco-hdlc vif *1*	*wanx* The serial interface you are configuring: one of **wan0** through **wan23**. The interface must already have been defined. *1* The identifier of the virtual interface. Currently, only one vif is supported for Cisco HDLC interfaces, and the identifier must be 1. The vif must already have been defined.
Serial Frame Relay	serial *wanx* frame-relay vif *dlci*	*wanx* The serial interface you are configuring: one of **wan0** through **wan23**. The interface must already have been defined. *dlci* The identifier of the virtual interface. For Frame Relay interfaces, this is the DLCI number for the interface. the range is 16 to 991. The vif must already have been defined.
Serial PPP	serial *wanx* ppp vif *1*	*wanx* The serial interface you are configuring: one of **wan0** through **wan23**. The interface must already have been defined. *1* The identifier of the virtual interface. Currently, only one vif is supported for point-to-point interfaces, and the identifier must be 1. The vif must already have been defined.
Tunnel	tunnel *tunx*	*tunx* An identifier for the tunnel interface you are defining. This may be **tun0** to **tun*x***, where ***x*** is a non-negative integer.

Interface Type	Syntax	Parameters
Virtual Tunnel	vti *vtix*	*vtix* An identifier for the virtual tunnel interface you are defining. This may be **vti0** to **vti*x***, where ***x*** is a non-negative integer. **Note:** This interface does not support IPv6.
VRRP	interface *parent-if* vrrp vrrp-group *group* interface	*parent-if* The type and identifier of the parent interface; for example, **ethernet eth0** or **bonding bond0**. *group* The VRRP group identifier. The name of the VRRP interface is not specified. The system internally constructs the interface name from the parent interface identifier plus the VRRP group number—for example, **eth0v99**, **eth0.15v99**, **bond0v99**, ot **bond0.15v99**. Note that VRRP interfaces support the same feature set as the parent interface does.
Wireless	wireless *wlanx*	*wlanx* The identifier for the wireless interface you are using. This may be **wlan0** to **wlan999**.
Wireless Modem	wirelessmodem *wlmx*	*wlmx* The identifier for the wireless modem interface you are using. This may be **wlm0** to **wlm999**.

Use the **set** form of this command to configure split-horizon and split-horizon poison-reverse on an interface running RIPng.

Use the **delete** form of this command to restore the default configuration.

Use the **show** form of this command to display split-horizon configuration.

Glossary of Acronyms

ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
AMI	Amazon Machine Image
API	Application Programming Interface
AS	autonomous system
ARP	Address Resolution Protocol
AWS	Amazon Web Services
BGP	Border Gateway Protocol
BIOS	Basic Input Output System
BPDU	Bridge Protocol Data Unit
CA	certificate authority
CCMP	AES in counter mode with CBC-MAC
CHAP	Challenge Handshake Authentication Protocol
CLI	command-line interface
DDNS	dynamic DNS
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol version 6

DLCI	data-link connection identifier
DMI	desktop management interface
DMZ	demilitarized zone
DN	distinguished name
DNS	Domain Name System
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
eBGP	external BGP
EBS	Amazon Elastic Block Storage
EC2	Amazon Elastic Compute Cloud
EGP	Exterior Gateway Protocol
ECMP	equal-cost multipath
ESP	Encapsulating Security Payload
FIB	Forwarding Information Base
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HDLC	High-Level Data Link Control
I/O	Input/Ouput
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IGP	Interior Gateway Protocol
IPS	Intrusion Protection System
IKE	Internet Key Exchange
IP	Internet Protocol
IPOA	IP over ATM

IPsec	IP security
IPv4	IP Version 4
IPv6	IP Version 6
ISP	Internet Service Provider
KVM	Kernel-Based Virtual Machine
L2TP	Layer 2 Tunneling Protocol
LACP	Link Aggregation Control Protocol
LAN	local area network
LDAP	Lightweight Directory Access Protocol
LLDP	Link Layer Discovery Protocol
MAC	medium access control
MIB	Management Information Base
MLPPP	multilink PPP
MRRU	maximum received reconstructed unit
MTU	maximum transmission unit
NAT	Network Address Translation
ND	Neighbor Discovery
NIC	network interface card
NTP	Network Time Protocol
OSPF	Open Shortest Path First
OSPFv2	OSPF Version 2
OSPFv3	OSPF Version 3
PAM	Pluggable Authentication Module
PAP	Password Authentication Protocol
PAT	Port Address Translation
PCI	peripheral component interconnect

PKI	Public Key Infrastructure
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PTMU	Path Maximum Transfer Unit
PVC	permanent virtual circuit
QoS	quality of service
RADIUS	Remote Authentication Dial-In User Service
RHEL	Red Hat Enterprise Linux
RIB	Routing Information Base
RIP	Routing Information Protocol
RIPng	RIP next generation
Rx	receive
S3	Amazon Simple Storage Service
SLAAC	Stateless Address Auto-Configuration
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
SONET	Synchronous Optical Network
SSH	Secure Shell
SSID	Service Set Identifier
STP	Spanning Tree Protocol
TACACS+	Terminal Access Controller Access Control System Plus
TBF	Token Bucket Filter
TCP	Transmission Control Protocol
TKIP	Temporal Key Integrity Protocol

ToS	Type of Service
TSS	TCP Maximum Segment Size
Tx	transmit
UDP	User Datagram Protocol
VHD	virtual hard disk
vif	virtual interface
VLAN	virtual LAN
VPC	Amazon virtual private cloud
VPN	Virtual Private Network
VRRP	Virtual Router Redundancy Protocol
WAN	wide area network
WAP	wireless access point
WPA	Wired Protected Access