VYATTA, INC. | Vyatta OFR

Vyatta OFR Command Reference

Vyatta
Suite 160
One Waters Park Drive
San Mateo, CA 94403
vyatta.com

ISSUE DATE: February 2007

DOCUMENT REVISION NO. Rel VC2 v01.

DOCUMENT PART NO. A0-0074-10-01

Table of Contents

Quick Reference to Commands

Use this section to help you quickly locate a command.

Quick List of Examples

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

Preface

This guide explains how to use the Vyatta OFR router, and how to use Vyatta OFR router commands in the command-line interface. It provides an overview of the router's functionality, highlighting core concepts, and a detailed description of each available command.

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

- Intended Audience
- Organization of This Guide
- Document Conventions
- Vyatta Publications

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

Organization of This Guide

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

- **Quick List of Examples**

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

- **Quick Guide to Configuration Statements**

  Use this section to quickly see the complete syntax of configuration statements.

This guide has the following chapters and appendixes:

Document Conventions

This guide contains advisory paragraphs and uses typographic conventions.

Advisory Paragraphs

This guide may use the following advisory paragraphs:

Warnings alert you to situations that may pose a threat to personal safety, as in the following example:

WARNING *Risk of injury. Switch off power at the main breaker before attempting to connect the remote cable to the service power at the utility box.*

Cautions alert you to situations that might cause harm to your system or damage to equipment, or that may affect service, as in the following example:

CAUTION *Risk of loss of service. Restarting a running router will interrupt service.*

Notes provide information you might need to avoid problems or configuration errors:

NOTE *You must create and configure network interfaces before enabling them for routing protocols.*

Tip: *Use tips to save time and effort.*

Tips (see left) provide helpful information for doing something in a faster or easier way, or for optimizing the performance of your system.

Typographic Conventions

In addition to advisory paragraphs, this document may use the following typographic conventions:

`Courier`	Courier font is used in command syntax sections and in special example paragraphs.
`boldface Courier`	Boldface Courier font is used to show something you enter at a command line.
boldface	Boldface font is used to represent commands or keywords inside a paragraph of ordinary text.
italics	Italic font is used to show arguments and variables, where you supply the value.
<key>	Angle brackets are used to indicate a key on your keyboard. Combinations of keys are joined by plus signs ("+"). An example is **<Ctrl>+<Alt>+<Del>**.
[*arg1*|*arg2*]	Square brackets enclose enumerated options for completing a syntax. The options are separated by a vertical bar. An example is **[enable|disable]**.
num1–numN	The typographic convention at left indicates a range of numbers. An example is 1–65535, which means 1 through 65535 inclusive.
arg1..argN	The typographic convention at left indicates a range of enumerated values. An example is **eth0..eth23**, which means **eth1**, **eth2**, **eth3**, and so on through **eth23**.
arg [*arg* ...]	The typographic convention at left indicates a value that can optionally represent a space-separated list of the same kind of element (for example, a space-separated list of IP addresses).

Vyatta Publications

The Vyatta technical library includes the following publications:

Vyatta OFR Quick Start Guide	Explains how to install the router software, and provides some basic configuration to get you started.
Vyatta OFR Configuration Guide	Explains router functions, and steps through sample configurations for every function.
Vyatta OFR Command Reference	Provides a complete description of each command in the CLI.

Chapter 1: Using the CLI

This chapter describes commands for using the CLI.

This chapter contains the following commands.

Command	Mode	Description
`? (help)`	Configuration Operational	Shows available options for completing a command.
`commit`	Configuration	Applies any uncommitted configuration changes.
`configure`	Operational	Switches to configuration mode.
`delete`	Configuration	Deletes a configuration node.
`edit`	Configuration	Navigates to the specified configuration node for editing.
`exit`	Configuration Operational	Exits from this level of use to the level above.
`help`	Configuration Operational	Displays information describing what a command does and how to use it.
`load`	Configuration	Loads configuration information from the specified file, discarding the current configuration.
`quit`	Configuration Operational	Exits from this level of use to the level above.
`run`	Configuration	Runs the specified operational command without leaving configuration mode.
`save`	Configuration	Saves the current configuration to the specified file.
`set`	Configuration	Creates a new configuration node, or changes a value in an existing configuration node.
`show`	Configuration	Displays configuration information (configuration mode) or system information (operational mode).
`top`	Configuration	Exits to the top level of configuration mode.
`up`	Configuration	Navigates up one level in the configuration tree.

See also the following commands in other chapters.

`init-floppy`	Operational	Formats a floppy diskette and prepares it to receive a configuration file. *See page 36.*
`rtrmgr`	Operational	Allows you to change the default location for configuration files. *See page 39.*
`init-floppy`	Operational	Formats a floppy diskette and prepares it to receive a configuration file. *See page 36.*

? (help)

Shows available options for completing a command.

Command Mode

Configuration mode.

Operational mode.

Syntax

```
?                      /* Lists available commands.
command ?              /* Lists options and parameters for the specified command.
```

Parameters

command	A command available in the current location.

Usage Guidelines

Use this command to list the commands currently available to you, or to see what parameters are available for a command.

Typing a question mark ("?") at the command prompt lists the commands currently available to you. The commands available will depend on what configuration you have added to the router.

Typing the question mark after a completed command lists the possible parameters for the command.

Examples

Example 1-1 iteratively applies the question mark to obtain the complete syntax for the **show route system forward** command.

Example 1-1 Iteratively determining command syntax using command-line help

```
root@R1> show route ?
Possible completions:
  <[Enter]>            Execute this command
  <prefix>             Show routing table information
  exact            Show routes exactly matching specified prefix
  next-hop             Show active prefixes with the specified
                      next hop
```

```
  prefix-length        Show active prefixes with the specified
                      prefix length
  protocol             Show routes learned through specified
                      protocol
  system               Show system routing table information
  |                    Pipe through a command
root@R1> show route system ?
Possible completions:
  <[Enter]>            Execute this command
  forward              Show system forwarding table
  |                    Pipe through a command
root@R1> show route system forward ?
Possible completions:
  <[Enter]>            Execute this command
  |                    Pipe through a command
root@R1> show route system forward
```

commit

Applies any uncommitted configuration changes.

Command Mode

Configuration mode.

Configuration Statement

```
commit
```

Parameters

None.

Usage Guidelines

Use this command to apply changes to configuration.

When you add configuration to the router, modify existing configuration, or delete configuration from the router, the changes you make must be committed before they take effect. To do this, you issue the **commit** statement.

If you try to exit or quit from configuration mode while there are still uncommitted configuration changes, the system will give you a warning. You will not be able to exit from configuration mode until you either commit the changes by issuing the **commit** statement, or you discard the changes using the **exit discard** statement (see page 13).

Until a configuration change is committed, the system marks the change when displaying the information.

Committing information can take time, depending on the complexity of the configuration and how busy the router is. Be prepared to wait for several seconds for the system to complete committing the information. The system will inform you when it has finished committing the information by issuing an "OK" response in the command line.

If two or more users are logged on to the router in configuration mode and one user changes the configuration, the other user(s) will receive a warning.

Examples

The following example commits configuration changes.

Example 1-2 Committing Changes

```
[edit interfaces ethernet eth0]
root@vyatta# commit
OK
[edit interfaces ethernet eth0]
root@vyatta# show
   address 172.16.0.65 {
      prefix-length: 24
   }
   address 172.16.0.63 {
      prefix-length: 24
   }

[edit interfaces ethernet eth0]
root@vyatta#
```

configure

Switches to configuration mode.

Command Mode

Operational mode.

Syntax

```
configure
```

Parameters

None.

Usage Guidelines

Use this command to switch to configuration mode, where you can modify aspects of router configuration.

When you are in configuration mode, the prompt pointer “>” changes to the pound sign “#” to indicate that you are in configuration mode (see Example 1-3).

Examples

Example 1-3 shows the system’s response to the configure command. In this example, notice how the command prompt changes when the user enters configuration mode.

Example 1-3 “configure”: Entering configuration mode

```
root@vyatta> configure
Entering configuration mode.
There are no other users in configuration mode.
[edit]
root@vyatta#
```

delete

Deletes a configuration node.

Command Mode

Configuration mode.

Configuration Statement

`delete` *path-to-config-node*

Parameters

path-to-config-node	The path to the part of the configuration to be deleted.

Usage Guidelines

Use this command to delete a part of configuration.

To do this, you delete the appropriate subtree from a configuration node. The deletion will be visible in the response to the **show** command. However, the information is not actually deleted until the change is committed using the **commit** command (see page 6).

Examples

Example 1-4 deletes the **authentication** node from OSPF configuration.

Example 1-4 Deleting configuration

```
root@R1# show protocols ospf4
    router-id: 10.1.0.54
    area 0.0.0.0 {
        interface eth0 {
            address 10.1.0.54 {
                authentication {
                    md5 1 {
                        password: "testmd5"
                    }
                }
            }
        }
    }
```

```
[edit]
root@R1# delete protocols ospf4 area 0.0.0.0 interface eth0
address 10.1.0.54 authentication
Deleting:
    authentication {
        md5 1 {
            password: "testmd5"
        }
    }

OK
[edit]
root@R1# commit
[edit]
OK
root@R1#
```

edit

Navigates to the specified configuration node for editing.

Command Mode

Configuration mode.

Configuration Statement

`edit` *path*

Parameters

path	The path to the node of configuration tree you want to edit.

Usage Guidelines

Use this command to navigate to a specific configuration subtree for editing. Once at that location, you can create a new configuration node, change configuration settings, or delete a configuration node.

- You can only edit a configuration node that has already been created. Configuration nodes are created and modified using the **set** command (see page 23).
- The changes you make do not take effect until they are committed using the **commit** command (see page 6).
- To delete a configuration node, use the **delete** command (see page 9).

After navigating to a branch of the configuration tree, the **show** command will display information for that node only.

Examples

Example 1-5 configures an Ethernet interface by navigating down the configuration tree to the node for the interface, and editing from that location. The resulting commands are much simpler than if they were issued from the top of the configuration tree.

This example begins in operational mode and enters configuration mode.

Example 1-5 Navigating with the “edit” command

```
root@vyatta> configure
Entering configuration mode.
There are no other users in configuration mode.
root@vyatta# edit interfaces ethernet eth0
[edit interfaces ethernet eth0]
root@vyatta# set description “my interface 1”
[edit interfaces ethernet eth0]
root@vyatta# set address 172.16.0.65 prefix-length 24
[edit interfaces ethernet eth0]
root@vyatta# show
>  description: “\“my interface 1\””
>  address 172.16.0.65 {
>     prefix-length: 24
>  }

[edit interfaces ethernet eth0]
root@vyatta# commit
OK
[edit interfaces ethernet eth0]
```

exit

Exits from this level of use to the level above.

This command is operationally equivalent to the **quit** command (see page 18).

Command Mode

Configuration mode.

Configuration Statement

```
exit [discard]
```

Parameters

discard	Applies only at the top level of configuration. Exits from configuration mode, discarding all uncommitted configuration changes.

Usage Guidelines

Use this command in configuration mode to move up one level of use:

- In configuration mode, using this command moves you up one level in the configuration tree.
- At the top level of configuration mode, using this command exits configuration mode, returning you to operational mode.

Use this command in operational mode to exit from the router shell to the UNIX command line.

If you try to exit from configuration mode while there are still uncommitted configuration changes, the system will give you a warning. You will not be able to exit from configuration mode until you either commit the changes by issuing the **commit** statement, or you discard the changes using the **exit discard** option. This is the only case where this option applies.

help

Displays information describing what a command does and how to use it.

Command Mode

Configuration mode.

Operational mode.

Syntax

`help` *command*

Parameters

command	Displays information about using the specified command.

Usage Guidelines

Use this command to display brief information about the usage of a command.

Examples

Example 1-6 gives an example of the usage of the **help** command. This example asks for help for the **show** command in configuration mode.

Example 1-6 The “help” command

```
root@R1# help show

The "show" command will display all or part of the router
configuration.
Without any parameters, the "show" command will display all of
the router
configuration below the current position in the command tree (See
the
"edit" command for how to move the current position).  The show
command
can also take a part of the configuration as parameters; it will
then show
only the selected part of the configuration.
Note that all configuration parameters that have default values
are not
```

```
displayed.

If the configuration has been modified, any changes not yet
committed will
be highlighted.  For example, if "show" displays:

  protocols {
      bgp {
>        peer 10.0.0.1 {
>           as: 65001
>        }
  }

then this indicates that the peer 10.0.0.1 has been created or
changed,
and the change has not yet been applied to the running router
configuration.

 --More--
```

load

Loads configuration information from the specified file, discarding the current configuration.

Command Mode

Configuration mode.

Configuration Statement

`load` *file-name*

Parameters

file-name	The name of the configuration file, including its location.

Usage Guidelines

Use this command to instruct the router to manually load configuration from a file.

Configuration can be loaded from the local hard disk, a TFTP server, an FTP server, or an HTTP server. Note that loading a configuration file causes the previous running configuration to be discarded.

You can save a configuration file to a location other than the configuration directory, as shown in Table 1-2.

Table 1-1 Specifying locations for the configuration file

Location	Specification
An absolute path	
A relative path	Relative paths are interpreted relative to the path configured in the **config-directory** parameter of the **rtrmgr** configuration node.
A TFTP server	Use the following syntax for *file-name*: **tftp://***ip-address/config-file* where *ip-address* is the IP address of the TFTP server, and *config-file* is the configuration file, including the path relative to the TFTP root directory.

Table 1-1 Specifying locations for the configuration file

Location	Specification
An FTP server	Use the following syntax for *file-name*: **ftp://***ip-address/config-file* where *ip-address* is the IP address of the FTP server, and *config-file* is the configuration file, including the path. If you use FTP, you will be prompted for a user name and password.
An HTTP server	use the following syntax for file-name: **http://***ip-address/config-fil*e where *ip-address* is the IP address of the HTTP server, and *config-file* is the configuration file, including the path.

Note that you cannot load an empty configuration file. The configuration file must contain at least one configuration node.

Examples

Example 1-9 loads the configuration file **my-config.boot** from the **/opt/vyatta/etc/config** directory.

Example 1-7 Loading configuration from a file

```
root@R1# load my-config.boot
[edit]
Load done.
root@R1#
```

quit

Exits from this level of use to the level above.

This command is operationally equivalent to the **exit** command (see page 13).

Command Mode

Configuration mode.

Operational mode.

Syntax

```
quit
```

Parameters

None.

Usage Guidelines

Use this command in configuration mode to move up one level of use:

- In configuration mode, using this command moves you up one level in the configuration tree.
- At the top level of configuration mode, using this command exits configuration mode, returning you to operational mode.

Use this command in operational mode to exit from the router shell to the UNIX command line.

If you try to quit from configuration mode while there are still uncommitted configuration changes, the system will give you a warning. You will not be able to exit from configuration mode until you either commit the changes by issuing the **commit** command (see page 6), or you discard the changes using the **exit discard** statement (see page 13).

run

Runs the specified operational command without leaving configuration mode.

Command Mode

Configuration mode.

Syntax

`run` *command*

Parameters

command	An operational command.

Usage Description

Use this command to run an operational command without leaving configuration mode.

Examples

Example 1-8 uses the **show host** command (an operational command) within configuration mode to view the system date and time.

Example 1-8 "run": Using an operational command within configuration mode

```
root@vyatta# run show host date
Wed Nov 30 16:36:58 PST 2005
[edit]
root@vyatta#
```

save

Saves the current configuration to the specified file.

Command Mode

Configuration mode.

Configuration Statement

`save` *file-name*

Parameters

file-name	The name of the file where the information is to be saved, including its location.

Usage Guidelines

Use this command to save the running configuration to a file.

The resulting file can later be loaded into the running router to replace the previous running configuration, using the **load** command (see page 16).

You can save a configuration file to a location other than the configuration directory, as shown in Table 1-2.

Table 1-2 Specifying locations for the configuration file

Location	Specification
An absolute path	
A relative path	Relative paths are interpreted relative to the path configured in the **config-directory** parameter of the **rtrmgr** configuration node.
A TFTP server	Use the following syntax for *file-name*: **tftp://***ip-address/config-file* where *ip-address* is the IP address of the TFTP server, and *config-file* is the configuration file, including the path relative to the TFTP root directory.

Table 1-2 Specifying locations for the configuration file

Location	Specification
An FTP server	Use the following syntax for *file-name*: **ftp://***ip-address/config-file* where *ip-address* is the IP address of the FTP server, and *config-file* is the configuration file, including the path. If you use FTP, you will be prompted for a user name and password.
An HTTP server	use the following syntax for file-name: **http://***ip-address/config-fil*e where *ip-address* is the IP address of the HTTP server, and *config-file* is the configuration file, including the path.

If you overwrite a configuration file, the router retains one backup, using a *file-name*~ convention. For example, if you save over **my-config.boot**, the router moves the previous file to **my-config.boot**~.

Note that the **save** command only writes committed changes. If you makes configuration changes, and try to save, the system warns you that you have uncommitted changes, and then saves only the committed changes.

Examples

Example 1-9 saves the running configuration into the file **my-config.boot** in the **/opt/vyatta/etc/config** directory.

Example 1-9 Saving configuration to a file

```
root@vyatta# save my-config.boot
[edit]
Save done.
root@vyatta# exit
[edit]
root@R1> show files /opt/vyatta/etc/config
total 24K
-rw-rw-r-- 1 vyatta xorp 2.8K Nov 28 10:30 config.boot
-rw-rw-r-- 1 vyatta xorp 2.8K Nov 27 14:32 config.boot~
-rw-rw-r-- 1 vyatta xorp 2.8K Nov 28 10:30 my-config.boot
-rw-rw-r-- 1 vyatta xorp 2.8K Nov 27 21:50 my-config.boot~
root@R1>
```

Example 1-10 saves the current running configuration to the file **my-config.boot** in the root directory of a TFTP server at 10.1.0.35.

Example 1-10 “save”: Saving configuration to a file on a TFTP server

```
root@vyatta# save tftp://10.1.0.35/my-config.boot
OK
[edit]
root@vyatta#
```

set

Creates a new configuration node, or changes a value in an existing configuration node.

Command Mode

Configuration mode.

Syntax

`set` *`parameter value`*

Parameters

parameter	The configuration node or property to be set, including the path to the node or property from the current location in the configuration tree. If the node or property does not exist, it is created. If the node or property already exists, the value is set to the new specified value.

value	The value to which the configuration node or property is to be set.
	Note that not all configuration nodes require values. For example,
	`set fea`
	does not require a value, while
	`set interfaces ethernet eth0`
	does require a value. See individual configuration items for details on the formats and supported values for each parameter.

Usage Guidelines

Use this command to add a configuration element to the current configuration—for example, to add a virtual interface to an interface. You can also use this command to set the value of an existing configuration item. When setting configuration values, note the following:

- The change does not take effect until the change is committed, using the **commit** command (see page 6).
- To navigate to a node for editing, use the the **edit** command (see page 11) or the **up** command (see page 30), the **exit** command (see page 13), or the **quit** command (see page 18) as appropriate.

You must add a configuration node before you can change it or even view it. Before you add any configuration nodes, the system is essentially “empty” except for a few pre-defined configuration nodes. Trying to view system configuration at this stage will show nothing except for very basic system configuration such as a default host name.

Once a configuration node has been added, you can modify it later using the **set** command (see page 23), or delete it using the **delete** command (see page 9).

Examples

Example 1-11 shows an example of the **set** command used to add new configuration for an Ethernet interface. The interface is created, along with a vif for the interface, and an IP address of 192.150.187.108 is applied to the vif. The network for the interface is defined as having a prefix length of 24.

After adding the configuration, the information is displayed and then committed.

Example 1-11 “set”: Adding an Ethernet interface

```
root@vyatta# set interfaces ethernet eth1 vif 0 address
192.150.187.108 prefix-length 24
OK
[edit]

root@vyatta# show interfaces ethernet eth1
>  vif 1 {
>      description: ""
>      address 192.150.187.108 {
>         prefix-length: 24
>         broadcast: 192.150.187.255
>      }
>  }

[edit]

root@vyatta# commit
OK

[edit]
root@vyatta#
```

show

Displays configuration information (configuration mode) or system information (operational mode).

Command Mode

Configuration mode.

Operational mode.

Syntax

```
show config-node        /* Configuration mode
show sub-command        /* Operational mode
```

Parameters

config-node	Available only in configuration mode. The configuration node you want to view, including the path relative to your current location in the configuration tree. The node must exist.
sub-command	Available only in operational mode. A valid operational **show** command; these vary with different router functionalities. See individual router functions for details.

Usage Guidelines

Use this command in configuration mode to display the configured state of the router. Use this command in operational mode to view various aspects of the running router.

In configuration mode, this command displays all existing configuration nodes and sub-nodes starting from your current location in the configuration tree. When used with a configuration path, this command displays the specified configuration node and all its sub-nodes. Default information is not shown.

There are a number of **show** commands in operational mode.

You can only view information for system functions that have been created and configured on the router. Therefore, the **show** commands actually available to you will vary depending on your configuration. Please see individual router functions for specific **show** commands.

If you try to show information for functions that have not been configured, the router gives an error. For example, if you try to use **show rip peers** before creating the **protocols rip** configuration node, the system responds with an error, as shown in Example 1-14.

Examples

Example 1-12 shows the **show** commands available in operational mode.

Example 1-12 Show commands available in operational mode

```
root@R1> show ?
Possible completions:
  arp                   Show Address Resolution Protocol information
  bridge                Show bridging information
  configuration         show current system configuration
  dhcp                  Show Dynamic Host Configuration Protocol
                       information
  files                 Show file information
  firewall              Show firewall information
  hardware              Show system hardware details
  host                  Show host information
  igmp                  Display information about IGMP
  interfaces            Show system interfaces
  log                   Show contents of master log file
  mfea                  Show IPv4 MFEA information
  nat                   Show Network Address Translation information
  ntp                   Show Network Time Protocol information
  package               Show information about system packages
  route                 Show routing table information
  snmp                  Show Simple Network Management Protocol
                       information
  system                Show system information
  tech-support          Consolidated tech-support report
  users                 Show user information
  version               Show software revision information
  vrrp                  Show Virtual Router Redundancy Protocol
                       information
vyatta@mercury> show
```

Example 1-13 shows the **show** command used in configuration mode. In this example, the configuration node displayed is the **service** node.

Example 1-13 Show command in configuration mode

```
root@vyatta# show service
dhcp-server {
   }
    dhcp {
    }
```

```
    http {
    }
    ssh {
    }
    telnet {
    }

[edit]
root@vyatta#
```

Example 1-14 shows the error displayed if you try to show information for functions that have not been configured. In this example, the **show rip peers** command is not recognized, because the **protocols rip** configuration node has not yet been created.

Example 1-14 Error in showing unconfigured functions

```
root@vyatta# show protocols
    ospf4 {
        router-id: 10.1.0.54
        area 0.0.0.0 {
            interface eth0 {
                address 10.1.0.54 {
                    authentication {
                        md5 1 {
                            password: "testmd5"
                        }
                    }
                }
            }
        }
    }

[edit]
root@vyatta# exit
[edit]
root@R1> show rip ?
syntax error, command "show rip" is not recognized.
root@R1> configure
Entering configuration mode.
There are no other users in configuration mode.
root@R1# set protocols rip
[edit]
root@R1# commit
[edit]
OK
root@R1> exit
```

```
[edit]
root@R1> show rip ?
Possible completions:
  peer                 Show RIP statistics for all peers
  statistics           -- No help available --
  status               -- No help available --
root@R1> show rip
```

Sometimes, the configuration information will be too long for your screen, and the screen will show the “More” indication where the information breaks.

- To display the next line of configuration information when the “More” indication is showing, press <**Enter**>.
- To page forward one page, press <**Space**>.
- To page backward, press **b**.
- When all the output has been displayed, the “END” flag appears beside the “More” indicator. Press **q** to exit from the “More” display, as shown in Example 1-15.

Example 1-15 Exiting a “More” screen

```
[edit]
--More-- (END) q
root@vyatta#
```

To turn off paging, pipe your command through the UNIX **no-more** option, as in Example 1-16.

Example 1-16 Exiting a “More” screen

```
root@vyatta> show route | no-more
```

top

Exits to the top level of configuration mode.

Command Mode

Configuration mode.

Configuration Statement

```
top
```

Parameters

None.

Usage Guidelines

Use this command to quickly navigate to the top level of configuration mode.

Examples

Example 1-17 navigates down through several nodes of the configuration tree, then uses the **top** command to jump directly to the top of the tree. In this example, notice how the **[edit]** line displays your location in the configuration tree.

Example 1-17 “top”: Navigating to the top of the configuration tree

```
root@vyatta# edit protocols rip interface eth0
[edit protocols/rip/interface/eth0]
root@vyatta# top
[edit]
root@vyatta#
```

up

Navigates up one level in the configuration tree.

Command Mode

Configuration mode.

Configuration Statement

up

Parameters

None.

Usage Guidelines

Use this command to navigate one level up in configuration mode.

Examples

Example 1-18 navigates down through several nodes of the configuration tree, then uses the **up** command to navigate successively higher in the tree. In this example, notice how the **[edit]** line displays your location in the configuration tree.

Example 1-18 "up": Navigating up through the configuration tree

```
root@vyatta# edit protocols rip interface eth0
[edit protocols/rip/interface/eth0]
root@vyatta# up
[edit protocols/rip/interface]
root@vyatta#
[edit protocols/rip/]
```

Chapter 2: System Management

This chapter describes commands required for basic system management tasks.

This chapter contains the following commands.

Command	Mode	Description
clear arp	Operational	Clears the ARP cache.
date	Operational	Allows you to manually set the system clock or synchronize it one time with an NTP server .
init-floppy	Operational	Formats a floppy diskette and prepares it to receive a configuration file.
mount	Operational	Mounts the floppy disk file system.
reboot	Operational	Reboots the router.
rtrmgr	Configuration	Allows you to change the default location for configuration files.
show arp	Operational	Displays the ARP cache.
show files	Operational	Lists the files in the specified directory.
show host	Operational	Displays host information for hosts reachable by the router.
show interfaces	Operational	Displays information about interfaces.
show ntp associations	Operational	Shows the status of configured NTP servers.
show system boot-messages	Operational	Displays boot messages generated by the kernel.
show system connections	Operational	Displays active network connections on the system.
show system kernel-messages	Operational	Displays messages in the kernel ring buffer.
show system memory	Operational	Displays system memory usage.
show system processes	Operational	Displays active system processes.
show system storage	Operational	Displays system file system usage and available storage space.
show version	Operational	Displays information about the version of router software.
system domain-name	Configuration	Defines the router’s domain.
system domain-search	Configuration	Defines a set of domains for domain completion.
system host-name	Configuration	Sets the host name for the router.

Command	Mode	Description
`system name-server`	Configuration	Specifies the DNS name servers available to the router.
`system ntp-server`	Configuration	Specifies the NTP servers to use when synchronizing the router’s clock.
`system static-host-mapping`	Configuration	Defines a static mapping between a host name and an IP address.
`system time-zone`	Configuration	Sets the time zone for the local system clock.

See also the following commands in other chapters.

`show interfaces ethernet`	Operational	Displays information or statistics about Ethernet interfaces. *See page 102.*
`show interfaces serial`	Operational	Displays information about a specific serial interface. *See page 126.*

clear arp

Clears the ARP cache.

Command Mode

Operational mode.

Syntax

```
clear arp [interface eth0..eth23 |
          address ipv4]
```

Parameters

interface	Clears the entire ARP cache for the specified Ethernet interface. The range of values is **eth0** to **eth23**.
address	Removes the ARP entry for the specified IP address from the ARP cache.

Usage Guidelines

Use this command to clear remove ARP entries associated with an Ethernet interface, or to remove the entry associated with a specific IP address from the ARP cache.

date

Allows you to manually set the system clock or synchronize it one time with an NTP server .

Command Mode

Operational mode.

Syntax

```
date {date-time |
      ntp ipv4}
```

Parameters

date-time	Manually sets the system time and date. The format is "*MMDDhhmm*[.*ss*]*YYYY*", where *MM* is a month from 01 to 12, *DD* is a day from 0 to 31, *hh* is an hour from 00 to 24, *mm* is minutes from 00 to 59, *ss* is seconds from 00 to 59, and *YYYY* is the year. Specifying seconds is optional; the other values are all required. The string must be enclosed in double quotes.
ntp	Instructs the system to synchronize the system time and date with the NTP server one time at the specified IP address. The server must be specified as an IPv4 address.

Usage Guidelines

Use this command to set the system clock.

When used with no option, this command manually sets the system clock to the specified date and time. When used with the **ntp** option, this command manually updates the system clock from the specified NTP server. The system echoes the set date and time on the console for you to verify.

Time zone cannot be set using this command. To set time zone, use the **system time-zone** command (see page 75).

You can configure the router to always automatically obtain the system date and time from one or more NTP servers using the **system ntp-server** command (see page 72).

init-floppy

Formats a floppy diskette and prepares it to receive a configuration file.

Command Mode

Operational mode.

Syntax

```
init-floppy
```

Parameters

None.

Usage Guidelines

Use this command to format a disk in the floppy disk drive.

The system puts a file system on the floppy disk and makes it accessible to the Vyatta system. It also saves a copy of the running configuration to **/mnt/floppy/config/config.boot**.

Initializing the floppy disk erases any previous data on the disk. The system reminds you of this, and provides a 5-second window in which you can quit out of the command by typing <Ctrl>+c.

Once the floppy disk has been formatted, you can save the **config.boot** configuration file to disk using the **save** command (see page 20).

Examples

Example 2-1 prepares a floppy disk for receiving a configuration file and saves the running configuration to **/mnt/floppy/config/config.boot**.

Example 2-1 "init-floppy": Preparing a floppy diskette for a configuration file

```
root@R1> init-floppy
This will erase all data on floppy /dev/fd0.
<CTRL>C to exit: 5
Formatting floppy /dev/fd0...

Floppy disk initialized.
root@R1>
```

mount

Mounts the floppy disk file system.

Command Mode

Operational mode.

Configuration Statement

```
mount floppy
```

Parameters

None.

Usage Guidelines

Use this command to mount the floppy disk file system.

reboot

Reboots the router.

Command Mode

Operational mode.

Syntax

`reboot`

Parameters

None.

Usage Guidelines

Use this command to reboot the router.

Examples

Example 2-2 reboots the router.

Example 2-2 "reboot": Rebooting the router

```
root@R1> reboot
The system is going down NOW !!
Sending SIGTERM to all processes.
Terminated
Sending SIGKILL to all processes.
Please stand by while rebooting the router.
```

rtrmgr

Allows you to change the default location for configuration files.

Command Mode

Configuration mode.

Syntax

`set rtrmgr` *`text`* …	Sets default configuration parameters for the XORP **rtrmgr** process.
`delete rtrmgr …`	The **rtrmgr** configuration node is mandatory and cannot be deleted. If you delete the **rtrmgr** node, configuration is reset to default.

Configuration Statement

```
rtrmgr {
   config-directory: text
}
```

Parameters

config-directory	Sets the default location of the configuration file. This location is where the Vyatta OFR will look to read the **config.boot** configuration file on startup. The default is **/opt/vyatta/etc/config**.

Usage Guidelines

Use this statement to change the directory where the router looks to load the **config.boot** configuration file on startup.

show arp

Displays the ARP cache.

Command Mode

Operational mode.

Syntax

```
show arp
```

Parameters

None.

Usage Guidelines

Use this command to see the entries in the ARP cache.

Table 2-1 shows possible ARP states.

Table 2-1 ARP states

State	Description
incomplete	Address resolution is currently being preformed on this neighbor entry.
reachable	Indicates that the neighbor is reachable. Positive confirmation has been received and the path to this neighbor is operational.
stale	More than the configured elapsed time has passed since reachability confirmation was received from this neighbor.
delay	More than the configured elapsed time has passed since reachability confirmation was received from this neighbor. This state allows TCP to confirm the neighbor. If not, a probe should be sent after the next delay time has elapsed.
probe	A solicitation has been sent and the router is waiting for a response from this neighbor.
failed	Neighbor reachability state detection failed.

Table 2-1 ARP states

State	Description
noarp	This is a pseudo-state, indicating that ARP is not used for this neighbor entry.
permanent	This is a pseudo-state indicating that this entry should not be cleared from the cache.
none	No state is defined.

Examples

Example 2-3 shows the ARP cache of router R1.

Example 2-3 “show arp”: Displaying the ARP cache

```
root@R1> show arp
MAC Address            IP Address      State   Interface
-----------            ----------      -----   ---------
00:12:D9:74:BE:91      172.16.215.1    reach   eth1
00:04:23:09:0F:79      10.1.0.1        reach   eth0


root@R1>
```

show configuration

Displays system configuration.

Command Mode

Operational mode.

Syntax

```
show [-all] configuration
```

Parameters

-all	Displays all configuration, including default values that would not normally be displayed.

Usage Guidelines

Use this command to list configuration information.

Using **show configuration** in operational is equivalent to using **show** in configuration mode. You can display any configuration node by specifying the path for the node. For example, show **configuration firewall** in operational mode is equivalent to **show firewall** in configuration mode.

Examples

Example 2-4 displays the **firewall** configuration node from operational mode.

Example 2-4 “show configuration”: Displaying the configuration tree in operational mode

```
root@R1> show configuration firewall
    log-martians: "enable"
    send-redirects: "disable"
    receive-redirects: "disable"
    ip-src-route: "disable"
    broadcast-ping: "disable"
    syn-cookies: "enable"

root@R1>
```

show files

Lists the files in the specified directory.

Command Mode

Operational mode.

Syntax

show files [*directory*]

Parameters

directory	The name of the directory, including the relative or absolute path to the directory.

Usage Guidelines

Use this command to list files.

When used with no option, this command lists files in the current directory. When a path is provided, this command lists files in the specified directory.

Examples

Example 2-5 lists the files in the **/usr** directory.

Example 2-5 "show files": Listing files in the file system

```
root@R1> show files /usr
total 48K
drwxr-xr-x  2 root root   12K Dec  7 09:33 bin
drwxr-xr-x  2 root root  4.0K Nov  3 11:26 games
drwxr-xr-x  2 root root  4.0K Nov  3 11:23 include
drwxr-xr-x 27 root root   12K Dec  7 09:33 lib
drwxrwsr-x 10 root staff 4.0K Sep 25 14:43 local
drwxr-xr-x  2 root root  4.0K Dec  7 09:42 sbin
drwxr-xr-x 47 root root  4.0K Dec  7 09:33 share
drwxrwsr-x  2 root src   4.0K Aug 28 10:59 src
root@R1>
```

show hardware cpu

Displays information about the router's processor.

Command Mode

Operational mode.

Syntax

```
show hardware cpu
```

Parameters

None.

Usage Guidelines

Use this command to view information about the processor used in the router's hardware platform.

Examples

Example 2-6 shows CPU information on router R1.

Example 2-6 "show hardware cpu": Showing CPU information

```
root@R1> show hardware cpu
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Celeron(R) CPU 2.00GHz
stepping        : 9
cpu MHz         : 1996.821
cache size      : 128 KB
fdiv_bug        : no
hlt_bug         : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 2
wp              : yes
```

```
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep
mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2
ss ht tm pbe cid xtpr
bogomips        : 3999.60
root@R1>
```

show hardware mem

Displays information about the memory used in the router's hardware platform.

Command Mode

Operational mode.

Syntax

```
show hardware mem
```

Parameters

None.

Usage Guidelines

Use this command to display information about the memory used in the router's hardware platform

Examples

Example 2-7 shows information about the memory used in router R1.

Example 2-7 "show hardware mem": Showing hardware memory information

```
root@R1> show hardware mem
MemTotal:       256280 kB
MemFree:        121384 kB
Buffers:         19240 kB
Cached:          64256 kB
SwapCached:          0 kB
Active:          77408 kB
Inactive:        38512 kB
HighTotal:           0 kB
HighFree:            0 kB
LowTotal:       256280 kB
LowFree:        121384 kB
SwapTotal:           0 kB
SwapFree:            0 kB
Dirty:               0 kB
Writeback:           4 kB
Mapped:          50112 kB
Slab:            16312 kB
```

```
CommitLimit:    128140 kB
Committed_AS:    49328 kB
PageTables:        752 kB
VmallocTotal:   770040 kB
VmallocUsed:      1740 kB
VmallocChunk:   768092 kB
root@R1>
```

show host

Displays host information for hosts reachable by the router.

Command Mode

Operational mode.

Syntax

```
show host [hostname | name | date | os}
```

Parameters

host-name	Shows DNS and IP address information about the specified host. This option can be used with either the host name or the IP address of the router. In either case, this command displays the name server canonical name of and IP address of the host, plus any configured aliases.
name	Shows the name configured for this router.
date	Shows the date and time according to the system clock.
os	Shows details about the router's operating system.

Usage Guidelines

Use this command to view information configured for the host.

The information displayed by this command can be configured using the **system host-name** command (see page 70).

Examples

Example 2-8 shows host information for router R2.

Example 2-8 "show host": Finding information about network hosts

```
root@R1> show host R2
Server:         10.0.0.31
Address:        10.0.0.31#53

Name:   R2.vyatta.com
Address: 10.1.0.3

root@R1>
```

Example 2-9 shows the name configured for router R1.

Example 2-9 "show host name": Finding the names of network hosts

```
root@R1> show host name
R1
root@R1>
```

Example 2-10 shows the date and time according to the system clock.

Example 2-10 "show host name": Showing the system date and time

```
root@R1> show host date
Sun Dec 10 01:04:49 PST 2006
root@R1>
```

Example 2-11 shows information about the operating system.

Example 2-11 "show host os": Showing operating system information

```
root@R1> show host date
Linux mercury 2.6.16 #1 Tue Dec 5 15:56:41 PST 2006 i686
   GNU/Linux
root@R1>
```

show interfaces

Displays information about interfaces.

Command Mode

Operational mode.

Syntax

```
show interfaces [system [enabled]]
```

Parameters

system	Displays all system interfaces known to the Linux kernel.
enabled	Shows only enabled interfaces known to the Linux kernel.

Usage Guidelines

Use this command to view configuration information and operational status for interfaces and vifs.

When used with no option, this statement displays information for all interfaces configured on the router. You can see specific information by using other, more detailed, versions of this command:

- To see information for Ethernet interfaces, use the **show interfaces ethernet** version. This command is described in full in “Chapter 3: Ethernet Interfaces, VLANs, and Bridging.”
- To see information for serial interfaces, use the **show interfaces serial** version. This command is described in full in “Chapter 4: Serial Interfaces.”

To see all the physical interfaces known to the operating system kernel, use the **system** option. This option differs from the other options in that the others show interfaces that have been configured on the router (and where the configuration has been committed), while this option shows all the interfaces that are available on your system. You can use this information to determine the interfaces you can configure (for example, how many Ethernet interfaces your system has, or whether it has serial interfaces). It will also show you the syntax for the interface types (Ethernet, serial, and so on).

- When used with no option, the **system** option shows all interfaces available for configuration.

- When used with the **enabled** option, the **system** option shows system interfaces that have been enabled through configuration.

Examples

Example 2-12 shows the first screen of output for **show interfaces system enabled**.

Example 2-12 “show interfaces”: Displaying interface information

```
root@R1> show interfaces system enabled
eth0      Link encap:Ethernet  HWaddr 00:30:48:84:B2:BC
          inet addr:10.1.0.54  Bcast:10.1.0.255
             Mask:255.255.255.0
          inet6 addr: fe80::230:48ff:fe84:b2bc/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:156611 errors:0 dropped:0 overruns:0 frame:0
         TX packets:8773 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
        RX bytes:15619584 (14.8 MiB)  TX bytes:1078150 (1.0 MiB)
          Base address:0xb000 Memory:f2100000-f2120000

eth1      Link encap:Ethernet  HWaddr 00:30:48:84:B2:BD
          inet addr:172.16.215.2  Bcast:172.16.215.255
             Mask:255.255.255.0
          inet6 addr: fe80::230:48ff:fe84:b2bd/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2252 errors:0 dropped:0 overruns:0 frame:0
         TX packets:5051 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
        RX bytes:144448 (141.0 KiB)  TX bytes:872198 (851.7 KiB)
          Base address:0xd100 Memory:f1000000-f1020000

eth2      Link encap:Ethernet  HWaddr 00:12:17:57:29:40
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 --More--
```

show ntp associations

Shows the status of configured NTP servers.

Command Mode

Operational mode.

Syntax

```
show ntp associations [no-resolve]
```

Parameters

no-resolve	Do not attempt to resolve IP addresses into domain names. Use this option to reduce the amount of time it takes for this command to return a result.

Usage Guidelines

Use this command to view the status of connections to configured NTP servers.

A line entry is given for each configured NTP server, showing the server’s IP address and how often the router is polling and updating to the NTP clock. An asterisk (*) next to the NTP server's IP address indicates successful synchronization with the NTP server.

When this command is used without the no-resolve option, the router will attempt to resolve all IP addresses in the configuration to DNS names. This can significantly increase the amount of time required for the command to return a result. To decrease the delay, use the **no-resolve** option.

NTP server connections are configured using the **system ntp-server** command (see page 72).

Examples

Example 2-13 shows the NTP server configured for R1.

Example 2-13 “show ntp associations”: Showing configured NTP servers

```
root@R1> show ntp associations
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 archive.vyatta. .INIT.          16 u  29h 1024    0    0.000    0.000 4000.00
root@R1>
```

show system boot-messages

Displays boot messages generated by the kernel.

Command Mode

Operational mode.

Syntax

```
show system boot-messages
```

Parameters

None.

Usage Guidelines

Use this command to see startup messages that have been generated by the kernel.

Examples

Example 2-14 shows the first screen of output for **show interfaces system enabled**.

Example 2-14 “show system boot-messages”: Displaying startup messages

```
root@R1> show system boot-messages
Linux version 2.6.16 (autobuild@phuket.vyatta.com) (gcc version
4.1.1) #1 Tue Dec 5 15:56:41 PST 2006
BIOS-provided physical RAM map:
 BIOS-e820: 0000000000000000 - 000000000009f800 (usable)
 BIOS-e820: 000000000009f800 - 00000000000a0000 (reserved)
 BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
 BIOS-e820: 0000000000100000 - 000000000fee0000 (usable)
 BIOS-e820: 000000000fee0000 - 000000000fee3000 (ACPI NVS)
 BIOS-e820: 000000000fee3000 - 000000000fef0000 (ACPI data)
 BIOS-e820: 000000000fef0000 - 000000000ff00000 (reserved)
 BIOS-e820: 00000000fec00000 - 0000000100000000 (reserved)
0MB HIGHMEM available.
254MB LOWMEM available.
found SMP MP-table at 000f5a20
On node 0 totalpages: 65248
  DMA zone: 4096 pages, LIFO batch:0
  DMA32 zone: 0 pages, LIFO batch:0
  Normal zone: 61152 pages, LIFO batch:15
```

```
  HighMem zone: 0 pages, LIFO batch:0
DMI 2.3 present.
Intel MultiProcessor Specification v1.4
    Virtual Wire compatibility mode.
OEM ID: OEM00000 Product ID: PROD00000000 APIC at: 0xFEE00000
 --More--
```

show system connections

Displays active network connections on the system.

Command Mode

Operational mode.

Syntax

`show system connections`

Parameters

None.

Usage Guidelines

Use this command to see what network connections are currently active on the network.

Examples

Example 2-15 shows the first screen of output for **show system connections**.

Example 2-15 “show system connections”: Displaying active connections

```
root@R1> show system connections
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address         Foreign Address
State
tcp       0      0 localhost:2912        *:*                LISTEN
tcp       0      0 localhost:3777        *:*                LISTEN
tcp       0      0 localhost:2177        *:*                LISTEN
tcp       0      0 localhost:1700        *:*                LISTEN
tcp       0      0 localhost:1893        *:*                LISTEN
tcp       0      0 localhost:4165        *:*                LISTEN
tcp       0      0 localhost:4744        *:*                LISTEN
tcp       0      0 localhost:34281       *:*                LISTEN
tcp       0      0 localhost:2862        *:*                LISTEN
tcp        0      0 localhost:sa-msg-port   *:*
LISTEN
tcp       0      0 localhost:4015        *:*                LISTEN
tcp       0      0 localhost:1327        *:*                LISTEN
tcp       0      0 *:www                *:*                 LISTEN
tcp       0      0 localhost:3312        *:*                LISTEN
```

```
tcp        0      0 localhost:3153          *:*                     LISTEN
tcp        0      0 localhost:2514          *:*                     LISTEN
tcp        0      0 localhost:2227          *:*                     LISTEN
tcp        0      0 localhost:4883          *:*                     LISTEN
tcp        0      0 localhost:1973          *:*                     LISTEN
tcp        0      0 localhost:4597          *:*                     LISTEN
tcp        0      0 localhost:2103          *:*                     LISTEN
 --More--
```

show system kernel-messages

Displays messages in the kernel ring buffer.

Command Mode

Operational mode.

Syntax

```
show system kernel-messages
```

Parameters

None.

Usage Guidelines

Use this command to see messages currently residing in the kernel ring buffer.

Examples

Example 2-16 shows the first screen of output for **show system kernel-messages**.

Example 2-16 “show system kernel-messages”: Displaying messages from the kernel

```
root@R1> show system kernel-messages
Linux version 2.6.16 (autobuild@phuket.vyatta.com) (gcc version
4.1.1) #1 Tue Dec 5 15:56:41 PST 2006
BIOS-provided physical RAM map:
 BIOS-e820: 0000000000000000 - 000000000009f800 (usable)
 BIOS-e820: 000000000009f800 - 00000000000a0000 (reserved)
 BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
 BIOS-e820: 0000000000100000 - 000000000fee0000 (usable)
 BIOS-e820: 000000000fee0000 - 000000000fee3000 (ACPI NVS)
 BIOS-e820: 000000000fee3000 - 000000000fef0000 (ACPI data)
 BIOS-e820: 000000000fef0000 - 000000000ff00000 (reserved)
 BIOS-e820: 00000000fec00000 - 0000000100000000 (reserved)
0MB HIGHMEM available.
254MB LOWMEM available.
found SMP MP-table at 000f5a20
On node 0 totalpages: 65248
  DMA zone: 4096 pages, LIFO batch:0
  DMA32 zone: 0 pages, LIFO batch:0
  Normal zone: 61152 pages, LIFO batch:15
```

```
  HighMem zone: 0 pages, LIFO batch:0
DMI 2.3 present.
Intel MultiProcessor Specification v1.4
    Virtual Wire compatibility mode.
OEM ID: OEM00000 Product ID: PROD00000000 APIC at: 0xFEE00000

 --More--
```

show system memory

Displays system memory usage.

Command Mode

Operational mode.

Syntax

```
show system memory
```

Parameters

None.

Usage Guidelines

Use this command to see how much memory is currently being used by the system, and how much is free.

Examples

Example 2-14 shows information about memory usage on router R1.

Example 2-17 “show system memory”: Displaying information about memory usage

```
root@R1> show system memory
total        used        free      shared     buffers      cached
Mem:       256280     136732     119548          0      19540      65772
Swap:             0           0           0
Total:       256280      136732      119548
root@R1>
```

show system processes

Displays active system processes.

Command Mode

Operational mode.

Syntax

```
show system processes [summary]
```

Parameters

summary	Provides a summary of process information.

Usage Guidelines

Use this command to see a list of processes currently running on the system.

When used with the **summary** option, this command shows a summary of system process information.

Examples

Example 2-18 shows the first screen of output for **show system processes**.

Example 2-18 “show system processes”: Displaying process information

```
root@R1> show system processes
PID TTY      STAT   TIME COMMAND
    1 ?        S      0:01 init [2]
    2 ?        SN     0:00 [ksoftirqd/0]
    3 ?        S<     0:00 [events/0]
    4 ?        S<     0:00 [khelper]
    5 ?        S<     0:00 [kthread]
    7 ?        S<     0:00 [kblockd/0]
   10 ?        S<     0:00 [khubd]
   68 ?        S      0:00 [pdflush]
   69 ?        S      0:00 [pdflush]
   71 ?        S<     0:00 [aio/0]
   70 ?        S      0:00 [kswapd0]
  656 ?        S<     0:00 [kseriod]
```

```
1481 ?        S<     0:00 [ata/0]
1484 ?        S<     0:00 [scsi_eh_0]
1486 ?        S<     0:00 [scsi_eh_1]
1723 ?        S      0:05 [kjournald]
1877 ?        S<s    0:00 udevd --daemon
2548 ?        S<     0:00 [kpsmoused]
3141 ?        Rs     0:00 /sbin/syslogd
3147 ?        Ss     0:00 /sbin/klogd -x
3190 ?        Ss     0:00 /usr/sbin/cron
--More--
```

show system storage

Displays system file system usage and available storage space.

Command Mode

Operational mode.

Syntax

`show system boot-messages`

Parameters

None.

Usage Guidelines

Use this command to see how much storage space is currently being used by the system, and how much is free.

Examples

Example 2-19 shows file system usage information for router R1.

Example 2-19 "show system storage": Displaying file system and storage information

```
root@R1> show system storage
Filesystem            Size  Used Avail Use% Mounted on
rootfs                953M  287M  618M  32% /
udev                   10M   28K   10M   1% /dev
/dev/hda1             953M  287M  618M  32% /
/dev/hda1             953M  287M  618M  32% /dev/.static/dev
tmpfs                 126M  4.0K  126M   1% /dev/shm
/dev/hda2           9.7M  1.5M  7.8M  17% /opt/vyatta/etc/config
root@R1>
```

show tech-support

Provides a consolidated report of system information.

Command Mode

Operational mode.

Syntax

```
show system tech-report
```

Parameters

None.

Usage Guidelines

Use this command to list a technical report providing consolidated information about system components and configuration.

This information is valuable for debugging and diagnosing system issues. You should provide the technical report whenever you open a case with Vyatta technical support.

Examples

Example 2-20 shows the first screen of a technical report.

Example 2-20 "show tech-support" Displaying consolidated system information

```
root@R1> show tech-support
----------------
OFR Version
----------------
Version:    1.1-1
Built by:   autobuild@vyatta.com
Built on:   200612060031 -- Wed Dec  6 00:31:13 UTC 2006
System booted: Fri Dec  8 15:36:39 PST 2006
Uptime: 19:42:44 up 1 day,  4:06,  1 user,  load average: 0.00,
0.11, 0.20
----------------
OFR Packages
----------------
Desired=Unknown/Install/Remove/Purge/Hold
```

```
|
Status=Not/Installed/Config-files/Unpacked/Failed-config/Half-i
nstalled
|/ Err?=(none)/Hold/Reinst-required/X=both-problems
(Status,Err: uppercase=bad)
||/ Name                Version                        Description
+++-=====================-==============================-====
========================================
ii  adduser                3.99                          Add
and remove users and groups
ii  apt                 0.6.46.2                     Advanced
front-end for dpkg
ii  apt-utils              0.6.46.2                      APT
utility programs
 --More--
```

show version

Displays information about the version of router software.

Command Mode

Operational mode.

Syntax

```
show version
```

Parameters

None.

Usage Guidelines

Use this command to display information about the version of router software the router is running.

Example 2-21 show sample output for the **show version** command.

Example 2-21 "show version": Displaying router software information

```
root@vyatta> show version
Version:    1.1-1
Built by:   autobuild@vyatta.com
Built on:   200612060031 -- Wed Dec  6 00:31:13 UTC 2006
System booted: Fri Dec  8 15:36:39 PST 2006
Uptime: 19:46:42 up 1 day,  4:10,  1 user,  load average: 0.00,
0.04, 0.15
root@vyatta>
```

system domain-name

Defines the router's domain.

Command Mode

Configuration mode.

Syntax

`set system domain-name …`	Creates or modifies the configuration node for the router's domain.
`delete system domain-name …`	Deletes domain configuration.

Configuration Statement

```
system {
   domain-name: text
}
```

Parameters

domain-name	Mandatory. The domain where the router resides. The format is a string containing letters, numbers, hyphens ("-") and a period.

Usage Guidelines

This statement is optional. Use this statement configure the router's domain—for example, **mydomain.com**.

system domain-search

Defines a set of domains for domain completion.

Command Mode

Configuration mode.

Syntax

`set system domain-search domain` *`text`* …	Adds a domain to the list of domains. Note that you cannot use set to change a domain. To change a domain, **delete** the incorrect domain and **set** a new one to replace it.
`delete system domain-search domain` *`text`* …	Deletes the specified domain from the list.

Configuration Statement

```
system {
   domain-search {
      domain: text [text ...]
   }
}
```

Parameters

domain	Mandatory. Multi-node. A domain name to be added to or deleted from the list of domains in the search order string. The format is a string specifying a domain, for example **mydomain.com**. Letters, numbers, hyphens (“-”) and a period (“.”) are allowed. You can enter up to six domains by issuing this command up to six times, to a maximum of 256 characters. Alternatively, up to six domains can be specified in a space-separated list, to a maximum of 256 characters.

Usage Guidelines

Use this statement to set the order for domain completions of DNS lookup requests.

When the router receives an unqualified host name, the domain names specified here appended to the host name to form a Fully Qualified Domain Name. The router tries each domain name in turn, in the order in which they were configured. If none of the resulting FQDNs succeeds, the name will not be resolved and an error will be reported.

You can specify up to six domains by issuing the **set** command multiple times. Alternatively, you can specify up to domain names in a space-separated list, to a maximum of 256 characters.

Note that you cannot use **set** to change a domain name in the list. To change an incorrect domain, delete it and replace it with a new one.

system host-name

Sets the host name for the router.

Command Mode

Configuration mode.

Syntax

`set system host-name` *`text`* …	Creates the configuration node for the router host name, or changes the router's host name. As the router is automatically provided with a default host name, this node will normally exist already.
`delete system host-name` *`text`* …	Resets the router's host name to the default.

Configuration Statement

```
system {
   host-name: text
}
```

Parameters

text	The name you want to give the router. Letters, numbers, and hyphens ("**-**") only are allowed. The default is "vyatta". If you delete the host name, or if you try to delete the **system** node, the host name reverts to the default.

Usage Guidelines

Use this statement to configure a host name for the router.

By default, the host name is preconfigured to "vyatta". If you delete the host name, or if you delete the **system** node, the default values are restored.

When you set this value, the command prompt changes to reflect the new host name. To see the change in the prompt, you must log out of the router shell and log back in again.

system name-server

Specifies the DNS name servers available to the router.

Command Mode

Configuration mode.

Syntax

`set system name-server ipv4 …`	Defines a new DNS name server. You can define multiple DNS servers by issuing the **set** command multiple times. You cannot use **set** to change the identifier of an existing name server. To change the IP address of a DNS server, **delete** the server configuration and **set** ia new one with the correct address.
`delete system name-server ipv4 …`	Removes a defined DNS name server.

Configuration Statement

```
system {
   name-server: ipv4 {}
}
```

Parameters

ipv4	Multi-node. The IPv4 address of a DNS name server to use for local name query requests. You can specify multiple DNS name servers by creating multiple instances of the **name-server** configuration node.

Usage Guidelines

Use this statement to specify DNS name servers for the router.

To add a DNS name server, use the **set** version of this statement. To remove a DNS name server, use the **delete** version of this statement. More than one name server can be specified by issuing the **set system name-server** statement multiple times.

To change the IP address for a DNS server, delete it and recreate it using the correct address.

system ntp-server

Specifies the NTP servers to use when synchronizing the router's clock.

Command Mode

Configuration mode.

Syntax

`set system ntp-server` *`ipv4`* …	Adds a server to the list of NTP servers. You can specify multiple NTP servers by issuing the **set** command multiple times. You cannot use set to change the address of an existing NTP server. To change the IP address of an NTP server, **delete** the server and **set** a new one to replace it.
`delete system ntp-server` *`ipv4`* …	Deletes the specified NTP server from the list of servers.

Configuration Statement

```
system {
   ntp-server: [ipv4|text] {}
}
```

Parameters

ntp-server	Multi-node. The IP address or host name of an NTP server. The router will automatically obtain the system date and time from the specified server(s). The default is **ntp.vyatta.com**. You can specify multiple NTP servers by creating multiple instances of the **name-server** configuration node.

Usage Guidelines

Use this statement to specify NTP servers for the router.

To add an NTP server, use the **set** version of this command. To remove an NTP server, use the **delete** version of this statement. More than one NTP server can be specified by issuing the **set system ntp-server** statement multiple times.

To change the IP address for an NTP server, delete it and recreate it using the correct address.

system static-host-mapping

Defines a static mapping between a host name and an IP address.

Command Mode

Configuration mode.

Syntax

`set system static-host-mapping host-name …`	Use **set** to create a new static mapping between a host name and an IP address, or to modify static mapping values. Note that you cannot use **set** to change the host name, as it is the identifier of the configuration node. To change the host name, **delete** the mapping entry and **set** a new one with the correct host name.
`delete system static-host-mapping host-name …`	Use **delete** to remove the **alias** portion of a mapping, or to remove the entire mapping entry. You cannot delete the **inet** value by itself, as it is mandatory.

Configuration Statement

```
system {
   static-host-mapping {
      host-name: text {
         inet: ipv4
         alias: text {}
      }
   }
}
```

Parameters

host-name	Multi-node. The fully qualified host name being statically mapped to an IP address (for example, **router1@mydomain.com**). Letters, numbers, periods (“.”) and hyphens (“-”) only are allowed. To define multiple mappings, set multiple **host-name** configuration nodes within the **static-host-mapping** node.
inet	Mandatory. The IPv4 address of the interface being statically mapped to the host name.

alias	Optional. Multi-node. An alias for the interface. Letters, numbers, and hyphens are allowed. You can define multiple aliases for a host by creating multiple **alias** configuration nodes.

Usage Guidelines

Use this statement to statically map a host name to an IP address and one or more aliases.

system time-zone

Sets the time zone for the local system clock.

Command Mode

Configuration mode.

Syntax

`set system time-zone` *`text`* …	Use **set** to set the time zone for the first time, or to change the time zone setting.
`delete system time-zone` *`text`* …	Use **delete** to remove the time zone setting. This restores the time zone to the default (GMT).

Configuration Statement

```
system {
   time-zone: text
}
```

Parameters

time-zone	A string representing the time-zone and offset from UTC, enclosed in double quotes. The format is "**GMT** [{+ | **-**}*h*]", where *h* is a number from 1 to 12 representing the hours offset from GMR. The string must be enclosed in double quotes. **Calculating offset from GMT:** Please see the "Usage Guidelines" section for this information. The following time zone names, enclosed in double quotes, are also accepted: **"Los Angeles"**: Sets the time zone to Los Angeles time. **"New York"**: Sets the time zone to New York time. **"Denver"**: Sets the time zone to Denver time. **"Chicago"**: Sets the time zone to Chicago time. **"Anchorage"**: Sets the time zone to Anchorage time. **"Honolulu"**: Sets the time zone to Honolulu time. **"Phoenix"**: Sets the time zone to Phoenix time. The default is "**GMT**", which uses UTC time exactly.

Usage Guidelines

Use this statement to set the time zone for the local system clock.

To do this, you specify the amount by which your time zone is offset from UTC (coordinated universal time). The offset you specify is added to UTC to produce the local time.

Note that the router uses POSIX-style offsets. The POSIX specification uses positive signs west of Greenwich—not positive signs east of Greenwich, which many other systems use. For example, an offset of "**GMT +4**" corresponds to 4 hours behind UTC (that is, west of Greenwich).

Chapter 3: Ethernet Interfaces, VLANs, and Bridging

This chapter lists the commands for configuring Ethernet interfaces, virtual interfaces (with optional VLAN tagging), the loopback interface, IP addresses, and bridging.

This chapter contains the following commands.

Command	Mode	Description
`interfaces`	Configuration	Sets configuration for interfaces.
`interfaces bridge`	Configuration	Defines a bridge group and its spanning tree parameters.
`interfaces ethernet`	Configuration	Defines an Ethernet interface and sets its characteristics.
`interfaces ethernet address`	Configuration	Defines an IP address on an Ethernet interface for non-802.1q packets.
`interfaces ethernet bridge-group`	Configuration	Assigns an interface to a bridge group.
`interfaces ethernet vif`	Configuration	Defines a virtual interface (vif) on an Ethernet interface for receiving 802.1q VLAN-tagged packets.
`interfaces ethernet vif address`	Configuration	Defines an IP address on a vif.
`interfaces ethernet vif bridge-group`	Configuration	Assigns a vif to a bridge group.
`interfaces loopback`	Configuration	Defines a loopback interface.
`interfaces loopback address`	Configuration	Defines an IP address on the loopback interface.
`show bridge`	Operational	Shows information for active bridge groups.
`show interfaces ethernet`	Operational	Displays information or statistics about Ethernet interfaces.

See also the following commands in other chapters.

`clear arp`	Operational	Clears the ARP cache. *See page 34.*
`show arp`	Operational	Displays the ARP cache. *See page 40.*
`show interfaces`	Operational	Displays information about interfaces. *See page 50.*

interfaces

Sets configuration for interfaces.

Command Mode

Configuration mode.

Syntax

`set interfaces …`	Creates the configuration node for a network interface and specifies whether to restore original configuration when the system is shut down.
`delete interfaces …`	Deletes any user configuration for the **interfaces** configuration node, restoring factory defaults.

Configuration Statement

```
interfaces {
   restore: [true|false]
}
```

Parameters

`restore`	Indicates whether to restore configuration to factory defaults when the router is shut down. Supported values are as follows: **true**: Restore original configuration when the router is shut down. **false**: Do not restore original configuration when the router is shut down. The default is **false**.

Usage Guidelines

Use this statement to specify configuration behavior on shutdown.

interfaces bridge

Defines a bridge group and its spanning tree parameters.

Command Mode

Configuration mode.

Syntax

`set interfaces bridge …`	Use **set** to create the **bridge** configuration node, which defines a bridge group to which interfaces and vifs may belong. You can also use **set** to overwrite bridge group properties.
`delete interfaces bridge …`	Use **delete** to delete a **bridge** configuration node, which removes the specified bridge group.

Configuration Statement

```
interfaces {
   bridge br0..br9 {
      description: text
      disable: [true|false]
      aging: 1-4294967296
      stp: [true|false]
      priority: 1-4294967296
      forwarding-delay: 1-4294967296
      hello-time: 1-4294967296
      max-age: 1-4294967296
   }
}
```

Parameters

bridge	Mandatory. The identifier for the bridge group. Supported identifiers are **br0** through **br09**.
description	Optional. A brief description for the bridge group.

disable	Optional. Enables or disables bridging on this interface. Supported values are as follows: **true**—Disables bridging on this interface, without discarding the configuration. **false**—Enables bridging on this interface. The default is **false**.
aging	Optional. Sets the length of time in seconds a MAC address will be kept in this bridge's forwarding database before the entry is aged out of the table. The range is 1 to 4294967295. The default is 300.
stp	Optional. Allows you to enable or disable the Spanning Tree Protocol on a per-bridge basis. Supported values are as follows: **true**: Enables Spanning Tree Protocol on this bridge. **false**: Disables Spanning Tree Protocol on this bridge. The default is **false**.
priority	Optional. Sets the forwarding priority of this bridge in the spanning tree. The default is 0.
forwarding-delay	Optional. The amount of time in seconds this bridge will keep listening and learning about the topology of the spanning tree after a topology change. After the forward delay interval has passed, the bridge transitions to the Forwarding state. The range is 1 to 4294967295. The default is 0.
hello-time	Optional. The interval in seconds at which this bridge will transmit "hello packets," which are messages that communicate the state of the spanning tree topology. On a spanning tree, hello packets are sent by the bridge that assumes itself to be the root bridge. The range is 1 to 4294967295. The default is 0.
max-age	Optional. The interval a bridge will wait to receive a hello packets before removing a neighboring bridge. The range is 1 to 4294967295. The default is 0.

Usage Guidelines

Use this statement to define a bridge and configure its bridging and Spanning Tree Protocol characteristics.

Note that you must create the bridge group (using this command) before you can assign interfaces to it.

interfaces ethernet

Defines an Ethernet interface and sets its characteristics.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet …`	Creates the configuration node for an Ethernet interface, or modifies configuration for the interface.
`delete interfaces ethernet …`	Deletes all configuration for the specified Ethernet interface.

Configuration Statement

```
interfaces {
   ethernet: eth0..eth23 {
      disable:[true|false]
      discard:[true|false]
      description:text
      mac: mac-addr
      mtu: 68–65535
      duplex: [full|half|auto]
      speed: [10|100|1000|auto]
}
```

Parameters

ethernet	Multi-node. An identifier for the Ethernet interface you are defining. This may be **eth0** to **eth23**, depending on what Ethernet interfaces that actually available on the system. You can create as many **ethernet** configuration nodes as there are Ethernet interfaces available on your system. To see the interfaces available to the system kernel, use the **system** option of the **show interfaces** command (see page 50).

disable	Optional. Enables or disables forwarding on this interface. Supported values are as follows: **true**—Disables forwarding on this interface, without discarding the configuration. **false**—Enables forwarding on this interface. The default is **false**.
discard	Optional. Specifies this interface as a discard interface. A discard interface is an interface that discards packets. You can configure local policies such that if a device comes under attack the attacking policies are forwarded out the discard interface. If desired, you can attach an output filter to the discard interface to log or count the packets as they egress. Otherwise, traffic is silently discarded. You can configure one discard interface per router. Supported values are as follows: **true**: This interface is the discard interface. **false**: This interface is not the discard interface. The default is **false**.
description	Optional. A mnemonic name or description for the interface. The default is an empty string.
mac	Optional. Sets the MAC address for the interface. MAC addresses on devices such as Ethernet devices are usually fixed, but in some cases it is possible to override the built-in default MAC address. The format should be appropriate for the interface type. For an Ethernet interface, this is six colon-separated 8-bit numbers in hexadecimal, for example: `00:0a:59:9a:f2:ba`
mtu	Optional. Sets the maximum transfer unit (MTU), in octets, for the interface as a whole. This will apply to all vifs defined for the interface. When forwarding, IPv4 packets larger than the MTU will be fragmented unless the DF bit is set. In that case, the packets will be dropped and an ICMP “Packet too big” message is returned to the sender. The range is 8 to 8818. If not set, fragmentation will never be performed.

duplex	Optional. Sets the duplexity of the interface. Supported values are as follows: **full**: This interface is to be full duplex. **half**: This interface is to be half duplex. **auto**: The router will autonegotiate the duplexity of the interface. The default is **auto**.
speed	Optional. Sets the speed of the interface. Supported values are as follows: **10**: 10 Mbps **100**: 100 Mbps **1000**: 1000 Mbps **auto**: The router will autonegotiate the speed of the interface. The default is **auto**.

Usage Guidelines

Use this statement to set the characteristics of Ethernet interfaces.

When the router starts up, it automatically discovers the physical interfaces available on the system and creates a loopback interface. Apart from the interfaces automatically created by the system, each level of interface, IP address, and vifs to be used must be explicitly created through configuration.

interfaces ethernet address

Defines an IP address on an Ethernet interface for non-802.1q packets.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *name* `address …`	Creates the configuration node for an IP address on an Ethernet interface, or modifies IP address configuration. Note that you cannot use **set** to change the address itself, as it is the identifier of a configuration node. To change an address delete the address and recreate it with the correct information.
`delete interfaces ethernet` *name* `address …`	Deletes all configuration for the specified address.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      address: [ipv4 | ipv6]{
         prefix-length: [0–32|0-128]
         broadcast: ipv4
         multicast-capable: [true|false]
         disable: [true|false]
      }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.
address	Multi-node. Defines an IPv4 or IPv6 address on this interface. You can define multiple IP addresses for a single interface, by creating multiple **address** configuration nodes.

prefix-length	Mandatory. Specifies the prefix length of the subnet connected to this interface. • For IPv4 addresses, the range is 0 to 32. • For IPv6 addresses, the range is 0 to 128.
broadcast	Gives the subnet broadcast address for the subnet corresponding to this address. Configuring this value is optional, as the system automatically calculates the broadcast address. You can use this option to override the automatically calculated broadcast address. • The broadcast address for IPv4 addresses must be an IPv4 address. • The broadcast address for IPv6 addresses must be an IPv6 address.
disable	Enables or disables this IP address for routing and forwarding. Supported values are as follows: **true**—Disables this IP address, without discarding the configuration. **false**—Enables this IP address. The default is **false**.

Usage Guidelines

Use this statement to define an IP address on an interface.

If you are not using 802.1q and you want to have multiple networks on the same physical interface, use this statement to define multiple IP addresses for the interface.

interfaces ethernet bridge-group

Assigns an interface to a bridge group.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *name* `bridge-group …`	Creates the **bridge-group** configuration node for an interface, or Modifies existing bridge group settings for an interface.
`delete interfaces ethernet` *name* `bridge-group …`	Deletes bridge group configuration for an interface.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23]
      bridge-group {
         bridge: br0..br9
         cost: 1-4294967296
         priority: 1-4294967296
      }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.
bridge-group	Assigns this network interface to the specified bridge group: the identifier will be **br0** through **br9**. The bridge group must already exist. To define a bridge group, use the **the interfaces bridge** command (see page 80). Note that membership in a Layer 2 bridge group precludes configuring IP settings (a Layer 3 protocol) for an interface.
cost	Optional. Specifies the path cost of this interface. An integer from 0 to 65535, where a higher number indicates a higher cost. The default is 0.

priority	Optional. Sets the order in which ports of equal cost are used. The default is 0.

Usage Guidelines

Use this statement to assign an interface to a bridge and set its cost and priority within the group.

Note that you must already have created the bridge group using the **interfaces bridge** command (see page 80).

interfaces ethernet vif

Defines a virtual interface (vif) on an Ethernet interface for receiving 802.1q VLAN-tagged packets.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *`name`* `vif` *`vlan-id`* …	Creates the configuration node for a vif, or modifies vif configuration. A vif on an Ethernet interface is always a VLAN interface, and the identifier of the vif of an Ethernet interface is its VLAN ID. Note that you cannot use **set** to change the VLAN ID for a vif, as it is the identifier of a configuration node. To change this information delete the vif and recreate it with the correct VLAN ID.
`delete interfaces ethernet` *`name`* `vif` *`vlan-id`* …	Deletes all configuration for the specified vif.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      vif 1-4096 {
         disable:[true|false]
      }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.

vif	Multi-node. The VLAN ID for the vif, for use with 802.1q VLAN tagging. Only tagged packets are received on vifs configured on Ethernet interfaces. The range is 1 to 4096. You can define more than one vif for a single interface by creating multiple **vif** configuration nodes.
disable	Optional. Enables or disables this vif. Supported values are as follows: **true**—Disables this vif, without discarding the configuration. **false**—Enables this vif. The default is **false**.

Usage Guidelines

Use this statement to define a virtual interface (vif) on an interface, or to enable or disable a vif.

In the Vyatta OFR router, most configuration can be applied either directly to the physical interface, or to a *virtual interface* (vif), which is a logical interface created for the physical interface. When the router starts up, it automatically detects the physical interfaces available on your device and creates configuration nodes for them. For example, on a system with two Ethernet interfaces, the router automatically creates configuration nodes for **eth0** and **eth1**.

Ethernet vifs are used only when 802.1Q VLANs are to be supported. In a basic Ethernet configuration, such as that for trial or evaluation or for a simple network topology, it will often be simplest and adequate to apply IP addresses directly to the physical interface.

Each physical interface can have multiple IP addresses assigned to it.If you want to have multiple networks on the same physical interface (that is, if you want to use multinetting, but not VLANs), simply create multiple **address** configuration nodes directly under the primary interface.

Note that, in statements other than **interface** statements, the notation for referring to a vif is *int.vif*—for example, **eth1.40**. When referring to a vif within an interface statement (**set interface**, **delete interface**, and **show interface** in configuration mode) the notation is **interface** *int* **vif** *vif*—for example, **set interface eth1 vif 40**.

interfaces ethernet vif address

Defines an IP address on a vif.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *`int.vif`* `address …`	Creates the configuration node for an IP address on an Ethernet vif, or modifies address configuration. Note that you cannot use **set** to change the address itself, as it is the identifier of a configuration node. To change an address, delete it and recreate it with the correct information.
`delete interfaces ethernet` *`int.vif`* `address …`	Deletes all configuration for the specified address.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      vif: 1–4096 {
         address: [ipv4 | ipv6]{
            prefix-length: [0–32|0-128]
            broadcast: ipv4
            multicast-capable: [true|false]
            disable: [true|false]
         }
      }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.
vif	The identifier (VLAN ID) of the Ethernet vif you are configuring. The vif must already have been defined.
address	Multi-node. Defines an IPv4 or IPv6 address on this vif. You can define multiple IP addresses for a single vif, by creating multiple **address** configuration nodes beneath the vif.
prefix-length	Mandatory. Specifies the prefix length of the subnet connected to this vif. • For IPv4 addresses, the range is 0 to 32. • For IPv6 addresses, the range is 0 to 128.
broadcast	Gives the subnet broadcast address for the subnet corresponding to this address. Configuring this value is optional, as the system automatically calculates the broadcast address. You can use this option to override the automatically calculated broadcast address. • The broadcast address for IPv4 addresses must be an IPv4 address. • The broadcast address for IPv6 addresses must be an IPv6 address.
disable	Enables or disables this IP address for routing and forwarding. Supported values are as follows: **true**—Disables this IP address, without discarding the configuration. **false**—Enables this IP address. The default is **false**.

Usage Guidelines

Use this statement to define an IP address on a vif.

interfaces ethernet vif bridge-group

Assigns a vif to a bridge group.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *`int.vif`* `bridge-group …`	Creates the **bridge-group** configuration node for a vif, or modifies existing bridge group settings for a vif.
`delete interfaces ethernet` *`int.vif`* `bridge-group …`	Deletes bridge group configuration for a vif.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth0]
      vif 1-4096
         bridge-group {
            bridge: br0..br9
            cost: 1-4294967296
            priority: 1-4294967296
         }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.
vif	The identifier (VLAN ID) of the Ethernet vif you are configuring. The vif must already have been defined.
bridge-group	Assigns this vif to the specified bridge group: the identifier will be **br0** through **br9**. The bridge group must already exist. To define a bridge group, use the **interfaces bridge** command (see page 80). Note that membership in a Layer 2 bridge group precludes configuring IP settings (a Layer 3 protocol) for vif.

cost	Optional. Specifies the path cost of this vif. An integer from 0 to 65535, where a higher number indicates a higher cost. The default is 0.
priority	Optional. Sets the order in which ports of equal cost are used. The default is 0.

Usage Guidelines

Use this statement to assign a vif to a bridge and set its cost and priority within the group.

Note that you must already have created the bridge group using the **interfaces bridge** command (see page 80).

interfaces loopback

Defines a loopback interface.

Command Mode

Configuration mode.

Syntax

`set interfaces loopback lo …`	Creates the configuration node for the loopback interface, or modifies loopback interface information.
`delete interfaces loopback lo …`	Deletes the loopback interface.

Configuration Statement

```
interfaces {
   loopback: lo {
      description: text
   }
}
```

Parameters

loopback	The identifier of the loopback interface: this is always **lo**.
description	A brief description for the interface.

Usage Guidelines

Use this statement to define the loopback interface.

The loopback interface is a special software-only interface that emulates a physical interface and allows the router to “connect” to itself. Packets routed to the loopback interface are rerouted back to the router and processed locally. Packets routed out the loopback interface but not destined for the loopback interface are dropped.

The loopback interface provides a number of advantages:

- As long as the router is functioning, the loopback interface is always up, and so is very reliable. As long as there is even one functioning link to the router, the loopback interface can be accessed. The loopback interface thus eliminates the need to try each IP address of the router until you find one that is still up.
- Because the loopback interface is always up, a routing session (such as a BGP session) can continue even if the outbound interface fails.
- You can simplify collection of management information by specifying the loopback interface as the interface for sending and receiving management information such as logs and SNMP traps.
- The loopback interface can be used as to increase security, by filtering incoming traffic using access control rules that specify the local interface as the only acceptable destination.
- In OSPF, you can advertise a loopback interface as an interface route into the network, regardless of whether physical links are up or down. This increases reliability, since the the routing traffic is more likely to be received and subsequently forwarded.
- In BGP, parallel paths can be configured to the loopback interface on a peer device. This provides improved load sharing.

interfaces loopback address

Defines an IP address on the loopback interface.

Command Mode

Configuration mode.

Syntax

`set interfaces loopback lo address …`	Creates an IP address for the loopback interface, or modifies loopback interface address information. Note that you cannot use **set** to change the address itself, as it is the identifier of a configuration node. To change an address, delete it and recreate it with the correct information.
`delete interfaces loopback lo address …`	Deletes this address on the loopback interface.

Configuration Statement

```
interfaces {
   loopback: lo {
      address: [ipv4|ipv6]{
         prefix-length: [0–32|0-128]
         broadcast: ipv4
         multicast-capable: [true|false]
         disable: [true|false]
      }
   }
}
```

Parameters

loopback	The identifier of the loopback interface: this is always **lo**.
address	Multi-node. Defines an IPv4 or IPv6 address on the loopback interface. You can define multiple IP addresses for the loopback interface, by creating multiple **address** configuration nodes.
prefix-length	Mandatory. Specifies the prefix length of the subnet connected to this vif. • For IPv4 addresses, the range is 0 to 32. • For IPv6 addresses, the range is 0 to 128.
broadcast	Gives the subnet broadcast address for the subnet corresponding to this address. Configuring this value is optional, as the system automatically calculates the broadcast address. You can use this option to override the automatically calculated broadcast address. • The broadcast address for IPv4 addresses must be an IPv4 address. • The broadcast address for IPv6 addresses must be an IPv6 address.
disable	Enables or disables this IP address for routing and forwarding. Supported values are as follows: **true**—Disables this IP address, without discarding the configuration. **false**—Enables this IP address. The default is **false**.

Usage Guidelines

Use this statement to define an IP address for the loopback interface.

The router automatically creates the loopback interface on startup, with an interface name of **lo**. You must configure an IP address for the interface. The IP address for the loopback interface must be unique, and must not be used by any other interface.

When configuring the router, it is good practice to take advantage of the loopback interface's reliability:

- The router's hostname should be mapped to the loopback interface address, rather than a physical interface.
- In OSPF and BGP, the router ID should be set to the loopback address.
- The network for the loopback interface can be small, since IP address space is not a consideration in this cse. Often a prefix of /32 is assigned.

NOTE *In some systems, the IP address 127.0.0.0 is assigned to the loopback interface by convention. However, in the Vyatta OFR the network 127.0.0.0/8 is reserved for XORP to communicate between processes. As a result, no IP address on this reserved network may be configured on any interface. Any other network may be assigned to the loopback interface.*

show bridge

Shows information for active bridge groups.

Command Mode

Operational mode.

Syntax

```
show bridge [bridge-group [macs | spanning-tree]]
```

Parameters

bridge-group	Displays information for the specified bridge group: one of **eth0** through **eth23**.
macs	Shows the MAC table for the specified bridge.
spanning-tree	Shows spanning tree information for the specified bridge.

Usage Guidelines

Use this command to display information about configured bridge groups.

When used with no option, this command displays information about all active bridge groups. When the identifier of a bridge group is provided, this command displays information for the specified bridge group. You can display the MAC table and Spanning Tree Protocol information for a bridge group.

show interfaces ethernet

Displays information or statistics about Ethernet interfaces.

Command Mode

Operational mode.

Syntax

```
show interfaces ethernet [eth0..eth23 [physical| vif vlan-id]]
```

Parameters

ethernet	Displays information for only Ethernet interfaces.
interface	Displays information for the specified Ethernet interface.
physical	Displays physical layer settings for the specified Ethernet interface.

Usage Guidelines

Use this command to view command and operational status of interfaces and vifs.

- When used with no argument, the **ethernet** option shows information for all Ethernet interfaces.
- When an interface name is supplied, the **ethernet** option shows information about the specified Ethernet interface only.
- When the **physical** argument is used, the **ethernet** option shows physical layer settings for the specified Ethernet interface.

Examples

Example 3-1 shows the first screen of output for **show interfaces ethernet**.

Example 3-1 “show interfaces ethernet”: Displaying Ethernet interface information

```
root@vyatta> show interfaces ethernet
eth0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc
pfifo_fast qlen 1000
    link/ether 00:18:fe:fa:16:18 brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    0          0        0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    0          0        0       0       0       0
eth1: <BROADCAST,MULTICAST,UP,10000> mtu 1500 qdisc pfifo_fast
qlen 1000
    link/ether 00:18:fe:fa:16:19 brd ff:ff:ff:ff:ff:ff
    inet 10.1.0.40/24 brd 10.1.0.255 scope global eth1
    inet6 fe80::218:feff:fefa:1619/64 scope link
       valid_lft forever preferred_lft forever
    RX: bytes  packets  errors  dropped overrun mcast
    641361     8860     0       0       0       21
    TX: bytes  packets  errors  dropped carrier collsns
    406342     3355     0       0       0       0
```

Example 3-2 shows the first screen of output for **show interfaces ethernet** *ethx* **physical**.

Example 3-2 “show interfaces ethernet *ethX* physical”: Displaying physical line characteristics for Ethernet interfaces

```
root@vyatta> show interfaces ethernet eth0 physical
Settings for eth0:
        Supported ports: [ MII ]
        Supported link modes:   10baseT/Half 10baseT/Full
                                100baseT/Half 100baseT/Full
                                1000baseT/Half 1000baseT/Full
        Supports auto-negotiation: Yes
        Advertised link modes:  10baseT/Half 10baseT/Full
                                100baseT/Half 100baseT/Full
                                1000baseT/Half 1000baseT/Full
        Advertised auto-negotiation: Yes
        Speed: Unknown! (65535)
        Duplex: Unknown! (255)
        Port: Twisted Pair
```

```
        PHYAD: 1
        Transceiver: internal
        Auto-negotiation: on
        Supports Wake-on: g
        Wake-on: d
        Current message level: 0x000000ff (255)
        Link detected: no
root@vyatta>
```

Chapter 4: Serial Interfaces

This chapter lists the commands for configuring Layer 2 encapsulation protocols on serial interfaces.

This chapter contains the following commands.

Command	Mode	Description
`clear interfaces serial`	Operational	Clears counters for serial interfaces
`interfaces serial`	Configuration	Specifies basic serial interface configuration, including Layer 2 encapsulation characteristics.
`interfaces serial cisco-hdlc`	Configuration	Defines the characteristics of Cisco High-Level Data Link Control encapsulation on a serial interface.
`interfaces serial e1-options`	Configuration	Specifies the physical line characteristics for E1 serial interfaces.
`interfaces serial frame-relay`	Configuration	Defines the characteristics of Frame Relay encapsulation on an interface.
`interfaces serial ppp`	Configuration	Defines the characteristics of Point-to-Point Protocol encapsulation on an interface.
`interfaces serial t1-options`	Configuration	Specifies the physical line characteristics for T1 serial interfaces.
`interfaces serial t3-options`	Configuration	Specifies the physical line characteristics for T3 serial interfaces.

See also the following commands in other chapters.

`show interfaces`	Operational	Displays information about interfaces. *See page 50.*
`interfaces serial cisco-hdlc vif firewall`	Operational	Applies named firewall instances (packet-filtering rule sets) to a Cisco HDLC–encapsulated serial interface. *See page 329.*
`interfaces serial frame-relay vif firewall`	Operational	Applies named firewall instances (packet-filtering rule sets) to a Frame Relay–encapsulated serial interface. *See page 332.*
`interfaces serial ppp vif firewall`	Operational	Applies named firewall instances (packet-filtering rule sets) to a Point-to-Point Protocol–encapsulated serial interface. *See page 335.*

clear interfaces serial

Clears counters for serial interfaces

Command Mode

Operational mode.

Syntax

```
clear interfaces serial interface counters {all | physical |
   cisco-hdlc | frame-relay | ppp}
```

Parameters

interface	The identifier of a configured serial interface.
all	Clears all counters for the specified serial interface.
physical	Clears counters related to the physical line settings for the specified interface.
cisco-hdlc	Clears counters related to Cisco HDLC settings for the specified interface.
frame-relay	Clears counters related to Frame Relay settings for the specified interface.
ppp	Clears counters related to Point-to-Point Protocol settings for the specified interface.

Usage Guidelines

Use this command to clear statistics for a specified serial interface.

At least one of the filters must be specified.

interfaces serial

Specifies basic serial interface configuration, including Layer 2 encapsulation characteristics.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *`name`* …	Use **set** to create the configuration node for a serial interface, or to to modify serial interface configuration. You can define multiple serial interfaces by creating multiple **serial** configuration nodes. Note that you cannot use **set** to change the name of the serial interface. To change the name of a serial interface, you must **delete** the old **serial** configuration node and create a new one.
`delete interfaces serial` *`name`* …	Use **delete** to delete configuration for a serial interface.

Configuration Statement

```
interfaces {
   serial [wan0..wan23] {
      encapsulation: [ppp|cisco-hdlc|frame-relay]
      description: text
   }
}
```

Parameters

encapsulation	Mandatory. The encapsulation type of the interface. Supported values are as follows: **ppp**: Uses Point-to-Point Protocol (PPP) encapsulation on the interface. **cisco-hdlc**: Uses Cisco High-Level Data Link Control (Cisco HDLC) encapsulation on the interface. **frame-relay**: Uses Frame Relay encapsulation on the interface.

description	Optional. A brief description for the serial interface. By default, the system auto-detects the card type and indicates it in the description.

Usage Guidelines

Use this command to specify the encapsulation type and physical line characteristics of traffic that will pass through this serial interface.

interfaces serial cisco-hdlc

Defines the characteristics of Cisco High-Level Data Link Control encapsulation on a serial interface.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *name* `cisco-hdlc …`	Use **set** to create the **cisco-hdlc** configuration node, or to modify Cisco HDLC encapsulation. Note that you cannot use **set** to change the identifier of configuration nodes. To change the identifier of a configuration node, you must **delete** the old configuration node and create a new one with the correct identifier.
`delete interfaces serial` *name* `cisco-hdlc …`	Use **delete** to delete configuration for Cisco HDLC encapsulation on this interface.

Configuration Statement

```
interfaces {
    serial [wan0..wan23] {
        cisco-hdlc {
            keepalives {
                require-rx: [enable|disable]
                timer: 10-60000
            }
            vif 1 {
                address {
                    local-address: ipv4
                    prefix-length: 0–32
                    remote-address: ipv4
                }
                description: text
            }
        }
    }
}
```

Parameters

keepalives	Sets the value for the keep-alive timeout. If the **rxinterval** timer expires without receiving a keep-alive message from the peer interface, the interface increments the **down-count** counter. If the **down-count** timer reaches the configured limit, the peer interface is declared down. All interfaces using the HDLC keep-alive mechanism must be configured with corresponding timers; that is, the **rxinterval** of the one peer must match the **txinterval** of the other.
require-rx	Require keep-alive messages for a link to be considered up. Supported values are as follows: **enable**: Require keep-alive messages. If keep-alive messages are not received, the peer interface is declared down. **disable**: Do not require keep-alive messages. The default is **disable**.
timer	The interval for keep-alive messages, in seconds. The range is 10 to 60000. The default is 10.
vif	The identifier of the virtual interface. Currently, only one vif is supported for Cisco HDLC interfaces, and the identifier must be **1**.
address	IP address information. Each serial vif can support exactly one IP address.
local-address	Mandatory. The IPv4 address for this vif.
prefix-length	Mandatory. The prefix defining the network served by this interface. The range is 0 to 32.
remote-address	Mandatory. An IPv4 address representing the network address.
description	Optional. A brief description for the interface. If the description contains spaces, it must be enclosed in double quotes.

Usage Guidelines

Use this statement to define the Cisco High-Level Data Link Control characteristics of the line.

Note that on Cisco HDLC interfaces, IP addresses are assigned to virtual interfaces, not directly to the interface. Currently, only one vif is supported, but multiple addresses may be defined for the vif.

The full identifier of an HDLC interface is *int* **cisco-hdlc vif** *vif*. For example, the full identifier of the HDLC vif on wan1 is **wan1 cisco-hdlc vif 1**. Note that subsequent to initial definition, the notation for referring to this is *int.vif*—that is, **wan1.1**.

interfaces serial e1-options

Specifies the physical line characteristics for E1 serial interfaces.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *`name`* `e1-options…`	Use **set** to configure physical line characteristics for an E1 serial interface, or to modify E1 serial interface configuration.
`delete interfaces serial` *`name`* `e1-options…`	Use **delete** to delete configuration for an E1 serial interface.

Configuration Statement

```
interfaces {
   serial [wan0..wan23] {
      e1-options {
         framing: [g704|g704-no-crc4|unframed]
         timeslots {
            start: [1–32]
            stop: [1–32]
         }
         mtu: 8–8188
         clock: [internal|external]
      }
   }
}
```

Parameters

framing	Optional. Sets the frame type for the interface. Supported values are as follows: **g704**: Sets the E1 frame type to use CRC4. **g704-no-crc**: Sets the E1 frame type not to use CRC4. **unframed**: Configures full-rate (2048 kbps) unchannelized E1 bandwidth for the line. The default is **g704**.

timeslots	Optional. Allows you to configure a fraction of a 32-port channelized E1 line. To do this, you assign a range of timeslots to the line.
start	The first timeslot in the range. The range of values is 1 to 32, where the value of **start** must be less than the value of **stop**. The default is 1.
stop	The last timeslot in the range. The range of values is 1 to 32, where the value of **start** must be less than the value of **stop**. The default is 32.
mtu	Optional. Sets the maximum transfer unit (MTU), in octets, for the interface as a whole. This will apply to all vifs defined for the interface. When forwarding, IPv4 packets larger than the MTU will be fragmented unless the DF bit is set. In that case, the packets will be dropped and an ICMP “Packet too big” message is returned to the sender. The range is 8 to 8188. If not set, fragmentation will never be performed. The default is 1500.
clock	Optional. Sets the timing source for the circuit. Supported values are as follows: **internal**: The interface will use the internal clock. **external**: The interface will use the external DTE TX and RX clock. The default is **external**.

Usage Guidelines

Use this command to specify the physical line characteristics of traffic that will pass through this E1 serial interface.

Configuring this option designates this interface as an E1 interface for transmitting signals in European digital transmission (E1) format. The E1 signal format carries information at a rate of 2.048 Mbps and can carry 32 channels of 64 Kbps each.

Currently, only high-density bipolar of order 3 (hdb3) line encoding is supported.

interfaces serial frame-relay

Defines the characteristics of Frame Relay encapsulation on an interface.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *name* `frame-relay …`	Use **set** to create the **frame-relay** configuration node, or to modify configuration for Frame Relay encapsulation. Note that you cannot use **set** to change the identifier of configuration nodes. To change the identifier of a configuration node, you must **delete** the old configuration node and create a new one with the correct identifier.
`delete interfaces serial` *name* `frame-relay …`	Use **delete** to delete configuration for Frame Relay encapsulation on this interface.

Configuration Statement

```
interfaces {
    serial [wan0..wan23] {
        frame-relay {
            signaling: [auto|ansi|q933|lmi]
            signaling-options {
                n391dte: 1–255
                n392dte: 1–100
                n393dte: 1–10
                t391dte: 5–30
            }
            vif [16..991] {
                address {
                    local-address: ipv4
                    prefix-length: 0–32
                    remote-address: ipv4
                }
                description: text
            }
        }
    }
}
```

Parameters

signaling	Specifies the Frame Relay signaling variant (LMI type). Supported values are as follows: **auto**: Autonegotiates the LMI type. **ansi**: Uses ANSI-617d Annex D LMI type. **q933**: Uses the Q.933 (ITU-T (CCIT) Q.933 annex A) LMI type. **lmi**: Uses Cisco proprietary LMI type. The default is auto.
signaling-options	Sets the Frame Relay signaling options.
n391dte	Sets the DTE full status message polling interval, which is the interval, in seconds, at which this interface expects a full status report from the DCE interface. All other status enquiries can be responded to with a keep-alive exchange only. The range is 1 to 255. The default is 6.
n392dte	Sets the DTE error threshold, which is the number of errors which, if they occur within the event count specified by the **n393dte** attribute, will cause the link to be declared down. The range is 1 to 100. The default is 6.
n393dte	Sets the DTE monitored event count. The range is 1 to 10. The default is 4.
t391dte	Sets the DTE keep-alive timer. This is the interval, in seconds, at which the interface sends out a keep-alive request to the DCE interface, which should respond with a keep-alive message. At the interval defined by the n391dte option, the DCE will send a full status report instead of just a keep-alive message. The range is 5 to 30. The default is 10.
vif	The identifier of the virtual interface. For Frame Relay interfaces, this is the DLCI number for the interface. The range is 16 to 991.
address	IP address information. Each serial vif can support exactly one IP address.
local-address	Mandatory. The IPv4 address for this vif.
prefix-length	Mandatory. The prefix defining the network served by this interface. The range is 0 to 32.

remote-address	Mandatory. An IPv4 address representing the network address.
description	Optional. A brief description for the interface. If the description contains spaces, it must be enclosed in double quotes.

Usage Guidelines

Use this statement to define Frame Relay settings on an interface. This consists primarily of defining the signaling variant, the PVC characteristics, and the keep-alive (health checking) characteristics of the line.

The full identifier of an Frame Relay interface is *int* **frame-relay vif** *vif*. For example, the full identifier of the Frame Relay vif 16 on wan0 is **wan0 frame-relay vif 16**. Note that subsequent to initial definition, the notation for referring to this is *int.vif*—that is, **wan0.16**.

interfaces serial ppp

Defines the characteristics of Point-to-Point Protocol encapsulation on an interface.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *`name`* `ppp …`	Use **set** to create the **ppp** configuration node, or to modify configuration for Point-to-Point Protocol encapsulation. Note that you cannot use **set** to change the identifier of configuration nodes. To change the identifier of a configuration node, you must **delete** the old configuration node and create a new one with the correct identifier.
`delete interfaces serial` *`name`* `ppp …`	Use **delete** to delete configuration for Point-to-Point Protocol encapsulation on this interface.

Configuration Statement

```
interfaces {
   serial [wan0..wan23] {
      ppp {
         authentication {
            type: [none|chap|pap]
            user-id: text
            password: text
            system-name: text
         }
         vif 1 {
            address {
               local-address: ipv4
               prefix-length: 0–32
               remote-address: ipv4
            }
            description: text
         }
      }
   }
}
```

Parameters

authentication	Sets the authentication parameters for the interface.
type	Sets the authentication type. Supported values are as follows: **none**: Authentication is not required on this interface. **chap**: Uses the Challenge Handshake Authentication Protocol (CHAP), as defined in RFC 1994. **pap**: Uses the Password Authentication Protocol (PAP). The client authenticates itself by sending a user ID and a password to the server, which the server compares to the password in its internal database.
user-id	Used with PAP. The user ID of the client.
password	Used with PAP. The password of the client.
system-name	Used with CHAP. The authenticator (that is, the server) sends a randomly generated "challenge" message to the client, along with this system name. The client uses the system name to look up the appropriate secret, combines it with the challenge, and encrypts the string using a one-way hashing function. The result is returned to the server along with the client's hostname. The server now performs the same computation, and acknowledges the client if it arrives at the same result.
vif	The identifier of the virtual interface. Currently, only one vif is supported for point-to-point interfaces, and the identifier must be **1**.
address	IP address information. Each serial vif can support exactly one IP address.
local-address	Mandatory. The IPv4 address for this vif.
prefix-length	Mandatory. The prefix defining the network served by this interface. The range is 0 to 32.
remote-address	Mandatory. An IPv4 address representing the network address.
description	Optional. A brief description for the interface. If the description contains spaces, it must be enclosed in double quotes.

Usage Guidelines

Use this statement to define Point-to-Point Protocol settings on an interface.

The full identifier of a Point-to-Point Protocol interface is *int* **ppp vif** *vif*. For example, the full identifier of the point-to-point vif on wan1 is **wan1 ppp vif 1**. Note that subsequent to initial definition, the notation for referring to this is *int.vif*—that is, **wan1.1**.

interfaces serial t1-options

Specifies the physical line characteristics for T1 serial interfaces.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *`name`* `t1-options…`	Use **set** to configure physical line characteristics for a T1 serial interface, or to modify T1 serial interface configuration.
`delete interfaces serial` *`name`* `t1-options…`	Use **delete** to delete configuration for a T1 serial interface.

Configuration Statement

```
interfaces {
   serial [wan0..wan23] {
      t1-options {
         lbo: [0-110ft|110-220fr|220-330ft|330-440ft|440-550ft]
         timeslots {
            start: [1–24]
            stop: [1–24]
         }
         mtu: 8–8188
         clock: [internal|external]
      }
   }
}
```

Parameters

lbo	Optional. Sets the maximum line build-out length. Supported values are as follows: **0–110ft**: The line will not exceed 110 feet in length. **110–220ft**: The line will be between 110 and 220 feet in length. **220–330ft**: The line will be between 220 and 330 feet in length. **330–440ft**: The line will be between 330 and 440 feet in length. **440–550ft**: The line will be between 440 and 550 feet in length. The default is **0-110ft**.
timeslots	Optional. Allows you to configure a fraction of a 24-port channelized T1 line. To do this, you assign a range of timeslots to the line.
start	The first timeslot in the range. The range of values is 1 to 24, where the value of **start** must be less than the value of **stop**. The default is 1.
stop	The last timeslot in the range. The range of values is 1 to 24, where the value of **start** must be less than the value of **stop**. The default is 24.
mtu	Optional. Sets the maximum transfer unit (MTU), in octets, for the interface as a whole. This will apply to all vifs defined for the interface. When forwarding, IPv4 packets larger than the MTU will be fragmented unless the DF bit is set. In that case, the packets will be dropped and an ICMP “Packet too big” message is returned to the sender. The range is 8 to 8188. If not set, fragmentation will never be performed. The default is 1500.
clock	Optional. Sets the timing source for the circuit. Supported values are as follows: **internal**: The interface will use the internal clock. **external**: The interface will use the external DTE TX and RX clock. The default is **external**.

Usage Guidelines

Use this command to specify the physical line characteristics of traffic that will pass through this T1 serial interface.

Configuring this option designates this interface as a T1 interface for transmitting digital signals in the T-carrier system used in the United States, Japan, and Canada. The T1 signal format carries 24 pulse code modulation (PCM) signals using time-division multiplexing (TDM) at an overall rate of 1.544 Mbps.

Currently, only bipolar 8-zero line coding is supported.

interfaces serial t3-options

Specifies the physical line characteristics for T3 serial interfaces.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *name* `t3-options…`	Use **set** to configure physical line characteristics for a T3 serial interface, or to modify T3 serial interface configuration.
`delete interfaces serial` *name* `t3-options…`	Use **delete** to delete configuration for a T3 serial interface.

Configuration Statement

```
interfaces {
    serial [wan0..wan23] {
        t3-options {
            framing: [c-bit|m13]
            line-coding: [ami|b8zs]
        }
    }
}
```

Parameters

framing	Optional. Sets the frame type for the interface. Supported values are as follows: **c-bit**: Sets the T3 frame type to C-bit parity **m13**: Sets the T3 frame type to M13. The default is **c-bit**.
line-coding	Optional. Sets the T3 line coding. Supported values are as follows: **ami**: Sets the line coding to alternate mark inversion (AMI). **b8zs**: Sets the line coding to bipolar 8-zero substitution. The default is **b8zs**.

Usage Guidelines

Use this command to specify the physical line characteristics of traffic that will pass through this T3 serial interface.

Configuring this option designates this interface as a T3 interface for transmitting digital signals in the T-carrier system used in the United States, Japan, and Canada. The T3 signal format carries multiple T1 channels multiplexed, resulting in transmission rates of up to 44.736 Mbit/s.

show interfaces serial

Displays information about a specific serial interface.

Command Mode

Operational mode.

Syntax

```
show interfaces serial [wan0..wan23
   {cisco-hdlc |
    frame-relay [pvc-list [active|inactive]| pvc [dcli]] |
    physical |
    ppp}]
```

Parameters

cisco-hdlc	Shows Cisco HDLC information for the specified serial interface.
frame-relay	Shows Frame Relay information for the specified serial interface.
pvc-list	Lists of Frame Relay permanent virtual circuits (PVCs). When used with no option, displays all configured PVCs.
active	Lists only active Frame Relay PVCs.
inactive	Lists only inactive Frame Relay PVCs.
pvc	Displays details for Frame Relay PVCs. When used with no option, displays information for all configured PVCs.
dlci	Displays details for just the specified Frame Relay DLCI.
ppp	Shows Point-to-Point protocol information for the specified serial interface.

Usage Guidelines

Use this command to view the operational status of a serial interface.

When used with no option, this command displays information for all available serial interfaces. If an interface is specified, you must also specify one of the **cisco-hdlc**, **frame-relay**, or **ppp** options.

Examples

Example 4-1 shows the first screen of output for **show interfaces serial**.

Example 4-1 “show interfaces serial”: Displaying serial interface information

```
vyatta@neptune> show interfaces serial
wan0: <NO-CARRIER,POINTOPOINT,NOARP,UP> mtu 1500 qdisc
pfifo_fast qlen 100
    link/ppp

    RX: bytes  packets  errors  dropped overrun mcast
    0          0        0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    0          0        0       0       0       0
```

Chapter 5: Basic Services

This chapter describes commands required to deploy basic protocol services such as DHCP, HTTP, SSH, and Telnet.

This chapter contains the following commands.

Command	Mode	Description
`clear dhcp leases`	Operational	Removes current DHCP leases.
`service dhcp relay`	Configuration	Configures the router to relay DHCP client messages to an off-net DHCP server.
`service dhcp-server`	Configuration	Configures the DHCP service on the router.
`service http`	Configuration	Configures HTTP as an access protocol on the router.
`service ssh`	Configuration	Configures SSH as an access protocol on the router.
`service telnet`	Configuration	Configures Telnet as an access protocol on the router.
`show dhcp leases`	Operational	Displays current DHCP lease information.
`show dhcp statistics`	Operational	Displays DHCP server statistics.

clear dhcp leases

Removes current DHCP leases.

Command Mode

Operational mode.

Syntax

```
clear dhcp leases [ipv4]
```

Parameters

ipv4	Clears the DHCP lease for the specified IP address.

Usage Guidelines

Use this command to remove DHCP leases.

When used with no option, this command clears all current leases. When an IP address is specified, this command clears the for the host at the specified address.

DHCP is configured using the the **service dhcp-server** command (see page 134).

service dhcp relay

Configures the router to relay DHCP client messages to an off-net DHCP server.

Command Mode

Configuration mode.

Syntax

`set service dhcp relay interface` *`text`* …	Use **set** to create a new DHCP relay agent, or to modify DHCP relay configuration. Note that you cannot use **set** to change the interface for an existing relay agent, or to change the identifiers of subordinate configuration nodes. To change this information, you must **delete** the entry and then **set** it again using the correct information.
`delete service dhcp relay interface` *`text`* …	Use **delete** to delete optional values for a DHCP relay agent. You cannot delete mandatory values within a configuration node.

Configuration Statement

```
service {
   dhcp {
      relay {
         interface: [all|eth0..eth23]{
            server: ipv4 {}
            relay-options {
               port: 1–65535
               max-size: 64–1400
               hop-count: 1-255
               relay-agents-packets: [discard|forward]
            }
         }
      }
   }
}
```

Parameters

interface	Mandatory. Multi-node. The interface to use to relaying DHCP client messages. You can relay DHCP client messages through more than one interface by creating multiple **interface** configuration nodes.
server	Mandatory. Multi-node. The IP address of the DHCP server. You can relay messages to more than one DHCP server, by creating multiple **server** configuration nodes.
relay-options	Optional. If relay options are configured, the router adds Relay Agent Information option (option 82) to the client-to-server packet, as specified by RFC 3046.
port	Optional. The port on this interface to be used for relaying DHCP client messages. The range is 1 to 65535. The default is 67.
max-size	Optional. The maximum size of the DHCP packet to be created after appending the relay agent information option. If, after appending the information, the packet would exceed this size, the packet is forwarded without appending the information. If this option is not configured, the router does not forward DHCP packets that exceed the MTU of the interface on which relaying is configured. The range is 64 to 1400. The default is 576.
hop-count	Optional. The time-to-live for outgoing relayed messages. The range is 1 to 255. The default is 10.
relay-agents-packets	Optional. Sets the reforwarding policy for a DHCP relay agent. This is the action the router will take if the DHCP message already contains relay information. Supported values are as follows: **discard**: If the packet already contains relay information, it will be discarded. **forward**: The packet will be forwarded regardless of whether it contains relay information. The default is **forward**.

Usage Guidelines

Use this command to configure the router as a DHCP relay agent.

A DHCP relay agent receives DHCP packets from DHCP clients and forwards them to a DHCP server. This allows you to place DHCP Clients and DHCP servers on different networks; that is, across router interfaces.

The relay agent is configured with addresses of DHCP servers to which they should relay client DHCP message. The relay agent intercepts the broadcast, sets the gateway address (the **giaddr** field of the DHCP packet) and, if configured, inserts the Relay Agent Information option (option 82) in the packet and forwards it to the DHCP server.

The DHCP server echoes the option back verbatim to the relay agent in server-to-client replies, and the relay agent strips the option before forwarding the reply to the client.

service dhcp-server

Configures the DHCP service on the router.

Command Mode

Configuration mode.

Syntax

`set service dhcp-server name` *`text`* …	Use **set** to create a new DHCP address pool, or to modify address pool configuration. Note that you cannot use **set** to change the name of an existing address pool, or change the identifiers of other configuration nodes. To change this information, you must **delete** the entry and then **set** it again using the correct information.
`delete service dhcp-server name` *`text`* …	Use **delete** to delete optional values for an address pool. You can also use **delete** to remove an entire address pool. If you delete the last address pool, DHCP will not be available as a service. Within an address pool, you cannot delete mandatory values, such as **interface** or **netmask**.

Configuration Statement

```
service {
   dhcp-server {
      name text {
         interface: eth0..eth23
         network-mask: 0–32
         start ipv4 {
            stop: ipv4
         }
         exclude: ipv4 {}
         static-mapping: text {
            ip-address: ipv4
            mac-address: macaddr
         }
         dns-server ipv4 {}
         default-router: ipv4
         wins-server ipv4 {}
```

```
                lease: 120-4294967296
                domain-name: text
                authoritative: [enable|disable]
            }
        }
    }
```

Parameters

name	Mandatory. Multi-node. Creates a DHCP server address pool with the specified name. You can define multiple address pools by creating multiple **name** configuration nodes, each with a different name.
interface	Mandatory. The router interface bound to this DHCP address pool. The interface must already be configured on the router.
network-mask	Mandatory. Defines the size of the subnet served by this pool of addresses. The range is 0 to 32.
start	Optional. Multi-node. The start address in an address range. This is the first address in the range that can be assigned. You can define multiple address ranges within an address pool, by creating multiple **start** configuration nodes.
stop	Mandatory. The stop address in this address range. This is the last address in the range that can be assigned.
exclude	Optional. Multi-node. Allows you to exclude an IP address from the address pool. The router will not assign these IP addresses to any devices. You can exclude multiple addresses within an address pool, by creating multiple **exclude** configuration nodes.
static-mapping	Optional. Multi-node. Allows you to statically map an IP address within an address pool to the MAC address of a device on the network. You can define multiple static mappings of this type by creating multiple **static-mapping** configuration nodes.
ip-address	Mandatory. The IP address to be statically assigned to the device.
mac-address	Mandatory. The MAC address to be statically mapped to the specified IP address.

dns-server	Optional. Multi-node. Gives the address of a DNS server that is available to DHCP clients on this subnet. You can specify more than one DNS server by issuing this statement multiple times. The format is an IP address.
default-router	Optional. Gives the address of the default router for DHCP clients on this subnet. The default router should be on the same subnet as the client. The format is an IP address.
wins-server	Optional. Multi-node. Gives the address of a NetBIOS Windows Internet Naming Server (WINS) available to DHCP clients on this subnet. The WINS server provides a name resolution services the Microsoft DHCP clients can use to correlate host names to IP addresses. You can specify more than one WINS server by issuing this statement multiple times. The format is an IP address.
lease	Optional. Specifies how long the address assigned by the DHCP server will be valid, in seconds. The range is 120 to 4294967296. The default is 86400 (24 hours).
domain-name	Optional. The client domain-name to configure. A domain name can include letters, numbers, hyphens (“-”), and one period (“.”).
authoritative	Optional. Enables and disables authoritative state. Supported values are as follows: **enable**: Enables authoritative state. **disable**: Disables authoritative state. The default is **disable**.

Usage Guidelines

Use this statement to configures a pool of addresses the router can use for Dynamic Host Configuration Protocol (DHCP).

At least one address pool must be configured for DHCP to be available as a service.

Each subnet requires a distinct address pool. A given interface can support more than one address pool (that is, more than one subnet), but it must have an IP address for each subnet it is supporting.

service http

Configures HTTP as an access protocol on the router.

Command Mode

Configuration mode.

Syntax

`set service http …`	Use **set** to enable or disable the HTTP service, or set the port to be used for HTTP.
`delete service http …`	Use **delete** to delete the specified port configuration, resetting to the default value. You can also use **delete** to delete the HTTP configuration node. This disables HTTP access to the router.

Configuration Statement

```
service {
   http {
      port: 1-65534
   }
}
```

Parameters

port	The port the system will use for the HTTP service. The range is 1 to 65534. The default is 80.

Usage Guidelines

Use this statement to configure the router to allow HTTP requests from remote systems to the local router.

Creating the HTTP configuration node enables HTTP as an access protocol. By default, the router uses port 80 for the HTTP service.

service ssh

Configures SSH as an access protocol on the router.

Command Mode

Configuration mode.

Syntax

`set service ssh …`	Use **set** to create the SSH configuration node. This enables the SSH service. You can also use **set** to set the port value or protocol version after the SSH configuration node has been created.
`delete service ssh …`	Use **delete** to delete the specified port or protocol version configuration, resetting to the default values. You can also use **delete** to delete the SSH configuration node. This disables SSH access to the router.

Configuration Statement

```
service {
   ssh {
      port: 1-65534
      protocol-version: [v1|v2|all]
   }
}
```

Parameters

port	The port the system will use for the SSH service. The range is 1 to 65534. The default is port 22.
protocol-version	Specifies which versions of SSH are enabled. Supported values are as follows: **v1**: SSH version 1 is enabled. **v2**: SSH version 2 is enabled. **all**: Both SSH version 1 and SSH version 2 are enabled. The default is **v2**.

Usage Guidelines

Use this statement to configure the router to allow SSH requests from remote systems to the local router.

Creating the SSH configuration node enables SSH as an access protocol. By default, the router uses port 22 for the SSH service, and SSH version 2 alone is used.

service telnet

Configures Telnet as an access protocol on the router.

Command Mode

Configuration mode.

Syntax

`set service telnet …`	Use **set** to create the **telnet** configuration node. This enables the Telnet service. You can also use **set** to set the port value after the Telnet configuration node has been created.
`delete service telnet …`	Use **delete** to delete the specified port or protocol version configuration, resetting to the default values. You can also use **delete** to delete the Telnet configuration node. This disables Telnet access to the router.

Configuration Statement

```
service {
   telnet {
      port: 1-65534
   }
}
```

Parameters

port	The port the system will use for the Telnet service. The range is 1 to 65534. The default is port 23.

Usage Guidelines

Use this statement to configure the router to accept Telnet as an access service to the router.

Creating the Telnet configuration node enables Telnet as an access protocol. By default, the router uses port 23 for the Telnet service.

show dhcp leases

Displays current DHCP lease information.

Command Mode

Operational mode.

Syntax

`show dhcp leases [pool *name*]`

Parameters

pool	Shows lease information for the specified address pool.

Usage Guidelines

Use this command to see current lease information for DHCP subscribers.

When used with no option, this command displays all current lease information. When address pool is provided, this command displays lease information for the specified address pool.

DHCP is configured using the the **service dhcp-server** command (see page 134).

show dhcp statistics

Displays DHCP server statistics.

Command Mode

Operational mode.

Syntax

```
show dhcp statistics [server-name]
```

Parameters

server-name	Shows statistics for the specified DHCP server.

Usage Guidelines

Use this command to see current lease information for DHCP subscribers.

When used with no option, this command displays all current lease information. When address pool is provided, this command displays lease information for the specified address pool.

The information shown includes the following:

- Number of DHCP requests
- Number of DHCP responses
- Total addresses in pool
- Number of addresses available
- Number of addresses assigned
- IP subnet(s) in pool
- Interface on which the DHCP pool is configured

DHCP is configured using the the **service dhcp-server** command (see page 134).

Chapter 6: Forwarding and Routing

This chapter lists commands for enabling and disabling forwarding, and for displaying general routing information.

This chapter contains the following commands.

Command	Mode	Description
`multicast mfea4`	Configuration	Enables or disables multicast forwarding for IPv4.
`multicast mfea6`	Configuration	Enables or disables multicast forwarding for IPv6.
`protocols fib2mrib`	Configuration	Enables or disables the FIB2MRIB module, which adds routing entries to the Multicast Routing Information Base.
`show mfea dataflow`	Operational	Displays information about IPv4 multicast forwarding data filters.
`show mfea interface`	Operational	Displays information about IPv4 multicast interfaces.
`show mfea6 dataflow`	Operational	Displays information about IPv6 multicast forwarding data filters.
`show mfea6 interface`	Operational	Displays information about IPv6 multicast interfaces.
`show route`	Operational	Displays information about routes stored in the routing table.

multicast mfea4

Enables or disables multicast forwarding for IPv4.

Command Mode

Configuration mode.

Syntax

`set multicast mfea4 …`	Use **set** to enable or disable multicast forwarding for IPv4.
`delete multicast mfea4 …`	Use **delete** to delete the **multicast mfea4** configuration node. This disables multicast forwarding for IPv4.

Configuration Statement

```
mfea4 {
    disable:bool
    interface: eth0..eth23
    traceoptions {
        flag {
            all {
                disable:bool
            }
        }
    }
}
```

Parameters

disable	Enables or disables multicast forwarding for IPv4. Supported values are: **true**—Disables multicast forwarding for IPv4, without discarding configuration. **false**—Enables multicast forwarding for IPv4. The default is **false**.

interface	Multi-node. The network interface to enable IPv4 multicast forwarding on. The network interface must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging” for information on creating and configuring network interfaces.) You can enable IPv4 multicast forwarding on more than one interface by creating multiple **interface** configuration nodes within the **multicast mfea4** node.
traceoptions	Sets the tracing and debugging options for IPv4 multicast forwarding.
flag	Specifies which tracing options are enabled.
all	Enables or disables all tracing options.
disable	Optional. Enables or disables debugging output for IPv4 multicast forwarding. Supported values are as follows: **true**—Disables debugging output for IPv4 multicast forwarding. **false**—Enables debugging output for IPv4 multicast forwarding. The default is **false**.

Usage Guidelines

Use this command to enable or disable multicast forwarding for IPv4.

Unicast forwarding is automatically enabled on the Vyatta OFR, but multicast forwarding must be explicitly enabled. You must enable multicast forwarding on each interface on which you intend to route multicast traffic.

multicast mfea6

Enables or disables multicast forwarding for IPv6.

Command Mode

Configuration mode.

Syntax

`set multicast mfea6 …`	Use **set** to enable or disable multicast forwarding for IPv6.
`delete multicast mfea6 …`	Use **delete** to delete the **multicast mfea6** configuration node. This disables multicast forwarding for IPv6.

Configuration Statement

```
mfea6 {
    disable:bool
    interface: eth0..eth23
    traceoptions {
        flag {
            all {
                disable:bool
            }
        }
    }
}
```

Parameters

disable	Enables or disables multicast forwarding for IPv6. Supported values are: **true**—Disables multicast forwarding for IPv6, without discarding configuration. **false**—Enables multicast forwarding for IPv6. The default is **false**.

interface	Multi-node. The network interface to enable IPv6 multicast forwarding on. The network interface must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging” for information on creating and configuring network interfaces.) You can enable IPv6 multicast forwarding on more than one interface by creating multiple **interface** configuration nodes within the **multicast mfea6** node.
traceoptions	Sets the tracing and debugging options for IPv6 multicast forwarding.
flag	Specifies which tracing options are enabled.
all	Enables or disables all tracing options.
disable	Optional. Enables or disables debugging output for IPv64 multicast forwarding. Supported values are as follows: **true**—Disables debugging output for IPv6 multicast forwarding. **false**—Enables debugging output for IPv6 multicast forwarding. The default is **false**.

Usage Guidelines

Use this command to enable or disable multicast forwarding for IPv64.

Unicast forwarding is automatically enabled on the Vyatta OFR, but multicast forwarding must be explicitly enabled. You must enable multicast forwarding on each interface on which you intend to route multicast traffic.

protocols fib2mrib

Enables or disables the FIB2MRIB module, which adds routing entries to the Multicast Routing Information Base.

Command Mode

Configuration mode.

Syntax

`set protocols fib2mrib …`	Use **set** to enable or disable the FIB2MRIB module.
`delete interfaces bridge …`	Use **delete** to delete the FIB2MRIB module. This means that unicast routing information will not be replicated into the Multicast RIB.

Configuration Statement

Parameters

disable	Enables or disables the FIB2MRIB module. Supported values are: **true**—Disables the FIB2MRIB module. **false**—Enables the FIB2MRIB module. The default is **false** (that is, when created, the module is automatically enabled).

Usage Guidelines

Use this command to enable or disable the FIB2MRIB.

If there are no unicast routing protocols configured in the router to supply the MRIB routes, then the FIB2MRIB module can be used to populate the MRIB. If the FIB2MRIB module is enabled, it will register with the Forwarding Engine Abstraction (FEA) to read the whole unicast forwarding table from the underlying system, and to receive notifications for all future modifications of that table. In other words, the FIB2MRIB's task is to replicate the unicast forwarding information on that router into the MRIB.

Examples

Example 6-1 creates the FIB2MRIB module. By default, this module is enabled when created.

Example 6-1 Populating the MRIB using the FIB2MRIB module

```
root@R1# set protocols fib2mrib
[edit]
root@R1# commit
[edit]
OK
root@R1# show protocols fib2mrib

[edit]
root@R1# show -all protocols fib2mrib
    disable: false

[edit]
root@R1#
```

show mfea dataflow

Displays information about IPv4 multicast forwarding data filters.

Command Mode

Operational mode.

Configuration Statement

```
show mfea dataflow
```

Parameters

None

Usage Guidelines

Use this command to view information about IPv4 multicast forwarding data filters.

This command is only available once the **multicast mfea4** configuration node has been created.

show mfea interface

Displays information about IPv4 multicast interfaces.

Command Mode

Operational mode.

Configuration Statement

`show mfea interface` *`int-name`* `[address` *`ipv4`*`]`

Parameters

int-name	Displays information for the specified Ethernet interface.
address	Displays information for the specified IPv4 multicast address.

Usage Guidelines

Use this command to view information about IPv4 multicast interfaces.

This command is only available once the **multicast mfea4** configuration node has been created.

show mfea6 dataflow

Displays information about IPv6 multicast forwarding data filters.

Command Mode

Operational mode.

Configuration Statement

```
show mfea6 dataflow
```

Parameters

None

Usage Guidelines

Use this command to view information about IPv6 multicast forwarding data filters.

This command is only available once the **multicast mfea6** configuration node has been created.

show mfea6 interface

Displays information about IPv6 multicast interfaces.

Command Mode

Operational mode.

Configuration Statement

```
show mfea6 interface int-name [address ipv6]
```

Parameters

int-name	Displays information for the specified Ethernet interface.
address	Displays information for the specified IPv6 multicast address.

Usage Guidelines

Use this command to view information about IPv6 multicast interfaces.

This command is only available once the **multicast mfea6** configuration node has been created.

show route

Displays information about routes stored in the routing table.

Command Mode

Operational mode.

Syntax

```
show route [[exact] prefix |
            protocol text |
            prefix-length 0–32 |
            next-hop {ipv4 | ipv6} |
            system [forward]]
```

Parameters

exact	Displays exact prefix matches only.
prefix	Lists all active prefixes matching the specified prefix. When used without the **exact** option, this includes both prefixes that are exact matches and prefixes that are longer than the specified prefix. The format is *ip-address/prefix-length*. For example, for a prefix of **10.0.0.0/8** all routes matching 10.0.0.0/8 or longer (**10.0.0.0/16**, **10.1.0.0/24**, and so on) are displayed.
protocol	Displays all the active prefixes in the RIB that were learned through the specified protocol. Supported values are as follows: **connected**: Displays directly connected routes. **static**: Displays static routes **bgp**: Displays both iBGP and eBGP routes. **ibgp**: Displays iBGP routes only. **ebgp**: Displays eBGP routes only. **rip**: Displays RIP routes. **ospf**: Displays OSPF routes.
prefix-length	Lists all active prefixes that have the specified prefix-length. The range is 0 to 32.
next-hop	Lists all the active prefixes that have the specified IPv4 or IPv6 address as the next hop.

system	Lists all routes in the system routing table.
forward	Lists all routes in the system forwarding table.

Usage Guidelines

Use this command to display route information.

When used with no option, this command lists all the active prefixes stored in the Routing Information Base (RIB), with summary information at the top. The summary information includes the following:

- Total routes. The number of prefixes in the RIB.
- Total paths. The number of routes in the RIB. This will be equal to the total routes unless there are routes with multiple next-hops.
- Routes in this view. The number of prefixes in the RIB matching the specified option.
- Paths in this view. The number of routes in the RIB matching the specified option. This will be equal to the total routes unless there are routes with multiple next-hops.

Examples

Example 6-2 shows all routes in the RIB using the default output format (brief).

Example 6-2 "show route": Displaying routes

```
root@vyatta> show route
Total routes: 13, Total paths: 13
10.0.0.0/8      [static(1)]     > to 192.168.2.1  via eth2/eth2
10.0.0.0/24     [connected(0)]  > to 10.0.0.50  via eth0/eth0
25.0.0.0/8      [ebgp(0)]       > to 10.0.0.100 via eth0/eth0
25.25.0.0/16    [ebgp(0)]       > to 10.0.0.100 via eth0/eth0
25.25.25.0/24   [ebgp(0)]       > to 10.0.0.100   via eth0/eth0
26.0.0.0/8      [ospf(1)]       > to 10.0.0.100 via eth0/eth0
26.26.0.0/16    [ospf(1)]       > to 10.0.0.100 via eth0/eth0
26.26.26.0/24   [ospf(1)]       > to 10.0.0.100 via eth0/eth0
27.0.0.0/8      [rip(2)]        > to 10.0.0.100 via eth0/eth0
27.27.0.0/16    [rip(2)]        > to 10.0.0.100 via eth0/eth0
27.27.27.0/24   [rip(2)]        > to 10.0.0.100 via eth0/eth0
172.16.0.0/14   [connected(0)]  > to 172.16.0.50  via eth1/eth1
192.168.2.0/24  [connected(0)]  > to 192.168.2.31 via eth2/eth2
```

Example 6-3 displays static routes.

Example 6-3 “show route”: Displaying static routes

```
root@vyatta> show route protocol static
Total routes: 13, Total paths: 13
Routes in this view: 1, Paths in this view: 1

10.0.0.0/8      [static(1)]    > to 192.168.2.1  via eth2/eth2
```

Example 6-4 displays routes with a prefix length of 16.

Example 6-4 “show route”: Displaying routes of a specified prefix length

```
root@vyatta> show route prefix-length 16
Total routes: 13, Total paths: 13
Routes in this view: 2, Paths in this view: 2

25.25.0.0/16    [ebgp(0)]       > to 10.0.0.100 via eth0/eth0
26.26.0.0/16    [ospf(1)]       > to 10.0.0.100 via eth0/eth0
27.27.0.0/16    [rip(2)]        > to 10.0.0.100 via eth0/eth0
```

Example 6-5 displays routes with a next hop of 10.0.0.100.

Example 6-5 “show route”: Displaying routes with a specified next hop

```
root@vyatta> show route next-hop 10.0.0.100
Total routes: 13, Total paths: 13
Routes in this view: 9, Paths in this view: 9

25.0.0.0/8      [ebgp(0)]       > to 10.0.0.100 via eth0/eth0
25.25.0.0/16    [ebgp(0)]       > to 10.0.0.100 via eth0/eth0
25.25.25.0/24   [ebgp(0)]      > to 10.0.0.100   via eth0/eth0
26.0.0.0/8      [ospf(1)]       > to 10.0.0.100 via eth0/eth0
26.26.0.0/16    [ospf(1)]       > to 10.0.0.100 via eth0/eth0
26.26.26.0/24   [ospf(1)]       > to 10.0.0.100 via eth0/eth0
27.0.0.0/8      [rip(2)]        > to 10.0.0.100 via eth0/eth0
27.27.0.0/16    [rip(2)]        > to 10.0.0.100 via eth0/eth0
27.27.27.0/24   [rip(2)]        > to 10.0.0.100 via eth0/eth0
```

Example 6-6 pipes the output of the **show route system forward** command through the UNIX **count** command, to display the total number of entries in the system forwarding table.

Example 6-6 "show route": Piping output through a UNIX command

```
root@vyatta> show route system forward | count

Count: 137937 lines
root@vyatta>
```

Chapter 7: Static Routes

This chapter lists the commands for configuring static routes on the Vyatta OFR.

A static route is a manually configured route, which in general cannot be updated dynamically from information the router learns about the network topology. However, if a link fails, the router will remove routes, including static routes, from the RIB that used that interface to reach the next hop.

This chapter contains the following commands.

Command	Mode	Description
`protocols static`	Configuration	Allows you to configure unicast and multicast static routes.

See also the following commands in other chapters.

`policy as-path-list`	Configuration	Allows you to create a list of AS paths, which can be referenced in BGP policy statements. *See page 268.*
`show route`	Operational	This chapter lists commands for enabling and disabling forwarding, and for displaying general routing information. *See page 155.*

protocols static

Allows you to configure unicast and multicast static routes.

Command Mode

Configuration mode.

Syntax

`set protocols static …`	Use **set** to create the **static** configuration node, or to change static route configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new configuration node with the correct information.
`delete protocols static …`	Use **delete** to delete the static configuration node altogether, to delete a specific route, or to delete an import policy.

Configuration Statement

```
protocols {
   static {
      disable: [true|false]
      route: ipv4net {
         next-hop: ipv4
         metric: 1-65535
      }
      interface-route: ipv4net {
         next-hop-interface: text
         next-hop-router: ipv4
         metric: 1-65535
      }
      import: text
   }
}
```

Parameters

disable	Specifies whether any static routes are installed or not. Supported values are as follows: **true**—Deletes the entire static routes configuration, but without removing configuration information. **false**—Enables the static routes configuration that has been specified. The default is **false**.
route	Multi-node. Defines a unicast route. The format is a destination subnet of the form *address/prefix*. You can define multiple routes by creating multiple **route** configuration nodes.
next-hop	Mandatory. The IPv4 address of the next-hop router toward the destination subnet.
metric	Optional. The routing metric or cost for this route. The format is a non-negative integer, where lower values indicate better routes. The metric for a static route is not directly used to decide which route to use, but may affect the choice of routes for protocols such as BGP or PIM-SM that indirectly use this information. For example, BGP uses the IGP metric to the next hop to decide between alternative routes as part of its decision process. The default metric is 1.
interface-route	Multi-node. Defines a interface-based static route. The format is a destination subnet of the form *address/prefix.* You can define multiple interface-based routes by creating multiple **interface-route** configuration nodes.
next-hop-interface	Mandatory. The name of the next-hop interface toward the destination subnet.
next-hop-router	Optional. The address of the next-hop router. The default is 0.0.0.0.

metric	Optional. The routing metric or cost for this route. The format is a non-negative integer, where lower values indicate better routes. The metric for a static route is not directly used to decide which route to use, but may affect the choice of routes for protocols such as BGP or PIM-SM that indirectly use this information. For example, BGP uses the IGP metric to the next hop to decide between alternative routes as part of its decision process. The default metric is 1.
import	The name of an import routing policy.

Usage Guidelines

Use this statement to configure static routes on the router, or to specify import policies to be applied to static routes. You can configure unicast and multicast routes.

Chapter 8: RIP

This chapter lists the commands for setting up the Routing Information Protocol (RIP) on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`protocols rip`		Allows you to configure RIP for IPv4 on the router.
`protocols ripng`		Allows you to configure RIP for IPv6 on the router.
`show rip peer`		Displays information for the RIP peers of this router.
`show rip statistics`		Displays RIP statistics.
`show rip status`		Displays RIP status.

See also the following commands in other chapters.

`policy as-path-list`	Configuration	Allows you to create a list of AS paths, which can be referenced in BGP policy statements. *See page 268.*
`show route`	Operational	Displays information about routes stored in the routing table. *See page 155.*

protocols rip

Allows you to configure RIP for IPv4 on the router.

Syntax

`set protocols rip …`	Use **set** to create the **rip** configuration node, or to modify RIP configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new node with the correct information.
`delete protocols rip …`	Use **delete** to delete the **rip** configuration node altogether, or to delete one of its subordinate nodes.

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
   rip {
      interface: text {
         address: ipv4 {
            metric: 0–16
            horizon:
               [none|split-horizon|split-horizon-poison-reverse]
            disable: [true|false]
            passive: [true|false]
            accept-non-rip-requests: [true|false]
            accept-default-route: [true|false]
            advertise-default-route: [true|false]
            route-timeout: 1–4294967296
            route-expiry-secs: 1–4294967296
            deletion-delay:1–4294967296
            route-deletion-secs: 1–4294967296
            triggered-delay: 1–4294967296
            triggered-jitter: 1–4294967296
            update-interval: 1–4294967296
            update-jitter: 1–4294967296
            request-interval: 1–4294967296
            interpacket-delay: 1–4294967296
            authentication {
               simple-password: text
```

```
                  md5: 0–255 {
                      password: text
                      start-time: YYYY-MM-DD.HH:MM
                      end-time: YYYY-MM-DD.HH:MM
                  }
              }
          }
      }
      import: text
      export: text
  }
}
```

Parameters

interface	Mandatory. Multi-node. The name of a network interface to be used by RIP for routing. The network interface must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging” for information on creating and configuring network interfaces.) You can enable RIP on more than one interface by creating multiple **interface** configuration nodes within the **rip** node. You can enable RIP on an individual vif, rather than an entire interface. To do this, refer to the vif using *int.vif* notation. For example to refer to vif 40 on interface eth0, use **eth0.40**.
address	Mandatory. Multi-node. An IPv4 address to be used by RIP for routing. RIP will peer with other routers using this address. The address must already be created and configured on the interface. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging” for information on configuring IP addresses.) You can enable RIP on more than one address by creating multiple **address** configuration nodes within the **interface** node.

metric	Optional. The metric or cost associated with routes received on this address. The metric is added to the cost in routes received before deciding between best routes to the same destination subnet. The sum of all the metrics across the entire RIP domain should be less than 16. The range is 0 to 16, where 16 means "infinity." The default is 1.
horizon	Optional. Specifies how the router should treat RIP updates to its neighbors. Valid values are as follows: **split-horizon-poison-reverse**: Announce routes back to neighbors from which they were learned with a metric of 16 (infinity). **split-horizon**: Omit the route in announcements to the neighbor from which the route was learned. **none**: Employs no strategy to eliminate failed routes. The default is **split-horizon-poison-reverse**. Under normal circumstances, this value is recommended.
disable	Optional. Determines whether RIP will exchange routes via this address. Supported values are as follows: **true**: Disables RIP routing on this address, without discarding configuration. **false**: Enables RIP routing on this address. The default is **false**.
passive	Optional. Determines whether RIP runs in passive mode on this address. Supported values are as follows: **true**: Operates in passive mode, where RIP will accept routes received on this address, but will not advertise any routes to neighbors via this address. **false**: RIP will both receive routes received on this address and advertise any routes to neighbors via this address. The default is **false**.

accept-non-rip-requests	Optional. Determines whether RIP will allow requests to be unicast, so that they can be sourced from non-RIP ports. Normal RIPv2 requests for routing updates are multicast to all neighbors and sourced from the RIP port. However, for monitoring purposes RIP also allows requests to be unicast, and then they can be sourced from non-RIP ports. Supported values are as follows: **true**: Accepts RIP requests from any UDP port. **false**: Does not accepts RIP requests from non-RIP ports. The default is **true**.
accept-default-route	Optional. Determines whether RIP should accept a default route if it receives one from a RIP neighbor. Supported values are as follows: **true**: Accepts a default route from a RIP neighbor. **false**: Does not accept a default route from a RIP neighbor. The default is **true**.
advertise-default-route	Optional. Determines whether RIP should advertise the default route. Supported values are as follows: **true**: Advertise the default route. **false**: Do not advertise the default route. The default is true.
route-timeout	Optional. Sets the route expiry interval. If no periodic or triggered update of a route from this neighbor has been received within this time interval, the route is considered to have expired. The range is 1 to 4294967296. The default is 180 seconds, which should not normally need to be changed.

route-expiry-secs	Optional. Determines how long the router maintains expired routes after their metric has reached infinity. After a route has expired (that is, after the route has been assigned an infinite metric), the router must keep a copy of it for a certain time so it can be reasonably confident it has told its neighbors that the route has expired. The range is 1 to 4294967296. The default is 120 seconds, which should not normally need to be changed.
deletion-delay	The delay, in seconds, before an expired route is deleted from the routing information base. The range is 1 to 4294967296. The default is 120.
triggered-delay	Optional. Sets the interval, in seconds, for the triggered update timer. When a router receives a modified route from a neighbor, it does not have to wait until the next periodic update to tell the other neighbors, but instead sends a triggered update. After a triggered update is sent, a timer is set for a random period in the interval specified by **triggered-jitter**. If other changes occur that would trigger updates before the timer expires, a single update is triggered when the timer expires. The range is 1 to 4294967296. The default is 3.
triggered-jitter	Optional. Sets the interval, in seconds, from within which the triggered update timer will randomly select an interval for triggered updates. The range is 0 to 100, where zero means use no random jitter (that is, always use the time specified in **triggered-delay**). The default is 66.
update-interval	Optional. The interval, in seconds, of routing updates. A RIP router will typically tell its neighbors its entire routing table every 30 seconds. To avoid self-synchronization of routing updates, the precise time interval between telling each neighbor about routing updates is randomly jittered, with the delay chosen in the interval specified by **update-jitter**. The range is 1 to 4294967296. The default is 30.

update-jitter	Optional. Sets the interval, in seconds, from within which the update timer will randomly select an interval for routing updates. The range is 0 to 100, where 0 means use no random jitter (that is, always use the time specified in **update-interval**). The default is 35.
request-interval	Optional. Determines how often a route update request may be sent. When a RIP router has no neighbors on a address, it may periodically send a request for a route update in case a neighbor appears. This timer determines how often such a request is re-sent. The range is 1 to 10000, and 0, which disables route update requests. The default is 30 seconds.
interpacket-delay	Optional. The default delay, in milliseconds, between back-to-back RIP packets when an update is sent that requires multiple packets to be sent. The range is 1 to 4294967296. The default is 50.
authentication	Optional. The authentication mechanism used to authorize RIP updates sent and received via this address.
simple-password	Optional. The password to be used for plaintext authentication on this address. The default is an empty string.
md5	Multi-node. An integer specifying the MD5 authentication key. The range is 0 to 255. You can define more than one MD5 authentication key by creating multiple **md5** configuration nodes.
password	The password to be used for this MD5 authentication key.
start-time	The start time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.
end-time	The end time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.

import	Optional. A RIP import policy defined using the **policy** statement. The import policy will be used to evaluate routing updates received by RIP. For policy terms that match, the defined action will be taken. Multiple policies can be configured using a comma-separated list of policy names.
export	Optional. A RIP export policy defined using the **policy** statement. The import policy will be used to evaluate routing updates sent to neighbors. For policy terms that match, the defined action will be taken. Multiple policies can be configured using a comma-separated list of policy names.

Usage Guidelines

Use this statement to configure RIP for IPv4 on the router. You can also use this statement to announce routes.

To announce routes, you export the routes that are to be announced, using the export parameter. You can export routes on directly connected networks or static routes using the **export** *policy-name* directive.

protocols ripng

Allows you to configure RIP for IPv6 on the router.

Syntax

`set protocols ripng …`	Use **set** to create the **ripng** configuration node, or to modify RIP configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new node with the correct information.
`delete protocols ripng …`	Use **delete** to delete the **ripng** configuration node altogether, or to delete one of its subordinate nodes.

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
   ripng {
      interface: text {
         address: ipv6 {
            metric: 0–16
            horizon:
               [none|split-horizon|split-horizon-poison-reverse]
            disable: [true|false]
            passive: [true|false]
            accept-non-rip-requests: [true|false]
            accept-default-route: [true|false]
            advertise-default-route: [true|false]
            route-timeout: 1–4294967296
            route-expiry-secs: 1–4294967296
            deletion-delay:1–4294967296
            route-deletion-secs: 1–4294967296
            triggered-delay: 1–4294967296
            triggered-jitter: 1–4294967296
            update-interval: 1–4294967296
            update-jitter: 1–4294967296
            request-interval: 1–4294967296
            interpacket-delay: 1–4294967296
            authentication {
```

```
                    simple-password: text
                    md5: 0–255 {
                        password: text
                        start-time: YYYY-MM-DD.HH:MM
                        end-time: YYYY-MM-DD.HH:MM
                    }
                }
            }
        }
        import: text
        export: text
    }
}
```

Parameters

interface	Mandatory. Multi-node. The name of a network interface to be used by RIP for routing. The network interface must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging" for information on creating and configuring network interfaces.) You can enable RIPng on more than one interface by creating multiple **interface** configuration nodes within the **rip** node. You can enable RIPng on an individual vif, rather than an entire interface. To do this, refer to the vif using *int.vif* notation. For example to refer to vif 40 on interface eth0, use **eth0.40**.
address	Mandatory. Multi-node. An IPv4 address to be used by RIP for routing. RIP will peer with other routers using this address. The address must already be created and configured on the interface. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging" for information on configuring IP addresses.) You can enable RIP on more than one address by creating multiple **address** configuration nodes within the **interface** node.

metric	Optional. The metric or cost associated with routes received on this address. The metric is added to the cost in routes received before deciding between best routes to the same destination subnet. The sum of all the metrics across the entire RIP domain should be less than 16. The range is 0 to 16, where 16 means “infinity.” The default is 1.
horizon	Optional. Specifies how the router should treat RIP updates to its neighbors. Valid values are as follows: **split-horizon-poison-reverse**: Announce routes back to neighbors from which they were learned with a metric of 16 (infinity). **split-horizon**: Omit the route in announcements to the neighbor from which the route was learned. **none**: Employs no strategy to eliminate failed routes. The default is **split-horizon-poison-reverse**. Under normal circumstances, this value is recommended.
disable	Optional. Determines whether RIP will exchange routes via this address. Supported values are as follows: **true**: Disables RIP routing on this address, without discarding configuration. **false**: Enables RIP routing on this address. The default is **false**.
passive	Optional. Determines whether RIP runs in passive mode on this address. Supported values are as follows: **true**: Operates in passive mode, where RIP will accept routes received on this address, but will not advertise any routes to neighbors via this address. **false**: RIP will both receive routes received on this address and advertise any routes to neighbors via this address. The default is **false**.

accept-non-rip-requests	Optional. Determines whether RIP will allow requests to be unicast, so that they can be sourced from non-RIP ports. Normal RIPv2 requests for routing updates are multicast to all neighbors and sourced from the RIP port. However, for monitoring purposes RIP also allows requests to be unicast, and then they can be sourced from non-RIP ports. Supported values are as follows: **true**: Accepts RIP requests from any UDP port. **false**: Does not accepts RIP requests from non-RIP ports. The default is **true**.
accept-default-route	Optional. Determines whether RIP should accept a default route if it receives one from a RIP neighbor. Supported values are as follows: **true**: Accepts a default route from a RIP neighbor. **false**: Does not accept a default route from a RIP neighbor. The default is **true**.
advertise-default-route	Optional. Determines whether RIP should advertise the default route. Supported values are as follows: **true**: Advertise the default route. **false**: Do not advertise the default route. The default is **true**.
route-timeout	Optional. Sets the route expiry interval. If no periodic or triggered update of a route from this neighbor has been received within this time interval, the route is considered to have expired. The range is 1 to 4294967296. The default is 180 seconds, which should not normally need to be changed.

route-expiry-secs	Optional. Determines how long the router maintains expired routes after their metric has reached infinity. After a route has expired (that is, after the route has been assigned an infinite metric), the router must keep a copy of it for a certain time so it can be reasonably confident it has told its neighbors that the route has expired. The range is 1 to 4294967296. The default is 120 seconds, which should not normally need to be changed.
deletion-delay	The delay, in seconds, before an expired route is deleted from the routing information base. The range is 1 to 4294967296. The default is 120.
triggered-delay	Optional. Sets the interval, in seconds, for the triggered update timer. When a router receives a modified route from a neighbor, it does not have to wait until the next periodic update to tell the other neighbors, but instead sends a triggered update. After a triggered update is sent, a timer is set for a random period in the interval specified by **triggered-jitter**. If other changes occur that would trigger updates before the timer expires, a single update is triggered when the timer expires. The range is 1 to 4294967296. The default is 3.
triggered-jitter	Optional. Sets the interval, in seconds, from within which the triggered update timer will randomly select an interval for triggered updates. The range is 0 to 100, where zero means use no random jitter (that is, always use the time specified in **triggered-delay**). The default is 66.
update-interval	Optional. The interval, in seconds, of routing updates. A RIP router will typically tell its neighbors its entire routing table every 30 seconds. To avoid self-synchronization of routing updates, the precise time interval between telling each neighbor about routing updates is randomly jittered, with the delay chosen in the interval specified by **update-jitter**. The range is 1 to 4294967296. The default is 30.

update-jitter	Optional. Sets the interval, in seconds, from within which the update timer will randomly select an interval for routing updates. The range is 0 to 100, where 0 means use no random jitter (that is, always use the time specified in **update-interval**). The default is 35.
request-interval	Optional. Determines how often a route update request may be sent. When a RIP router has no neighbors on a address, it may periodically send a request for a route update in case a neighbor appears. This timer determines how often such a request is re-sent. The range is 1 to 10000, and 0, which disables route update requests. The default is 30 seconds.
interpacket-delay	Optional. The default delay, in milliseconds, between back-to-back RIP packets when an update is sent that requires multiple packets to be sent. The range is 1 to 4294967296. The default is 50.
authentication	Optional. The authentication mechanism used to authorize RIP updates sent and received via this address.
simple-password	Optional. The password to be used for plaintext authentication on this address. The default is an empty string.
md5	Multi-node. An integer specifying the MD5 authentication key. The range is 0 to 255. You can define more than one MD5 authentication key by creating multiple **md5** configuration nodes.
password	The password to be used for this MD5 authentication key.
start-time	The start time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.
end-time	The end time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.

import	Optional. A RIP import policy defined using the **policy** statement. The import policy will be used to evaluate routing updates received by RIP. For policy terms that match, the defined action will be taken. Multiple policies can be configured using a comma-separated list of policy names.
export	Optional. A RIP export policy defined using the **policy** statement. The import policy will be used to evaluate routing updates sent to neighbors. For policy terms that match, the defined action will be taken. Multiple policies can be configured using a comma-separated list of policy names.

Usage Guidelines

Use this statement to configure RIP for IPv6 on the router. You can also use this statement to announce routes.

To announce routes, you export the routes that are to be announced, using the export parameter. You can export routes on directly connected networks or static routes using the **export** *policy-name* directive.

show rip peer

Displays information for the RIP peers of this router.

Command Mode

Operational mode.

Syntax

`show rip peer [statistics [`*`ipv4`*` | `*`ipv6`*` | all]]`

Parameters

ipv4	Displays peer statistics for the specified IPv4 address.
ipv6	Displays peer statistics for the specified IPv6 address.
all	Displays peer statistics for all RIP interfaces on the router.

Usage Guidelines

Use this command to display information about RIP peers.

show rip statistics

Displays RIP statistics.

Command Mode

Operational mode.

Syntax

`show rip statistics [ipv4 | ipv6 | all]`

Parameters

ipv4	Displays RIP statistics for the specified IPv4 address.
ipv6	Displays RIP statistics for the specified IPv6 address.
all	Displays statistics for all RIP interfaces on the router.

Usage Guidelines

Use this command to display RIP statistics for interfaces configured for RIP.

show rip status

Displays RIP status.

Command Mode

Operational mode.

Syntax

show rip status [*ipv4* | *ipv6* | all}

Parameters

ipv4	Displays RIP status for the specified IPv4 address.
ipv6	Displays RIP status for the specified IPv6 address.
all	Displays status for all RIP interfaces on the router.

Usage Guidelines

Use this command to see the status of RIP on the router.

Chapter 9: OSPF

This chapter lists the commands for configuring OSPF on the router.

This chapter contains the following commands.

Command	Mode	Description
`protocols ospf4`	Configuration	Configures OSPF on the router.
`show ospf4 database`	Operational	Displays the OSPF LSA database.
`show ospf4 database area`	Operational	Displays the OSPF LSA database for the specified area.
`show ospf4 database summary`	Operational	Displays summary output for the OSPF LSA database.
`show ospf4 database summary area`	Operational	Displays summary output for the specified area in the OSPF LSA database.
`show ospf4 neighbor`	Operational	Displays information about OSPF neighbors of this router.

See also the following commands in other chapters.

`policy as-path-list`	Configuration	Allows you to create a list of AS paths, which can be referenced in BGP policy statements. *See page 268.*
`show route`	Operational	Displays information about routes stored in the routing table. *See page 155.*

protocols ospf4

Configures OSPF on the router.

Command Mode

Configuration mode.

Syntax

`set protocols ospf4 …`	Use **set** to create the **ospf** configuration node, or to modify OSPF configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new configuration node with the correct information.
`delete protocols ospf4 …`	Use **delete** to delete the **ospf** configuration node altogether, or to delete one of its subordinate nodes.

Configuration Statement

```
protocols {
   ospf4 {
      router-id: ipv4
      rfc1583-compatibility: [true|false]
      ip-router-alert: [true|false]
      traceoptions {
         flag {
            all {
               disable:[true|false]
            }
         }
      }
      area: ipv4 {
         area-type:[normal|stub|nssa]
         default-lsa {
            disable:[true|false]
            metric: 1-4294967296
         }
         summaries {
            disable:[true|false]
         }
         area-range: ipv4net {
            advertise:[true|false]
```

```
            }
            virtual-link: ipv4 {
                transit-area: ipv4
                hello-interval:1–65535
                router-dead-interval: 1–4294967295
                retransmit-interval: 1–65535
                transit-delay:0–3600
                authentication {
                    simple-password:text
                    md5: 0–255 {
                        password: text
                        start-time: YYYY-MM-DD.HH:MM
                        end-time: YYYY-MM-DD.HH:MM
                        max-time-drift: 0–65534,65535
                    }
                }
            }
            interface: text {
                link-type:[broadcast|p2p|p2m]
                address: ipv4 {
                    priority:0–255
                    hello-interval:1–65535
                    router-dead-interval: 1–4294967296
                    interface-cost:1–65535
                    retransmit-interval: 1–65535
                    transit-delay:0–3600
                    authentication {
                        simple-password:text
                        md5: 0–255 {
                            password: text
                            start-time: YYYY-MM-DD.HH:MM
                            end-time: YYYY-MM-DD.HH:MM
                            max-time-drift: 0–65534,65535
                        }
                    }
                    passive: [true|false]
                    neighbor: ipv4 {
                        router-id: ipv4
                    }
                    disable: [true|false]
                }
            }
        }
        import: text
        export: text
    }
}
```

Parameters

router-id	Mandatory. The identifier of this router. This is a unique 32-bit number in IP address format that is assigned to each router running the OSPF protocol. This number uniquely identifies this router within the OSPF domain. It is good practice to set the OSPF router ID to the address of the loopback interface, since the loopback interface is the most reliable interface on the router.
rfc1583-compatibility	Indicates whether handling of AS external routes should comply with RFC 1583. Supported values are as follows: **true**: Comply with RFC 1583. **false**: Do not comply with RFC 1583. The default is **false**.
ip-router-alert	Optional. Indicates whether to send the IP router alert option in packets. Supported values are as follows: **true**: Send the IP router alert option in packets. **false**: Do not send the IP router alert option in packets. The default is **false**.
traceoptions	Sets the tracing and debugging options for OSPF.
flag	Specifies which tracing options are enabled.
all	Enables or disables all tracing options.
disable	Optional. Enables or disables debugging output for OSPF. Supported values are as follows: **true**: Disables debugging output for OSPF. **false**: Enables debugging output for OSPF. The default is **false**.
area	Mandatory. Multi-node. An IPv4 address uniquely identifying the OSPF area with which you want to associate the attached network. To configure the router as an Area Border Router, associate the router with more than one area by issuing this statement multiple times.

area-type	Mandatory. The type of the area. Supported values are as follows: **normal**: This is a normal OSPF area: one that is neither a stub area nor a not-so-stubby area. **stub**: This is a stub area: one where no external link-state advertisements (type 5 LSAs) are allowed. Any routers in a stub area must be configured with this option. **nssa**: This is a not-so-stubby area (NSSA): one where type 3 and 4 summary link-state advertisements (LSAs) are prevented from being sent into the specified area. In an NSSA, no inter-area routes are allowed. The default is **normal**.
default-lsa	Specifies characteristics of the default route.
disable	Enables and disables originating the default route in stubby or not-so-stubby areas. Supported values are as follows: **true**: Allow the default route to originate in stubby or not-so-stubby areas. **false**: Do not allow the default route to originate in stubby or not-so-stubby areas. The default is **true**.
metric	Provides the metric for the default route. The range is 0 to 4294967295. The default is 0.
summaries	Specifies whether route summaries should be generated into stubby and not-so-stubby areas.
disable	Enables and disables route summary generation into stubby and not-so-stubby areas. **true**: Do not generate summaries into stubby and not-so-stubby areas. **false**: Generate summaries into stubby and not-so-stubby areas. The default is **false**.
area-range	Optional. Multi-node The network for generating route summaries. **Area Border Routers only**. For an area, summarize a range of IP addresses when sending summary link advertisements into other areas. To summarize multiple ranges, include multiple area-range statements.

	NSSAs. Generate AS-External (Type 5) LSAs into other areas. The specified prefixes are used to aggregate external routes learned within the area when the routes are advertised to other areas. To specify multiple prefixes, include multiple **area-range** statements. All external routes learned within the area that do not fall into one of the prefixes are advertised individually to other areas. By default, Area Border Routers do not summarize routes being sent from one area to other areas, but rather send all routes explicitly. The format is *ipv4/prefix*. You can define multiple area ranges by creating multiple **area-range** configuration nodes.
advertise	Mandatory. Indicates whether to advertise type 3 summary link-state advertisements (LSAs). **true**: Causes the area to generate type 3 summary link-state advertisements (LSAs). **false**: Causes the area to suppress type 3 summary LSAs, hiding the area's component networks from other networks. This effectively creates a route filter. The default is **true**.
virtual-link	Multi-node. The IP address of the router in the backbone area that you are creating the virtual link to. This router becomes the virtual link neighbor. You can define multiple virtual links by creating multiple **virtual-link** configuration nodes.
transit-area	Optional. The area through which the virtual link will transit. The format is a 32-bit area identifier.
hello-interval	Optional. Specifies the interval in seconds between hello packets sent over the virtual link. The range is 1 to 65535. The default is 10.
router-dead-interval	Optional. Specifies the time in seconds that neighboring routers will wait to detect hello packets from the virtual link before declaring the router down. The range is 1 to 4294967295 seconds. The default is 40 (four times the hello interval).

retransmit-interval	Optional. Specifies the time in seconds to wait for an acknowledgement, after which the router retransmits an LSA packet to its neighbors. The range is 1 to 65535. The default is 5.
transit-delay	Optional. The interface transit delay, in seconds. Indicates the estimated time in seconds required to send a link-state advertisement on this interface. The range is 0 to 3600. The default is 1.
authentication	Optional. The authentication mechanism used to authorize OSPF updates sent from this address.
simple-password	Optional. The password to be used for plaintext authentication on this address. The default is an empty string.
md5	Multi-node. An integer specifying the MD5 authentication key. The range is 0 to 255. You can define more than one MD5 authentication key by creating multiple **md5** configuration nodes.
password	The password to be used for this MD5 authentication key.
start-time	The start time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.
end-time	The end time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.
max-time-drift	Sets the maximum time drift, in seconds, among all routers. The range is 0 to 65535, where 65535 means unlimited time drift is allowed.
interface	Mandatory. Multi-node. Enables OSPF hellos on the specified interface and attempts to discover OSPF neighbors. You can enable OSPF on multiple interfaces by creating multiple **interface** nodes. You can enable OSPF on an individual vif, rather than an entire interface. To do this, refer to the vif using *int.vif* notation. For example to refer to vif 40 on interface eth0, use **eth0.40**.

link-type	Mandatory. Specifies the correct interface type for this physical interface. The following values are supported: **broadcast**: This is an interface that supports broadcast mode (such as a LAN link). **p2p**: This is an interface that supports point-to-point mode (such as a PPP interface or a point-to-point logical interface on Frame Relay). **p2m**: This is an interface that supports point-to-multipoint mode (such as an NBMA interface). The default is broadcast.
address	Mandatory. Multi-node. Configures an IP address on this interface for use with OSPF traffic. You can define multiple OSPF-enabled addresses on an interface by creating multiple **address** nodes. The IP address must have already been configured on a created network interface.
priority	Optional. Sets the OSPF router priority for this address. The OSPF priority is used in determining whether the router becomes the designated router for this network. If all routers have the same priority, the first router activated on the network becomes the DR and the second router activated on the network becomes the BDR. The range is 0 to 255, where a router with priority 0 can never become the designated router. The default is 128.
hello-interval	Optional. Specifies the interval in seconds between hello packets sent over the interface you are configuring. The range is 1 to 65535. The default is 10.
router-dead-interval	Optional. Specifies the time in seconds that neighboring routers will wait to detect hello packets from the interface you are configuring before declaring the router down. The range is 1 to 4294967295 seconds. The default is 40 (four times the hello interval).
interface-cost	Optional. The link-state metric (OSPF cost) that you want advertised in the link-state advertisement (LSA) as the cost of sending packets over this interface. The range is 1 to 65535. The default is 1.

retransmit-interval	Optional. Specifies the time in seconds to wait for an acknowledgement, after which the router retransmits an LSA packet to its neighbors. The range is 1 to 65535. The default is 5.
transit-delay	Optional. The interface transit delay, in seconds. Indicates the estimated time in seconds required to send a link-state advertisement on this interface. The range is 0 to 3600. The default is 1.
authentication	Optional. The authentication mechanism used to authorize OSPF updates sent from this address.
simple-password	Optional. The password to be used for plaintext authentication on this address. The default is an empty string.
md5	Optional. Multi-node. An integer specifying the MD5 authentication key. The range is 0 to 255. You can define more than one MD5 authentication key by creating multiple **md5** configuration nodes.
password	Optional. The password to be used for this MD5 authentication key.
start-time	The start time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.
end-time	The end time of the interval when this authentication key and password will be valid. The format is *YYYY-MM-DD.HH*:*MM*.
max-time-drift	Sets the maximum time drift, in seconds, among all routers. The range is 0 to 65535, where 65535 means unlimited time drift is allowed.
passive	Optional. Sets the address into a loopback state. Supported values are as follows: **true**: Set the address into a loopback state. **false**: Do not set the address into a loopback state. The default is **false**.

neighbor	Optional. Multi-node. The IP address of a router to be designated as an OSPF neighbor. This value should be configured for nonbroadcast interfaces, which will not send broadcast packets to dynamically discover their neighbors. To specify multiple neighbors, create multiple **neighbor** configuration nodes.
router-id	Mandatory. The OSPF router ID of the neighbor router. An IPv4 address.
disable	Optional. Enables or disables OSPF on this address. Supported options are as follows: **true**: Disables OSPF on this address, without discarding configuration. **false**: Enables OSPF on this address. The default is **false**.
import	Optional. The name of a routing policy defined using the **policy** statement. A routing policy that is applied to OSPF as an import will be used to evaluate all routing updates that OSPF receives from its neighbors. Routes that match the routing policy will have the specified action taken; this can include **reject**, which would block the route from being installed in the routing table. Multiple policies can be configured using a comma-separated list of policy names.
export	Optional. The name of a routing policy defined using the **policy** statement. A routing policy that is configured as an export in OSPF will evaluate policy criteria for routing updates sent by this router to its OSPF neighbors. This could include exporting routes from other protocols, like static, into the OSPF routing updates. Multiple policies can be configured using a comma-separated list of policy names.

Usage Guidelines

Use this statement to configure OSPF on the router.

show ospf4 database

Displays the OSPF LSA database.

Command Mode

Operational mode.

Syntax

```
show ospf4 database [router | network | netsummary | asbrsummary |
   external | nssa] [brief | detail]
```

Parameters

router	Shows router (Type 1) LSAs in the LSA database.
network	Shows network (Type 2) LSAs in the LSA database.
netsummary	Shows network summary (Type 3) LSAs in the LSA database.
asbrsummary	Shows ASBR-summary (Type 4) LSAs in the LSA database.
external	Shows AS-external (Type 5) LSAs in the LSA database.
nssa	Shows NSSA (Type 7) LSAs in the LSA database.
brief	Displays brief output.
detail	Displays detailed output.

Usage Guidelines

Use this command to view the contents of the OSPF LSA database.

Only one option can be specified at a time.

show ospf4 database area

Displays the OSPF LSA database for the specified area.

Command Mode

Operational mode.

Syntax

```
show ospf4 database area area-id [router | network | netsummary |
   asbrsummary | external | nssa] [brief | detail]
```

Parameters

area-id	The ID of the area for which you want to view LSA information.
router	Shows router (Type 1) LSAs for the specified area.
network	Shows network (Type 2) LSAs for the specified area.
netsummary	Shows network summary (Type 3) LSAs for the specified area.
asbrsummary	Shows ASBR-summary (Type 4) LSAs for the specified area.
external	Shows AS-external (Type 5) LSAs for the specified area.
nssa	Shows NSSA (Type 7) LSAs for the specified area.
brief	Displays brief output.
detail	Displays detailed output.

Usage Guidelines

Use this command to see the contents of the OSPF LSA database for a specified area.

Only one option can be specified at a time.

show ospf4 database summary

Displays summary output for the OSPF LSA database.

Command Mode

Operational mode.

Syntax

```
show ospf4 database summary [router | network | netsummary |
   asbrsummary | external | nssa] [brief | detail]
```

Parameters

router	Shows summary output for router (Type 1) LSAs in the LSA database.
network	Shows summary output for network (Type 2) LSAs in the LSA database.
netsummary	Shows summary output for network summary (Type 3) LSAs in the LSA database.
asbrsummary	Shows summary output for ASBR-summary (Type 4) LSAs in the LSA database.
external	Shows summary output for AS-external (Type 5) LSAs in the LSA database.
nssa	Shows summary output for NSSA (Type 7) LSAs in the LSA database.
brief	Displays brief output.
detail	Displays detailed output.

Usage Guidelines

Use this command to see summary output for the OSPF LSA database.

Only one option can be specified at a time.

show ospf4 database summary area

Displays summary output for the specified area in the OSPF LSA database.

Command Mode

Operational mode.

Syntax

```
show ospf4 database summary area area-id [router | network |
   netsummary | asbrsummary | external | nssa] [brief | detail]
```

Parameters

area-id	The ID of the area for which you want to view LSA information.
router	Shows summary output for router (Type 1) LSAs for the specified area.
network	Shows summary output for network (Type 2) LSAs for the specified area.
netsummary	Shows summary output for network summary (Type 3) LSAs for the specified area.
asbrsummary	Shows summary output for ASBR-summary (Type 4) LSAs for the specified area.
external	Shows summary output for AS-external (Type 5) LSAs for the specified area.
nssa	Shows summary output for NSSA (Type 7) LSAs for the specified area.
brief	Displays brief output.
detail	Displays detailed output.

Usage Guidelines

Use this command to see summary output for the specified area in the OSPF LSA database.

Only one option can be specified at a time.

.

show ospf4 neighbor

Displays information about OSPF neighbors of this router.

Command Mode

Operational mode.

Syntax

`show ospf4 neighbor` *`neighbor`* `[brief | detail]`

Parameters

neighbor	Displays information about the specified neighbor.
brief	Displays brief output.
detail	Displays detailed output.

Usage Guidelines

Use this command to see information about OSPF neighbors to this router.

When used without specifying a neighbor, information is shown for all neighbors to this router. When a neighbor is specified, information is shown for just the specified neighbor.

Chapter 10: BGP

This chapter lists the commands for setting up the Border Gateway Protocol on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`clear bgp`	Operational	Resets BGP peer information.
`protocols bgp`	Configuration	Allows you to configure BGP on the router.
`show bgp peers`	Operational	Displays information about BGP peerings.
`show bgp routes`	Operational	Displays BGP route information.

See also the following commands in other chapters.

`policy as-path-list`	Configuration	Allows you to create a list of AS paths, which can be referenced in BGP policy statements. *See page 268.*
`show route`	Operational	Displays information about routes stored in the routing table. *See page 155.*

clear bgp

Resets BGP peer information.

Command Mode

Operational mode.

Syntax

`clear bgp [`*`ipv4`*` | `*`as`*`]`

Parameters

ipv4	Resets peer information for the peer at the specified IP address.
as	Resets all BGP peers within the specified autonomous system.

Usage Guidelines

Use this command on a router running BGP to reset information for BGP peers.

protocols bgp

Allows you to configure BGP on the router.

Command Mode

Configuration mode.

Syntax

`set protocols bgp …`	Use **set** to create the **bgp** configuration node, or to modify BGP configuration. Note that you cannot use **set** to change the identifier of a configuration node. In BGP, this includes peer identifiers, as well as the network address identifying an originated route. To change this information, delete the old node and create a new configuration node with the correct information.
`delete protocols bgp …`	Use **delete** to delete the **bgp** configuration node altogether, or to delete one of its subordinate nodes.

Configuration Statement

```
protocols {
    bgp {
        bgp-id: ipv4
        local-as: 1-65535
        route-reflector {
            cluster-id: ipv4
            disable: [true|false]
        }
        confederation {
            identifier: 1-4294967296
            disable: [true|false]
        }
        damping {
            half-life: 1-4294967296
            max-suppress: 1-4294967296
            reuse: 1-4294967296
            suppress: 1-4294967296
            disable: [true|false]
        }
        peer: text {
            local-ip: ipv4
            as: 1-65535
```

```
                    next-hop: ipv4
                    next-hop6: ipv6
                    holdtime: 0,3–65535
                    delay-open-time: 1–4294967296
                    client: [true|false]
                    confederation-member: [true|false]
                    prefix-limit {
                       maximum: 1–4294967296
                       disable: [true|false]
                    disable: [true|false]
                    ipv4-unicast: [true|false]
                    ipv4-multicast: [true|false]
                    ipv6-unicast: [true|false]
                    ipv6-multicast: [true|false]
                 }
                 traceoptions {
                    flag {
                       verbose {
                          disable: [true|false]
                       }
                       all {
                          disable: [true|false]
                       }
                       message-in {
                          disable: [true|false]
                       }
                       message-out {
                          disable: [true|false]
                       }
                       state-change {
                          disable: [true|false]
                       }
                       policy-configuration {
                          disable: [true|false]
                       }
                    }
                    import: text
                    export: text
                 }
              }
           }
```

Parameters

bgp-id	Mandatory. The BGP identifier for the BGP instance on this router. It is typically set to one of the router's IP addresses, and it is normally required that this be globally unique. It is considered good practice to set the BGP router ID to the address of the loopback interface, as this is the most reliable interface on the router. The required format of the BGP ID is a dotted-decimal IPv4 address, as mandated by the BGP specification.
local-as	Mandatory. The autonomous system number for the domain in which this router resides. Any peers of this router must be configured to know this AS number—if there is a mismatch, a peering will not be established. The range is 1 to 65535.
route-reflector	Optional. Makes this router a route reflector.
cluster-id	Mandatory. A network address uniquely identifying the the route reflection cluster in an internal BGP group.
disable	Optional. Enables or disables route reflection for this router. Supported values are as follows: **true**: Disables route reflection on this router, without discarding configuration. **false**: Enables route reflection on this router. The default is **false**.
confederation	Optional. Makes this router part of a confederation.
identifier	Mandatory. The confederation to which this router will belong.
disable	Optional. Enables or disables this router's membership in the confederation. Supported values are as follows: **true**: Disables this router's membership in the confederation, without discarding configuration. **false**: Enables this router's membership in the confederation. The default is **false**.
damping	Optional. Configures route flap damping.

half-life	Optional. The time in minutes after which the flapping penalty is decreased. After a route has been assigned a penalty, the penalty is decreased by half after the half-life period. Subsequently, the penalty is reduced every 5 seconds. The range is 1 to 45. The default is 15.
max-suppress	Optional. The maximum time in minutes a route can be suppressed. The range is 1 to 20000. The default is 60.
reuse	Optional. The reuse threshold. If the penalty for a flapping route falls below this value, the route is unsuppressed. The range is 1 to 20000. The default is 750.
suppress	Optional. The suppression threshold. A route is suppressed when its penalty exceeds this limit. The range is 1 to 20000. The default is 3000.
disable	Optional. Enables or disables damping on this BGP instance. Supported values are as follows: **true**: Disables damping, without discarding configuration. **false**: Enables damping. The default is **false**.
peer	Optional. Multi-node. Configures a BGP peering association with another router. The format is the IPv4 unicast address of the router being peered with. • For eBGP peerings, the peer identifier will normally be the IP address of the peer router on the interface over which BGP traffic is to be exchanged. • For iBGP peerings, the peer identifier will normally be an IP address bound to that router's loopback interface. You can define multiple peers for this router by creating multiple **peer** configuration nodes.
peer-port	Mandatory. The port on the remote peer over which the TCP connection between the peers will be established. Currently this must be 179.
local-port	Mandatory. The port on the local peer over which the TCP connection between the peers will be established. Currently this must be 179.
local-ip	Mandatory. The IPv4 address that the remote peer should use for BGP connections to this peer.
as	Mandatory. The AS that the remote peer belongs to. This must be the AS number that the peer advertises for itself, or the peering will not be established. The range is 1 to 65535.

next-hop	Mandatory. The IPv4 address that will be sent as the next-hop router address in routes sent to this peer.
next-hop6	Optional. The IPv6 address that will be sent as the next-hop router address in routes sent to this peer.
holdtime	Optional. The holdtime in seconds that the router should use when negotiating the connection with this peer. If no message is received from a BGP peer during the negotiated holdtime, the peering will be shut down. Supported values are 0 (wait forever), or 3 to 65535. The default is 90.
delay-open-time	Optional. How long in seconds this router should wait before sending an OPEN message to this peer. This allows the remote peer time to send the first OPEN message. The range is 0 to 65535, where 0 means send the OPEN message immediately. The default is 0.
client	Optional. Identifies this peer as a client or non-client for of the cluster's route reflector. Supported values are as follows: **true**: The peer is a client of the route reflector. **false**: The peer is a non-client of the route reflector. The default is **false**.
confederation-member	Optional. Identifies the peer as a member or non-member of the confederation. Supported values are as follows: **true**: This router is a confederation member. **false**: This router is a not a confederation member. The default is **false**.
prefix-limit	Optional. Provides the ability to disallow a peer if the number of prefixes received from that peer exceeds a threshold.
maximum	The maximum number of prefixes that will be accepted from the peer before disallowing it. The range is 1 to 4294967294. The default is 250000.
disable	Optional. Enables or disables prefix filtering on this BGP instance. Supported values are as follows: **true**: Disables prefix filtering, without discarding configuration. **false**: Enables prefix filtering. The default is **false**.

disable	Optional. Enables or disables this peer. Supported values are as follows: **true**: Disables this peer, without discarding the configuration. **false**: Enables this peer. The default is **false**.
ipv4-unicast	Optional. Enables or disables BGP negotiation multi-protocol support allowing IPv4 unicast routes to be exchanged. Supported values are as follows: **true**: Allows IPv4 unicast route exchange. **false**: Disallows IPv4 unicast route exchange. The default is **true**.
ipv4-multicast	Enables or disables BGP negotiation multi-protocol support allowing IPv4 multicast routes to be exchanged. Supported values are as follows: **true**: Allows IPv4 multicast route exchange. **false**: Disallows IPv4 multicast route exchange. The default is **false**.
ipv6-unicast	Optional. Enables or disables BGP negotiation multi-protocol support allowing IPv6 unicast routes to be exchanged. Supported values are as follows: **true**: Allows IPv6 unicast route exchange. **false**: Disallows IPv6 unicast route exchange. The default is **true**.
ipv6-multicast	Enables or disables BGP negotiation multi-protocol support allowing IPv4 multicast routes to be exchanged. Supported values are as follows: **true**: Allows IPv6 multicast route exchange. **false**: Disallows IPv6 multicast route exchange. The default is **false**.
traceoptions	Optional. Enables or disables tracing (debugging) information for BGP.
flag	Selectively defines the options for which tracing is to be enabled.
verbose	Optional. Allows you to request extra detail in debug messages.

disable	Optional. Enables or disables verbose tracing. Supported values are as follows: **true**: Disables verbose tracing, without discarding the configuration. **false**: Enables verbose tracing. The default is **false**.
all	Optional. Allows you to apply tracing for all options at once.
disable	Optional. Enables or disables all tracing options at once. Supported values are as follows: **true**: Disables all trace options, without discarding the configuration. **false**: Enables all trace options. The default is **false**.
message-in	Optional. Allows you to apply tracing to inbound messages only.
disable	Optional. Enables or disables tracing on inbound messages only. Supported values are as follows: **true**: Disables tracing on inbound messages, without discarding the configuration. **false**: Enables tracing on inbound messages. The default is **false**.
message-out	Optional. Allows you to apply tracing to outbound messages only.
disable	Optional. Enables or disables tracing on outbound messages only. Supported values are as follows: **true**: Disables tracing on outbound messages, without discarding the configuration. **false**: Enables tracing on outbound messages. The default is **false**.
state-change	Optional. Allows you to apply tracing to forwarding state machine (FSM) state change messages only.
disable	Optional. Enables or disables tracing on FSM state-change messages. Supported values are as follows: **true**: Disables tracing on FSM state-change messages, without discarding the configuration. **false**: Enables tracing on FSM state-change messages. The default is **false**.

policy-configuration	Optional. Allows you to apply tracing to BGP policy configuration only.
disable	Optional. Enables or disables tracing on outbound messages only. Supported values are as follows: **true**: Disables tracing on outbound messages only, without discarding the configuration. **false**: Enables tracing on outbound messages only. The default is **false**.
export	Optional. The name of the export policy you configured for BGP. The import policy will be used to evaluate routing updates received by BGP. For policy terms that match, the defined action will be taken. Multiple policies can be configured using a comma-separated list of policy names.
import	Optional. The name of the import policy you configured for BGP. The import policy will be used to evaluate routing updates sent to peers. For policy terms that match, the defined action will be taken. Multiple policies can be configured using a comma-separated list of policy names.

Usage Guidelines

Use this statement to configure BGP on the router.

Note that it is preferable to use the loopback address as the interface addresses of an iBGP peer. That way, as long as the peer is reachable by any interface, the peer stays up and traffic can be routed correctly. For an eBGP peer, the actual interface address is used, since, if that interface the session should drop.

show bgp peers

Displays information about BGP peerings.

Command Mode

Operational mode.

Syntax

`show bgp peers [detail [`*`peer`*`]]`

Parameters

detail	Displays detailed information for all BGP peers.
peer	Displays detailed information for the specified BGP peer.

Usage Guidelines

Use this command on a router running BGP to display the status of BGP peerings. The information displayed will include information about all BGP peerings that have been configured.

When used without the **detail** option, this command displays a short list that are configured, irrespective of whether the peering is in established state or not. The **detail** parameter provides additional information, either for all peers or for the specified peer.

The output of this command can be piped through another command using the UNIX pipe operator (“|”).

Examples

Example 10-1 shows sample output of **show bgp peers** without the detail option.

Example 10-1 “show bgp peers”: Displaying a list of BGP peers

```
root@vyatta> show bgp peers detail
Peer 1: local 192.150.187.112/179 remote 69.110.224.158/179
Peer 2: local 192.150.187.112/179 remote 192.150.187.2/179
Peer 3: local 192.150.187.112/179 remote 192.150.187.78/179
Peer 4: local 192.150.187.112/179 remote 192.150.187.79/179
Peer 5: local 192.150.187.112/179 remote 192.150.187.109/179
```

show bgp routes

Displays BGP route information.

Command Mode

Operational mode.

Syntax

```
show bgp routes [ipv4 [summary |
                      detail |
                      unicast [summary | detail] |
                      multicast [summary | detail]]]
                [ipv6 [summary |
                      detail |
                      unicast [summary | detail] |
                      multicast [summary | detail]]]
```

Parameters

ipv4	Displays IPv4 BGP route information.
summary	Summarizes the specified IPv4 BGP route information.
detail	Displays detailed IPv4 BGP peers information.
unicast	Displays displays information about IPv4 unicast BGP routes.
summary	Summarizes the specified IPv4 unicast BGP route information.
detail	Displays detailed IPv4 BGP unicast peers information.
multicast	Displays displays information about IPv4 multicast BGP routes.
summary	Summarizes the specified IPv4 multicast BGP route information.
detail	Displays detailed IPv4 BGP multicast peers information.

ipv6	Displays IPv6 BGP route information.
summary	Summarizes the specified IPv6 BGP route information.
detail	Displays detailed IPv6 BGP peers information.
unicast	Displays displays information about IPv6 unicast BGP routes.
summary	Summarizes the specified IPv6 unicast BGP route information.
detail	Displays detailed IPv6 BGP unicast peer information.
multicast	Displays displays information about IPv6 multicast BGP routes.
summary	Summarizes the specified IPv6 multicast BGP route information.
detail	Displays detailed IPv6 BGP multicast peer information.

Usage Guidelines

Use this command on a router running BGP to display locally configured BGP routes and BGP routes this router has received from its peers.

When used with no option, this command displays all BGP routes with an intermediate amount of detail. The **ipv4** option displays IPv4 routes, and the **ipv6** option displays IPv6 routes.

On a router with a full Internet routing table (in excess of 100,000 routes), this command can produce a large amount of output.(“|”), as follows:

show bgp routes | match *prefix*

where *prefix* is the route prefix, as in the following example:

```
show bgp routes | match "10.0.0.0"
```

Chapter 11: IGMP and MLD

This chapter lists the commands for setting up Internet Group Management Protocol and Multicast Listener Discovery protocol on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
protocols igmp	Configuration	Configures IGMP on the router.
protocols mld	Configuration	Configures MLD on the router.
show igmp group	Operational	Displays information about IGMP group membership.
show igmp interface	Operational	Displays information about IGMP interfaces.
show mld group	Operational	Displays information about MLD group membership.
show mld interface	Operational	Displays information about MLD interfaces.

See also the following commands in other chapters.

policy as-path-list	Configuration	Allows you to create a list of AS paths, which can be referenced in BGP policy statements. *See page 268.*
show route	Operational	Displays information about routes stored in the routing table. *See page 155.*

protocols igmp

Configures IGMP on the router.

Syntax

`set protocols igmp …`	Use **set** to create the **igmp** configuration node, or to modify IGMP configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new node with the correct information.
`delete protocols igmp …`	Use **delete** to delete the **igmp** configuration node altogether, or to delete one of its subordinate nodes.

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    igmp {
        disable:[true|false]
        interface: eth0..eth23 {
            disable:[true|false]
            version:1-3
            enable-ip-router-alert-option-check: [true|false]
            query-interval: 1-1024
            query-last-member-interval: 1-1024
            query-response-interval: 1-1024
            robust-count: 2-10
        }
        traceoptions {
            flag {
                all {
                    disable:[true|false]
                }
            }
        }
    }
}
```

Parameters

disable	Enables or disables IGMP on this router. Supported values are as follows: **true**: Disables IGMP, without discarding the configuration. **false**: Enables IGMP. The default is **false**.
interface	Mandatory. Multi-node. The name of an Ethernet interface to be monitored by IGMP for the presence of multicast receivers. The network interface must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging" for information on creating and configuring network interfaces.) You can enable IGMP on more than one interface by creating multiple **interface** configuration nodes within the **igmp** node.
disable	Enables or disables IGMP on this interface. Supported values are as follows: **true**: Disables IGMP, without discarding the configuration. **false**: Enables IGMP. The default is **false**.
version	Specifies which version of IGMP to support. Make sure that the hosts on the network support the same version. Supported values are as follows: **1**: IGMPv1 **2**: IGMPv2 **3**: IGMPv3 The default is **2**.
enable-ip-router-alert-option-check	Specifies whether to check for the IP Router Alert option in IP packets. The Router Alert option is IP option 20, specified in RFC 2113. It can be used to alert transit routers to more closely examine the contents of an IP packet. Supported values are as follows: **true**: The router will check to see if the IP Router Alert option is flagged. **false**: The router will not check to see if the IP Router Alert option is flagged. The default is **false**.

query-interval	Directs the router to send IGMP host-query messages at the specified interval. The range is 1 to 1024, in seconds. The default is 125.
query-last-member-interval	The maximum response time, in seconds, to wait for a response to a group-specific query sent in answer to leave-group messages. It is also the interval between group-specific query messages. When the router receives an IGMPv2 leave message or an IGMPv3 state change report, it sends out a query and expects a response within the time specified by this value. Using a lower value enables members to leave groups more quickly.
query-response-interval	The maximum response time, in seconds, to wait for a host to respond to a group membership query. If the responder does not answer within this interval, the router deletes the group. This value can only be configured for IGMPv2 and IGMPv3. It does not apply to IGMPv1. Using a lower value enables members to join and leave groups more quickly.
robust-count	The number of times that the router should resend each IGMP message from this interface. IGMP sends messages over UDP, which is inherently unreliable. To increase reliability, the message can be resent. The higher the robustness count, the higher the reliability for the messages. The range is 2 to 10. The default is 2.
traceoptions	Sets the tracing and debugging options for IGMP.
flag	Specifies which tracing options are enabled.
all	All tracing options.
disable	Enables or disables the specified tracing options. Supported values are as follows: **true**: Disables tracing. **false**: Enables tracing. The default is **false**.

Usage Guidelines

Use this command to configure IGMP on the router for IPv4 interfaces. To configure this routing type on IPv6 interfaces, use the the **protocols mld** command (see page 219).

In the configuration, each interface that is intended to have multicast listeners must be configured separately. The **traceoptions** section is used to explicitly enable log information that can be used for debugging purposes.

protocols mld

Configures MLD on the router.

Syntax

`set protocols mld …`	Use **set** to create the **mld** configuration node, or to modify MLD configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new node with the correct information.
`delete protocols mld …`	Use **delete** to delete the **mld** configuration node altogether, or to delete one of its subordinate nodes.

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
    mld {
        disable:[true|false]
        interface: eth0..eth23 {
            vif text {
                disable:[true|false]
                version:1-2
                enable-ip-router-alert-option-check: [true|false]
                query-interval: 1-1024
                query-last-member-interval: 1-1024
                query-response-interval: 1-1024
                robust-count: 2-10
            }
        }
        traceoptions {
            flag {
                all {
                    disable:[true|false]
                }
            }
        }
    }
}
```

Parameters

disable	Enables or disables MLD on this router. Supported values are as follows: **true**: Disables MLD, without discarding the configuration. **false**: Enables MLD. The default is **false**.
interface	Mandatory. Multi-node. The name of an Ethernet interface to be monitored by MLD for the presence of multicast receivers. The network interface must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging" for information on creating and configuring network interfaces.) You can enable MLD on more than one interface by creating multiple **interface** configuration nodes within the **igmp** node.
vif	Mandatory. Multi-node. The name of a virtual interface to be monitored by MLD for the presence of multicast receivers. The vif must already be created and configured. (See Chapter 3: Ethernet Interfaces, VLANs, and Bridging" for information on creating and configuring network interfaces.) You can enable MLD on more than one vif by creating multiple **vif** configuration nodes within the **igmp interface** node.
disable	Enables or disables MLD on this interface. Supported values are as follows: **true**: Disables MLD, without discarding the configuration. **false**: Enables MLD. The default is **false**.
version	Specifies which version of MLD to support. Make sure that the hosts on the network support the same version. Supported values are as follows: **1**: MLDv1 **2**: MLDv2 The default is **1**.

enable-ip-router-alert-option-check	Specifies whether to check for the IP Router Alert option in IP packets. The Router Alert option is IP option 20, specified in RFC 2113. It can be used to alert transit routers to more closely examine the contents of an IP packet. Supported values are as follows: **true**: The router will check to see if the IP Router Alert option is flagged. **false**: The router will not check to see if the IP Router Alert option is flagged. The default is **false**.
query-interval	Directs the router to send MLD host-query messages at the specified interval. The range is 1 to 1024, in seconds. The default is 125.
query-last-member-interval	The maximum response time, in seconds, to wait for a response to a group-specific query sent in answer to leave-group messages. It is also the interval between group-specific query messages. When the router receives a leave message or an state change report, it sends out a query and expects a response within the time specified by this value. Using a lower value enables members to leave groups more quickly.
query-response -interval	The maximum response time, in seconds, to wait for a host to respond to a group membership query. If the responder does not answer within this interval, the router deletes the group. This value can only be configured for IGMPv2 and IGMPv3. It does not apply to IGMPv1. Using a lower value enables members to join and leave groups more quickly.
robust-count	The number of times that the router should resend each IGMP message from this interface. IGMP sends messages over UDP, which is inherently unreliable. To increase reliability, the message can be resent. The higher the robustness count, the higher the reliability for the messages. The range is 2 to 10. The default is 2.
traceoptions	Sets the tracing and debugging options for MLD.
flag	Specifies which tracing options are enabled.
all	All tracing options.

disable	Enables or disables the specified tracing options. Supported values are as follows: **true**: Disables tracing. **false**: Enables tracing. The default is **false**.

Usage Guidelines

Use this command to configure MLD on the router for IPv6 vifs. To configure this routing type on IPv4 interfaces and vifs, use the the **protocols igmp** command (see page 215).

In the configuration, each vif that is intended to have multicast listeners must be configured separately. The **traceoptions** section is used to explicitly enable log information that can be used for debugging purposes.

show igmp group

Displays information about IGMP group membership.

Command Mode

Operational mode.

Syntax

```
show igmp group
```

Parameters

None.

Usage Guidelines

Use this command to view information about IGMP group membership.

The information displayed includes the following:

- Source. This is the multicast source address in the case of source-specific IGMP Join entries. Alternatively, this is set to **0.0.0.0** in case of any-source IGMP join entries.
- LastReported. This contains the address of the most recent receiver that responded to an IGMP Join message.
- Timeout. This field shows the number of seconds until the next time the router will query for host members (that is, before the router will send an IGMP Query message for this particular entry).
- Version. The version of IGMP being used.
- State. The state of the interface.

show igmp interface

Displays information about IGMP interfaces.

Command Mode

Operational mode.

Syntax

```
show igmp interface [address]
```

Parameters

address	Displays IP address information for IGMP interfaces.

Usage Guidelines

Use this command to view information about IGMP interfaces.

- When used with no option, this command displays the state of the interface, the querier for the interface, the timeout value, the IGMP version being used, and the number of groups listening.
- When used with the **address** option, the command displays the primary and secondary (if any) addresses enabled for IGMP.

show mld group

Displays information about MLD group membership.

Command Mode

Operational mode.

Syntax

```
show mld group
```

Parameters

None.

Usage Guidelines

Use this command to view information about MLD group membership.

The information displayed includes the following:

- Source. This is the multicast source address in the case of source-specific MLD Join entries. Alternatively, this is set to **0.0.0.0** in case of any-source MLD Join entries.
- LastReported. This contains the address of the most recent receiver that responded to an MLD Join message.
- Timeout. This field shows the number of seconds until the next time the router will query for host members (that is, before the router will send an MLD Query message for this particular entry).
- Version. The version of MLD being used.
- State. The state of the interface.

show mld interface

Displays information about MLD interfaces.

Command Mode

Operational mode.

Syntax

```
show mld interface [address]
```

Parameters

address	Displays IP address information for MLD interfaces.

Usage Guidelines

Use this command to view information about MLD interfaces.

- When used with no option, this command displays the state of the interface, the querier for the interface, the timeout value, the MLD version being used, and the number of groups listening.
- When used with the **address** option, the command displays the primary and secondary (if any) addresses enabled for MLD.

Chapter 12: PIM Sparse-Mode

This chapter lists the commands for setting up Protocol Independent Multicast on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`protocols pimsm4`	Configuration	Allows you to configure PIM-SM for IPv4 on the router.
`protocols pimsm6`	Configuration	Allows you to configure PIM-SM for IPv6 on the router.
`show pim bootstrap`	Operational	Displays information about the IPv4 bootstrap zones that are currently in use.
`show pim bootstrap rps`	Operational	Displays information about IPv4 Candidate RP information received by the bootstrap mechanism.
`show pim interface address`	Operational	Displays information about IPv4 PIM-SM network interfaces.
`show pim interface address`	Operational	Displays address information about IPv4 PIM-SM network interfaces.
`show pim join`	Operational	Displays information about IPv4 PIM-SM multicast routing state.
`show pim mfc`	Operational	Displays information about IPv4 PIM multicast forwarding entries installed in the MFEA.
`show pim mrib`	Operational	Displays information about the MRIB used by IPv4 PIM.
`show pim neighbors`	Operational	Displays information about this router’s IPv4 PIM neighbor routers.
`show pim rps`	Operational	Displays information about the Candidate RP set for IPv4 PIM-SM.
`show pim scope`	Operational	Displays information about the IPv4 PIM scope zones for this router.
`show pim6 bootstrap`	Operational	Displays information about the IPv6 bootstrap zones that are currently in use.
`show pim6 bootstrap rps`	Operational	Displays information about IPv6 Candidate RP information received by the bootstrap mechanism.
`show pim6 interface`	Operational	Displays information about IPv6 PIM-SM network interfaces.

Command	Mode	Description
`show pim6 interface address`	Operational	Displays address information about IPv6 PIM-SM network interfaces.
`show pim6 join`	Operational	Displays information about IPv6 PIM-SM multicast routing state.
`show pim6 mfc`	Operational	Displays information about IPv6 PIM multicast forwarding entries installed in the MFEA.
`show pim6 mrib`	Operational	Displays information about the MRIB used by IPv6 PIM.
`show pim6 neighbors`	Operational	Displays information about this router's IPv6 PIM neighbor routers.
`show pim6 rps`	Operational	Displays information about the Candidate RP set for IPv6 PIM-SM.
`show pim6 scope`	Operational	Displays information about the IPv64 PIM scope zones for this router.

See also the following commands in other chapters.

`show route`	Operational	Displays information about routes stored in the routing table. *See page 155.*

protocols pimsm4

Allows you to configure PIM-SM for IPv4 on the router.

Syntax

`set protocols pimsm4 …`	Use **set** to create the **pimsm4** configuration node, or to modify PIM-SM configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new node with the correct information.
`delete protocols pimsm4 …`	Use **delete** to delete the **pimsm4** configuration node altogether, or to delete one of its subordinate nodes.

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
   pimsm4 {
      disable:[true|false]
      interface eth0..eth23 {
         disable: [true|false]
         enable-ip-router-alert-option-check: [true|false]
         dr-priority: 1-255
         hello-period: 1-18724
         hello-triggered-delay: 1-255
         alternative-subnet ipv4net {}
      }
      static-rps {
         rp ipv4 {
            group-prefix ipv4net {
               rp-priority: 0-255
               hash-mask-len: 4-32
            }
         }
      }
      bootstrap {
         disable: [true|false]
         cand-bsr {
            scope-zone ipv4net{
```

```
                    is-scope-zone: [true|false]
                    cand-bsr-by-vif-name: 1-4096
                    cand-bsr-by-vif-addr: ipv4
                    bsr-priority: 0-255
                    hash-mask-len: 4-32
                }
            }
            cand-rp {
                group-prefix: ipv4net {
                    is-scope-zone: [true|false]
                    cand-bsr-by-vif-name: 1-4096
                    cand-bsr-by-vif-addr: ipv4
                    rp-priority: 0-255
                    rp-holdtime: 0-65535
                }
            }
        }
        switch-to-spt-threshold {
            disable: [true|false]
            interval: 3-2147483647
            bytes: 0–4294967296
        }
        traceoptions {
            flag {
                all {
                    disable: [true|false]
                }
            }
        }
    }
}
```

Parameters

disable	Optional. Enables or disables PIM-SM for IPv4 on the router. Supported values are: **true**: Disables PIM-SM for IPv4 on the router, without discarding the configuration. **false**: Enables PIM-SM for IPv4 on the router. The default is **false**.

interface	Mandatory. Multi-node. The name of the Ethernet interface on which you are enabling PIM-SM for IPv4. The network interface must already be created and configured with an IPv4 address. (See “Chapter 3: Ethernet Interfaces, VLANs, and Bridging” for information on creating and configuring network interfaces.) You can enable PIM-SM on more than one interface by creating multiple **interface** configuration nodes within the **pimsm4** node.
disable	Optional. Enables or disables PIM-SM for IPv4 on this interface. Supported values are: **true**: Disables PIM-SM for IPv4 on this interface, without discarding the configuration. **false**: Enables PIM-SM or IPv4 on this interface. The default is **false**.
enable-ip-router-alert-option-check	Optional. Specifies whether to check for the IP Router Alert option in IP packets. The Router Alert option is IP option 20, as specified in RFC 2113. It can be used to alert transit routers to more closely examine the contents of an IP packet. Supported values are as follows: **true**: The router will check to see if the IP Router Alert option is flagged. **false**: The router will not check to see if the IP Router Alert option is flagged. The default is **false**.
dr-priority	Optional. This router’s Designated Router (DR) priority for this interface. The PIM router on this subnet with the highest value of DR priority will become the DR for the subnet. The range is 0 to 255. The default is 1.
hello-period	Optional. The interval, in seconds, at which the router sends hello messages to neighbors. Hello messages are automatically sent on bootup. After that, hello messages will be sent at this interval. The range is 1 to 18724. The default is 30.
hello-triggered-delay	Optional. Sets the randomized triggered delay, in seconds, for hello messages. When the router learns a new generation ID (PIM-SM GenID) for a neighbor, the router unicasts a hello message to the neighbor after this delay. This triggers the neighbor to establish neighborship with all routers as soon as possible. The range is 1 to 255. The default is 5.

alternative-subnet	Optional. Multi-node.Used to associate additional IP subnets with a network interface. The format is an IPv4 network in *address/prefix* format. One use of this directive is to make incoming traffic with a non-local source address appear as if it is coming from a local subnet. Typically, this is needed as a work-around solution when unidirectional interfaces such as satellite links are used for receiving traffic. You can define more than one alternative subnet by creating multiple **alternative-subnet** configuration nodes. This directive should be used with extreme care, because it is possible to create forwarding loops.
static-rps	Manually configures PIM rendezvous point (RP) router information. A PIM-SM router must either have some RPs configured as static RPs, or it must run the PIM-SM bootstrap mechanism (see the **bootstrap** directive). One or more RPs can be configured. It is important that all routers in a PIM domain make the same choice of RP for the same multicast group, so generally they should be configured with the same RP information.
rp	Multi-node. The IPv4 address of a router that will be a static RP. At least one RP must be specified. You can define more than one static RP by creating multiple **rp** configuration nodes.
group-prefix	Multi-node. The range of multicast addresses for which the specified router is willing to be the RP. The format is an IPv4 network in *address/prefix* format. You can define more than one set of multicast addresses for a static RP by creating multiple **group-prefix** configuration nodes.
rp-priority	Optional. The priority of the RP for this multicast group. If multiple RP routers are known for a particular multicast group, then the one with the most specific group prefix will be used. If more than one router has an equally specific group prefix, then the one with the highest RP priority is used. See also **hash-mask-len**. The range is 0 to 255. The default is 192.

hash-mask-len	Optional. The number of bits in the group IP address to which the hash function will be applied. If multiple routers all have the most specific group prefixes and the highest RP priority, then to balance load a hash function is used to choose the RP. At the same time, it is usually desirable for closely associated multicast groups to use the same RP. Thus the hash function is only applied to the first *n* bits of the group IP address, ensuring that if two groups have the same first *n* bits, they will hash to the same RP address. The **hash-mask-len** parameter specifies the value of *n*. The range is 4 to 32. The default is 30. Typically this value should not be changed. If it is modified, then all PIM-SM routers must be configured with the same value.
bootstrap	Configures the automatic bootstrapping of PIM RP router information using the PIM bootstrap router mechanism. A PIM-SM router must either run the PIM-SM bootstrap mechanism, or have at least one RP configured as a static RP (see the **static-rps** directive).
disable	Optional. Indicates whether or not the router will run the PIM-SM automatic bootstrap mechanism. Supported values are as follows: **true**: The router will not run the PIM-SM automatic bootstrap mechanism, but the configuration will be preserved. **false**: The router will run the PIM-SM automatic bootstrap mechanism. The default is **false**.
cand-bsr	Optional. Designates this router as a candidate to be the BootStrap Router (BSR) for this PIM-SM domain. The router will become the BSR only if it wins the BSR election process. At least one scope zone must be specified for a candidate BSR router.
scope-zone	Multi-node. Defines one multicast group prefix for which this router is willing to be BSR. The format is an IPv4 network in *address/prefix* format. At least one scope zone is mandatory for a candidate BSR router. You can define more than one scope zone by creating multiple **scope-zone** configuration nodes.

is-scope-zone	Optional. Indicates whether this multicast group prefix defines a multicast scope zone. Supported values are as follows: **true**—This multicast group prefix defines a multicast scope zone. **false—**This multicast group prefix merely represents a range of multicast groups for which this router is willing to be BSR. The default is false.
cand-bsr-by-vif-name	Mandatory. The name of the vif whose IP address will be used in the PIM bootstrap messages.
cand-bsr-by-vif-addr	Optional. The address to be used in the PIM bootstrap messages.
bsr-priority	Optional. The BSR priority for this router. This value will be used in the PIM-SM BSR election process. For each scope-zone, the candidate bootstrap router with the highest BSR priority will be chosen to be BSR. The range is 0 to 255. The default is 1.
hash-mask-len	Optional. The number of bits in the group IP address to which the hash function will be applied. The BSR mechanism announces a list of Candidate RPs (C-RPs) for each scope zone to the other routers in the scope zone. To balance load, those routers then use a hash function to choose the RP for each multicast group from amongst the C-RPs. However, it is usually desirable for closely associated multicast groups to use the same RP. Thus the hash function is only applied to the first *n* bits of the group IP address, ensuring that if two groups have the same first *n* bits, they will hash to the same RP address. The **hash-mask-len** parameter specifies the value of *n*. The range is 4 to 32. The default is 30. Typically this value should not be changed. If it is modified, then all PIM-SM routers must be configured with the same value.
cand-rp	Optional. Designates this router as a candidate to be an RP for this PIM-SM domain.It will become an RP only if the BSR elects it to be. At least one group prefix must be specified for this router to function as an RP.

group-prefix	The range of multicast addresses for which the specified router is willing to be the RP. The format is an IPv4 network in *address*/*prefix* format. At least one group prefix must be specified for this router to function as an RP. You can define more than one set of multicast addresses by creating multiple **group-prefix** configuration nodes.
is-scope-zone	Optional. Indicates whether this multicast group prefix defines a multicast scope zone. Supported values are as follows: **true**: This multicast group prefix defines a multicast scope zone. **false**: This multicast group prefix merely represents a range of multicast groups for which this router is willing to be RP. The default is **false**.
cand-rp-by-vif-name	Mandatory. The name of the vif whose IP address will be used as the RP address if this router becomes an RP.
cand-bsr-by-vif-addr	Optional. The address to be used as the RP address if this router becomes an RP.
rp-priority	Optional. The priority of the specified RP router for this group prefix. If multiple RP routers are known for a particular multicast group, then the one with the most specific group prefix will be used. If more than one router has an equally specific group prefix, then the one with the highest RP priority is used. See also **hash-mask-len**. The range is 0 to 255. The default is 192.
rp-holdtime	Optional. The holdtime, in seconds, that this router will advertise when talking to the BSR. If the BSR has not heard a Candidate RP Advertisement from this router for *rp-holdtime* seconds, then the BSR will conclude it is dead, and will remove it from the set of possible RPs. The range is 0 to 65535. The default is 150.
switch-to-spt-threshold	Optional. Allows you to specify a bitrate threshold at a last-hop router or RP for switching from the RP tree to the shortest-path tree.

disable	Optional. Enables or disables bitrate-based switching to the shortest-path tree. Supported values are as follows: **true**: Disables bitrate-based switching to the shortest-path tree, without discarding configuration. **false**: Enables bitrate-based switching to the shortest-path tree. The default is **false**.
interval	Optional. The measurement interval, in seconds, for measuring the bitrate of traffic from a multicast sender. The measurement interval should normally not be set too small: values greater than ten seconds are recommended. The range 3 is 2147483647. The default is 100.
bytes	Optional. The maximum number of bytes from a multicast sender that can be received in *interval* seconds. If this threshold is exceeded, the router will attempt to switch to the shortest-path tree from that multicast sender. If you want shortest-path switch to happen immediately after the first packet is forwarded, set this value to 0. The range is 0 to 4294967296. The default is 0.
traceoptions	Optional. Sets the tracing and debugging options for PIM-SM for IPv4.
flag	Optional. Specifies which tracing options are enabled.
all	Optional. All tracing options.
disable	Optional. Enables or disables the specified tracing options. Supported values are as follows: **true**: Disables tracing. **false**: Enables tracing. The default is **false**.

Usage Guidelines

Use this command to configure PIM Sparse-Mode multicast routing for IPv4 interface/vifs.

protocols pimsm6

Allows you to configure PIM-SM for IPv6 on the router.

Syntax

`set protocols pimsm6 …`	Use **set** to create the **pimsm6** configuration node, or to modify PIM-SM configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new node with the correct information.
`delete protocols pimsm6 …`	Use **delete** to delete the **pimsm6** configuration node altogether, or to delete one of its subordinate nodes.

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
   pimsm6 {
      disable:[true|false]
      interface eth0..eth23 {
         disable: [true|false]
         enable-ip-router-alert-option-check: [true|false]
         dr-priority: 1-255
         hello-period: 1-18724
         hello-triggered-delay: 1-255
         alternative-subnet ipv6net {}
      }
      static-rps {
         rp ipv4 {
            group-prefix ipv6net {
               rp-priority: 0-255
               hash-mask-len: 8-128
            }
         }
      }
      bootstrap {
         disable: [true|false]
         cand-bsr {
            scope-zone ipv6net{
```

```
                    is-scope-zone: [true|false]
                    cand-bsr-by-vif-name: text
                    cand-bsr-by-vif-addr: ipv6
                    bsr-priority: 0-255
                    hash-mask-len: 8-128
                 }
              }
              cand-rp {
                 group-prefix: ipv6net {
                    is-scope-zone: [true|false]
                    cand-bsr-by-vif-name: text
                    cand-bsr-by-vif-addr: ipv6
                    rp-priority: 0-255
                    rp-holdtime: 0-65535
                 }
              }
           }
           switch-to-spt-threshold {
              disable: [true|false]
              interval: 3-2147483647
              bytes: 0–4294967296
           }
           traceoptions {
              flag {
                 all {
                    disable: [true|false]
                 }
              }
           }
        }
     }
```

Parameters

disable	Optional. Enables or disables PIM-SM for IPv6 on the router. Supported values are: **true**: Disables PIM-SM for IPv6 on the router, without discarding the configuration. **false**: Enables PIM-SM for IPv6 on the router. The default is **false**.

interface	Mandatory. Multi-node. The name of the Ethernet interface on which you are enabling PIM-SM for IPv6. The network interface must already be created and configured with an IPv6 address. (See “Chapter 3: Ethernet Interfaces, VLANs, and Bridging” for information on creating and configuring network interfaces.) You can enable PIM-SM on more than one interface by creating multiple **interface** configuration nodes within the **pimsm6** node.
disable	Optional. Enables or disables PIM-SM for IPv6 on this interface. Supported values are: **true**: Disables PIM-SM for IPv6 on this interface, without discarding the configuration. **false**: Enables PIM-SM or IPv6 on this interface. The default is **false**.
enable-ip-router-alert-option-check	Optional. Specifies whether to check for the IP Router Alert option in IP packets. The Router Alert option is IP option 20, as specified in RFC 2113. It can be used to alert transit routers to more closely examine the contents of an IP packet. Supported values are as follows: **true**: The router will check to see if the IP Router Alert option is flagged. **false**: The router will not check to see if the IP Router Alert option is flagged. The default is **false**.
dr-priority	Optional. This router’s Designated Router (DR) priority for this interface. The PIM router on this subnet with the highest value of DR priority will become the DR for the subnet. The range is 0 to 255. The default is 1.
hello-period	Optional. The interval, in seconds, at which the router sends hello messages to neighbors. Hello messages are automatically sent on bootup. After that, hello messages will be sent at this interval. The range is 1 to 18724. The default is 30.
hello-triggered-delay	Optional. Sets the randomized triggered delay, in seconds, for hello messages. When the router learns a new generation ID (PIM-SM GenID) for a neighbor, the router unicasts a hello message to the neighbor after this delay. This triggers the neighbor to establish neighborship with all routers as soon as possible. The range is 1 to 255. The default is 5.

alternative-subnet	Optional. Multi-node.Used to associate additional IP subnets with a network interface. The format is an IPv6 network in *address/prefix* format. One use of this directive is to make incoming traffic with a non-local source address appear as if it is coming from a local subnet. Typically, this is needed as a work-around solution when unidirectional interfaces such as satellite links are used for receiving traffic. You can define more than one alternative subnet by creating multiple **alternative-subnet** configuration nodes. This directive should be used with extreme care, because it is possible to create forwarding loops.
static-rps	Manually configures PIM rendezvous point (RP) router information. A PIM-SM router must either have some RPs configured as static RPs, or it must run the PIM-SM bootstrap mechanism (see the **bootstrap** directive). One or more RPs can be configured. It is important that all routers in a PIM domain make the same choice of RP for the same multicast group, so generally they should be configured with the same RP information.
rp	Multi-node. The IPv6 address of a router that will be a static RP. At least one RP must be specified. You can define more than one static RP by creating multiple **rp** configuration nodes.
group-prefix	Multi-node. The range of multicast addresses for which the specified router is willing to be the RP. The format is an IPv6 network in *address/prefix* format. You can define more than one set of multicast addresses for a static RP by creating multiple **group-prefix** configuration nodes.
rp-priority	Optional. The priority of the RP for this multicast group. If multiple RP routers are known for a particular multicast group, then the one with the most specific group prefix will be used. If more than one router has an equally specific group prefix, then the one with the highest RP priority is used. See also **hash-mask-len**. The range is 0 to 255. The default is 192.

hash-mask-len	Optional. The number of bits in the group IP address to which the hash function will be applied. If multiple routers all have the most specific group prefixes and the highest RP priority, then to balance load a hash function is used to choose the RP. At the same time, it is usually desirable for closely associated multicast groups to use the same RP. Thus the hash function is only applied to the first *n* bits of the group IP address, ensuring that if two groups have the same first *n* bits, they will hash to the same RP address. The **hash-mask-len** parameter specifies the value of *n*. The range is 4 to 32. The default is 30. Typically this value should not be changed. If it is modified, then all PIM-SM routers must be configured with the same value.
bootstrap	Configures the automatic bootstrapping of PIM RP router information using the PIM bootstrap router mechanism. A PIM-SM router must either run the PIM-SM bootstrap mechanism, or have at least one RP configured as a static RP (see the **static-rps** directive).
disable	Optional. Indicates whether or not the router will run the PIM-SM automatic bootstrap mechanism. Supported values are as follows: **true**: The router will not run the PIM-SM automatic bootstrap mechanism, but the configuration will be preserved. **false**: The router will run the PIM-SM automatic bootstrap mechanism. The default is **false**.
cand-bsr	Optional. Designates this router as a candidate to be the BootStrap Router (BSR) for this PIM-SM domain. The router will become the BSR only if it wins the BSR election process. At least one scope zone must be specified for a candidate BSR router.
scope-zone	Multi-node. Defines one multicast group prefix for which this router is willing to be BSR. The format is an IPv6 network in *address/prefix* format. At least one scope zone is mandatory for a candidate BSR router. You can define more than one scope zone by creating multiple **scope-zone** configuration nodes.

is-scope-zone	Optional. Indicates whether this multicast group prefix defines a multicast scope zone. Supported values are as follows: **true**: This multicast group prefix defines a multicast scope zone. **false**: This multicast group prefix merely represents a range of multicast groups for which this router is willing to be BSR. The default is **false**.
cand-bsr-by-vif-name	Mandatory. The name of the vif whose IP address will be used in the PIM bootstrap messages.
cand-bsr-by-vif-addr	Optional. The address to be used in the PIM bootstrap messages.
bsr-priority	Optional. The BSR priority for this router. This value will be used in the PIM-SM BSR election process. For each scope-zone, the candidate bootstrap router with the highest BSR priority will be chosen to be BSR. The range is 0 to 255. The default is 1.
hash-mask-len	Optional. The number of bits in the group IP address to which the hash function will be applied. The BSR mechanism announces a list of Candidate RPs (C-RPs) for each scope zone to the other routers in the scope zone. To balance load, those routers then use a hash function to choose the RP for each multicast group from amongst the C-RPs. However, it is usually desirable for closely associated multicast groups to use the same RP. Thus the hash function is only applied to the first *n* bits of the group IP address, ensuring that if two groups have the same first *n* bits, they will hash to the same RP address. The **hash-mask-len** parameter specifies the value of *n*. The range is 4 to 32. The default is 30. Typically this value should not be changed. If it is modified, then all PIM-SM routers must be configured with the same value.
cand-rp	Optional. Designates this router as a candidate to be an RP for this PIM-SM domain.It will become an RP only if the BSR elects it to be. At least one group prefix must be specified for this router to function as an RP.

group-prefix	The range of multicast addresses for which the specified router is willing to be the RP. The format is an IPv6 network in *address/prefix* format. At least one group prefix must be specified for this router to function as an RP. You can define more than one set of multicast addresses by creating multiple **group-prefix** configuration nodes.
is-scope-zone	Optional. Indicates whether this multicast group prefix defines a multicast scope zone. Supported values are as follows: **true**: This multicast group prefix defines a multicast scope zone. **false**: This multicast group prefix merely represents a range of multicast groups for which this router is willing to be RP. The default is **false**.
cand-rp-by-vif-name	Mandatory. The name of the vif whose IP address will be used as the RP address if this router becomes an RP.
cand-bsr-by-vif-addr	Optional. The address to be used as the RP address if this router becomes an RP.
rp-priority	Optional. The priority of the specified RP router for this group prefix. If multiple RP routers are known for a particular multicast group, then the one with the most specific group prefix will be used. If more than one router has an equally specific group prefix, then the one with the highest RP priority is used. See also **hash-mask-len**. The range is 0 to 255. The default is 192.
rp-holdtime	Optional. The holdtime, in seconds, that this router will advertise when talking to the BSR. If the BSR has not heard a Candidate RP Advertisement from this router for *rp-holdtime* seconds, then the BSR will conclude it is dead, and will remove it from the set of possible RPs. The range is 0 to 65535. The default is 150.
switch-to-spt-threshold	Optional. Allows you to specify a bitrate threshold at a last-hop router or RP for switching from the RP tree to the shortest-path tree.

disable	Optional. Enables or disables bitrate-based switching to the shortest-path tree. Supported values are as follows: **true**: Disables bitrate-based switching to the shortest-path tree, without discarding configuration. **false**: Enables bitrate-based switching to the shortest-path tree. The default is **false**.
interval	Optional. The measurement interval, in seconds, for measuring the bitrate of traffic from a multicast sender. The measurement interval should normally not be set too small: values greater than ten seconds are recommended. The range 3 is 2147483647. The default is 100.
bytes	Optional. The maximum number of bytes from a multicast sender that can be received in *interval* seconds. If this threshold is exceeded, the router will attempt to switch to the shortest-path tree from that multicast sender. If you want shortest-path switch to happen immediately after the first packet is forwarded, set this value to 0. The range is 0 to 4294967296. The default is 0.
traceoptions	Optional. Sets the tracing and debugging options for PIM-SM for IPv6.
flag	Optional. Specifies which tracing options are enabled.
all	Optional. All tracing options.
disable	Optional. Enables or disables the specified tracing options. Supported values are as follows: **true**: Disables tracing. **false**: Enables tracing. The default is **false**.

Usage Guidelines

Use this command to configure PIM Sparse-Mode multicast routing for IPv6 interface/vifs.

show pim bootstrap

Displays information about the IPv4 bootstrap zones that are currently in use.

Command Mode

Operational mode.

Syntax

```
show pim boostrap
```

Parameters

None.

Usage Guidelines

Use this command to display information about IPv4 PIM bootstrap routers.

show pim bootstrap rps

Displays information about IPv4 Candidate RP information received by the bootstrap mechanism.

Command Mode

Operational mode.

Syntax

```
show pim boostrap rps
```

Parameters

None.

Usage Guidelines

Use this command to display IPv4 Candidate RP information received by the bootstrap.

show pim interface

Displays information about IPv4 PIM-SM network interfaces.

Command Mode

Operational mode.

Syntax

```
show pim interface
```

Parameters

None.

Usage Guidelines

Use this command to display information about the network interfaces that have been configured for IPv4 PIM-SM.

show pim interface address

Displays address information about IPv4 PIM-SM network interfaces.

Command Mode

Operational mode.

Syntax

```
show pim interface address
```

Parameters

None.

Usage Guidelines

Use this command to display address information for network interfaces that have been configured for IPv4 PIM-SM.

show pim join

Displays information about IPv4 PIM-SM multicast routing state.

Command Mode

Operational mode.

Syntax

```
show pim join
```

Parameters

None.

Usage Guidelines

Use this command to display multicast state information for IPv4 PIM-SM interfaces.

show pim mfc

Displays information about IPv4 PIM multicast forwarding entries installed in the MFEA.

Command Mode

Operational mode.

Syntax

```
show pim mfc
```

Parameters

None.

Usage Guidelines

Use this command to display information about IPv4 PIM multicast forwarding entries that are installed in the multicast forwarding engine.

show pim mrib

Displays information about the MRIB used by IPv4 PIM.

Command Mode

Operational mode.

Syntax

```
show pim mrib
```

Parameters

None.

Usage Guidelines

Use this command to display information about the Multicast Routing Information Base (MRIB) used by IPv4 PIM.

show pim neighbors

Displays information about this router’s IPv4 PIM neighbor routers.

Command Mode

Operational mode.

Syntax

```
show pim neighbors
```

Parameters

None.

Usage Guidelines

Use this command to see the IPv4 PIM neighbors for this router.

show pim rps

Displays information about the Candidate RP set for IPv4 PIM-SM.

Command Mode

Operational mode.

Syntax

```
show pim rps
```

Parameters

None.

Usage Guidelines

Use this command to display Candidate RP set information for IPv4 PIM-SM.

show pim scope

Displays information about the IPv4 PIM scope zones for this router.

Command Mode

Operational mode.

Syntax

```
show pim neighbors
```

Parameters

None.

Usage Description

Use this command to see information about this router’s scoped zones for IPv4 PIM-SM.

show pim6 bootstrap

Displays information about the IPv6 bootstrap zones that are currently in use.

Command Mode

Operational mode.

Syntax

```
show pim6 boostrap
```

Parameters

None.

Usage Guidelines

Use this command to display information about IPv6 PIM bootstrap routers.

show pim6 bootstrap rps

Displays information about IPv6 Candidate RP information received by the bootstrap mechanism.

Command Mode

Operational mode.

Syntax

```
show pim6 boostrap rps
```

Parameters

None.

Usage Guidelines

Use this command to display IPv6 Candidate RP information received by the bootstrap.

show pim6 interface

Displays information about IPv6 PIM-SM network interfaces.

Command Mode

Operational mode.

Syntax

```
show pim6 interface
```

Parameters

None.

Usage Guidelines

Use this command to display information about the network interfaces that have been configured for IPv6 PIM-SM.

show pim6 interface address

Displays address information about IPv6 PIM-SM network interfaces.

Command Mode

Operational mode.

Syntax

```
show pim6 interface address
```

Parameters

None.

Usage Guidelines

Use this command to display address information for network interfaces that have been configured for IPv6 PIM-SM.

show pim6 join

Displays information about IPv6 PIM-SM multicast routing state.

Command Mode

Operational mode.

Syntax

```
show pim6 join
```

Parameters

None.

Usage Guidelines

Use this command to display multicast state information for IPv6 PIM-SM interfaces.

show pim6 mfc

Displays information about IPv6 PIM multicast forwarding entries installed in the MFEA.

Command Mode

Operational mode.

Syntax

```
show pim6 mfc
```

Parameters

None.

Usage Guidelines

Use this command to display information about IPv6 PIM multicast forwarding entries that are installed in the multicast forwarding engine.

show pim6 mrib

Displays information about the MRIB used by IPv6 PIM.

Command Mode

Operational mode.

Syntax

```
show pim6 mrib
```

Parameters

None.

Usage Guidelines

Use this command to display information about the Multicast Routing Information Base (MRIB) used by IPv6 PIM.

show pim6 neighbors

Displays information about this router's IPv6 PIM neighbor routers.

Command Mode

Operational mode.

Syntax

```
show pim6 neighbors
```

Parameters

None.

Usage Guidelines

Use this command to see the IPv6 PIM neighbors for this router.

show pim6 rps

Displays information about the Candidate RP set for IPv6 PIM-SM.

Command Mode

Operational mode.

Syntax

```
show pim6 rps
```

Parameters

None.

Usage Guidelines

Use this command to display Candidate RP set information for IPv6 PIM-SM.

show pim6 scope

Displays information about the IPv64 PIM scope zones for this router.

Command Mode

Operational mode.

Syntax

```
show pim6 neighbors
```

Parameters

None.

Usage Description

Use this command to see information about this router's scoped zones for IPv6 PIM-SM.

Chapter 13: Routing Policies

This chapter lists the commands you can use to create routing policies.

This chapter contains the following command.

Command	Mode	Description
`policy as-path-list`	Configuration	Allows you to create a list of AS paths, which can be referenced in BGP policy statements.
`policy community-list`	Configuration	Allows you to create a list of BGP communities, which can be referenced in BGP policy statements.
`policy network4-list`	Configuration	Allows you to create a list of IPv4 networks, which can be referenced in policy statements.
`policy network6-list`	Configuration	Allows you to create a list of IPv6 networks, which can be referenced in policy statements.
`policy policy-statement`	Configuration	Allows you to define policies that can be applied to routing protocols.

policy as-path-list

Allows you to create a list of AS paths, which can be referenced in BGP policy statements.

Command Mode

Configuration mode.

Configuration Statement

```
policy {
   as-path-list: text {
      elements: text
   }
}
```

Parameters

as-path-list	Multi-node. Names a list of AS paths, which you can use in a routing policy match condition. The name may include numbers, letters, and hyphens only. You can define multiple AS path lists by creating multiple **as-path-list** configuration nodes.
elements	A regular expression defining a list of AS paths. Regular expressions must be enclosed in double quotes.

Usage Guidelines

Use this command to create a named list of AS paths, which you can use in BGP policy statement.

The name configured here is referred to in the match condition(s) of the policy statement.

policy community-list

Allows you to create a list of BGP communities, which can be referenced in BGP policy statements.

Command Mode

Configuration mode.

Configuration Statement

```
policy {
   community-list: text {
      elements: text
   }
}
```

Parameters

community-list	Multi-node. Names a list of BGP communities, which you can use in a routing policy match condition. The name may include numbers, letters, and hyphens only. You can define multiple community lists by creating multiple **community-list** configuration nodes.
elements	A community identifier or a space-separated list of community identifiers surrounded by enclosed in double quotes.

Usage Guidelines

Use this command to create a named list of BGP communities, which you can use in BGP policy statements.

The name configured here is referred to in the match condition(s) of the policy statement.

policy network4-list

Allows you to create a list of IPv4 networks, which can be referenced in policy statements.

Command Mode

Configuration mode.

Configuration Statement

```
policy {
   network4-list: text {
      elements: text
   }
}
```

Parameters

network4-list	Multi-node. Names a list of IPv4 networks, which you can use in a routing policy match condition. The name may include numbers, letters, and hyphens only. You can define multiple network lists by creating multiple **network4-list** configuration nodes.
elements	A regular expression defining a list of IPv4 networks. Regular expressions must be enclosed in double quotes.

Usage Guidelines

Use this command to create a named list of IPv4 networks, which you can use a in a routing policy statement.

The name configured here is referred to in the match condition(s) of the policy statement.

policy network6-list

Allows you to create a list of IPv6 networks, which can be referenced in policy statements.

Command Mode

Configuration mode.

Configuration Statement

```
policy {
   network6-list: text {
      elements: text
   }
}
```

Parameters

network6-list	Multi-node. Names a list of IPv6 networks, which you can use in a routing policy match condition. The name may include numbers, letters, and hyphens only. You can define multiple network lists by creating multiple **network6-list** configuration nodes.
elements	A regular expression defining a list of IPv6 networks. Regular expressions must be enclosed in double quotes.

Usage Guidelines

Use this command to create a named list of IPv6 networks, which you can use a in a routing policy statement.

The name configured here is referred to in the match condition(s) of the policy statement.

policy policy-statement

Allows you to define policies that can be applied to routing protocols.

Command Mode

Configuration mode.

Configuration Statement

```
policy {
    policy-statement: text {
        term: text {
            from {
                protocol: text
                network4: ipv4net
                network6: ipv6net
                network4-list: text
                network6-list: text
                prefix-length4: 0–32-range
                prefix-length6: 0–128-range
                nexthop4: ipv4-range
                nexthop6: ipv6-range
                as-path: text
                as-path-list: text
                community: text
                community-list: text
                neighbor: ipv4-range
                origin: [0|1|2]
                med: int-range
                localpref: int-range
                metric: 1–65535-range
                external: [type-1|type-2]
                tag: int-range
            }
            to {
                network4: ipv4net
                network6: ipv6net
                network4-list: text
                network6-list: text
                prefix-length4: 0–32-range
                prefix-length6: 0–128-range
                nexthop4: ipv4-range
                nexthop6: ipv6-range
                as-path: text
                as-path-list: text
                community: text
```

```
            neighbor: ipv4-range
            origin: int
            med: int-range
            localpref: int-range
            was-aggregated: bool
            metric: 1–65535-range
            external: [type-1|type-2]
            tag: int-range
        }
        then {
            action: [accept|reject]
            trace: int
            nexthop4: next-hop
            nexthop6: ipv6
            as-path-prepend: int
            as-path-expand: int
            community: text
            community-add: text
            community-del: text
            origin: int
            med: int
            med-remove: [true|false]
            localpref: int
            aggregate-prefix-len: int
            aggregate-brief-mode: int
            metric: 1–65535
            external: [type-1|type-2]
            tag: int
        }
    }
```

Parameters

Not every policy criterion in the **from**, **to**, and **then** parts of the term can be applied to every routing protocol; the applicable criteria vary with the protocol.

This section all lists parameters, regardless of their applicability. To see which options apply to which protocol, please see Table 13-3 in the Usage Guidelines.

policy-statement	Mandatory. Multi-node. Defines a named routing policy statement. You can define multiple policy statements by creating multiple **policy-statement** configuration nodes.
term	Mandatory. Multi-node. A unique numeric identifier for the term within this policy statement. You can define multiple policy terms by creating multiple **term** configuration nodes.
from	Defines a match condition for a route based on information about the source contained in the routing update. All specified criteria must match for the match condition to succeed.
protocol	The source protocol. Supported values are as follows: **connected**: The route is to a directly connected network. **static**: The route is a static route. **bgp**: The route was learned through BGP. **rip**: The route was learned through RIP. **ospf**: The route was learned through OSPF.
network4	Match the route based on its source IPv4 network. The format is *address/prefix*.
network6	Match the route based on its source IPv6 network. The format is *address/prefix*.
network4-list	Match the route based a named list of IPv4 networks. The list is defined and named using the the **policy network4-list** command (see page 270).
network6-list	Match the route based a named list of IPv6 networks. The list is defined and named using the the **policy network6-list** command (see page 271).
prefix-length4	Match the route based on its IPv4 prefix length. The range is 0 to 32.

prefix-length6	Match the route based on its IPv6 prefix length. The range is 0 to 128.
nexthop4	Match the route based on the next-hop address specified in the route announcement. The format is a match expression based on an IPv4 address.
nexthop6	Match the route based on the next-hop address specified in the route announcement. The format is a match expression based on an IPv6 address.
as-path	Match the route based on its AS path. This is a regular expression directly defining a BGP AS path filter, for example **“100 10”**. Regular expressions must be enclosed in double quotes.
as-path-list	Match the route based on an AS path regular expression defined under the specified name.
community	Match the route based on its BGP communities. The format is a community identifier or a space-separated list of community identifiers enclosed in double quotes. The router recognizes the following BGP well-known communities as per RFC 1997: **NO_EXPORT**: All routes received carrying a communities attribute containing this value are not advertised outside a BGP confederation boundary (a stand-alone autonomous system that is not part of a confederation should be considered a confederation itself). **NO_ADVERTISE**: All routes received carrying a communities attribute containing this value are not advertised to other BGP peers. **NO_SUBCONFED**: All routes received carrying a communities attribute containing this value are not advertised to external BGP peers (this includes peers in other members autonomous systems inside a BGP confederation).
community-list	Match the route based a named community list networks. The list is defined and named using the the **policy community-list** command (see page 269).

neighbor	Match the route based on the address of one or more BGP peers. The address can be a directly connected or an indirectly connected peer. The format is a match expression based on IPv4 addresses.
origin	Match the route based on an integer representing the value of the BGP ORIGIN attribute, which is the origin of the AS path information. Supported values are as follows: **0**: IGP **1**: EGP **2**: Incomplete
med	Match the route based on the multiple exit discriminator (MED). The format is a match expression based on the MED.
localpref	Match the route based on the value of the BGP LOCAL_PREF attribute. The format is a match expression based on the value of the LOCAL_PREF attribute, which is a number from 0 to 4294967295.
metric	Match the route based on its metric. The format is a match expression based on the value of the metric.
external	Sets the type of the external OSPF route. The format is a match expression based on the following values: **type-1**: Type 1 external OSPF route. **type-2**: Type 2 external OSPF route.
tag	Match the route based on its tag. The format is a match expression based on the value of the tag.
to	Defines a match condition for a route based on information about the destination in the routing update. All specified criteria must match for the match condition to succeed.
network4	Match the route based on its destination IPv4 network. The format is *address/prefix*.
network6	Match the route based on its destination IPv6 network. The format is *address/prefix*.
network4-list	Match the route based a named list of IPv4 networks. The list is defined and named using the the **policy network4-list** command (see page 270).

network6-list	Match the route based a named list of IPv6 networks. The list is defined and named using the the **policy network6-list** command (see page 271).
prefix-length4	Match the route based on its IPv4 prefix length. The range is 0 to 32.
prefix-length6	Match the route based on its IPv6 prefix length. The range is 0 to 128.
nexthop4	Match the route based on the next-hop address specified in the route announcement. The format is a match expression based on an IPv4 address.
nexthop6	Match the route based on the next-hop address specified in the route announcement. The format is a match expression based on an IPv6 address.
as-path	Match the route based on its AS path. This is a regular expression directly defining a BGP AS path filter. Regular expressions must be enclosed in double quotes.
as-path-list	Match the route based on an AS path regular expression defined under the specified name.
community	Match the route based on its communities. The format is a community identifier or a space-separated list of community identifiers enclosed in double quotes. The router recognizes the following BGP well-known communities as per RFC 1997: **NO_EXPORT**: All routes received carrying a communities attribute containing this value are not advertised outside a BGP confederation boundary (a stand-alone autonomous system that is not part of a confederation should be considered a confederation itself). **NO_ADVERTISE**: All routes received carrying a communities attribute containing this value are not advertised to other BGP peers. **NO_SUBCONFED**: All routes received carrying a communities attribute containing this value are not advertised to external BGP peers (this includes peers in other members autonomous systems inside a BGP confederation).

community-list	Match the route based a named community list networks. The list is defined and named using the the **policy community-list** command (see page 269).
neighbor	Match the route based on the address of one or more BGP peers. The address can be a directly connected or an indirectly connected peer. The format is a match expression based on IPv4 addresses.
origin	Match the route based on an integer representing the value of the BGP ORIGIN attribute, which is the origin of the AS path information. Supported values are as follows: **0**: IGP **1**: EGP **2**: Incomplete
med	Match the route based on the multiple exit discriminator (MED). The format is a match expression based on the MED.
localpref	Match the route based on the value of the BGP LOCAL_PREF attribute. The format is a match expression based on the value of the LOCAL_PREF attribute, which is a number from 0 to 4294967295.
was-aggregated	Match the route based on the value of the ATOMIC_AGGREGATED attribute. This will be true if this route contributed to origination of an aggregate. The format is a match expression based on the value of the ATOMIC_AGGREGATED attribute.
metric	Match the route based on its metric. The format is a match expression based on the value of the metric.
external	Sets the type of the external OSPF route. The format is a match expression based on the following values: **type-1**: Type 1 external OSPF route. **type-2**: Type 2 external OSPF route.
tag	Match the route based on its tag. The format is a match expression based on the value of the tag.
then	Defines the set of actions to be taken if all match conditions succeed. The default action is **accept** routes; that is, all routes are implicitly accepted.

action	How to process routes matching the criteria. Supported actions are as follows: **accept**: Accept the route and propagate it. After a route is accepted, no other terms in the routing policy and no other routing policies are evaluated. This is the default action. **reject**: Reject the route and do not propagate it. After a route is rejected, no other terms in the routing policy and no other routing policies are evaluated.
trace	Sets the level of detail for tracing. The range is 0 to 3, where 0 disables tracing and 3 provides the highest level of detail. The default is 0.
nexthop4	Specifies the next hop. Supported values are as follows: **self**: The next-hop address will be replaced with the local IP address used for BGP adjacency. Note that a router cannot install routes with itself as the next hop. *ipv4*: The next-hop address will be replaced with the specified IPv4 address. **peer-address**: Valid only for import policies. The next-hop address will be replaced with the IP address of the peer from which this route was received. This option is primarily used by BGP to enforce using the peer's IP address for advertised routes. It is meaningful only when the next hop is the advertising router or another directly connected router.
nexthop6	Specifies the next hop. Supported values are as follows: **self**: The next-hop address will be replaced with the local IP address used for BGP adjacency. Note that a router cannot install routes with itself as the next hop. *ipv6*: The next-hop address will be replaced with the specified IPv6 address. **peer-address**: Valid only for import policies. The next-hop address will be replaced with the IP address of the peer from which this route was received. This option is primarily used by BGP to enforce using the peer's IP address for advertised routes. It is meaningful only when the next hop is the advertising router or another directly connected router.

as-path-prepend	Affixes the specified AS number(s) at the beginning of the AS path. If specifying more than one AS number, surround the space-separated list with quotation marks. This action adds AS numbers to **as-path** sequences only; it does not add AS numbers to **as-path-list** sequences.
as-path-expand	Extracts the last AS number in the existing AS path and affix that AS number to the beginning of the AS path *n* times, where *n* is the specified integer. The AS number is added before the local AS number has been added to the path. This action adds AS numbers to **as-path** sequences only; it does not add AS numbers to **as-path-list** sequences. The range is 0 to 32.
community	Replaces any communities that were in the route with the specified communities. The format is a community identifier or a space-separated list of community identifiers surrounded by enclosed in double quotes. The router recognizes the following BGP well-known communities as per RFC 1997: **NO_EXPORT**: All routes received carrying a communities attribute containing this value are not advertised outside a BGP confederation boundary (a stand-alone autonomous system that is not part of a confederation should be considered a confederation itself). **NO_ADVERTISE**: All routes received carrying a communities attribute containing this value are not advertised to other BGP peers. **NO_SUBCONFED**: All routes received carrying a communities attribute containing this value are not advertised to external BGP peers (this includes peers in other members autonomous systems inside a BGP confederation).
community-add	Adds the specified communities to the set of communities in the route. To specify more than one community, use a space-separated list of community names, surrounded by quotation marks.
community-del	Deletes the specified communities from the set of communities in the route. To specify more than one community, use a space-separated list of community names, surrounded by quotation marks.

origin	Sets the value of the BGP ORIGIN attribute to the specified integer.
med	Sets the multiple exit discriminator (MED) to the specified value.
med-remove	Specifies whether or not the multiple exit discriminator (MED) should be removed. Supported values are as follows: **true**: Remove the MED. **false**: Do not remove the MED.
localpref	Sets the BGP LOCAL_PREF attribute to the specified value.
aggregate-prefix-len	Sets the aggregate prefix length to the specified value.
aggregate-brief-mode	Does not generate AS_SETs for aggregate routes.
metric	Set the metric to the specified value.
external	Set the type of external OSPF route to one of the following types: **type-1**: Type 1 external OSPF route. **type-2**: Type 2 external OSPF route.
tag	Set the tag to the specified value.

Usage Guidelines

Use this command to configure routing policies. Once the policy is defined, it must be explicitly applied to the routing protocol using the **import** and/or **export** directives in routing protocol configuration.

A policy consists of some number of *policy statements*. A policy statement consists of some number of *terms*. Each term is structured as follows:

- **from**. The **from** statement in a term describes the match conditions for the source of the route.
- **to**. The **to** statement in a term describes the match conditions for the destination of the route.
- **then**. The **then** statement in a term defines the actions that will be taken if all match conditions are met.

For the defined actions to be taken, all criteria in all defined match conditions must be met. If any criterion in a match condition is not met the match condition fails, and if any of multiple match conditions fails the match fails.

Criteria Operators

Some of the match criteria defined in **from** and **to** policy-statement terms can use operators in addition to the criteria value. For example, a **from** policy-statement term could include a **prefix-length4 > 24** statement. This would match routes with a prefix length greater than 24. In this case, the greater-than sign (“>”) is the operator.

If no operator is explicitly defined, each criterion has a default operator value. For example, by default, the operator for **prefix-length4** is equals (“==”).

Table 13-1 shows the definitions for policy operators.

Table 13-1 Operator Definitions

Operator	Example	Description
:	10.10.35.0**:**10.10.35.254	Specifies a range of values, such as a range of numbers or IP addresses. Example: “Is an IPv4 address between 10.10.35.0 and 10.10.35.254, inclusive.”
==	==15	Is equal to. Example: “is equal to 15.”
!=	!=0	Is not equal to. Example: “Is not equal to 0.”
<	<15	Is less than. Example: “Is less than 15.”
>	>12	Is greater than. Example: “Is greater than 12.”
<=	<=12	Is less than or equal to. Example: “Is less than or equal to 12.”
>=	>=12	Is greater than or equal to. Example: “Is greater than or equal to 12.”

The following criteria allow operators.

Table 13-2 Matching criteria allowing operators

Criterion	Matching Operators Allowed
localpref	all
med	all
metric	all
neighbor	all
network4	all
network4-list	all

Table 13-2 Matching criteria allowing operators

Criterion	Matching Operators Allowed
nexthop	all
origin	all
prefix-length	all
tag	all

Protocol-Specific Criteria

Not every criterion in the **from**, **to**, and **then** parts of the term can be applied to every routing protocol; the applicable criteria vary with the protocol. Table 13-3 shows which options apply to which protocols.

Table 13-3 Policy Options Applicable per Protocol

from	BGP	RIP	RIPng	OSPF	Static
protocol	✕	✕	✕	✕	✕
network4	✕	✕	✕	✕	✕
network6	✕	✕	✕	✕	✕
network4-list	✕	✕	✕	✕	✕
network6-list	✕	✕	✕	✕	✕
prefix-length4	✕	✕	✕	✕	✕
prefix-length6	✕	✕	✕	✕	✕
nexthop4	✕	✕		✕	
nexthop6	✕		✕		
as-path	✕				
as-path-list	✕				
community	✕				
community-list	✕				
neighbor	✕				
origin	✕				
med	✕				

Table 13-3 Policy Options Applicable per Protocol

localpref	×				
metric		×	×	×	×
external				×	
tag		×	×	×	
to	BGP	RIP	RIPng	OSPF	Static
network4	×	×	×	×	×
network6	×	×	×	×	×
network4-list	×	×	×	×	×
network6-list	×	×	×	×	×
prefix-length4	×	×	×	×	×
prefix-length6	×	×	×	×	×
nexthop4	×	×		×	
nexthop6	×		×		
as-path	×				
as-path-list	×				
community	×				
community-list	×				
neighbor	×				
origin	×				
med	×				
localpref	×				
was-aggregated	×				
metric		×	×	×	
external				×	
tag		×	×	×	

Table 13-3 Policy Options Applicable per Protocol

then	BGP	RIP	RIPng	OSPF	Static
action	✕	✕	✕	✕	✕
trace	✕	✕	✕	✕	✕
nexthop4	✕	✕		✕	
nexthop6	✕		✕		
as-path-prepend	✕				
as-path-expand	✕				
community	✕				
community-add	✕				
community-del	✕				
origin	✕				
med	✕				
med-remove	✕				
localpref	✕				
aggregate-prefix-len	✕				
aggregate-brief-mode	✕				
metric		✕	✕	✕	
external				✕	
tag		✕	✕	✕	

Regular Expressions

Regular expressions provide the ability to perform pattern matching are used to parse data sets within AS path lists and community lists. In general, a regular expression takes the following form:

<regex-term><operator>

where *<regex-term>* is a string to be matched, and *<operator>* is one of the operators shown in Table 13-1.

Note that operators must occur immediately after *<regex-term>* with no intervening space, with the following exceptions:

- The vertical bar operator ("|") and hyphen ("-") operator, both of which are placed between two terms
- Parentheses, which enclose <*regex-term*>s.

Table 13-4 shows the regular expression operators supported in policy statements.

Table 13-4 Regular expression operators

Operator	Description
{*m*,*n*}	At least *m* and at most *n* repetitions of *regex-term*. Both *m* and *n* must be positive integers, and *m* must be smaller than *n*.
{*m*}	Exactly *m* repetitions of *regex-term*. *m* must be a positive integer.
{*m*,}	*m* or more repetitions of *regex-term*. *m* must be a positive integer.
*	Zero or more repetitions of *regex-term*. This is equivalent to {0,}.
+	One or more repetitions of *regex-term*. This is equivalent to {1,}.
?	Zero or one repetition of *regex-term*. This is equivalent to {0,1}.
|	One of the two *regex-term* on either side of the vertical bar.
-	Between a starting and ending range, inclusive.
^	Character at the beginning of an AS path regular expression. This character is added implicitly; therefore, the use of it is optional.
$	Character at the end of an AS path regular expression. This character is added implicitly; therefore, the use of it is optional.
( )	A group of *regex-term*s that are enclosed in the parentheses. If enclosed in quotation marks with no intervening space ("()"), indicates a null. Intervening space between the parentheses and the *regex-term* is ignored.

Table 13-4 Regular expression operators

Operator	Description
[]	Set of characters. One character from the set can match. To specify the start and end of a range, use a hyphen (-).
^	NOT operator.

Chapter 14: VRRP

This chapter lists the commands for setting up the Virtual Router Redundancy Protocol on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`clear vrrp`	Operational	Restarts the VRRP process on the router, setting all interface statistics to zero.
`interfaces ethernet vrrp`	Configuration	Allows you to configure a VRRP group on an Ethernet interface.
`interfaces ethernet vif vrrp`	Configuration	Allows you to configure a VRRP group on a vif.
`show vrrp`	Operational	Displays VRRP information about VRRP groups.

clear vrrp

Restarts the VRRP process on the router, setting all interface statistics to zero.

Command Mode

Operational mode.

Syntax

`clear vrrp[`*`eth0..eth23`*`]`

Parameters

eth0..eth23	Clears VRRP statistics for the specified interface.

Usage Guidelines

Use this command to clear VRRP statistics.

Issuing this command restarts the VRRP process on the router. In doing this, it sets all VRRP statistics to zero.

- When used with no option, this command resets VRRP statistics for all configured interfaces.
- When an interface is specified, this command resets statistics for just the specified interface.

Examples

Example 14-1 clears VRRP statistics on interface **eth0**.

Example 14-1 “clear vrrp”: Clearing VRRP statistics from an interface.

```
root@vyatta> clear vrrp eth0
OK
root@vyatta>
```

show vrrp

Displays VRRP information about VRRP groups.

Command Mode

Operational mode.

Syntax

```
show vrrp
```

Parameters

None.

Usage Guidelines

Use this command to see information about VRRP groups, including current VRRP elections and statistics.

interfaces ethernet vrrp

Allows you to configure a VRRP group on an Ethernet interface.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *`name`* `vrrp …`	Use **set** to create the **vrrp** configuration node for an interface, or to modify VRRP configuration.
`delete interfaces ethernet` *`name`* `vrrp …`	Use **delete** to delete the **vrrp** configuration node for an interface.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      vrrp {
         vrrp-group: 1-255
         virtual-address: ipv4
         authentication:text
         advertise-interval: 1-255
         preempt:[true|false]
         priority: 1-255
      }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring. The interface must already be defined.
vrrp	Enables VRRP on the interface.
vrrp-group	Defines a VRRP group on the interface. The group identifier is an integer that uniquely identifies a cluster of interfaces being managed by the VRRP process. The range is 1 to 255. The default is 1.

virtual-address	Mandatory. The virtual IP address (VIP) of the virtual router. This will become the shared IP address of the group, which will float from one real router to another if the master router fails.
authentication	Optional. The plaintext password the interface will use to authenticate itself as a member of the group.
advertise-interval	Optional. The interval in seconds between VRRP advertisement packets. All routers in this VRRP group must use the same advertisement interval. The range is 1 to 255. The default is 1.
preempt	Optional. Allows a high-priority VRRP backup router to assert itself as master over a lower-priority master router. Supported values are as follows: **true**: Allow the master router to be preempted by a backup router with higher priority. **false**: Do not allow the master router to be preempted by a backup router with higher priority. The default is **true**; that is, the master router can be preempted by a backup router with higher priority.
priority	Mandatory. Sets the priority of a real router, which determines the likelihood of its being elected the master router in a cluster of VRRP routers. The range of values for the VRRP backup router(s) is from 3 to 254. The VRRP master router must have the highest priority, and typically has a priority of 255. The default is 1.

Usage Guidelines

Use this statement to define a VRRP group on an interface. The implementation is currently restricted to one VRRP group per interface, regardless of whether the group is defined at the physical interface level or the vif level.

The group identifier is an integer that uniquely identifies a cluster of interfaces being managed by the VRRP process. The group identifier together with a virtual IP address (the VIP) uniquely define an interface on the virtual router.

The group identifier used to construct a virtual MAC address for the virtual router. The five highest-order octets of the MAC address are specified in the RFC for VRRP (RFC 2338) as “00-00-5E-00-01.” VRRP inserts the 8-byte group identifier as the lowest-order octet required to complete the MAC address. If you specify a group identifier of less than 8 bytes, the system prepends the necessary leading zeros to create a well-formed octet.

The same group identifier/VIP pair must be used by all interfaces providing redundancy for one another. Unless interfaces have the same group identifier and VIP, they will not communicate.

Interfaces being mapped to the VIP must be on the same subnet as the VIP, but should not have the identical IP address.

It is possible to configure a VIP to have the same address as a real interface on the router. In this case, that router is said to “own” the VIP, and it must be configured with the highest possible priority so that it automatically becomes the master. However, this should be avoided, because conflicts can arise over which of the real router or the virtual router should respond to ARPs and other requests directed at the VIP. In any case, no backup router can have the same IP address as the VIP.

To signal that it is still in service, the master router sends MAC-level multicast “heartbeat” packets called *advertisements* to the LAN segment, using the IP multicast address **224.0.0.18**, using **port 112** (VRRP’s well-known port). These advertisements confirm the health of the master to backup routers in the cluster, and contain other VRRP information, such as the master’s priority.

If the master fails to send advertisements for some interval (the “Master is Dead” timer), the master is considered out of service, and the VRRP process triggers failover to the backup router. In this case, the backup router with the highest priority value becomes the new master router.

The advertise interval on the master router is typically one-third of the Master is Dead timer on the backup router(s).

Each VRRP router can be configured with a priority between 1 and 255. The router with the highest priority is elected as the master router of the VRRP cluster.

The VRRP standard (RFC 2338) specifies that a router owning the virtual IP should be assigned a priority of 255, which automatically elects the router owning the VIP as master. If you configure a VIP that is the real IP address of an interface on a router, you must set the priority of that router as 255. In any case, the priority of the master router is typically set to 255.

The backup router can be left with the default priority. However, if you have more than one backup router, you should set different priorities to ensure election occurs correctly when required.

The VRRP advertisements sent out by the master router include the master router’s priority. If preemption is enabled, a backup router with a higher priority than the current master will “preempt” the master, and become the master itself. This might occur, for example, if a new backup router is brought online, while a lower-priority backup is acting as master.

A backup router preempts the master by beginning to send out its own VRRP advertisements. The master router examines these, and discovers that the backup router has a higher priority than itself. The master then stops sending out advertisements, while the backup continues to send, thus making itself the new master.

interfaces ethernet vif vrrp

Allows you to configure a VRRP group on a vif.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *`int.vif`* `vrrp …`	Use **set** to create the **vrrp** configuration node for a vif, or to modify VRRP configuration.
`delete interfaces ethernet` *`int.vif`* `vrrp …`	Use **delete** to delete the **vrrp** configuration node of a vif. Note that the **vrrp-group** node is mandatory, and therefore cannot be deleted. If you delete the **vrrp-group** node, the system creates a new VRRP group with a group ID of 1.

Configuration Statement

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      vif vlan-id {
      }
   }
}
```

Parameters

vif	The VLAN ID of the vifa. The vif must already be defined. Note the notation for referring to the vif is *int.vif*. For example, to configure VRRP on vif 40 or eth1, use the statement **set interfaces ethernet eth1.40 vrrp**....
vrrp	Enables VRRP on the vif.

vrrp-group	Defines a VRRP group on the vif. The group identifier is an integer that uniquely identifies a cluster of interfaces being managed by the VRRP process. At least one VRRP group must be defined. If you do not define one, or if you delete the last existing VRRP group, the system creates a **vrrp-group** node with a group ID of 1. The range is 1 to 255. The default is 1.
virtual-address	Mandatory. The virtual IP address (VIP) of the virtual router. This will become the shared IP address of the group, which will float from one real router to another if the master router fails.
authentication	Optional. The plaintext password the vif will use to authenticate itself as a member of the group.
advertise-interval	Optional. The interval in seconds between VRRP advertisement packets. All routers in this VRRP group must use the same advertisement interval. The range is 1 to 255. The default is 1.
preempt	Optional. Allows a high-priority VRRP backup router to assert itself as master over a lower-priority master router. Supported values are as follows: **true**: Allow the master router to be preempted by a backup router with higher priority. **false**: Do not allow the master router to be preempted by a backup router with higher priority. The default is **true**; that is, the master router can be preempted by a backup router with higher priority.
priority	Mandatory. Sets the priority of a real router, which determines the likelihood of its being elected the master router in a cluster of VRRP routers. The range of values for the VRRP backup router(s) is from 3 to 254. The VRRP master router must have the highest priority, and typically has a priority of 255. The default is 1.

Usage Guidelines

Use this statement to define a VRRP group on a vif. The implementation is currently restricted to one VRRP group per interface, regardless of whether the group is defined at the physical interface level or the vif level.

The group identifier is an integer that uniquely identifies a cluster of interfaces being managed by the VRRP process. The group identifier together with a virtual IP address (the VIP) uniquely define a vif on the virtual router.

The group identifier used to construct a virtual MAC address for the virtual router. The five highest-order octets of the MAC address are specified in the RFC for VRRP (RFC 2338) as “00-00-5E-00-01.” VRRP inserts the 8-byte group identifier as the lowest-order octet required to complete the MAC address. If you specify a group identifier of less than 8 bytes, the system prepends the necessary leading zeros to create a well-formed octet.

The same group identifier/VIP pair must be used by all interfaces providing redundancy for one another. Unless interfaces have the same group identifier and VIP, they will not communicate.

Interfaces being mapped to the VIP must be on the same subnet as the VIP, but should not have the identical IP address.

It is possible to configure a VIP to have the same address as a real interface on the router. In this case, that router is said to “own” the VIP, and it must be configured with the highest possible priority so that it automatically becomes the master. However, this should be avoided, because conflicts can arise over which of the real router or the virtual router should respond to ARPs and other requests directed at the VIP. In any case, no backup router can have the same IP address as the VIP.

To signal that it is still in service, the master router sends MAC-level multicast “heartbeat” packets called *advertisements* to the LAN segment, using the IP multicast address **224.0.0.18**, using **port 112** (VRRP’s well-known port). These advertisements confirm the health of the master to backup routers in the cluster, and contain other VRRP information, such as the master’s priority.

If the master fails to send advertisements for some interval (the “Master is Dead” timer), the master is considered out of service, and the VRRP process triggers failover to the backup router. In this case, the backup router with the highest priority value becomes the new master router.

The advertise interval on the master router is typically one-third of the Master is Dead timer on the backup router(s).

Each VRRP router can be configured with a priority between 1 and 255. The router with the highest priority is elected as the master router of the VRRP cluster.

The VRRP standard (RFC 2338) specifies that a router owning the virtual IP should be assigned a priority of 255, which automatically elects the router owning the VIP as master. If you configure a VIP that is the real IP address of an interface on a router, you must set the priority of that router as 255. In any case, the priority of the master router is typically set to 255.

The backup router can be left with the default priority. However, if you have more than one backup router, you should set different priorities to ensure election occurs correctly when required.

The VRRP advertisements sent out by the master router include the master router's priority. If preemption is enabled, a backup router with a higher priority than the current master will "preempt" the master, and become the master itself. This might occur, for example, if a new backup router is brought online, while a lower-priority backup is acting as master.

A backup router preempts the master by beginning to send out its own VRRP advertisements. The master router examines these, and discovers that the backup router has a higher priority than itself. The master then stops sending out advertisements, while the backup continues to send, thus making itself the new master.

Chapter 15: NAT

This chapter lists the commands for setting up NAT on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`clear nat counters`	Operational	Resets counters for active NAT rules.
`clear nat translations`	Operational	Clears state information associated with the specified NAT rule(s).
`service nat`	Configuration	Configures NAT on the router.
`show nat rules`	Operational	Lists configured NAT rules.
`show nat statistics`	Operational	Displays statistics for NAT.

clear nat counters

Resets counters for active NAT rules.

Command Mode

Operational mode.

Syntax

```
clear nat counters
```

Parameters

None.

Usage Guidelines

Use this command to reset counters for NAT translation rules. Counters are reset for all rules.

clear nat translations

Clears state information associated with the specified NAT rule(s).

Command Mode

Operational mode.

Syntax

```
clear nat translations
```

Parameters

None.

Usage Guidelines

Use this rule to clear state information associated with all NAT rules.

service nat

Configures NAT on the router.

Command Mode

Configuration mode.

Syntax

`set service nat …`	Use **set** to create the **nat** configuration node or modify NAT configuration. Note that you cannot use **set** to change the number of a NAT rule, as it is the identifier of a configuration node. To change the number of a NAT rule, delete the rule and create it again with the correct number.
`delete service nat …`	Use **delete** to delete a NAT rule or one of a rule's subordinate configuration nodes, or to delete the **nat** configuration node altogether.

Configuration Statement

```
service {
   nat {
      rule: 1-1024 {
         type: [source|destination]
         translation-type: [static|dynamic|masquerade]
         inbound-interface: text
         outbound-interface: text
         protocols: [tcp|udp|icmp|all]
         source {
            address: ipv4
            network: ipv4net
            port-number: 1-4294967296 {}
            port-name: [http|ftp|smtp|telnet|ssh|dns|snmp] {}
            port-range {
               start: 1-4294967296
               stop: 1-4294967296
            }
         }
         destination {
            address: ipv4
            network: ipv4net
```

```
                port-number: 1-4294967296 {}
                port-name: [http|ftp|smtp|telnet|ssh|dns|snmp] {}
                port-range {
                    start: 1-4294967296
                    stop: 1-4294967296
                }
            inside-address {
                address: ipv4
                network: ipv4net
            }
            outside-address {
                address: ipv4
                network: ipv4net
                range {
                    start: ipv4
                    stop: ipv4
                }
            }
        }
    }
}
```

Parameters

rule	Mandatory. Multi-node. A numeric identifier for the rule. The range is 1–1024. Note that in the current implementation, the table of NAT rules is not sorted numerically. NAT rules are evaluated *in the sequence in which they were configured*, regardless of the rule number. (This is different from the firewall feature, where rules are evaluated in sequence according to rule number.)
type	Mandatory. Indicates whether this rule is translating the source IP or the destination IP. Note that this is dependent on the direction of the interface. The supported values are as follows: **source**: This rule translates the source network address. Typically “source” rules are applied to outbound packets. **destination**: This rule translates the destination network address. Typically “destination” rules are applied to inbound packets.

translation-type	Mandatory. Specifies whether the rule will apply static mapping, dynamic many-to-one mapping, or masquerade mapping. Supported values are as follows: **static**: The rule applies one-to-one static mapping. **dynamic**: The rule applies dynamic many-to-one mapping. **masquerade**: The rule uses a router interface IP address for source NAT only.
inbound-interface	Mandatory for destination NAT. The inbound Ethernet or serial interface. Destination NAT (DNAT) translation will be performed on traffic received on this interface. You can specify an individual vif, rather than an entire interface. To do this, refer to the vif using *int.vif* notation. For example to refer to vif 40 on interface eth0, use **eth0.40**.
outbound-interface	Mandatory for source NAT. The outbound Ethernet or serial interface. Source NAT (SNAT) translation will be performed on traffic transmitted from this interface. You can specify an individual vif, rather than an entire interface. To do this, refer to the vif using *int.vif* notation. For example to refer to vif 40 on interface eth0, use **eth0.40**.
protocols	Optional. The protocols on which to perform NAT. Supported values are as follows: **tcp**: Performs NAT on TCP traffic only. **udp**: Performs NAT on UDP traffic only. **icmp**: Performs NAT on ICMP traffic only. **all**: Performs NAT on all protocol traffic. The default is **all**.
source	Optional. Defines the source for this NAT rule. • Source addresses are defined by specifying just one of **address** or **network**. • Source ports can only be defined when the specified protocol is TCP or UDP. Source ports are defined by specifying just one of **port-number**, **port-name**, or **port-range**.

address	Mandatory. The IP address to be included as the "source" address in the IP header. • For source NAT, this will be the "inside" (private) IP address or subnet. • For destination NAT this will be the "outside" (public) IP address or subnet.
network	The source network. The format is *ip-address*/*prefix*. The default is "any," which is represented as **0/0**.
port-number	Specifies a port by number (for example, port **80**). The range is 0 to 65535.
port-name	Specifies a port using the protocol literal. The following protocol literals are supported: • **http** (maps to port 80) • **ftp** (maps to port 20 and 21) • **smtp** (maps to port 25) • **telnet** (maps to port 23) • **ssh** (maps to port 22) • **dns** (maps to port 53) • **snmp** (maps to port 161)
port-range	Defines a range of consecutive ports for the source. The range is 1 to 4294967296.
start	Mandatory. The start port for the source port range. The range is 1 to 4294967296, where **start** must be lower than **stop**.
stop	Mandatory. The stop port for the source port range. The range is 1 to 4294967296, where **start** must be lower than **stop**.
destination	Optional. Defines the destination for this NAT rule. • Destination addresses are defined by specifying just one of **address** or **network**. • Destination ports can only be defined when the specified protocol is TCP or UDP. Source ports are defined by specifying just one of **port-number**, **port-name**, or **port-range**.

network	The destination network. The format is *ip-address*/*prefix*, where *ip-address* is an IP address and *prefix* is a number from 0 to 32.
address	The destination IPv4 address. When you have an inbound destination static NAT when traffic comes to the
port-number	Specifies a port by number (for example, port **80**). The range is 0 to 65535.
port-name	Specifies a port using the protocol literal. The following protocol literals are supported: • **http** (maps to port 80) • **ftp** (maps to port 20 and 21) • **smtp** (maps to port 25) • **telnet** (maps to port 23) • **ssh** (maps to port 22) • **dns** (maps to port 53) • **snmp** (maps to port 161)
port-range	Defines a range of consecutive ports.
start	Mandatory. The start port for the destination port range. The range is 1 to 4294967296, where **start** must be lower than **stop**.
stop	Mandatory. The stop port for the destination port range. The range is 1 to 4294967296, where **start** must be lower than **stop**.
inside-address	Defines the “inside” IP address for destination NAT rules with a translation type of **static**. Mandatory for destination NAT rules with a translation type of **static**. Forbidden otherwise. Destination rules ingress from the untrusted to the trusted network. For static NAT rules, the inside address defines the IP address of the host on the trusted network. This is the address that will be substituted for the original destination IP address on packets sent to the OFR.
address	An IP address.
network	A network. The format is *ip-address*/*prefix*., where *ip-address* is an IP address and *prefix* is a number from 0 to 32.

outside-address	Defines the "outside" IP address for source NAT rules with a translation type of **static** or **dynamic**. Mandatory for source NAT rules with a translation type of **static** or **dynamic**. Forbidden otherwise. Source rules egress from the trusted to the untrusted network. For static and dynamic source NAT rules, the outside address defines the IP address that faces the untrusted network. This is the address that will be substituted in for the original source IP address in packets egressing to the untrusted network. An outside address is not required for source rules with a translation type of **masquerade**, because for masquerade source rules the original source IP address is replaced with the IP address of the outbound interface.
address	An IP address.
network	A network. The format is *ip-address*/*prefix*, where *ip-address* is an IP address and *prefix* is a number from 0 to 32.
range	Defines a range of consecutive IP addresses. Make sure the "start" address is lower than the "stop" address.
start	Mandatory. The start address.
stop	Mandatory. The stop address.

Usage Guidelines

Use this statement to configure NAT.

In this release, you must create explicit NAT rules for each direction of traffic. For example, if you configure a one-to-one static source NAT rule and you want inbound traffic to match the NAT rule, you must explicitly create a matching destination NAT rule.

The "source" and "destination" attributes are relative to the interface they are applied to. For example, an outbound interface will process traffic as it leaves the interface. If the type of its rule is "source," it will change the source IP address.

This configuration structure does not currently support port rewriting (for example, where packets destined for port 80 are rewritten to be destined for 8080).

show nat rules

Lists configured NAT rules.

Command Mode

Operational mode.

Syntax

```
show nat rules [dynamic|static]
```

Parameters

dynamic	Displays only dynamic NAT rules.
static	Displays only static NAT rules.

Usage Guidelines

Use this command to display the NAT rules you have configured. You can use this command for troubleshooting, to confirm whether traffic is matching the NAT rules as expected.

When used with no option, this command displays all rules in the NAT rule table. Otherwise, you can choose to display just dynamic NAT or just static NAT rules.

show nat statistics

Displays statistics for NAT.

Command Mode

Operational mode.

Syntax

```
show nat statistics
```

Parameters

None.

Usage Guidelines

Use this command to display current statistics for NAT.

Chapter 16: Firewall

This chapter lists the commands for setting up firewall functionality on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`clear firewall name counters`	Operational	Clears all statistics associated with the specified firewall rule set.
`firewall`	Configuration	Configures a firewall instance (a named rule set) to use in packet filtering.
`interfaces ethernet firewall`	Configuration	Applies named firewall instances (packet-filtering rule sets) to an Ethernet interface.
`interfaces ethernet vif firewall`	Configuration	Applies named firewall instances (packet-filtering rule sets) to a virtual interface.
`interfaces serial cisco-hdlc vif firewall`	Configuration	Applies named firewall instances (packet-filtering rule sets) to a Cisco HDLC–encapsulated serial interface.
`interfaces serial frame-relay vif firewall`	Configuration	Applies named firewall instances (packet-filtering rule sets) to a Frame Relay–encapsulated serial interface.
`interfaces serial ppp vif firewall`	Configuration	Applies named firewall instances (packet-filtering rule sets) to a Point-to-Point Protocol–encapsulated serial interface.
`show firewall`	Operational	Shows the list of rules associated with a specific firewall instance.

clear firewall name counters

Clears all statistics associated with the specified firewall rule set.

Command Mode

Operational mode.

Syntax

clear firewall name *firewall-name* counters

Parameters

firewall-name	The name of the firewall rule set where statistics are to be cleared.

Usage Guidelines

Use this command to clear the statistics associated with a specific firewall rule set.

firewall

Configures a firewall instance (a named rule set) to use in packet filtering.

Command Mode

Configuration mode.

Syntax

`set firewall …`	Use **set** to create the **firewall** configuration node, or to modify the configuration of a firewall rule set. Note that you cannot use **set** to change the identifier of a configuration node. Specifically, you cannot use **set** to change the number of a firewall rule. To change the number of a firewall rule, delete the rule and create it again with the correct identifier.
`delete firewall …`	Use **delete** to delete a firewall rule set or one of a rule set's subordinate configuration nodes, or to delete the **firewall** configuration node altogether.

Configuration Statement

```
firewall {
   log-martians: [enable|disable]
   send-redirects: [enable|disable]
   receive-redirects: [enable|disable]
   ip-src-route: [enable|disable]
   broadcast-ping: [enable|disable]
   syn-cookies: [enable|disable]
   name: text {
      description: text
      rule: 1-1024 {
         protocol: [all|tcp|udp|icmp|igmp|ipencap|gre|esp|ah|
            ospf|pim|vrrp]
         icmp {
            type: text {
            code: text
         }
         state {
            established: [enable|disable]
            new: [enable|disable]
            related: [enable|disable]
            invalid: [enable|disable]
```

```
                        }
                    action: [accept|drop|reject]
                    log: [enable|disable]
                    source {
                        address: ipv4
                        network: ipv4net
                        range {
                            start: ipv4
                            stop: ipv4
                        }
                        port-number: 1-65535
                        port-name: [http|ftp|smtp|telnet|ssh|dns|snmp]
                        port-range {
                            start: 1-65535
                            stop: 1-65535
                        }
                    }
                    destination {
                        address: ipv4
                        network: ipv4net
                        range {
                            start: ipv4
                            stop: ipv4
                        }
                        port-number: 1-65535
                        port-name: [http|ftp|smtp|telnet|ssh|dns|snmp]
                        port-range {
                            start: 1-65535
                            stop: 1-65535
                        }
                    }
                }
            }
        }
```

Parameters

log-martians	Optional. Directs whether to log packets with impossible addresses. Supported values are as follows: **enable:** Records packets with impossible addresses in the log. **disable:** Does not record packets with impossible addresses in the log. The default is **enable**.

send-redirects	Optional. Directs whether to allow sending of ICMP redirects. Sending a redirect will potentially alter the routing table of the host or router to which the redirect is sent. Supported values are as follows: **enable:** Allows ICMP redirects to be sent. **disable:** Does not allow ICMP redirects to be sent. The default is **disable**.
receive-redirects	Optional. Directs whether to accept ICMP redirects. ICMP redirects can allow an arbitrary sender to forge packets and alter the router's routing table. This can leave the router open to a man-in-the-middle attack. Supported values are as follows: **enable:** Permits packets with ICMP redirects. **disable:** Denies packets with ICMP redirects. The default is **disable**.
ip-src-route	Optional. Directs whether to permit or deny packets with the Loose Source Route or Strict Source Route IP options. Source routing allows applications to override the routing tables and specify one or more intermediate destinations for outgoing datagrams. This capability is sometimes used for troubleshooting, but renders the network vulnerable to attacks where network traffic is transparently directed to a centralized collection point for packet capture. Supported values are as follows: **enable:** Permits packets with source routing IP options set. **disable:** Drops packets with source routing IP options set. The default is **disable**.
broadcast-ping	Optional. Directs whether the router will respond to ICMP Echo request messages sent to an IP broadcast address. Supported values are as follows: **enable:** The router will respond to ICMP Echo requests sent to the broadcast address. **disable:** The router will ignore ICMP Echo requests sent to the broadcast address. The default is **disable**.

syn-cookies	Optional. Enabling this option can help protect the router from a TCP SYN Flood Denial of Service (DoS) attack. To start a TCP connection, a source sends a SYN (synchronize/start) packet. The destination sends back a SYN ACK (synchronize acknowledge). Then the source sends an ACK (acknowledge), and the connection is established. This is referred to as the "TCP three-way handshake." After a destination server sends a SYN ACK, it uses a connection queue to keep track of the connections waiting to be completed. An attacker can fill up the connection queue by generating phony TCP SYN packets from random IP addresses at a rapid rate. When the connection queue is full, all subsequent TCP services are denied. When this option is enabled, the router creates a hash entry when it receives a SYN packet, and returns a SYN ACK cookie only, without retaining all the SYN information. When it receives the ACK from the client, it validates it against the hash and, if it is valid, rebuilds the SYN packet information and accepts the packet. **enable:** Enables TCP SYN cookies option. **disable:** Disables TCP SYN cookies option. The default is **enable**.
name	The name of this firewall instance. A firewall instance consists of a rule set of up to 1024 rules. Following the 1024 configurable rules is an implicit "deny all" rule.
description	A brief description for this firewall instance. If the description contains spaces, enclose it in double quotes.
rule	Mandatory. Defines a firewall rule within the rule set. The argument is the rule number, which specifies the order in which this rule appears in the firewall rule table. The range is 1 to 1024. Each rule must have a unique rule number. Note that rules are evaluated in sequence according to rule number. This is different from the NAT feature, where rules are evaluated in the order in which they were configured, regardless of rule number. Note that after the final user-defined rule is executed, an implicit rule of **deny all** takes effect.

protocol	Optional. Defines the protocol to which the firewall rule applies. Packets using this protocol will “match” the rule. **Note:** The protocol must be specified for the source or the destination, but not both. Supported values are as follows: **all:** This rule applies to packets of all protocols. **tcp:** This rule applies to TCP packets only. **udp:** This rule applies to UDP packets only. **icmp:** This rule applies to ICMP packets only. **igmp:** This rule applies to IGMP packets only. **ipencap:** This rule applies to IP-in-IP packets only. **gre:** This rule applies to GRE packets only. **esp:** This rule applies to ESP packets only. **ah:** This rule applies to AH packets only. **ospf:** This rule applies to OSPF packets only. **pim:** This rule applies to PIM packets only. **vrrp:** This rule applies to VRRP packets only. The default is **all**.
icmp	Optional. Defines the ICMP types this packet applies to—for example Echo Request or Echo Reply. Packets having this ICMP type will “match” the rule.
type	Mandatory.A valid ICMP type code from 0 to 255; for example, **8** (Echo Request), or **0** (Echo Reply), or the keyword **all**. The default is all. For a list of ICMP codes and types, see “Appendix A: ICMP Types.”
code	Optional. The ICMP type code associated with this ICMP type. The range is 0 to 255. For a list of ICMP codes and types, see “Appendix A: ICMP Types.”
state	Specifies the kind of packets this rule will be applied to. You can enable multiple states.

established	This rule will be applied to packets that are part of a connection that has seen packets in both directions (for example, a reply packet, or an outgoing packet on a connection that has been replied to). Supported values are as follows: **enable:** Allow packets that are part of an established connection. **disable:** Block packets that are part of an established connection. The default is **disable**.
new	This rule will be applied to packets creating new connections. For TCP, this will be packets with the SYN flag set. Supported values are as follows: **enable:** Allow packets that are part of a new connection. **disable:** Block packets that are part of a new connection. The default is **disable**.
related	This rule will be applied to a packet that is related to, but not part of, an existing connection, such as an ICMP error. Supported values are as follows: **enable:** Allow packets that are part of a related connection. **disable:** Block packets that are part of a related connection. The default is **disable**.
invalid	This rule will be applied to packets that could not be identified for some reason. These might include the router running out of resource, or ICMP errors that do not correspond to any known connection. Generally these packets should be dropped. Supported values are as follows: **enable:** Allow packets that are part of an invalid connection. **disable:** Block packets that are part of an invalid connection. The default is **disable**.
action	Mandatory. The action to perform on packets that match the criteria specified in this firewall rule. Only one action can be defined for a rule. Supported values are as follows: **accept:** Accepts and forwards packets matching the criteria. **drop:** Silently drops packets matching the criteria. **reject:** Drops packets matching the criteria with a TCP reset.

log	Any actions taken will be logged. Supported values are as follows: **enable:** Log when action is taken. **disable:** Do not log when action is taken. The default is **disable**.
source	Optional. Defines the source for this firewall rule. • Source addresses are defined by specifying just one of **address**, **network**, or **range**. • Source ports can only be defined when the specified protocol is TCP or UDP. Source ports are defined by specifying just one of **port-number**, **port-name**, or **port-range**.
address	An IPv4 address.
network	The source network. The format is *ip-address*/*prefix*. The default is "any," which is represented as **0/0**.
range	Defines a range of contiguous addresses for the source.
start	Mandatory. The start address for the source address range.
stop	Mandatory. The stop address for the source address range.
port-number	Specifies a port by number (for example, port **80**).
port-name	Specifies a port using the protocol literal. The following protocol literals are supported: • **http** (maps to port 80) • **ftp** (maps to port 20 and 21) • **smtp** (maps to port 25) • **telnet** (maps to port 23) • **ssh** (maps to port 22) • **dns** (maps to port 53) • **snmp** (maps to port 161) You can specify more than one protocol using a comma-separated list, for example **http,ssh,telnet**.
port-range	Defines a range of consecutive ports for the source. The range is 0 to 65535.

start	Mandatory. The start port for the source port range. The range is 1 to 65535, where **start** must be a lower port number than **stop**.
stop	Mandatory. The stop port for the source port range. The range is 1 to 65535, where **start** must be a lower port number than **stop**.
destination	Defines the destination for this firewall rule. • Destination addresses are defined by specifying just one of **address**, **network**, or **range**. • Destination ports can only be defined when the specified protocol is TCP or UDP. Destination ports are defined by specifying just one of **port-number**, **port-name**, or **port-range**.
address	The destination IPv4 address.
network	Defines the destination network. The format is *ip-address*/*prefix*. The default is “any”, which is represented as 0/0.
range	Defines a range of contiguous addresses as the destination.
start	Mandatory. The start address for the destination address range.
stop	Mandatory. The stop address for the destination address range.
port-number	Specifies a port by number (for example, port **80**).
port-name	Specifies a port using the protocol literal. The following protocol literals are supported: • **http** (maps to port 80) • **ftp** (maps to port 20 and 21) • **smtp** (maps to port 25) • **telnet** (maps to port 23) • **ssh** (maps to port 22) • **dns** (maps to port 53) • **snmp** (maps to port 161) You can specify more than one protocol using a comma-separated list, for example **http,ssh,telnet**.
port-range	Defines a range of consecutive ports.

start	Mandatory. The start port for the destination port range. The range is 1 to 65535, where **start** must be a lower port number than **stop**.
stop	Mandatory. The stop port for the destination port range. The range is 1 to 65535, where **start** must be a lower port number than **stop**.

Usage Guidelines

Use this statement to configure firewall.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface using the **interfaces ethernet firewall** command (see page 324).

Note that after the final user-defined rule is executed, an implicit rule of “deny all” takes effect.

interfaces ethernet firewall

Applies named firewall instances (packet-filtering rule sets) to an Ethernet interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif using this command.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *`name`* `firewall …`	Use **set** to specify the rule sets to be applied to an Ethernet interface.
`delete interfaces ethernet` *`name`* `firewall …`	Use **delete** to remove a packet filter (or all packet filters) from an interface.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      firewall {
         in {
            name: text
         }
         out {
            name: text
         }
         local {
            name: text
         }
      }
   }
}
```

Parameters

interface	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.

firewall	Applies a named firewall rule set to the interface. One rule set can be applied to each of the following: • Inbound packets • Outbound packets • Packets destined for this router itself
in	The specified rule set will be applied to packets entering this interface.
name	Applies the specified rule set to packets entering this interface.
out	The specified rule set will be applied to packets leaving this interface.
name	Applies the specified rule set to packets leaving this interface.
local	The specified rule set will be applied to packets destined for this router itself.
name	Applies the specified rule set to packets destined for this router.

Usage Guidelines

Use this statement to apply the rule set defined for a firewall instance to a vif.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif using this command.

To use the firewall feature, you define a firewall rule set as a named firewall instance, using the **firewall** command (see page 315). You then apply the firewall instance to interfaces and/or vifs using this statement or the **interfaces ethernet vif firewall** command (see page 326). Once applied, the instance acts as a packet filter.

The firewall instance will filter packets in one of the following ways, depending on what you specify when you apply it to the interface:

- **in.** If you apply the rule set as **in**, the firewall will filter packets entering the interface.
- **out.** If you apply the rule set as **out**, the firewall will filter packets leaving the interface.
- **local.** If you apply the rule set as **local**, the firewall will filter packets destined for this router itself.

For each interface, you can apply up to three firewall instances: one firewall **in** instance, one firewall **out** instance, and one firewall **local** instance.

Make sure the firewall instance you apply to an interface is already defined, or you may experience unintended results. If you apply a firewall instance that does not exist to an interface, the implicit firewall rule of **allow all** will be applied.

To define firewall rule sets, use the **firewall** command (see page 315).

interfaces ethernet vif firewall

Applies named firewall instances (packet-filtering rule sets) to a virtual interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

Command Mode

Configuration mode.

Syntax

`set interfaces ethernet` *name* `vif` *name* `firewall …`	Use **set** to specify the rule sets to be applied to a vif on an Ethernet interface.
`delete interfaces ethernet` *name* `vif` *name* `firewall …`	Use **delete** to remove a packet filter (or all packet filters) from a virtual.

Configuration Statement

```
interfaces {
   ethernet [eth0..eth23] {
      vif text {
         firewall {
            in {
               name: text
            }
            out {
               name: text
            }
            local {
               name: text
            }
         }
      }
   }
}
```

Parameters

ethernet	The Ethernet interface you are configuring: one of **eth0** through **eth23**. The interface must already have been defined.

vif	The vif you are configuring. The vif must already have been defined.
firewall	Applies a named firewall rule set to the vif. One rule set can be applied to each of the following: • Inbound packets • Outbound packets • Packets destined for this router itself
in	The specified rule set will be applied to packets entering this vif.
name	Applies the specified rule set to packets entering this vif.
out	The specified rule set will be applied to packets leaving this vif.
name	Applies the specified rule set to packets leaving this vif.
local	The specified rule set will be applied to packets destined for this router itself.
name	Applies the specified rule set to packets destined for this router.

Usage Guidelines

Use this statement to apply the rule set defined for a firewall instance to the vif of an Ethernet interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

To use the firewall feature, you define a firewall rule set as a named firewall instance, using the **firewall** command (see page 315). You then apply the firewall instance to interfaces and/or vifs using a statement like this one. Once applied, the instance acts as a packet filter.

The firewall instance will filter packets in one of the following ways, depending on what you specify when you apply it to the vif:

- **in.** If you apply the rule set as **in**, the firewall will filter packets entering the vif.
- **out.** If you apply the rule set as **out**, the firewall will filter packets leaving the vif.
- **local.** If you apply the rule set as **local**, the firewall will filter packets destined for this router itself.

For each vif, you can apply up to three firewall instances: one firewall **in** instance, one firewall **out** instance, and one firewall **local** instance.

Make sure the firewall instance you apply to a vif is already defined, or you may experience unintended results. If you apply a firewall instance that does not exist to vif, the implicit firewall rule of **allow all** will be applied.

To define firewall rule sets, use the **firewall** command (see page 315).

interfaces serial cisco-hdlc vif firewall

Applies named firewall instances (packet-filtering rule sets) to a Cisco HDLC–encapsulated serial interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *name* `cisco-hdlc vif` *name* `firewall …`	Use **set** to specify the firewall rule sets to be applied to a Cisco HDLC–encapsulated serial interface.
`delete interfaces serial` *name* `cisco-hdlc vif` *name* `firewall …`	Use **delete** to remove a packet filter (or all packet filters) from the vif configuration node of a Cisco HDLC–encapsulated serial interface.

Configuration Statement

```
interfaces {
    serial [wan0..wan9] {
        vif 1 {
            cisco-hdlc {
                firewall {
                    in {
                        name: text
                    }
                    out {
                        name: text
                    }
                    local {
                        name: text
                    }
                }
            }
        }
    }
}
```

Parameters

serial	The serial interface you are configuring: one of **wan0** through **wan9**. The interface must already have been defined.
vif	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.
cisco-hdlc	Identifies this interface as a Cisco HDLC–encapsulated interface.
firewall	Applies a named firewall rule set to the interface. One rule set can be applied to each of the following: • Inbound packets • Outbound packets • Packets destined for this router itself
in	The specified rule set will be applied to packets entering this interface.
name	Applies the specified rule set to packets entering this interface.
out	The specified rule set will be applied to packets leaving this interface.
name	Applies the specified rule set to packets leaving this interface.
local	The specified rule set will be applied to packets destined for this router itself.
name	Applies the specified rule set to packets destined for this router.

Usage Guidelines

Use this statement to apply the rule set defined for a firewall instance to the vif of a Cisco HDLC–encapsulated serial interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

To use the firewall feature, you define a firewall rule set as a named firewall instance, using the **firewall** command (see page 315). You then apply the firewall instance to interfaces and/or vifs using this statement or one like it. Once applied, the instance acts as a packet filter.

The firewall instance will filter packets in one of the following ways, depending on what you specify when you apply it to the interface:

- **in.** If you apply the rule set as **in**, the firewall will filter packets entering the interface.
- **out.** If you apply the rule set as **out**, the firewall will filter packets leaving the interface.
- **local.** If you apply the rule set as **local**, the firewall will filter packets destined for this router itself.

For each interface, you can apply up to three firewall instances: one firewall **in** instance, one firewall **out** instance, and one firewall **local** instance.

Make sure the firewall instance you apply to an interface is already defined, or you may experience unintended results. If you apply a firewall instance that does not exist to an interface, the implicit firewall rule of **allow all** will be applied.

To define firewall rule sets, use the **firewall** command (see page 315).

interfaces serial frame-relay vif firewall

Applies named firewall instances (packet-filtering rule sets) to a Frame Relay–encapsulated serial interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *name* `frame-relay vif` *name* `firewall …`	Use **set** to specify the firewall rule sets to be applied to a Frame Relay–encapsulated serial interface.
`delete interfaces serial` *name* `frame-relay vif` *name* `firewall …`	Use **delete** to remove a packet filter (or all packet filters) from the vif configuration node of a Frame Relay–encapsulated serial interface.

Configuration Statement

```
interfaces {
    serial [wan0..wan9] {
        vif 16-991 {
            frame-relay {
                firewall {
                    in {
                        name: text
                    }
                    out {
                        name: text
                    }
                    local {
                        name: text
                    }
                }
            }
        }
    }
}
```

Parameters

serial	The serial interface you are configuring: one of **wan0** through **wan9**. The interface must already have been defined.
vif	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.
cisco-hdlc	Identifies this interface as a Cisco HDLC–encapsulated interface.
firewall	Applies a named firewall rule set to the interface. One rule set can be applied to each of the following: • Inbound packets • Outbound packets • Packets destined for this router itself
in	The specified rule set will be applied to packets entering this interface.
name	Applies the specified rule set to packets entering this interface.
out	The specified rule set will be applied to packets leaving this interface.
name	Applies the specified rule set to packets leaving this interface.
local	The specified rule set will be applied to packets destined for this router itself.
name	Applies the specified rule set to packets destined for this router.

Usage Guidelines

Use this statement to apply the rule set defined for a firewall instance to the vif of a Frame Relay–encapsulated serial interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

To use the firewall feature, you define a firewall rule set as a named firewall instance, using the **firewall** command (see page 315). You then apply the firewall instance to interfaces and/or vifs using this statement or one like it. Once applied, the instance acts as a packet filter.

The firewall instance will filter packets in one of the following ways, depending on what you specify when you apply it to the interface:

- **in.** If you apply the rule set as **in**, the firewall will filter packets entering the interface.
- **out.** If you apply the rule set as **out**, the firewall will filter packets leaving the interface.
- **local.** If you apply the rule set as **local**, the firewall will filter packets destined for this router itself.

For each interface, you can apply up to three firewall instances: one firewall **in** instance, one firewall **out** instance, and one firewall **local** instance.

Make sure the firewall instance you apply to an interface is already defined, or you may experience unintended results. If you apply a firewall instance that does not exist to an interface, the implicit firewall rule of **allow all** will be applied.

To define firewall rule sets, use the **firewall** command (see page 315).

interfaces serial ppp vif firewall

Applies named firewall instances (packet-filtering rule sets) to a Point-to-Point Protocol–encapsulated serial interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

Command Mode

Configuration mode.

Syntax

`set interfaces serial` *name* `ppp vif` *name* `firewall …`	Use **set** to specify the firewall rule sets to be applied to a Point-to-Point Protocol–encapsulated serial interface.
`delete interfaces serial` *name* `ppp vif` *name* `firewall …`	Use **delete** to remove a packet filter (or all packet filters) from the vif configuration node of a Point-to-Point Protocol–encapsulated serial interface.

Configuration Statement

```
interfaces {
   serial wan0..wan9 {
      vif 1 {
         ppp {
            firewall {
               in {
                  name: text
               }
               out {
                  name: text
               }
               local {
                  name: text
               }
            }
         }
      }
   }
}
```

Parameters

serial	The serial interface you are configuring: one of **wan0** through **wan9**. The interface must already have been defined.
vif	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.
cisco-hdlc	Identifies this interface as a Point-to-Point Protocol–encapsulated interface.
firewall	Applies a named firewall rule set to the interface. One rule set can be applied to each of the following: • Inbound packets • Outbound packets • Packets destined for this router itself
in	The specified rule set will be applied to packets entering this interface.
name	Applies the specified rule set to packets entering this interface.
out	The specified rule set will be applied to packets leaving this interface.
name	Applies the specified rule set to packets leaving this interface.
local	The specified rule set will be applied to packets destined for this router itself.
name	Applies the specified rule set to packets destined for this router.

Usage Guidelines

Use this statement to apply the rule set defined for a firewall instance to the vif of a Point-to-Point Protocol–encapsulated serial interface.

A firewall has no effect on traffic traversing the router or destined to the router until it has been applied to an interface or a vif.

To use the firewall feature, you define a firewall rule set as a named firewall instance, using the **firewall** command (see page 315). You then apply the firewall instance to interfaces and/or vifs using this statement or one like it. Once applied, the instance acts as a packet filter.

The firewall instance will filter packets in one of the following ways, depending on what you specify when you apply it to the interface:

- **in.** If you apply the rule set as **in**, the firewall will filter packets entering the interface.
- **out.** If you apply the rule set as **out**, the firewall will filter packets leaving the interface.
- **local.** If you apply the rule set as **local**, the firewall will filter packets destined for this router itself.

For each interface, you can apply up to three firewall instances: one firewall **in** instance, one firewall **out** instance, and one firewall **local** instance.

Make sure the firewall instance you apply to an interface is already defined, or you may experience unintended results. If you apply a firewall instance that does not exist to an interface, the implicit firewall rule of **allow all** will be applied.

To define firewall rule sets, use the **firewall** command (see page 315).

show firewall

Shows the list of rules associated with a specific firewall instance.

Command Mode

Operational mode.

Syntax

```
show firewall rule-set [no-resolve |
                        statistics |
                        detail [rule rule-num]]
```

Parameters

rule-set	The name of the firewall rule set.
no-resolve	Do not attempt to resolve IP addresses into domain names. Use this option to reduce the amount of time it takes for this command to return a result.
statistics	Displays counters for the specified firewall rule set.
detail	Displays detailed information about the specified rule set.
rule	Displays detailed information about the specified individual rule.

Usage Guidelines

Use this command to display the rules associated with a specific firewall rule set.

When this command is used without the **no-resolve** option, the router will attempt to resolve all IP addresses in the configuration to DNS names. This can significantly increase the amount of time required for the command to return a result. To minimize the delay, use the **no-resolve** option.

The **statistics** option displays the current values of all counters associated with the specified firewall rule set.

Chapter 17: User Authentication

This chapter lists the commands available for setting up user accounts and user authentication.

This chapter contains the following commands.

Command	Mode	Description
`system login`	Configuration	Allows you to create user accounts and set up user authentication.
`show users`	Operational	Shows which users are currently logged on.

system login

Allows you to create user accounts and set up user authentication.

Command Mode

Configuration mode.

Syntax

`set system login …`	Use **set** to create the **login** configuration node, or to change user authentication configuration. Note that you cannot use set to change a user name or the IP address of a RADIUS server, as these are identifiers of configuration nodes. To change this information, delete the configuration node and create a new one with the correct identifier.
`delete system login …`	Use **delete** to delete a user or a RADIUS server, or to delete the **login** configuration node altogether. Note that the **login** configuration node is a mandatory node, so deleting this node simply resets it to default values.

Configuration Statement

```
system {
   login {
      user text {
         full-name: text
         authentication {
            plaintext-password: text
            encrypted-password: text
         }
      }
      radius-server ipv4 {
         port: 1–65534
         secret: text
         timeout: 1–4294967296
      }
   }
}
```

Parameters

user	Multi-node. Creates a user account, or changes user information. The user name must be unique within the router. The string may be up to 32 characters, which may include alphanumeric characters and hyphens. You can define multiple users to be authenticated using the router's internal mechanism, by creating multiple **user** configuration nodes.
full-name	The complete name of the user. This may include alphanumeric characters, space, and hyphen. Strings that include spaces must be enclosed in double quotes.
authentication	Specifies the authentication method(s) that the user can use to log on to the router. You can assign more than one authentication method to a given user.
plaintext-password	The user's password as you enter it in plain text. The system encrypts the plain-text password using Message Digest 5 and stores the encrypted version internally. When you display user information, you see the encrypted password, shown as the value of the encrypted-password attribute.
encrypted-password	The encrypted version of the plain-text password that was specified for this user. The password is specified in plain-text as the value of the **plaintext-password** attribute, then encrypted using Message Digest 5 and the encrypted version is stored internally. When you display user information, you see the encrypted password, shown as the value of this attribute.
radius-server	Multi-node. The IP address of a remote authentication server running the RADIUS protocol. This server can be used to authenticate multiple users. You can define multiple RADIUS servers, by creating multiple **radius-server** configuration nodes.
port	The port for RADIUS traffic. The default is1812.

secret	Mandatory. A password, as recorded on the RADIUS server. This may include alphanumeric characters, space, and special characters. Strings that include spaces must be enclosed in double quotes.
timeout	Optional. A time period in seconds after which the next RADIUS server should be queried. If no other RADIUS servers remain to be queried, the login request fails. The default is 2.

Usage Guidelines

Use this statement to configure user authentication on the router.

The Vyatta OFR supports either of the following options for user account management:

- A local user database ("login" authentication).
- RADIUS authentication server

The system creates two login user accounts by default: user **vyatta** and user **root**. The user account **vyatta** can be deleted, but the user account **root** is protected and cannot be deleted. The default password for each is **vyatta**.

By default, users are authenticated first using the local user database ("login" authentication). If this fails, the system looks for a configured RADIUS server. If found, the router queries the RADIUS server using the supplied RADIUS secret. After the query is validated, the server authenticates the user from information in its database.

You supply login user passwords and RADIUS secrets in plain text. After configuration is committed, the system encrypts them and stores the encrypted version internally. When you display user configuration, only the encrypted version of the password or secret is displayed.

The argument for each of the login class sub-statements is a regular expression as defined in POSIX 1003.2. If the regular expression contains any spaces, operators, or wildcard characters, you must enclose it in double quotes.

show users

Shows which users are currently logged on.

Command Mode

Operational mode.

Syntax

```
show users
```

Parameters

None.

Usage Guidelines

Use this command what users are currently logged on to the router.

Chapter 18: Logging

This chapter lists the commands used for system logging.

This chapter contains the following commands.

Command	Mode	Description
`delete log file`	Operational	Deletes the specified log file, including all its archive files.
`show log`	Operational	Displays the contents of the specified log file.
`show log directory`	Operational	Displays a list of files in the logging directory.
`system syslog`	Configuration	Allows you to configure system logging on the router.

delete log file

Deletes the specified log file, including all its archive files.

Command Mode

Operational mode.

Syntax

`delete log file` *`file-name`*

Parameters

file-name	Deletes the specified user-defined file in the **/var/log** directory, including all its archive files.

Usage Guidelines

Use this command to delete a log file.

Log files are created in the **/var/log** directory. When you issue this command, the specified file and all associated archive files are deleted from this directory.

Note that deleting the log file does not stop the system from logging events. If you use this command while the system is logging events, old log events will be deleted, but events after the delete operation will be recorded in the new file. To delete the file altogether, first disable logging to the file using the **system syslog** command (see page 350), and then delete it.

show log

Displays the contents of the specified log file.

Command Mode

Operational mode.

Syntax

`show log [file` *`file-name`*`]`

Parameters

file *file-name*	Displays the contents of the specified file in the **/var/log** directory.

Usage Guidelines

Use this command to view the contents of a log file.

When used with no option, this command displays the contents of the main log file (**/var/log/messages**), which is the default file to which the system writes syslog messages.

When the **file** *file-name* is specified, this command displays the contents of the specified user-defined file in the **/var/log/user** directory.

show log directory

Displays a list of files in the logging directory.

Command Mode

Operational mode.

Syntax

```
show log directory
```

Parameters

None.

Usage Guidelines

Use this command to list the user-defined log files currently stored in the **/log/var/user** directory.

This the directory where user-defined log files are stored. Syslog messages are written either to the messages file, or to a file with a name specified during syslog configuration using the **system syslog** command (see page 350).

system syslog

Allows you to configure system logging on the router.

Command Mode

Configuration mode.

Syntax

`set system syslog …`	Use **set** to create the **syslog** configuration node, or to modify system logging configuration. Note that you cannot use **set** to change the name of a file, a host, or a user, as these are identifiers of configuration nodes. To change this information, delete the old node and recreate it using the correct identifier.
`delete system syslog …`	Use **delete** to delete a log destination, or to delete the **syslog** configuration node altogether.

Configuration Statement

```
system {
    syslog {
        console {
            facility: text {
                level: text
            }
        }
        file: text {
            facility: text {
                level: text
            }
            archive {
                files: 1-4294967296
                size: 1-4294967296
            }
        }
        host: text {
            facility: text {
                level: text
            }
        }
        user: text {
```

```
            facility: text {
               level: text
            }
         }
      }
   }
```

Parameters

console	Sends the specified log messages to the console.
facility	Multi-node. The syslog facility for which log messages are being collected. Please see the Usage Guidelines for supported facilities. You can send the log messages of multiple facilities to the console by creating multiple **facility** configuration nodes within the **console** node.
level	The minimum severity of log message that will be reported. Supported values are **emerg**, **alert**, **crit**, **err**, **warning**, **notice**, **info**, and **debug**. By default, messages of **err** severity are logged to the console. Please see the Usage Guidelines for the meanings of these levels.
file	Multi-node. Writes the specified log messages to the specified file in the **/var/log** directory in the local file system. File names can include numbers, letters, and hyphens. You can send log messages to multiple files by creating multiple **file** configuration nodes.
facility	Multi-node. The router component for which log messages are being collected. Please see the Usage Guidelines for supported logging facilities. You can send the log messages of multiple facilities to the console by creating multiple **facility** configuration nodes within the **file** node.
level	The minimum severity of log message that will be reported. Supported values are **emerg**, **alert**, **crit**, **err**, **warning**, **notice**, **info**, **debug**. By default, messages of **warning** severity are logged to file. Please see the Usage Guidelines for the meanings of these levels.
archive	Changes the settings for log file archiving for the specified file.
files	Sets the maximum number of archive files that will be maintained for this log file. After the maximum has been reached, logs will be rotated with the oldest file overwritten. The default is 5.

size	Sets the maximum size in bytes of archive files for this log file. After the maximum has been reached, the file will be closed and archived in compressed format. The default is 0, which means unlimited.
host	Multi-node. Sends the specified log messages to a host. The host must be running the syslog protocol. Host names can include numbers, letters, and hyphens ("-"). You can send log messages to multiple hosts by creating multiple **host** configuration nodes.
facility	Multi-node. The router component for which log messages are being collected. Please see the Usage Guidelines for supported facilities. You can send the log messages of multiple facilities to the console by creating multiple **facility** configuration nodes within the **host** node.
level	The minimum severity of log message that will be reported. Supported values are **emerg**, **alert**, **crit**, **err**, **warning**, **notice**, **info**, **debug**. By default, messages of **err** severity are logged to hosts. Please see the Usage Guidelines for the meanings of these levels.

Usage Guidelines

Use this command to configure the router's system logging utility (syslog).

Using this command, you can set the destinations for log messages from different routing components (facilities) and specify what severity of message should be reported for each facility.

Log messages generated by the Vyatta OFR router will be associated with one of the following levels of severity.

Table 18-1 Syslog message severities

Severity	Meaning
emerg	Emergency. A general system failure or other serious failure has occurred, such that the router is unusable.
alert	Alert. Immediate action is required to prevent the system from becoming unusable—for example, because a network link has failed, or the database has become compromised.
crit	Critical. A critical condition exists, such as resource exhaustion—for example, the system is out of memory, CPU processing thresholds are being exceeded, or a hardware failure has occurred.

Table 18-1 Syslog message severities

err	Error. An error condition has occurred, such as a failed system call. However, the system is still functioning.
warning	Warning. An event has occurred that has the potential to cause an error, such as invalid parameters being passed to a function. This situation should be monitored.
notice	Notice. A normal but significant event has occurred, such as an unexpected event. It is not an error, but could potentially require attention.
info	Informational. Normal events of interest are being reported as they occur.
debug	Debug level. Trace-level information is being provided.

The Vyatta OFR supports standard syslog facilities. These are as follows:

Table 18-2 Syslog facilities

Facility	Description
auth	Authentication and authorization
authpriv	Non-system authorization
cron	Cron daemon
daemon	System daemons
kernel	Kernel
lpr	Line printer spooler
mail	Mail subsystem
mark	Timestamp
news	USENET subsystem
security	Security subsystem
syslog	System logging
user	Application processes
uucp	UUCP subsystem
local0	Local facility 0
local1	Local facility 1

Table 18-2 Syslog facilities

local2	Local facility 2
local3	Local facility 3
local4	Local facility 4
local5	Local facility 5
local6	Local facility 6
local7	Local facility 7
*	All facilities excluding “mark”

Messages are written either to the main log file (the default) or to a file that you specify. The main log file is created in the **/var/log** directory, to the **messages** file. User-defined log files are written to the **/var/log/user** directory.

NOTE *Currently, log severities can only be changed for user-defined files. The main log file at /var/log/messages always uses log severity **warning**.*

The router uses standard UNIX log rotation to prevent the file system from filling up with log files. When log messages are written to a file, the system will write up to 500 KB of log messages into the file *logfile*, where *logfile* is either the system-defined **messages** file, or a name you have assigned to the file. When *logfile* reaches its maximum size, the system closes it and compresses it into an archive file. The archive file is named *logfile***.0.gz**.

At this point, the logging utility opens a new *logfile* file and begins to write system messages to it. When the new log file is full, the first archive file is renamed *logfile***.1.gz** and the new archive file is named *logfile***.0.gz**. The system archives log files in this way until a maximum number of log files exists. By default, this is five (that is, up to *logfile***.4.gz**), where *logfile***.0.gz** always represents the most recent file. After this, the oldest log archive file is deleted as it is overwritten by the next oldest file.

To change the properties of log file archiving, configure the **system syslog archive** node:

- Use the **size** parameter to specify the maximum size of the log file.
- Use the **files** parameter to specify the maximum number of archive files to be maintained.

Chapter 19: SNMP

This chapter lists the commands for setting up the Simple Network Management Protocol on the Vyatta OFR.

This chapter contains the following commands.

Command	Mode	Description
`protocols snmp`	Configuration	Defines SNMP community and trap information for the router.
`clear snmp statistics`	Operational	Resets all SNMP statistics on the router to zero.
`show snmp`	Operational	Displays information about SNMP configuration.
`show snmp statistics`	Operational	Displays packet-level SNMP counters and statistics.

clear snmp statistics

Resets all SNMP statistics on the router to zero.

Command Mode

Operational mode.

Syntax

```
clear snmp [statistics]
```

Parameters

None.

Usage Guidelines

Use this command to reset all SNMP counters and statistics.

protocols snmp

Defines SNMP community and trap information for the router.

Command Mode

Configuration mode.

Syntax

`set protocols snmp …`	Use **set** to create the **snmp** configuration node, or to modify SNMP configuration. Note that you cannot use **set** to change the identifier of a configuration node. To change this information, delete the old node and create a new one with the correct identifier.
`delete protocols snmp …`	Use **delete** to delete SNMP configuration.

Configuration Statement

```
protocols {
    snmp {
        community: text {
            client: ipv4 {}
            network: ipv4net {}
            authorization: [ro|rw]
        }
        contact: text
        description: text
        location: text
        trap-target: ipv4 {}
        mibs {
            mib-module: text {
                abs-path:text
                mib-index:int
            }
        }
    }
}
```

Parameters

community	Optional. Multi-node. Defines an SNMP community. The argument is the community string to be used to authorize SNMP managers making requests of this router. Letters, numbers, and hyphens are supported. You can define more than one community by creating multiple **community** configuration nodes. By default, no community string is defined.
authorization	Optional. Specifies the privileges this community will have. Supported values are as follows: **ro**: This community can view router information, but not change it. **rw**: This community has read-write privileges. The default authorization privilege is **ro**. Deleting the **authorization** statement resets the privilege level to the default (**ro**).
client	Optional. Multi-node. The SNMP clients in this community that are authorized to access the server. You can define more than one client by creating the **client** configuration node multiple times. If no client or network is defined, then any client presenting the correct community string will have read-only access to the router. If any client or network is defined then only explicitly listed clients and/or networks will have access to the router.
network	Optional. Multi-node. The network of SNMP clients in this community that are authorized to access the server. You can define more than one network by creating the **network** configuration node multiple times. If no client or network is defined, then any client presenting the correct community string will have read-only access to the router. If any client or network is defined then only explicitly listed clients and/or networks will have access to the router.
contact	Optional. Records contact information for this SNMP community. This is stored as MIB-2 system information in the **snmpd.conf** configuration file. Letters, numbers, and hyphens are supported.
description	Optional. Records a brief description for this SNMP community. This is stored as MIB-2 system information in the **snmpd.conf** configuration file. Letters, numbers, and hyphens are supported.

location	Optional. Records the location of this SNMP community. This is stored as MIB-2 system information in the **snmpd.conf** configuration file. Letters, numbers, and hyphens are supported.
trap-target	Optional. Multi-node. The IP address of the destination for SNMP traps. You can specify multiple destinations for SNMP traps by creating multiple **trap-target** configuration nodes. Or, you can enter a space-separated list of IP addresses.
mibs	Specifies information about included MIB modules.
mib-module	Optional. Multi-node. Allows you to add a MIB module. The argument is the MIB module file name, without the file extension. You can add multiple MIB modules by creating multiple **mib-module** configuration nodes.
abs-path	The absolute path to the MIB module.
mib-index	This is for internal use only. The default is “0”.

Usage Guidelines

Use this statement to specify information about which SNMP communities this router should respond to, about SNMP MIBs that should be loaded, about the router’s location and contact information, and about destinations for SNMP traps.

show snmp

Displays information about SNMP configuration.

Command Mode

Operational mode.

Syntax

```
show snmp
```

Parameters

None.

Usage Guidelines

Use this command to see how SNMP has been configured on the router.

show snmp statistics

Displays packet-level SNMP counters and statistics.

Command Mode

Operational mode.

Syntax

```
show snmp [statistics]
```

Parameters

None.

Usage Guidelines

Use this command to view packet-level counters and statistics for SNMP.

Table 19-1 shows the statistics that are maintained for received packets.

Table 19-1 SNMP statistics about received packets

Input—Information about received packets	
Packets	Total number of messages delivered to the SNMP entity from the transport service.
Bad versions	Total number of messages delivered to the SNMP entity that were for an unsupported SNMP version.
Bad community names	Total number of messages delivered to the SNMP entity that used an SNMP community name not known to the entity.
Bad community uses	Total number of messages delivered to the SNMP entity that represented an SNMP operation that was not allowed by the SNMP community named in the message.
ASN parse errors	Total number of ASN.1 or BER errors encountered by the SNMP entity when decoding received SNMP messages.
Too bigs	Total number of SNMP PDUs delivered to the SNMP entity with an error status field of tooBig.
No such names	Total number of SNMP PDUs delivered to the SNMP entity with an error status field of noSuchName.
Bad value	Total number of SNMP PDUs delivered to the SNMP entity with an error status field of badValue.

Table 19-1 SNMP statistics about received packets

Input—Information about received packets	
Read onlys	Total number of valid SNMP PDUs delivered to the SNMP entity with an error status field of readOnly. Only incorrect implementations of SNMP generate this error.
General errors	Total number of SNMP PDUs delivered to the SNMP entity with an error status field of genErr.
Total requests varbinds	Total number of MIB objects retrieved successfully by the SNMP entity as a result of receiving valid SNMP GetRequest and GetNext PDUs.
Total set varbinds	Total number of MIB objects modified successfully by the SNMP entity as a result of receiving valid SNMP SetRequest PDUs.
Get requests	Total number of SNMP GetRequest PDUs that have been accepted and processed by the SNMP entity.
Get nexts	Total number of SNMP GetNext PDUs that have been accepted and processed by the SNMP entity.
Set requests	Total number of SNMP SetRequest PDUs that have been accepted and processed by the SNMP entity.
Get responses	Total number of SNMP GetResponse PDUs that have been accepted and processed by the SNMP entity.
Traps	Total number of SNMP traps generated by the SNMP entity.

Table 19-2 shows the statistics that are maintained for transmitted packets.

Table 19-2 SNMP statistics about transmitted packets

Output—-Information about transmitted packets	
Packets	Total number of messages passed from the SNMP entity to the transport service.
Too bigs	Total number of SNMP PDUs generated by the SNMP entity with an error status field of tooBig.
No such names	Total number of SNMP PDUs delivered to the SNMP entity with an error status field of noSuchName.
Bad values	Total number of SNMP PDUs generated by the SNMP entity with an error status field of badValue.
General errors	Total number of SNMP PDUs generated the SNMP entity with an error status field of genErr.

Table 19-2 SNMP statistics about transmitted packets

Output—-Information about transmitted packets	
Get requests	Total number of SNMP GetRequest PDUs generated by the SNMP entity.
Get nexts	Total number of SNMP GetNext PDUs generated by the SNMP entity.
Set requests	Total number of SNMP SetRequest PDUs generated by the SNMP entity.
Get responses	Total number of SNMP GetResponse PDUs generated by the SNMP entity.
Traps	Total number of SNMP traps generated by the SNMP entity.

Examples

Example 19-1 shows sample output for the **show snmp statistics** command:

Example 19-1 “show snmp statistics”: Viewing SNMP statistics

```
root@vyatta> show snmp statistics
SNMP statistics:
  Input:
    Packets: 246213, Bad versions: 12, Bad community names: 12,
    Bad community uses: 0, ASN parse errors: 96,
    Too bigs: 0, No such names: 0, Bad values: 0,
    Read onlys: 0, General errors: 0,
    Total request varbinds: 227084, Total set varbinds: 67,
    Get requests: 44942, Get nexts: 190371, Set requests: 10712,
    Get responses: 0, Traps: 0,
  Output:
    Packets: 246093, Too bigs: 0, No such names: 31561,
    Bad values: 0, General errors: 2,
    Get requests: 0, Get nexts: 0, Set requests: 0,
    Get responses: 246025, Traps: 0
```

Chapter 20: Diagnostics and Debugging

This chapter lists supported commands that can be used for diagnostics and debugging.

This chapter contains the following commands.

Command	Mode	Description
`ping`	Operational	Sends ICMP ECHO_REQUEST packets to IPv4 network hosts.
`ping6`	Operational	Sends ICMP ECHO_REQUEST packets to IPv6 network hosts.
`traceroute`	Operational	Displays the route packets take to an IPv4 network host.
`traceroute6`	Operational	Displays the route packets take to an IPv6 network host.

See also the following commands in other chapters.

`reboot`	Operational	Reboots the router. *See page 38.*
`show system boot-messages`	Operational	Displays boot messages generated by the kernel. *See page 54.*
`show system connections`	Operational	Displays active network connections on the system. *See page 56.*
`show system kernel-messages`	Operational	Displays messages in the kernel ring buffer. *See page 58.*
`show system memory`	Operational	Displays system memory usage. *See page 60.*
`show system storage`	Operational	Displays system file system usage and available storage space. *See page 63.*
`show tech-support`	Operational	Provides a consolidated report of system information. *See page 64.*
`show version`	Operational	Displays information about the version of router software. *See page 66.*

ping

Sends ICMP ECHO_REQUEST packets to IPv4 network hosts.

Syntax

`ping` *host* `[-c` *count*`] [-i` *interval*`] [-s` *packetsize*`] [-t` *ttl*`] [-w` *timeout*`] [-M` *hint*`]`

Parameters

host	The host being pinged. Can be specified either as name (if DNS is being used on the network) or as an IPv4 address.
-c *count*	Stop after sending (and receiving) *count* ECHO_RESPONSE packets.
-i *interval*	The time in seconds to wait before sending the next packet.
-t *ttl*	Sets the IP time to live value in the packets. The range is 1 to 255. The default is 64.
-s *packetsize*	Specifies the number of data bytes to be sent.
-w *wait*	Sets the in seconds to wait for a response.
-M *hint*	Selects the path MTU Discovery strategy. The default hint is "'do' set the DF flag".

Usage Guidelines

The `ping` command is used to test whether a network host is reachable or not.

The `ping` command uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway. ECHO_REQUEST datagrams (pings) have an IP and ICMP header, followed by a "`struct timeval`" and then an arbitrary number of pad bytes used to fill out the packet.

To interrupt the ping command, press **<Ctrl>+c**.

When using ping for fault isolation, it should first be run on the local host, to verify that the local network interface is up and running. Then, hosts and gateways further and further away should be "pinged". Round-trip times and packet loss statistics are computed.

If duplicate packets are received, they are not included in the packet loss calculation, although the round-trip time of these packets is used in calculating the minimum/average/maximum round-trip time numbers. When the specified number of packets have been sent (and received) or if the program is terminated, a brief summary is displayed.

Examples

Example 20-1 shows sample output of the `ping` command:

Example 20-1 Sample output of “ping”

```
root@vyatta> ping 10.3.0.2
PING 10.3.0.2 (10.3.0.2): 56 data bytes
64 bytes from 10.3.0.2: icmp seq=0 ttl=64 time=0.281 ms
64 bytes from 10.3.0.2: icmp seq=1 ttl=64 time=0.244 ms
64 bytes from 10.3.0.2: icmp seq=2 ttl=64 time=0.302 ms
64 bytes from 10.3.0.2: icmp seq=3 ttl=64 time=0.275 ms
Command interrupted!
```

ping6

Sends ICMP ECHO_REQUEST packets to IPv6 network hosts.

Syntax

`ping` *host* `[-c` *count*`] [-i` *interval*`] [-s` *packetsize*`] [-t` *ttl*`] [-w` *timeout*`]`
`[-M` *hint*`]`

Parameters

host	The host being pinged. Can be specified either as name (if DNS is being used on the network) or as an IPv6 address.
-c *count*	Stop after sending (and receiving) *count* ECHO_RESPONSE packets.
-i *interval*	The time in seconds to wait before sending the next packet.
-t *ttl*	Sets the IP time to live value in the packets. The range is 1 to 255. The default is 64.
-s *packetsize*	Specifies the number of data bytes to be sent.
-w *wait*	Sets the in seconds to wait for a response.
-M *hint*	Selects the path MTU Discovery strategy. The default hint is "'do' set the DF flag".

Usage Guidelines

The `ping` command is used to test whether an IPv6 network host is reachable or not.

The `ping` command uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway. ECHO_REQUEST datagrams (pings) have an IP and ICMP header, followed by a "`struct timeval`" and then an arbitrary number of pad bytes used to fill out the packet.

To interrupt the ping command, press **<Ctrl>+c**.

When using ping for fault isolation, it should first be run on the local host, to verify that the local network interface is up and running. Then, hosts and gateways further and further away should be "pinged". Round-trip times and packet loss statistics are computed.

If duplicate packets are received, they are not included in the packet loss calculation, although the round-trip time of these packets is used in calculating the minimum/average/maximum round-trip time numbers. When the specified number of packets have been sent (and received) or if the program is terminated, a brief summary is displayed.

Examples

Example 20-1 shows sample output of the **ping6** command:

Example 20-2 Sample output of “ping6”

```
root@vyatta> ping6 ::10.4.1.1
PING 10.3.0.2 (::10.4.1.1): 56 data bytes
64 bytes from ::10.4.1.1: icmp seq=0 ttl=64 time=0.281 ms
64 bytes from ::10.4.1.1: icmp seq=1 ttl=64 time=0.244 ms
64 bytes from ::10.4.1.1: icmp seq=2 ttl=64 time=0.302 ms
64 bytes from ::10.4.1.1: icmp seq=3 ttl=64 time=0.275 ms
Command interrupted!
root@vyatta>
```

traceroute

Displays the route packets take to an IPv4 network host.

Syntax

`traceroute` *host*

Parameters

host	The host that is the destination for the packets. Can be specified either as name (if DNS is being used on the network) or as an IPv4 address.

Usage Guidelines

Traceroute utilizes the IP protocol time to live (“ttl”) field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to some host to track the route a set of packets follows.It attempts to trace the route an IP packet would follow to some internet host by launching UDP probe packets with a small ttl then listening for an ICMP “time exceeded” reply from a gateway.

traceroute6

Displays the route packets take to an IPv6 network host.

Syntax

`traceroute` *host*

Parameters

host	The host that is the destination for the packets. Can be specified either as name (if DNS is being used on the network) or as an IPv6 address.

Usage Guidelines

Traceroute utilizes the IP protocol time to live ("ttl") field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to some host to track the route a set of packets follows.It attempts to trace the route an IP packet would follow to some internet host by launching UDP probe packets with a small ttl then listening for an ICMP "time exceeded" reply from a gateway.

Chapter 21: Software Upgrades

This chapter lists commands for using the Vyatta OFR’s software upgrade mechanism.

This chapter contains the following commands.

Command	Mode	Description
`delete package`	Operational	Removes one or more previously installed software components from the system.
`install package`	Operational	Installs one or more specific packages onto the router.
`show package info`	Operational	Displays information about packages that are available in the software repository.
`show package installed`	Operational	Lists software packages that have already been installed.
`show package statistics`	Operational	Displays statistics bout update packages residing on your system.
`system package`	Configuration	Specifies the information needed for automatic software updates.
`update package`	Operational	Upgrades installed packages.
`update package-list`	Operational	Updates the list of packages available to install.

delete package

Removes one or more previously installed software components from the system.

Command Mode

Operational mode.

Syntax

`delete package *pkg-name* [*pkg-name* ...]`

Parameters

pkg-name	Mandatory. The name of a package. You can specify more than one package using a space-separated list.

Usage Guidelines

Use this command to remove previously installed software packages from the system.

All packages matching the specified package name(s) are removed. You must supply at least one package name.

If there are packages that depend on the package being removed, the system removes all dependent packages. You cannot remove a package without removing packages that depend on it.

Package removal may take some time to complete, and the system displays a progress indicator during removal. You can cancel package removal at any time, by pressing <Ctrl>-c.

install package

Installs one or more specific packages onto the router.

Command Mode

Operational mode.

Syntax

`install package` *`pkg-name`* `[`*`pkg-name ...`*`]`

Parameters

pkg-name	At least one package name must be specified. You can specify more than one package using a space-separated list.

Usage Guidelines

Use this command to install software packages onto the router.

All packages matching the specified package name(s) are downloaded, and then installed. You must supply at least one package name. If a package matching the specified package name is already installed, the system will report an error.

The system will retrieve the most recent version of the specified package from the software archive. If this package depends on another package, the system will resolve the dependencies and install any other required packages.

Package installation may take some time to complete, and the system displays a progress indicator during installation. You can cancel installation at any time, by pressing <Ctrl>-c.

show package info

Displays information about packages that are available in the software repository.

Command Mode

Operational mode.

Syntax

`show package info pkg-name [pkg-name ...] |`

Parameters

pkg-name	Shows detailed information about all packages matching the specified package name. You can specify more than one package in a space-separated list.

Usage Guidelines

Use this command to display information about software packages available for upgrading the router software.

The router maintains a list packages that are available in all configured repositories; you can force this list to synchronize with the repository using the **update package-list** command (see page 383).

show package installed

Lists software packages that have already been installed.

Command Mode

Operational mode.

Syntax

```
show package installed [pkg-name [pkg-name ...]] |
```

Parameters

pkg-name	Show information for just the specified installed package. You can specify more than one package in a space-separated list.

Usage Guidelines

Use this command to display information about software packages you have already installed into the system.

show package statistics

Displays statistics bout update packages residing on your system.

Command Mode

Operational mode.

Syntax

```
show package statistics
```

Parameters

None.

Usage Guidelines

Use this command to display information about software packages residing on your system.

The information displayed includes the number of packages, the number of dependencies between packages, and so on.

system package

Specifies the information needed for automatic software updates.

Command Mode

Configuration mode.

Syntax

`set system package repository` *`text`* …	Creates or modifies a software update repository (location).
`delete package repository` *`text`* …	Deletes the specified software update repository (location).

Configuration Statement

```
system {
   package {
      repository: text {
         description: text
         url: text
         component: text
      }
   }
}
```

Parameters

repository	Multi-node. The version number of the software. For example, `repository 1.1` You can define more than one software repository by creating multiple **repository** nodes.
description	A brief description for the repository.
url	Mandatory. The full URL of the server hosting the software repository, including the path if required.

component	Multi-node. The repository component names. You can configure more than one component within a repository by creating multiple **component** nodes. The stock components are **main** and **security**.

Usage Guidelines

Use this command to specify the information needed to obtain software updates from the Vyatta software archive.

Vyatta OFR packages are stored in the Vyatta software repository. Access to this repository is available with a support contract.

update package

Upgrades installed packages.

Command Mode

Operational mode.

Syntax

```
update package [pkg-name [pkg-name ...]
```

Parameters

pkg-name	Upgrades only the specified package, plus any dependencies. You can specify more than one package using a space-separated list.

Usage Guidelines

Use this command to upgrade your system software.

When used with no option, this command upgrades all installed packages, including any necessary dependencies.

Packages are downloaded from the repository and upgraded in the correct order. Packages are upgraded to the most recent version available in the repository, provided all dependencies can be satisfied. Packages for which dependencies cannot be satisfied, or that have conflicts with installed software, are not "kept back" and not installed.

Before running this command, you should use the **show package info** command to confirm the complete list of packages that will be upgraded.

update package-list

Updates the list of packages available to install.

Command Mode

Operational mode.

Syntax

```
update package-list
```

Parameters

None.

Usage Guidelines

Use this command to update the list of packages that are available in the repository.

The router maintains its own list of available packages; issuing this command synchronizes the router’s list with the configured software repository.

Updating the package list may take some time to complete, especially if there are many packages or your connection to the repository is slow. You can cancel the update at any time, by pressing <Ctrl>-c.

Appendix A: ICMP Types

This appendix lists the ICMP types defined by the Internet Assigned Numbers Authority (IANA).

The Internet Assigned Numbers Authority (IANA) has developed a standard that maps a set of integers and standard literal strings onto ICMP types.Table A-1 lists the ICMP types defined by the IANA.

Table A-1 ICMP types

ICMP Type	Literal
0	echo-reply
3	unreachable
4	source-quench
5	redirect
6	alternate-address
8	echo
9	router-advertisement
10	router-solicitation
11	time-exceeded
12	parameter-problem
13	timestamp-reply
14	timestamp-request
15	information-request
16	information-reply
17	mask-request
18	mask-reply
31	conversion-error
32	mobile-redirect
33	where-are-you
34	i-am-here
35	mobile-regist-request
36	mobile-regist-response
37	domainname-request

Table A-1 ICMP types

ICMP Type	Literal
38	domainname-response
39	skip
40	photuris

Appendix A: Regular Expressions

This appendix describes the regular expressions that can be recognized by the Vyatta OFR.

The Vyatta OFR supports POSIX-style regular expressions.

POSIX expressions are an extension of standard UNIX regular expressions. A regular expression is a string representing a pattern that describes or matches a set of strings.

In these regular expressions, most characters (*literals*) match themselves and nothing else. For example, “a” matches “a”, “ab” matches “ab”, and so on. A small set of characters (*metacharacters*) carry special meaning. Table B-1 describes supported metacharacters.

Table B-1 Regular expression metacharacters

Metacharacter	Meaning
.	Matches any single character. Note that the dot does not match a newline character. For example: • .at matches “aat”, “bat”, “cat”, and so on.
[ ]	Matches any single character included within the brackets. You can also specify a range of characters using the hyphen. Individual characters can be mixed with ranges. For example: Examples: • [abc] matches either “a”, “b”, or “c”. It does not match “ab” or “abc”. • [a-d] matches “a”, “b”, “c”, or “d”. • [a-dqrs] matches “a”, “b”, “c”, “d”, “q”, “r”, or “s”, and so does [a-dq-s]. If you want to match the hyphen character itself (“-”), position it as either the very first or the very last character in the list; for example: • [-abc] or [abc-] matches “-”, “a”, “b”, or “c”. Otherwise, it is interpreted as a range separator. If you want to match the square brackets themselves, place the right (closing) square bracket first in the list, followed by the left (opening) square bracket, as follows: • [ ] [ab] matches “[”, “]”, “a”, or “b”.
[^]	Matches any single character that is NOT included within the brackets. Individual characters can be mixed with ranges. Examples: • [^abc] matches any single character OTHER than “a”, “b”, or “c”. • [^a-z] matches any single character that is not a lowercase letter. • [^] matches all expressions matching .at except “bat”.
\ (\)	Creates a “block” or sub-expression from the enclosed characters. For example: • \(at\) matches “at” only. • [pb]\(at\)h matches “path” and “bath”.
^	To be matched, the specified character or block must occur at the beginning of a line. • ^[hc]at matches “hat” and “cat”, but only at the beginning of a line.

Table B-1 Regular expression metacharacters

<table>
<tr><th>Metacharacter</th><th>Meaning</th></tr>
<tr><td>$</td><td>To be matched, the specified character or block must occur at the end of a line.
• [hc]at$ matches “hat” and “cat”, but only at the end of a line.
• ^$ matches blank lines.</td></tr>
<tr><td>*</td><td>Matches 0 or more instances of the preceding single character, for example:
• [hc]* matches “”, “h”, and “c”
• [hc]*at matches “”, “h”, “c”, “at”, “hat”, “cat”, “hcat”, “chat”, “hhat”, “ccat”, and so on.</td></tr>
<tr><td>+</td><td>Matches 1 or more instances of the preceding single character or block, for example:
• [hc]+ matches “h” and “c”, “hh”, “hc”, “cc”, “ch”, and so on, but not “”.
• [hc]+at matches “”, “h”, “c”, “hat”, “cat”, “hcat”, “chat”, “hhat”, “ccat”, and so on, but not “at”.</td></tr>
<tr><td>?</td><td>Matches 0 or 1 instances of the preceding single character or block, for example:
• [hc]? matches “”, “h” and “c”.
• [hc]?at matches “at”,“hat”, and “cat”.</td></tr>
<tr><td>|</td><td>Alternation operator. Matches either the expression before or the expression after the operator. For example:
• abc|def matches either “abc” or “def”.</td></tr>
<tr><td>\</td><td>The escape character. If a metacharacter is to be included as part of the search string, it must be escaped by preceding it with the backslash. This includes the backslash itself. For example:
• bat\. matches “bat.”.
• \\dev matches “\dev”.</td></tr>
</table>

To account for differences between the organization of character sets in different implementations, the POSIX standard defines a number of *classes* or categories of characters. Table B-2 lists POSIX classes.

Table B-2 POSIX classes

Class	Equivalent to:
[:upper:]	[A-Z] Upper case letters.
[:lower:]	[a-z] Lower case letters.
[:alpha:]	[A-Za-z] Upper and lower case letters.

Table B-2 POSIX classes

Class	Equivalent to:
[:digit:]	[0-9] Digits.
[:alnum:]	[A-Za-z0-9] Digits, and upper and lower case letters.
[:xdigit:]	[0-9A-Fa-f] Hexadecimal digits.
[:punct:]	[.,!?:...] Punctuation.
[:blank:]	[\t] Space and <Tab>.
[:space:]	[\t\n\r\f\v] Characters generating white space.
[:cntrl:]	Control characters.
[:graph:]	[^\t\n\r\f\v] Printed characters.
[:print:]	[^ \t\n\r\f\v] Printed characters and blank space.

Quick Guide to Configuration Statements

Use this section to quickly see the complete syntax of configuration statements.

The Vyatta OFR supports the following configuration statements:

- firewall
- interfaces
- multicast
- policy
- protocols
- rtrmgr
- service
- system

firewall

```
firewall {
   log-martians: [enable|disable]
   send-redirects: [enable|disable]
   receive-redirects: [enable|disable]
   ip-src-route: [enable|disable]
   broadcast-ping: [enable|disable]
   syn-cookies: [enable|disable]
   name: text {
      description: text
      rule: 1–1024 {
         protocol: [all|tcp|udp|icmp|igmp|ipencap|gre|esp|ah|
            ospf|pim|vrrp]
         icmp {
            type: text {
            code: text
         }
         state {
            established: [enable|disable]
            new: [enable|disable]
            related: [enable|disable]
            invalid: [enable|disable]
            }
         action: [accept|drop|reject]
         log: [enable|disable]
         source {
            address: ipv4
            network: ipv4net
            range {
               start: ipv4
               stop: ipv4
            }
            port-number: 1–65535
            port-name: [http|ftp|smtp|telnet|ssh|dns|snmp]
            port-range {
               start: 1–65535
               stop: 1–65535
            }
         }
         destination {
            address: ipv4
            network: ipv4net
            range {
               start: ipv4
               stop: ipv4
            }
```

```
                port-number: 1–65535
                port-name: [http|ftp|smtp|telnet|ssh|dns|snmp]
                port-range {
                   start: 1–65535
                   stop: 1–65535
                }
             }
          }
       }
    }
```

interfaces

```
interfaces {
   restore: [true|false]
   loopback: lo {
      description: text
      address: [ipv4|ipv6]{
         prefix-length: [0–32|0-128]
         broadcast: ipv4
         multicast-capable: [true|false]
         disable: [true|false]
      }
   }
   bridge br0..br9 {
      description: text
      disable: [true|false]
      aging: 1–4294967296
      stp: [true|false]
      priority: 1–4294967296
      forwarding-delay: 1–4294967296
      hello-time: 1–4294967296
      max-age: 1–4294967296
   }
   ethernet: eth0..eth23 {
      disable:[true|false]
      discard:[true|false]
      description:text
      mac: mac-addr
      mtu: 68–65535
      duplex: [full|half|auto]
      speed: [10|100|1000|auto]
      address: [ipv4 | ipv6]{
         prefix-length: [0–32|0-128]
         broadcast: ipv4
         multicast-capable: [true|false]
         disable: [true|false]
      }
      bridge-group {
         bridge: br0..br9
         cost: 1–4294967296
         priority: 1–4294967296
      }
      vrrp {
         vrrp-group: 1–255
         virtual-address: ipv4
         authentication:text
         advertise-interval: 1–255
```

```
            preempt:[true|false]
            priority: 1–255
         }
         firewall {
            in {
               name: text
            }
            out {
               name: text
            }
            local {
               name: text
            }
         }
         vif 1–4096 {
            disable:[true|false]
            address: [ipv4 | ipv6]{
               prefix-length: [0–32|0-128]
               broadcast: ipv4
               multicast-capable: [true|false]
               disable: [true|false]
            }
            bridge-group {
               bridge: br0..br9
               cost: 1–4294967296
               priority: 1–4294967296
            }
            vrrp {
               vrrp-group: 1–255
               virtual-address: ipv4
               authentication:text
               advertise-interval: 1–255
               preempt:[true|false]
               priority: 1–255
            }
            firewall {
               in {
                  name: text
               }
               out {
                  name: text
               }
               local {
                  name: text
               }
            }
      serial [wan0..wan9] {
```

```
encapsulation: [ppp|cisco-hdlc|frame-relay]
card-type: [sangoma-t1/e1|sangoma-t3]
description: text
t1-options {
   lbo: [0-110ft|110-220fr|220-330ft|330-440ft|440-550ft]
   timeslots {
      start: [1–24]
      stop: [1–24]
   }
   mtu: 8–8188
   clock: [internal|external]
}
e1-options {
   framing: [g704|g704-no-crc4|unframed]
   timeslots {
      start: [1–32]
      stop: [1–32]
   }
   mtu: 8–8188
   clock: [internal|external]
}
t3-options {
   framing: [c-bit|m13]
   line-coding: [ami|b8zs]
}
ppp {
   authentication {
      type: [none|chap|pap]
      user-id: text
      password: text
      system-name: text
   }
   vif 1 {
      address {
         local-address: ipv4
         prefix-length: 0–32
         remote-address: ipv4
      }
      description: text
      firewall {
         in {
            name: text
         }
         out {
            name: text
         }
         local {
```

```
                name: text
            }
        }
    }
}
cisco-hdlc {
    keepalives {
        require-rx: [enable|disable]
        timer: 10-60000
    }
    vif 1 {
        address {
            local-address: ipv4
            prefix-length: 0–32
            remote-address: ipv4
        }
        description: text
        firewall {
            in {
                name: text
            }
            out {
                name: text
            }
            local {
                name: text
            }
        }
    }
}
frame-relay {
    signaling: [auto|ansi|q933|lmi]
    signaling-options {
        n39ldte: 1–255
        n392dte: 1–100
        n393dte: 1–10
        t391dte: 5–30
    }
    vif [16..991] {
        address {
            local-address: ipv4
            prefix-length: 0–32
            remote-address: ipv4
        }
        description: text
        firewall {
            in {
```

```
            name: text
         }
         out {
            name: text
         }
         local {
            name: text
         }
      }
   }
}
}
}
```

multicast

```
multicast {
   mfea4 {
      disable:bool
      interface: eth0..eth23
      traceoptions {
         flag {
            all {
               disable:bool
            }
         }
      }
   }
   mfea6 {
      disable:bool
      interface: eth0..eth23
      traceoptions {
         flag {
            all {
               disable:bool
            }
         }
      }
   }
}
```

policy

```
policy {
   policy-statement: text {
      term: text {
         from {
            protocol: text
            network4: ipv4net
            network6: ipv6net
            network4-list: text
            network6-list: text
            prefix-length4: 0–32-range
            prefix-length6: 0–128-range
            nexthop4: ipv4-range
            nexthop6: ipv6-range
            as-path: text
            as-path-list: text
            community: text
            community-list: text
            neighbor: ipv4-range
            origin: [0|1|2]
            med: int-range
            localpref: int-range
            metric: 1–65535-range
            external: [type-1|type-2]
            tag: int-range
         }
         to {
            network4: ipv4net
            network6: ipv6net
            network4-list: text
            network6-list: text
            prefix-length4: 0–32-range
            prefix-length6: 0–128-range
            nexthop4: ipv4-range
            nexthop6: ipv6-range
            as-path: text
            as-path-list: text
            community: text
            neighbor: ipv4-range
            origin: int
            med: int-range
            localpref: int-range
            was-aggregated: bool
            metric: 1–65535-range
            external: [type-1|type-2]
            tag: int-range
```

```
        }
        then {
           action: [accept|reject]
           trace: int
           nexthop4: next-hop
           nexthop6: ipv6
           as-path-prepend: int
           as-path-expand: int
           community: text
           community-add: text
           community-del: text
           origin: int
           med: int
           med-remove: [true|false]
           localpref: int
           aggregate-prefix-len: int
           aggregate-brief-mode: int
           metric: 1–65535
           external: [type-1|type-2]
           tag: int
        }
     }
  }
  community-list: text {
     elements: text
  }
  community-list: text {
     elements: text
  }
  network6-list: text {
     elements: text
  }
}
```

protocols

```
protocols
   bgp {
      bgp-id: ipv4
      local-as: 1–65535
      route-reflector {
          cluster-id: ipv4
          disable: [true|false]
      }
      confederation {
          identifier: 1–4294967296
          disable: [true|false]
      }
      damping {
          half-life: 1–4294967296
          max-suppress: 1–4294967296
          reuse: 1–4294967296
          suppress: 1–4294967296
          disable: [true|false]
      }
      peer: text {
         peer-port: 1–4294967296
         local-port: 1–4294967296
         local-ip: text
         as: 1–65535
         next-hop: ipv4
         holdtime: 0,3–65535
         delay-open-time: 1–4294967296
         client: [true|false]
         confederation-member: [true|false]
         prefix-limit {
            maximum: 1–4294967296
            disable: [true|false]
         disable: [true|false]
         ipv4-unicast: [true|false]
         ipv4-multicast: [true|false]
         }
      traceoptions {
         flag {
            verbose {
               disable: [true|false]
            }
            all {
               disable: [true|false]
            }
            message-in {
```

```
                disable: [true|false]
            }
            message-out {
                disable: [true|false]
            }
            state-change {
                disable: [true|false]
            }
            policy-configuration {
                disable: [true|false]
            }
        }
        import: text
        export: text
    }
}
ospf4 {
    router-id: ipv4
    RFC1538Compatibility: [true|false]
    ip-router-alert: [true|false]
    traceoptions {
    flag {
        all {
            disable:[true|false]
        }
    }
    area: ipv4 {
        area-type:[normal|stub|nssa]
        default-lsa {
            disable:[true|false]
            metric: 1–4294967296
        }
        summaries {
            disable:[true|false]
        }
        area-range: ipv4net {
            advertise:[true|false]
        }
        virtual-link: ipv4 {
            transit-area: ipv4
            hello-interval:1–65535
            router-dead-interval: 1–4294967295
            retransmit-interval: 1–65535
            transit-delay:0–3600
            authentication {
                simple-password:text
                md5: 0–255 {
                    password: text
```

```
                    start-time: YYYY-MM-DD.HH:MM
                    end-time: YYYY-MM-DD.HH:MM
                    max-time-drift: 0–65534,65535
                }
            }
        }
        interface: text {
            link-type:[broadcast|p2p|p2m]
            address: ipv4 {
                priority:0–255
                hello-interval:1–65535
                router-dead-interval: 1–4294967296
                interface-cost:1–65535
                retransmit-interval: 1–65535
                transit-delay:0–3600
                authentication {
                    simple-password:text
                    md5: 0–255 {
                        password: text
                        start-time: YYYY-MM-DD.HH:MM
                        end-time: YYYY-MM-DD.HH:MM
                        max-time-drift: 0–65534,65535
                    }
                }
                passive: [true|false]
                neighbor: ipv4 {
                    router-id: ipv4
                }
                disable: [true|false]
            }
        }
    }
    import: text
    export: text
}
rip {
    interface: text {
        address: ipv4 {
            metric: 1–16
            horizon:
                [none|split-horizon|split-horizon-poison-reverse]
            disable: [true|false]
            passive: [true|false]
            accept-non-rip-requests: [true|false]
            accept-default-route: [true|false]
            advertise-default-route: [true|false]
            route-expiry-secs: 1–4294967296
            route-deletion-secs: 1–4294967296
```

```
                triggered-update-min-secs: 1–4294967296
                triggered-update-max-secs: 1–4294967296
                table-announce-min-secs: 1–4294967296
                table-announce-max-secs: 1–4294967296
                table-request-secs: 1–4294967296
                interpacket-delay-msecs: 1–4294967296
                authentication {
                    simple-password: text
                    md5: 0–255 {
                        password: text
                        start-time: YYYY-MM-DD.HH:MM
                        end-time: YYYY-MM-DD.HH:MM
                    }
                }
            }
        }
        import: text
        export: text
    }
    snmp {
        mib-module: text {
            abs-path:text
            mib-index:int
        }
        community: text {
            authorization: [ro|rw]
            client: ipv4 {}
        }
        contact: text
        description: text
        location: text
        trap-target: ipv4 {}
    }
    static {
        disable: [true|false]
        route: ipv4net {
            next-hop: ipv4
            metric: 1–65535
        }
        interface-route: ipv4net {
            next-hop-interface: text
            next-hop-router: ipv4
            metric: 1–65535
        }
        import: text
    }
}
```

rtrmgr

```
rtrmgr {
   config-directory: text
}
```

service

```
service {
   dhcp-server {
      name text {
         interface: eth0..eth23
         network-mask: 0–32
         start ipv4 {
            stop: ipv4
         }
         exclude: ipv4 {}
         static-mapping: text {
            ip-address: ipv4
            mac-address: macaddr
         }
         dns-server ipv4 {}
         default-router: ipv4
         wins-server ipv4 {}
         lease: 120–4294967296
         domain-name: text
         authoritative: [enable|disable]
      }
   }
   http {
      port: 1–65534
   }
   ssh {
      port: 1–65534
      protocol-version: [v1|v2|all]
   }
   telnet {
      port: 1–65534
   }
   nat {
      rule: 1–1024 {
         type: [source|destination]
         translation-type: [static|dynamic|masquerade]
         inbound-interface: text
         outbound-interface: text
         protocols: [tcp|udp|icmp|all]
         source {
            address: ipv4
            network: ipv4net
            port-number: 1–4294967296 {}
            port-name: [http|ftp|smtp|telnet|ssh|dns|snmp] {}
            port-range {
               start: 1–4294967296
```

```
                  stop: 1–4294967296
               }
            }
            destination {
               address: ipv4
               network: ipv4net
               port-number: 1–4294967296 {}
               port-name: [http|ftp|smtp|telnet|ssh|dns|snmp] {}
               port-range {
                  start: 1–4294967296
                  stop: 1–4294967296
               }
            inside-address {
               address: ipv4
               network: ipv4net
            }
            outside-address {
               address: ipv4
               network: ipv4net
               range {
                  start: ipv4
                  stop: ipv4
               }
            }
         }
      }
   }
```

system

```
system {
   disable: [true | false]
   host-name: text
   domain-name: text
   domain-search {
      domain: text [text ...]
   }
   name-server: ipv4 {}
   time-zone: text
   ntp-server: [ipv4|text] {}
   static-host-mapping {
      host-name: text {
         inet: ipv4
         alias: text {}
      }
   }
   login {
      user text {
         full-name: text
         authentication {
            plaintext-password: text
            encrypted-password: text
         }
      }
      radius-server ipv4 {
         port: 1–65534
         secret: text
         timeout: 1–4294967296
      }
   }
   syslog {
      console {
         facility: text {
            level: text
         }
      }
      file: text {
         facility: text {
            level: text
         }
         archive {
            files: 1–4294967296
            size: 1–4294967296
         }
      }
```

```
        host: text {
            facility: text {
                level: text
            }
        }
        user: text {
            facility: text {
                level: text
            }
        }
    }
    package {
        repository: text {
            description: text
            url: text
            component: text
        }
    }
}
```

Glossary

AS	*See* Autonomous System.
Autonomous System	A routing domain that is under one administrative authority, and which implements its own routing policies. A key concept in BGP.
BGP	Border Gateway Protocol.
Bootstrap Router	A PIM-SM router that chooses the RPs for a domain from amongst a set of candidate RPs.
BSR	*See* Bootstrap Router.
Candidate RP	A PIM-SM router that is configured to be a candidate to be an RP. The Bootstrap Router will then choose the RPs from the set of candidates.
Dynamic Route	A route learned from another router via a routing protocol such as RIP or BGP.
EGP	*See* Exterior Gateway Protocol.
Exterior Gateway Protocol	A routing protocol used to route between Autonomous Systems. The main example is BGP.
IGMP	Internet Group Management Protocol. *TBD*
IGP	*See* Interior Gateway Protocol.
Interior Gateway Protocol	A routing protocol used to route within an Autonomous System. Examples include RIP, OSPF and IS-IS.
MLD	Multicast Listener Discovery protocol. TBD
MRIB	*See* Multicast RIB.

Multicast RIB	The part of the RIB that holds multicast routes. These are not directly used for forwarding, but instead are used by multicast routing protocols such as PIM-SM to perform RPF checks when building the multicast distribution tree.
OSPF	Open Shortest Path First. An IGP routing protocol based on a link-state algorithm. Used to route within medium to large networks.
PIM-SM	Protocol Independent Multicast, Sparse-Mode TBD
Rendezvous Point	A router used in PIM-SM as part of the rendezvous process by which new senders are grafted on to the multicast tree.
Reverse Path Forwarding	Many multicast routing protocols such as PIM-SM build a multicast distribution tree based on the best route back from each receiver to the source, hence multicast packets will be forwarded along the reverse of the path to the source.
RIB	*See* Routing Information Base.
RIP	Routing Information Protocol. *TBD*
Routing Information Base	The collection of routes learned from all the dynamic routing protocols running on the router. Subdivided into a Unicast RIB for unicast routes and a Multicast RIB.
RP	*See* Rendezvous Point.
RPF	*See* Reverse Path Forwarding.
Static Route	A route that has been manually configured on the router.
xorpsh	XORP command shell.
xorp rtrmgr	XORP router manager process.