

VYATTA, INC.

| **Vyatta System**

# Basic System

## REFERENCE GUIDE

Using the CLI

System Management

User Management

Logging

SNMP

Installation and Upgrades



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

**COPYRIGHT**

Copyright © 2005–2009 Vyatta, Inc. All rights reserved.

Vyatta reserves the right to make changes to software, hardware, and documentation without notice. For the most recent version of documentation, visit the Vyatta web site at [vyatta.com](http://vyatta.com).

**PROPRIETARY NOTICES**

Vyatta is a registered trademark of Vyatta, Inc.

VMware, VMware ESX, and VMware server are trademarks of VMware, Inc.

All other trademarks are the property of their respective owners.

ISSUE DATE: February 2009

DOCUMENT REVISION. VC5 v03

RELEASED WITH: VC5.0.2

PART NO. A0-0110-10-0002

# Table of Contents

Quick Reference to Commands .....	ix
Quick List of Examples .....	xii
<b>Preface .....</b>	<b>xiv</b>
Intended Audience .....	xv
Organization of This Guide .....	xv
Document Conventions .....	xvi
Advisory Paragraphs .....	xvi
Typographic Conventions .....	xvi
Vyatta Publications .....	xvii
<b>Chapter 1 Using the CLI .....</b>	<b>1</b>
CLI Features .....	2
Accessing the CLI .....	2
The Vyatta CLI and the System Shell .....	3
User Privilege Levels .....	3
“Admin” Role .....	3
“Operator” Role .....	4
The Root User .....	5
Command Modes .....	5
Command Prompts .....	6
Using Special Characters in Commands .....	6
Command Completion .....	8
Command History .....	9
Command Editing .....	10
Displaying Long Output .....	10
Filtering Command Output .....	11
Working with Configuration .....	12
Entering and Exiting Configuration Mode .....	12

---

Configuration Hierarchy .....	13
Navigating in Configuration Mode .....	13
Viewing Configuration .....	14
Adding or Modifying Configuration .....	15
Deleting Configuration .....	16
Committing Configuration Changes .....	17
Discarding Configuration Changes .....	17
Saving Configuration .....	18
Loading a Saved Configuration .....	19
Booting from a Saved Configuration .....	20
Running an Operational Command from Configuration Mode .....	20
Displaying Configuration from Operational Mode .....	20
Basic CLI Commands .....	22
commit .....	23
configure .....	25
delete .....	26
discard .....	28
edit .....	30
exit .....	32
load .....	34
run .....	36
save .....	37
set .....	40
show .....	42
show configuration .....	44
top .....	46
up .....	47
<b>Chapter 2 System Management .....</b>	<b>48</b>
Basic System Configuration .....	49
Configuring Host Information .....	49
Host Name .....	50
Domain .....	51
IP Address .....	51
Default Gateway .....	52
Aliases .....	53
Configuring DNS .....	53
DNS Name Servers .....	54
Domain Search Order .....	55

---

Configuring Date and Time .....	56
Setting the Date .....	57
Manually Synchronizing with an NTP Server .....	57
Setting the Time Zone .....	57
Using NTP for Automatic Synchronization .....	59
Monitoring System Information .....	60
Showing Host Information .....	60
Showing the Date and Time .....	61
System Management Commands .....	62
clear arp address <ipv4> .....	64
clear arp interface <ethx> .....	65
clear console .....	66
clear interfaces counters .....	67
init-floppy .....	68
reboot .....	70
set date .....	71
show arp .....	73
show date .....	75
show files .....	76
show hardware cpu .....	77
show hardware dmi .....	79
show hardware mem .....	81
show hardware pci .....	83
show host .....	85
show interfaces .....	87
show license .....	89
show ntp .....	92
show system boot-messages .....	94
show system connections .....	96
show system kernel-messages .....	98
show system memory .....	100
show system processes .....	101
show system storage .....	103
show system uptime .....	104
show system usb .....	105
show tech-support .....	106
show version .....	108
system domain-name <domain> .....	112
system domain-search domain <domain> .....	113
system gateway-address <address> .....	115
system host-name <name> .....	116
system name-server <address> .....	117

system ntp-server <name> . . . . .	118
system options reboot-on-panic <value> . . . . .	120
system static-host-mapping host-name <name> . . . . .	122
system time-zone <zone> . . . . .	124
terminal . . . . .	126
<b>Chapter 3 User Management . . . . .</b>	<b>127</b>
User Management Configuration . . . . .	128
User Management Overview . . . . .	128
Login Authentication . . . . .	128
RADIUS Authentication . . . . .	129
Order of Authentication . . . . .	129
Creating "Login" User Accounts . . . . .	130
Configuring for a RADIUS Server . . . . .	131
Viewing Authentication Information . . . . .	132
User Management Commands . . . . .	134
system login . . . . .	135
system login radius-server <address> . . . . .	136
system login user <user> . . . . .	138
system login user <user> authentication . . . . .	140
system login user <user> full-name <name> . . . . .	142
system login user <user> level <level> . . . . .	144
show users . . . . .	146
<b>Chapter 4 Logging . . . . .</b>	<b>147</b>
Logging Configuration . . . . .	148
Logging Overview . . . . .	148
Logging Facilities . . . . .	148
Log Destinations . . . . .	149
Log File Locations and Archiving . . . . .	150
Log Severities . . . . .	150
Logging Configuration Example . . . . .	151
Enabling and Disabling Logging for Specific Features . . . . .	152
Logging Commands . . . . .	153
delete log file . . . . .	154
show log . . . . .	155
show log directory . . . . .	156
show log tail . . . . .	157
system syslog . . . . .	158
system syslog console facility <facility> level <level> . . . . .	162

system syslog file <filename> archive .....	164
system syslog file <filename> facility <facility> level <level> .....	166
system syslog global archive .....	168
system syslog global facility <facility> level <level> .....	170
system syslog host <hostname> facility <facility> level <level> .....	172
system syslog user <userid> facility <facility> level <level> .....	174
<b>Chapter 5 SNMP .....</b>	<b>176</b>
SNMP Configuration .....	177
SNMP Overview .....	177
MIB Objects .....	177
Traps .....	178
SNMP Commands .....	178
SNMP Versions .....	178
SNMP MIBs .....	178
SNMP Configuration Examples .....	179
Defining the SNMP Community .....	179
Specifying Trap Destinations .....	180
SNMP Commands .....	182
protocols snmp .....	183
protocols snmp community <community> .....	184
protocols snmp community <community> authorization <auth> .....	185
protocols snmp community <community> client <ipv4> .....	187
protocols snmp community <community> network <ipv4net> .....	189
protocols snmp contact <contact> .....	191
protocols snmp description <desc> .....	192
protocols snmp location <location> .....	193
protocols snmp trap-target <ipv4> .....	194
<b>Chapter 6 Installation and Upgrade .....</b>	<b>195</b>
Installation and Upgrade Overview .....	196
Installation and Upgrade Commands .....	197
full-upgrade .....	198
install-system .....	200
system package .....	201
system package auto-sync <days> .....	202
system package repository <repository> .....	203
system package repository <repository> components <component> .....	205
system package repository <repository> description <desc> .....	207
system package repository <repository> distribution <dist> .....	209
system package repository <repository> password <password> .....	211

---

system package repository <repository> url <url> .....	213
system package repository <repository> username <username> .....	215
<b>Appendix A SNMP MIB Support .....</b>	<b>217</b>
<b>Glossary of Acronyms .....</b>	<b>219</b>



# Quick Reference to Commands

Use this section to help you quickly locate a command.

clear arp address <ipv4> . . . . .	64
clear arp interface <ethx> . . . . .	65
clear console . . . . .	66
clear interfaces counters . . . . .	67
commit . . . . .	23
configure . . . . .	25
delete . . . . .	26
delete log file . . . . .	154
discard . . . . .	28
edit . . . . .	30
exit . . . . .	32
full-upgrade . . . . .	198
init-floppy . . . . .	68
install-system . . . . .	200
load . . . . .	34
protocols snmp . . . . .	183
protocols snmp community <community> . . . . .	184
protocols snmp community <community> authorization <auth> . . . . .	185
protocols snmp community <community> client <ipv4> . . . . .	187
protocols snmp community <community> network <ipv4net> . . . . .	189
protocols snmp contact <contact> . . . . .	191
protocols snmp description <desc> . . . . .	192
protocols snmp location <location> . . . . .	193
protocols snmp trap-target <ipv4> . . . . .	194
reboot . . . . .	70
run . . . . .	36
save . . . . .	37
set . . . . .	40
set date . . . . .	71
show . . . . .	42
show arp . . . . .	73
show configuration . . . . .	44

---

show date	75
show files	76
show hardware cpu	77
show hardware dmi	79
show hardware mem	81
show hardware pci	83
show host	85
show interfaces	87
show license	89
show log	155
show log directory	156
show log tail	157
show ntp	92
show system boot-messages	94
show system connections	96
show system kernel-messages	98
show system memory	100
show system processes	101
show system storage	103
show system uptime	104
show system usb	105
show tech-support	106
show users	146
show version	108
system domain-name <domain>	112
system domain-search domain <domain>	113
system gateway-address <address>	115
system host-name <name>	116
system login	135
system login radius-server <address>	136
system login user <user>	138
system login user <user> authentication	140
system login user <user> full-name <name>	142
system login user <user> level <level>	144
system name-server <address>	117
system ntp-server <name>	118
system options reboot-on-panic <value>	120
system package	201
system package auto-sync <days>	202
system package repository <repository>	203
system package repository <repository> components <component>	205
system package repository <repository> description <desc>	207
system package repository <repository> distribution <dist>	209
system package repository <repository> password <password>	211

---

system package repository <repository> url <url> . . . . .	213
system package repository <repository> username <username> . . . . .	215
system static-host-mapping host-name <name> . . . . .	122
system syslog . . . . .	158
system syslog console facility <facility> level <level> . . . . .	162
system syslog file <filename> archive . . . . .	164
system syslog file <filename> facility <facility> level <level> . . . . .	166
system syslog global archive . . . . .	168
system syslog global facility <facility> level <level> . . . . .	170
system syslog host <hostname> facility <facility> level <level> . . . . .	172
system syslog user <userid> facility <facility> level <level> . . . . .	174
system time-zone <zone> . . . . .	124
terminal . . . . .	126
top . . . . .	46
up . . . . .	47

# Quick List of Examples

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

Example 1-1	Entering configuration mode . . . . .	12
Example 1-2	"commit": Committing configuration changes . . . . .	24
Example 1-3	"configure": Entering configuration mode . . . . .	25
Example 1-4	"delete": Deleting configuration . . . . .	27
Example 1-5	"discard": Discarding configuration changes . . . . .	28
Example 1-6	"edit": Navigating in the configuration tree . . . . .	31
Example 1-7	Loading configuration from a file . . . . .	35
Example 1-8	"run": Running an operational command in configuration mode . . . . .	36
Example 1-9	"save": Saving configuration to a file . . . . .	38
Example 1-10	"save": Saving configuration to a file on a TFTP server . . . . .	39
Example 1-11	"set": Adding a configuration node . . . . .	41
Example 1-12	"show": Displaying configuration information . . . . .	43
Example 1-13	"show configuration": Displaying the first screen of the configuration tree in operational mode . . . . .	44
Example 1-14	"top": Navigating to the top of the configuration tree . . . . .	46
Example 1-15	"up": Navigating up through the configuration tree . . . . .	47
Example 2-13	Showing the system host name . . . . .	60
Example 2-14	Showing the system date and time . . . . .	61
Example 2-15	"init-floppy": Initializing a floppy diskette for saving configuration files . . . . .	69
Example 2-16	"reboot": Rebooting the system . . . . .	70
Example 2-17	Set the date and time directly . . . . .	72
Example 2-18	Set the date and time using an NTP server . . . . .	72
Example 2-19	"show arp": Displaying the ARP cache . . . . .	74
Example 2-20	"show date": Displaying the system date and time . . . . .	75
Example 2-21	"show files <directory>": Displaying file information . . . . .	76
Example 2-22	"show hardware cpu": Showing CPU information . . . . .	77

---

Example 2-23	"show hardware dmi": Showing DMI information	79
Example 2-24	"show hardware mem": Showing memory information	81
Example 2-25	"show hardware pci": Showing PCI bus information	83
Example 2-26	"show host lookup": Looking up network hosts	85
Example 2-27	"show host name": Showing network host names	86
Example 2-28	"show host date": Showing the system date and time	86
Example 2-29	"show host os": Showing operating system information	86
Example 2-30	"show interfaces": Displaying interface information	88
Example 2-31	"show license": Displaying license information	90
Example 2-32	"show ntp": Showing configured NTP servers	92
Example 2-33	"show system boot-messages": Displaying startup messages	94
<b>Example 2-34</b>	<b>"show system connections": Displaying active connections</b>	<b>96</b>
Example 2-35	"show system kernel-messages": Displaying messages from the kernel	98
<b>Example 2-36</b>	<b>"show system memory": Displaying information about memory usage</b>	<b>100</b>
Example 2-37	"show system processes": Displaying process information	101
Example 2-38	"show system storage": Displaying file system and storage information	103
Example 2-39	"show system storage": Displaying file system and storage information	104
Example 2-40	"show system usb": Displaying USB peripheral information	105
Example 2-41	"show tech-support" Displaying consolidated system information	107
Example 2-42	"show version": Displaying a summary of version information	109
Example 2-43	"show version all": Displaying software package version information	109
Example 2-44	"show version added": Displaying information about added software packages	110
Example 6-1	"Upgrading Vyatta software with full-upgrade"	199

# Preface

This guide explains how to use basic features of the Vyatta system. It describes the available commands and provides configuration examples.

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 aid to help you find the information you are looking for:

- **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.

This guide has the following chapters and appendixes:

Chapter	Description	Page
Chapter 1: Using the CLI	This chaptersection provides an overview of the Vyatta command-line interface (CLI), which is the primary user interface to the Vyatta system.	1
Chapter 2: System Management	This chapter describes Vyatta system features for basic system management tasks, such as setting host information, working with the ARP cache, and setting the system date and time.	48
Chapter 3: User Management	This chapter explains how to set up user accounts and user authentication.	127
Chapter 4: Logging	This chapter describes the Vyatta system logging mechanism.	147
Chapter 5: SNMP	This chapter describes the Vyatta system's support for SNMP.	176

Chapter 6: Installation and Upgrade	This chapter describes the commands for installing and upgrading Vyatta software.	195
Chapter A: SNMP MIB Support	This appendix lists the standard MIBs and traps supported by the Vyatta system.	217
Glossary of Acronyms		219

## Document Conventions

This guide contains advisory paragraphs and uses typographic conventions.

### Advisory Paragraphs

This guide uses 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 system 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.*

### Typographic Conventions

This document uses the following typographic conventions:

Courier	Examples, command-line output, and representations of configuration nodes.
<b>boldface Courier</b>	In an example, your input: something you type at a command line.



<b>boldface</b>	In-line commands, keywords, and file names .
<i>italics</i>	Arguments and variables, where you supply a value.
<key>	A key on your keyboard. Combinations of keys are joined by plus signs (“+”). An example is <Ctrl>+<Alt>+<Del>.
[ <i>arg1</i>   <i>arg2</i> ]	Enumerated options for completing a syntax. An example is [enable   disable].
<i>num1–numN</i>	A inclusive range of numbers. An example is 1–65535, which means 1 through 65535.
<i>arg1..argN</i>	A range of enumerated values. An example is eth0..eth3, which means eth0, eth1, eth2, and eth3.
<i>arg</i> [ <i>arg ...</i> ] <i>arg</i> , [ <i>arg</i> ,...]	A value that can optionally represent a list of elements (a space-separated list in the first case, and a comma-separated list in the second case).

## Vyatta Publications

More information about the Vyatta system is available in the Vyatta technical library, and on [www.vyatta.com](http://www.vyatta.com) and [www.vyatta.org](http://www.vyatta.org).

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

# Chapter 1: Using the CLI

This chaptersection provides an overview of the Vyatta command-line interface (CLI), which is the primary user interface to the Vyatta system.

This chapter presents the following topics:

- CLI Features
- Basic CLI Commands

# CLI Features

---

This section presents the following topics:

- Accessing the CLI
- The Vyatta CLI and the System Shell
- User Privilege Levels
- Command Modes
- Command Prompts
- Using Special Characters in Commands
- Command Completion
- Command History
- Command Editing
- Displaying Long Output
- Filtering Command Output
- Working with Configuration
- Running an Operational Command from Configuration Mode
- Displaying Configuration from Operational Mode

## Accessing the CLI

To access the command-line interface (CLI), you log on to the Vyatta system, either directly through the console port, or remotely using an SSH or Telnet session.

- From the system's console.
- Remotely, using SSH or Telnet

After the startup messages complete, the login prompt appears:

```
vyatta login:
```

Log on using the the user ID and password of a defined user account.

By default, the system has two predefined user accounts:

- **root**. This user has administrator-level privileges, which allows execution of all Vyatta and operating system commands. Command completion and CLI help show all commands and files, including operating system commands and files. The root user cannot be deleted.

```
User ID: root  
Default password: vyatta
```

- **vyatta.** This user has administrator-level privileges, which allows execution of all Vyatta and operating system commands. Command completion and CLI help show only Vyatta commands.

User ID: **vyatta**  
Default password: **vyatta**

**NOTE** You can change user accounts using operating system commands, but the changes will not persist across reboots. For persistent changes to user account information, use the Vyatta CLI.

## The Vyatta CLI and the System Shell

The CLI of the Vyatta system includes two kinds of commands:

- Vyatta-specific commands for operating and configuring the Vyatta system.
- Commands provided by the operating system shell in which the Vyatta CLI operates.

The commands you can execute depend on your user role. However, any command you are able to execute can be run from within the Vyatta CLI.

## User Privilege Levels

The Vyatta system supports two user roles:

- Admin level
- Operator level

The system also supports a special root user account. This account has an admin role; however, command completion and CLI help for the root user show both Vyatta commands plus operating system commands and files.

This section presents the following topics:

- “Admin” Role
- “Operator”Role
- The Root User

### “Admin” Role

Admin users have full access to the Vyatta CLI. Admin users can view, configure, and delete information, and execute all Vyatta operational commands. Admin users can also execute all operating system shell commands and constructs.

The default user **vyatta** is an admin user.

To create an admin user, issue the following commands in configuration mode:

```
vyatta@vyatta# set system login user user-name level admin

vyatta@vyatta# set system login user user-name authentication
plaintext-password password

vyatta@vyatta# commit
```

where *user-name* is the ID of the user account you are creating and *password* is the password you are assigning to the user.

Although operating system shell commands are always available to admin users, they are not shown when these users use command completion to query the CLI for available commands. This is because there are several hundred operating system shell commands and constructs available at any time: showing all available operating system shell commands would make it very difficult to distinguish available Vyatta CLI commands.

Adminusers can see available commands by entering **help** at the command prompt.

You can remove the restriction on command completion by setting the **VYATTA\_RESTRICTED\_MODE** environment variable to **none**:

```
export VYATTA_RESTRICTED_MODE=none
```

This removes command completion restriction for all users, regardless of privilege level.

## “Operator” Role

Operator users have read-only access to configuration plus the ability to execute Vyatta operational commands. Operator users can view in operational mode (using **show** commands), configure their terminal settings (using the **terminal** command), and exit from the Vyatta CLI (using the **exit** command). Operator users cannot enter configuration mode; however they can display configuration by issuing the **show configuration** command in operational mode.

Basic commands for displaying information (for example, **show configuration** plus the “pipe” commands, such as **more**, for managing display output) are available. Commands that use control constructs (such as **if**, **for**, and so on), list operators (such as “;”, “&”, and so on), and redirection are not available to operator users.

To create an operator user, issue the following command:

```
vyatta@vyatta# set system login user user-name level operator

vyatta@vyatta# set system login user user-name authentication
plaintext-password password

vyatta@vyatta# commit
```

where *user-name* is the ID of the user account you are creating and *password* is the password you are assigning to the user.

Operating system shell commands are not available to operator users and consequently, the list of commands returned using command completion for operator-level users is restricted to Vyatta commands.

You can remove the restriction on command completion by setting the **VYATTA\_RESTRICTED\_MODE** environment variable to **none**:

```
export VYATTA_RESTRICTED_MODE=none
```

This removes command completion restriction for all users, regardless of privilege level.

## The Root User

The Vyatta system supports a special user named **root**.

The root user is an admin user who has full access to the system. In addition, the root user has an expanded view of the operating system shell commands and constructs: when the root user queries the system for available commands using command completion, all operating system shell commands and constructs are displayed along with Vyatta CLI commands.

The root user is predefined by default and cannot be deleted from the system. The privilege level of the root user cannot be changed: any **level** setting configured for the root user is ignored.

## Command Modes

There are two command modes in the Vyatta CLI: operational mode and configuration mode.

- Operational mode provides access to operational commands for showing and clearing information and enabling or disabling debugging, as well as commands for configuring terminal settings, loading and saving configuration, and restarting the system.
- Configuration provides access to commands for creating, modifying, deleting, committing and showing configuration information, as well as commands for navigating through the configuration hierarchy.

When you log on to the system, the system is in operational mode.

- To enter configuration mode from operational mode, issue the **configure** command.
- To return to operational mode from configuration mode, issue the **exit** command. If there are uncommitted configuration changes, you must either commit the changes using the **commit** command, or discard the changes using the **discard** command (or **exit discard**), before you can exit to operational mode.

Issuing the **exit** command in operational mode logs you out of the system.

## Command Prompts

The command prompts show you where you are in the CLI, what user account you are logged on under, and the hostname of the system you are logged onto.

Table 1-1 shows some examples of command prompts and what they mean.

Table 1-1 Command prompts

The prompt shows this	And means this
vyatta@R1:~\$	User: vyatta Hostname: R1 Command mode: Operational mode
R1:~\$	User: root Hostname: R1 Command mode: Operational mode
vyatta@R1#	User: vyatta Hostname: R1 Command mode: Configuration mode
root@R1#	User: root Hostname: R1 Command mode: Configuration mode

## Using Special Characters in Commands

The Vyatta FusionCLI management interface is based on the GNU Bash shell. When entering a command at the command prompt, keep in mind that some characters have special meaning to the shell. For example, one such special character is the space character, which denotes the end of a token in a command, as shown below

```
prompt> show interfaces ethernet
```

In this example, the space characters separate the command line into three components: “show,” “interfaces,” and “ethernet.”

If you want to enter string that includes a literal character understood by the shell as a special character, you must enclose the character in double quotation marks. For example, if you want to enter a string that includes a space, you must enclose the string in double quotation marks as shown below:

```
prompt> set firewall name TEST description "external inbound"
```

In this example, the space within the string “external inbound” is within quotes and therefore loses its special meaning as a token separator.

Another example of a special character is the “pipe” character (also called the vertical bar, “|”), which separates two commands and means that the output of the left-hand side command should be processed using the right-hand side command, as shown in the following example:

```
prompt> show interfaces | match eth
```

In this example, the pipe character tells the shell to execute the **show interfaces** command and then process the output using the **match eth** command; as a result, only lines that contain the string “eth” will be displayed. As for the space character, if you want a literal vertical bar in a command component, you must enclose it in double quotation marks.

In addition to the space and vertical bar, the following characters have special meaning for the shell:

- ampersand (“&”)
- semi-colon (“;”)
- comma (“,”)
- left parenthesis (“(”)
- right parenthesis (“)”)
- left angle bracket (“<”)
- right angle bracket (“>”)
- backslash (“\”)
- pound sign (“#”)

In general, if you are unsure what characters are special, a good rule of thumb is to enclose anything that is not alphanumeric within double quotation marks.

Note that within a quoted string, you can include a literal quote mark by preceding it with a backslash, as follows:

```
"some \"quotes\" within quotes"
```

Of course, the rules become more complex if you want a literal backslash. As a general rule, try to avoid using quotation marks or backslashes as literal configuration values.



## Command Completion

You can have the system auto-complete a command syntax by entering any of the following at the command prompt:

Table 1-2 CLI Help Keystrokes

Type this:	To see this:
<Tab>	<p>Auto-completes a command.</p> <ul style="list-style-type: none"> <li>• If the command is unambiguous, the system generates the next token in the syntax.</li> <li>• If more than one completion is possible, the system displays the set of next possible tokens.</li> </ul> <p>(Note that the space following a command or keyword counts as a token.)</p> <p>Pressing &lt;Tab&gt; a second time generates CLI help for the current set of tokens.</p>
?	<p>Pressing the question mark key (" ? ") also generates command completion. To enter a literal question mark, first enter &lt;Ctrl&gt;+v, then the question mark.</p>
<Tab> <Alt>-?	<p>For non-root users, displays all available Vyatta commands and provides command completion. For the root user, displays all available Vyatta and operating system shell commands and provides command completion.</p>

The following example finds all available commands.

```
vyatta@R1:~$ <Tab>
add          format          reboot          terminal
clear        init-floppy        release        traceroute
configure    install-system    remove         traceroute6
connect      loopback          renew          undebug
debug        no                set            update
delete       ping              show           vpn
disconnect   ping6             telnet
vyatta@R1:~$
```

The following example requests command completion for the typed string **sh**. In this example, the command to be completed is unambiguous.

---

```
vyatta@R1~$: sh<Tab>
vyatta@R1~$: show
```

---

The following example requests command completion for the typed string `s`. In this case, there is more than one command that could complete the entry and the system lists all valid completions.

---

```
vyatta@R1~$: s<Tab>
set          show
```

---

Note that neither the `<Tab>` key nor the `<Alt>+?` key combination provides a help function when double-quoted. When used within double quotes, the `<Tab>` key generates a tab character and the `<Alt>+?` key combination generates a question mark (“?”) character.

## Command History

The Vyatta system shell supports a command history, where commands you execute are stored in an internal buffer and can be re-executed or edited.

Table 1-3 shows the most important history keystrokes.

Table 1-3 Command history keystrokes

Type this	To do this
<Up-Arrow> <Control>-p	Move to the previous command.
<Down-Arrow> <Control>-n	Move to the next command.

## Command Editing

The Vyatta system shell supports **emacs**-style command editing.

Table 1-4 shows the most important editing keystrokes.

Table 1-4 Command-Line Editing Keystrokes

Type this	To do this
<Left-Arrow> <Control>-b	Move backward in the command line.
<Right-Arrow> <Control>-f	Move forward in the command line.
<Control>-a	Move to the beginning of the command line.
<Control>-e	Move the end of the command line.
<Control>-d	Delete the character directly under the cursor.
<Control>-t	Toggle (swap) the character under the cursor with the character immediately preceding it.
<Control>-<Space>	Mark the current cursor position.
<Control>-w	Delete the text between the mark and the current cursor position, copying the deleted text to the cut buffer.
<Control>-k	“Kill” (delete) from the cursor to the end of the line, copying the deleted text into the cut buffer.
<Control>-y	“Yank” (paste) from the cut buffer into the command line, inserting it at the cursor location.

## Displaying Long Output

If the information being displayed is too long for your screen, the screen will show the “More” indication where the information breaks.

Table 1-5 shows the keystrokes for controlling the display of information in a “More” screen.

Table 1-5 Display options within a “More” screen

To do this	Press this
Exit “More”	q Q

Table 1-5 Display options within a “More” screen

Scroll down one whole screen.	<Space> f <Ctrl>+f
Scroll up one whole screen	b <Ctrl>+b
Scroll down one-half screen.	d <Ctrl>+d
Scroll up one-half screen	u <Ctrl>+u
Scroll down one line.	<Enter> e <Ctrl>+e <Down Arrow>
Scroll up one line.	y <Ctrl>+y <Up Arrow>
Scroll down to the bottom of the output.	G
Scroll up to the top of the output.	g
Display detailed help for “More”.	h

## Filtering Command Output

The Vyatta system can pipe the output of commands into selected operating system shell commands to filter what is displayed on the console. Commands are piped into the filters using the vertical bar pipe operator (“|”).

Table 1-6 shows the pipe commands implemented for the Vyatta system.

Table 1-6 “Pipe” filter commands

Type this	To do this
count	Count occurrences.
match <i>pattern</i>	Show only text that matches the specified pattern.
more	Paginate output
no-match <i>pattern</i>	Show only text that does not match the specified pattern.
no-more	Don't paginate output.

## Working with Configuration

This section presents the following topics:

- Entering and Exiting Configuration Mode
- Configuration Hierarchy
- Navigating in Configuration Mode
- Viewing Configuration
- Adding or Modifying Configuration
- Deleting Configuration
- Committing Configuration Changes
- Discarding Configuration Changes
- Saving Configuration
- Loading a Saved Configuration
- Booting from a Saved Configuration

## Entering and Exiting Configuration Mode

To enter configuration mode, use the **configure** command in operational mode.

### Example 1-1 Entering configuration mode

---

```
vyatta@vyatta:~$ configure
[edit]
vyatta@vyatta#
```

---

Once in configuration mode, the command prompt changes from this:

```
user@host:~$
```

to this:

```
user@host: #
```

---

To exit configuration mode, use the **exit** command from the top level of configuration.

If you have changed configuration, you must either **commit** changes or discard them using the **exit discard** command.

## Configuration Hierarchy

Vyatta system configuration is organized as a hierarchy of configuration statements, with a hierarchical tree of *nodes* similar to the directory structure on a UNIX file system. There are three kinds of statements:

- Configuration nodes. These can be either:
  - Single nodes (just one instance can be created; for example, the **rip** protocol node)
  - Multi-nodes (more than one instance can be created; for example, **address** nodes)
- Attribute statements. These set the values or characteristics for parameters within a node.

From the system's point of view, a configuration node is different from a simple configuration attribute statement. A configuration *attribute statement* takes the form *attribute value*, as in the following example.

---

```
protocol-version v2
```

---

A configuration *node* always has an enclosing pair of braces, which may be empty, as in the following example:

---

```
dns-server ipv4 {}
```

---

or non-empty, as in the following example:

---

```
ssh {  
    allow-root false  
}
```

---

## Navigating in Configuration Mode

You can tell where you are in the configuration tree by the **[edit]** prompt, which is context-sensitive.

At the top of the configuration tree, the [edit] prompt displays like this:

```
[edit]
```

When you are in another location, the edit prompt displays your location by showing the node hierarchy in order, like this:

```
[edit protocols bgp]
```

Table 1-5 shows the commands for navigating in configuration mode.

Table 1-7 Commands for navigating in configuration mode

Command	Result
<code>edit config-node</code>	Navigates to the specified configuration node for editing. The node must already be created the the configuration committed.
<code>exit</code>	Jumps to the top of the configuration tree. If you are already at the top of the configuration tree, exit from configuration mode and return to operational mode.
<code>top</code>	Jumps to the top of the configuration tree.
<code>up</code>	Moves up one node in the configuration tree.

Using the **edit** command lets you navigate to the part of the hierarchy that you are interested in and execute commands relative to your location. This saves typing if you need to work on a particular part of the configuration hierarchy.

The following example navigates to the configuration node for the Ethernet interface eth2. Once you have navigated to the node, you can show configuration directly without specifying the full path.

```
vyatta@R1# edit interfaces ethernet eth2
[edit interfaces ethernet eth2]
vyatta@R1# show
  hw-id 00:13:46:e6:f6:87
[edit interfaces ethernet eth2]
vyatta@R1#
```

## Viewing Configuration

Use the **show** command in configuration mode to display configuration. You can restrict the display to a particular node by specifying the path to the node.

For example, the following example shows configuration for all configured interfaces.

```
user@host# show interfaces
  ethernet eth0 {
    address 10.1.0.62/24
    hw-id 00:40:63:e2:e4:00
```

```
    }  
    ethernet eth1 {  
        address 172.16.234.23/25  
        hw-id 00:40:63:e2:e3:dd  
        vrrp {  
            virtual-address 172.16.99.99  
            vrrp-group 20  
        }  
    }  
    loopback lo {  
    }  
}
```

---

The following example shows configuration only for the Ethernet interface eth0.

---

```
vyatta@R1# show interfaces ethernet eth0  
address 10.1.0.62/24  
hw-id 00:40:63:e2:e4:00
```

---

When the display is too large for one screen, it stops with one screen displayed. In this case:

- Press <Enter> to display the next line.
- Press <space> to display the next screen.
- Press q to interrupt the display and return to the command prompt.

## Adding or Modifying Configuration

Add new configuration by creating a configuration node, using the **set** command in configuration mode. Modify existing configuration using the **set** command in configuration mode. For example:

---

```
vyatta@R1# set interfaces ethernet eth2 address 192.168.1.100/24  
[edit]  
vyatta@R1#
```

---

Then use the **show** command to see the change:

---

```
vyatta@R1# show interfaces ethernet eth2  
+address 192.168.1.100/24
```



```
hw-id 00:13:46:e6:f6:87
[edit]
vyatta@R1#
```

---

Note the “+” in front of the new statement. This shows that this statement has been added to the configuration but the change is not yet committed. The change does not take effect until configuration is committed using the **commit** command.

You can modify configuration from the root of the configuration tree or use the **edit** command to navigate to the part of the tree where you want to change or add.

The configuration tree is nearly empty when you first start up, except for a few automatically configured nodes. You must create a node for any functionality you want to configure on the system. When a node is created, any default values that exist for its attributes are applied to the node.

One thing you can’t do with the **set** command is change the identifier of a node for which there can be multiple instances (a “multi-node”), such as a DNS server or an IP address for an interface. If a multi-node has an incorrect identifier, you must delete the node and recreate it with the correct identifier.

## Deleting Configuration

Use the **delete** command to delete configuration statement or a complete configuration node. For example:

---

```
vyatta@R1# delete interfaces ethernet eth2 address
192.168.1.100/24
[edit]
```

---

Then use the **show** command to see the change:

---

```
vyatta@R1# show interfaces ethernet eth2
-address 192.168.1.100/24
hw-id 00:13:46:e6:f6:87
[edit]
```

---

Note the “-” in front of the deleted statement. This shows that this statement has been deleted from the configuration but the change is not yet committed. The change does not take effect until configuration is committed using the **commit** command.

Some configuration nodes are mandatory; these cannot be deleted. Some configuration nodes are mandatory, but have default values; if you delete one of these nodes, the default value is restored.

## Committing Configuration Changes

In the Vyatta system, configuration changes do not take effect until you commit them using the **commit** command.

---

```
vyatta@R1# commit
[edit]
```

---

Uncommitted changes are flagged with either a plus sign (for added or modified changes) or a minus sign (for deleted changes). Once you commit the changes, the sign disappears, as in the following example:

---

```
vyatta@R1# show interfaces ethernet eth2
-address 192.168.1.100/24
  hw-id 00:13:46:e6:f6:87
[edit]
vyatta@R1# commit
[edit]
vyatta@R1# show interfaces ethernet eth2
  hw-id 00:13:46:e6:f6:87
[edit]
```

---

## Discarding Configuration Changes

You cannot exit from configuration mode with uncommitted configuration changes; you must either commit the changes or discard them. If you don't want to commit the changes, you can discard them using the **exit discard** command.

---

```
vyatta@R1# exit
Cannot exit: configuration modified.
Use 'exit discard' to discard the changes and exit.
[edit]
vyatta@R1# exit discard
vyatta@R1:~$
```

---

## Saving Configuration

The running configuration can be saved using the **save** command in configuration mode. By default, configuration is saved to the file **config.boot** in the standard configuration directory.

- For hard disk installs the configuration directory is **/opt/vyatta/etc/config**
- For installations running off LiveCD, the configuration directory is **/media/floppy/config**.

---

```
vyatta@R1# save
Saving configuration to '/opt/vyatta/etc/config/config.boot'...
Done
[edit]
vyatta@R1#
```

---

You can save configuration to a different location by specifying a different file name.

---

```
vyatta#R1 save testconfig
Saving configuration to '/opt/vyatta/etc/config/testconfig'...
Done
[edit]
vyatta@R1#
```

---

You can also save a configuration file to a location path other than the standard configuration directory **/opt/vyatta/etc/config**, by specifying a different path. You can save to a hard drive, compact Flash or USB device by including the drive identifier in the path.

Note that the **save** command writes only committed changes. If you try to save uncommitted changes the system warns you that it is saving only the committed changes.

Table 1-8 shows the syntax for file specification for various circumstances.

Table 1-8 Specifying locations for the configuration file

Location	Specification
An absolute path	Use standard UNIX file specification.
A relative path	Specify the path name relative to the location configured for the the config-directory parameter of the rtrmgr configuration node.

Table 1-8 Specifying locations for the configuration file

Location	Specification
TFTP server	Use the following syntax for <i>file-name</i> : <code>tftp://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the TFTP server, and <i>config-file</i> is the configuration file, including the path relative to the TFTP root directory.
FTP server	Use the following syntax for <i>file-name</i> : <code>ftp://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the FTP server, and <i>config-file</i> is the configuration file, including the path. If you use FTP, you will be prompted for a user name and password.
HTTP server	Use the following syntax for <i>file-name</i> : <code>http://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the HTTP server, and <i>config-file</i> is the configuration file, including the path.

If you are running the system from LiveCD, configuration can be saved only to floppy disk. If you do not save your running configuration to floppy disk, any changes you have made will be lost after reboot.

Before you can save configuration to a floppy disk, you must initialize the floppy disk using the **init-floppy** command in operational mode.

## Loading a Saved Configuration

To load a previously saved configuration use the **load** command in configuration mode. By default, the system reads the file from the standard configuration directory. By default this is **/opt/vyatta/etc/config.**

```
vyatta@R1# load testconfig
Loading config file /opt/vyatta/etc/config/testconfig...
Done
[edit]
vyatta@R1#
```

A loaded configuration is automatically committed and becomes the active configuration.

## Booting from a Saved Configuration

If you want the file to be automatically read the next time the system starts, you must save it as **config.boot** in the standard configuration directory. By default

- For hard disk installs the configuration directory is **/opt/vyatta/etc/config**
- For installations running off of a LiveCD, the configuration directory is **/media/floppy/config**.

## Running an Operational Command from Configuration Mode

You can run an operational command without leaving configuration mode using the **run** command, as in the following example:

---

```
vyatta@R1# run show system processes summary
20:45:46 up 1 day, 10:16, 3 users, load average: 0.00, 0.00,
0.00
[edit]
vyatta@R1#
```

---

## Displaying Configuration from Operational Mode

You can display configuration information without leaving operational mode using the **show configuration** command, as in the following example:

---

```
vyatta@R1:~$ show configuration
interfaces {
    ethernet eth0 {
        address 192.168.1.77/24
        hw-id 00:0c:29:68:b3:9f
    }
    ethernet eth1 {
        hw-id 00:0c:29:68:b3:a9
    }
    loopback lo {
    }
}
service {
    ssh {
        allow-root false
    }
}
```

---

```
    }  
  }  
  system {  
    gateway-address 192.168.1.254  
    host-name R1  
    login {  
      user root {  
        authentication {  
          encrypted-password *****  
        }  
      }  
    }  
  }  
}
```

---

# Basic CLI Commands

This chapter contains the following commands.

Configuration Commands	
commit	Applies any uncommitted configuration changes.
delete	Deletes a configuration node.
edit	Navigates to a subnode in the configuration tree for editing.
exit	Navigates up one level of use.
load	Loads a saved configuration.
run	Runs an operational command without leaving configuration mode.
save	Saves the running configuration to a file.
set	Creates a new configuration node, or modifies a value in an existing configuration node.
show	Displays configuration information in configuration mode.
top	Exits to the top level of configuration mode.
up	Navigates up one level in the configuration tree.
Operational Commands	
configure	Enters configuration mode.
exit	Navigates up one level of use.
init-floppy	Formats a floppy diskette and prepares it to receive a configuration file. <i>See page 68 in Chapter 2: System Management.</i>
show arp	Displays the system's ARP cache. <i>See page 73 in Chapter 2: System Management</i>
show configuration	Displays system configuration from operational mode.

# commit

Applies any uncommitted configuration changes.

---

## Syntax

**commit**

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to apply changes to configuration.

When you add configuration to the system, modify existing configuration, or delete configuration from the system, 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 32).

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 system is. Be prepared to wait for several seconds for the system to complete committing the information.

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



---

## Examples

Example 1-2 shows an uncommitted deletion which is then committed. In this example, note how the uncommitted deletion is flagged with a minus sign (“-”), which disappears after the change is committed.

### Example 1-2 “commit”: Committing configuration changes

---

```
vyatta@vyatta# show interfaces ethernet eth2
-address 192.168.1.100/24
 hw-id 00:13:46:e6:f6:87
[edit]
vyatta@vyatta# commit
[edit]
vyatta@vyatta# show interfaces ethernet eth2
 hw-id 00:13:46:e6:f6:87
[edit]
```

---

# configure

Enters configuration mode.

---

## Syntax

**configure**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to enter configuration mode from operational mode. In configuration mode, you can add, delete, and modify configuration information.

When you are in configuration mode, the command prompt changes to mark the change in command mode.

---

## Examples

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

Example 1-3 "configure": Entering configuration mode

---

```
vyatta@vyatta:~$ configure
[edit]
vyatta@vyatta#
```

---

# delete

Deletes a configuration node.

---

## Syntax

**delete** *config-node*

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

---

<i>config-node</i>	The configuration node to be deleted, including the full path, separated by spaces, through the configuration hierarchy to the node.
--------------------	--

---

---

## Default

None.

---

## Usage Guidelines

Use this command to delete a part of configuration. To do this, you delete the appropriate subnode of a configuration node.

If you show configuration before it is committed, you will see the deleted statement flagged with a minus sign (“-”); the statement disappears after the configuration change is committed.

Some configuration nodes and statements are mandatory; these nodes or statements cannot be deleted. Some configuration statements are mandatory but have default values; if you delete one of these statements, the default value is restored.

---

## Examples

Example 1-4 deletes a DNS server from system configuration.

Example 1-4 “delete”: Deleting configuration

---

```
vyatta@vyatta# show system name-server <Tab>
10.0.0.30 10.0.0.31 10.0.0.32
[edit]
vyatta@vyatta# delete system name-server 10.0.0.32
[edit]
vyatta@vyatta# show system name-server <Tab>
10.0.0.30 10.0.0.31
[edit]
```

---

# discard

Discards any uncommitted configuration changes.

---

## Syntax

**discard**

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to discard all uncommitted changes to configuration.

---

## Examples

Example 1-5 shows an uncommitted deletion and an uncommitted addition which are then discarded. In this example, note how the uncommitted deletion (flagged with a minus sign “-”) and the uncommitted addition (flagged with a plus sign “+”), disappear after the **discard** command is invoked.

Example 1-5 “discard”: Discarding configuration changes

---

```
vyatta@vyatta# show interfaces ethernet eth2
-address 192.168.1.100/24
+address 192.168.1.101/24
  hw-id 00:13:46:e6:f6:87
[edit]
vyatta@vyatta# discard
Changes have been discarded
[edit]
```

```
vyatta@vyatta# show interfaces ethernet eth2  
address 192.168.1.100/24  
hw-id: 00:13:46:e6:f6:87  
[edit]
```

---

# edit

Navigates to a subnode in the configuration tree for editing.

---

## Syntax

**edit** *path*

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

---

<i>path</i>	The path to the node of configuration tree you want to edit.
-------------	--

---

---

## Default

None.

---

## Usage Guidelines

Use this command to navigate to a specific configuration subnode for editing. The **[edit]** prompt changes dynamically to mark your place in the configuration tree.

Once at that location, any actions you take such as showing, creating, or deleting configuration are relative to your location in the tree.

You can only navigate to a configuration node that has already been created and committed. Configuration nodes are created and modified using the **set** command (see page 40) and are committed using the **commit** command (see page 23).

---

## Examples

The following example begins at the top of the configuration tree in configuration mode and navigates to the **system login** configuration node. Once at the **system login** node, a **show** command displays just the contents of the **login** node.

In this example, notice how the prompt changes to mark the location in the configuration tree.

---

### Example 1-6 “edit”: Navigating in the configuration tree

---

```
[edit]
vyatta@vyatta# edit system login
[edit system login]
vyatta@vyatta# show
user mike {
    authentication {
        encrypted-password $1$hccJixQo$V6sL5hDl6CUmVZvaHlvTf0
        plaintext-password " "
    }
}
user root {
    authentication {
        encrypted-password $1$$Ht7gBYnxI1xCd0/JOnodh.
    }
}
user vyatta {
    authentication {
        encrypted-password $1$$Ht7gBYnxI1xCd0/JOnodh.
    }
}
[edit system login]
```

---



# exit

Navigates up one level of use.

- From a configuration subnode, jumps to the top of the configuration tree.
- From the top of the configuration tree, exits to operational mode.
- From operational mode, exits the system.

---

## Syntax

**exit** [**discard**]

---

## Command Mode

Configuration mode.

Operational mode.

---

## Configuration Statement

None.

---

## Parameters

---

<b>discard</b>	Applies when exiting from configuration mode to operational mode with uncommitted configuration changes. Allows you to exit from configuration mode by discarding all configuration changes.
----------------	--

---

---

## Default

None.

---

## Usage Guidelines

Use this command from a subnode in the configuration tree to navigate to the top of the configuration tree.

Use this command from the top of the configuration tree to exit from configuration mode to operational mode.

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** command with the **discard** option. This is the only case where this option applies.

Use this command in operational mode to exit from the system.

# load

Loads a saved configuration.

---

## Syntax

**load** *file-name*

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

<i>file-name</i>	The name of the configuration file, including the full path to its location.
------------------	--

---

## Default

None.

---

## Usage Guidelines

Use this command to manually load a configuration previously saved to a file.

The loaded configuration becomes the active (running) configuration and the previous running configuration is discarded.

Configuration can be loaded from a hard disk (including a Flash disk or USB device), a TFTP server, an FTP server, or an HTTP server. Note that you cannot load an empty configuration file; the configuration file must contain at least one configuration node.

The default configuration directory is **/opt/vyatta/etc/config**.

The following table shows the syntax for file specification for different file locations.

Table 1-9 Specifying locations for the configuration file

Location	Specification
An absolute path	Use standard UNIX file specification.

Table 1-9 Specifying locations for the configuration file

Location	Specification
A relative path	Specify the path name relative to the default configuration directory.
TFTP server	Use the following syntax for <i>file-name</i> : <code>tftp://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the TFTP server, and <i>config-file</i> is the configuration file, including the path relative to the TFTP root directory.
FTP server	Use the following syntax for <i>file-name</i> : <code>ftp://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the FTP server, and <i>config-file</i> is the configuration file, including the path. If you use FTP, you will be prompted for a user name and password.
HTTP server	use the following syntax for <i>file-name</i> : <code>http://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the HTTP server, and <i>config-file</i> is the configuration file, including the path.

---

## Examples

Example 1-7 loads the configuration file **testconfig** from the default configuration directory.

Example 1-7 Loading configuration from a file

---

```
vyatta@vyatta# load testconfig
Loading config file /opt/vyatta/etc/config/testconfig...
Done
[edit]
vyatta@vyatta#
```

---

## run

Runs an operational command without leaving configuration mode.

---

### Syntax

**run** *command*

---

### Command Mode

Configuration mode.

---

### Configuration Statement

None.

---

### Parameters

---

<i>command</i>	The operational command to be executed.
----------------	---

---

---

### Default

None.

---

### Usage Guidelines

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

---

### Examples

Example 1-8 executes the **show date** command (an operational command) from configuration mode.

Example 1-8 “run”: Running an operational command in configuration mode

---

```
vyatta@vyatta# run show date
Sun Dec 16 23:34:06 GMT 2007
[edit]
vyatta@vyatta#
```

---

## save

Saves the running configuration to a file.

---

### Syntax

**save** *file-name*

---

### Command Mode

Configuration mode.

---

### Configuration Statement

None.

---

### Parameters

---

<i>file-name</i>	The name of the file where the information is to be saved, including the path to the file.
------------------	--

---

---

### Default

None.

---

### Usage Guidelines

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

The resulting file can later be loaded into the running system to replace the previous running configuration, using the **load** command (see page 34). A non-absolute path is interpreted relative to the default configuration directory, which is **/opt/vyatta/etc/config**.

The following table shows the syntax for file specification for different file locations.

Table 1-10 Specifying locations for the configuration file

Location	Specification
An absolute path	Use standard UNIX file specification.
A relative path	Specify the path name relative to the default configuration directory.

Table 1-10 Specifying locations for the configuration file

Location	Specification
TFTP server	Use the following syntax for <i>file-name</i> : <code>tftp://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the TFTP server, and <i>config-file</i> is the configuration file, including the path relative to the TFTP root directory.
FTP server	Use the following syntax for <i>file-name</i> : <code>ftp://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the FTP server, and <i>config-file</i> is the configuration file, including the path. If you use FTP, you will be prompted for a user name and password.
HTTP server	use the following syntax for file-name: <code>http://ip-address/config-file</code> where <i>ip-address</i> is the IP address of the HTTP server, and <i>config-file</i> is the configuration file, including the path.

If you overwrite a configuration file, the system retains one backup, using a *file-name~* convention. For example, if you save over **my-config.boot**, the system 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** in the default configuration directory, exits from configuration mode, and displays the set of files stored in the configuration directory.

Example 1-9 “save”: Saving configuration to a file

```
vyatta@vyatta# save
Saving configuration to '/opt/vyatta/etc/config/config.boot'...
Done
[edit]
vyatta@vyatta# exit
vyatta@vyatta:~$ 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
```

```
-rw-rw-r-- 1 vyatta xorp 2.8K Nov 27 21:50 my-config~  
vyatta@vyatta:~$
```

---

Example 1-10 saves the current running configuration to the file **my-config** 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

---

```
vyatta@vyatta# save tftp://10.1.0.35/my-config  
Saving configuration to 'tftp://10.1.0.35/my-config'...  
Done  
[edit]  
vyatta@vyatta#
```

---



# set

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

---

## Syntax

To create a new configuration node, the syntax is as follows:

**set** *config-node* [*identifier*]

To set an attribute within a configuration node, the syntax is as follows:

**set** *config-node* [*identifier*] *attribute* [*value*]

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

<i>config-node</i>	The configuration node to be created or modified, including the full path, separated by spaces, through the configuration hierarchy to the node.
<i>identifier</i>	The identifier of the configuration node. Mandatory if the configuration node has an identifier; forbidden otherwise.
<i>attribute</i>	The configuration attribute or property to be set. If the attribute statement does not exist, it is created. If the attribute statement already exists, the value is set to the new value.
<i>value</i>	The new value of the attribute. Mandatory if the attribute statement requires a value; forbidden otherwise.

---

## Default

None.

---

## Usage Guidelines

Use this command to add a configuration element to the current configuration—for example, to enable a routing protocol or define an interface.

You can also use this command to modify the value of an existing configuration item. When setting configuration values, note that the change does not take effect until the change is committed, using the **commit** command (see page 23).

Once a configuration node has been added, you can modify it later using the **set** command (see page 40), or delete it using the **delete** command (see page 26).

---

## Examples

Example 1-11 adds a configuration node for an Ethernet interface and commits the change.

Example 1-11 “set”: Adding a configuration node

---

```
vyatta@vyatta# set interfaces ethernet eth1 address
192.150.187.108/24
[edit]
vyatta@vyatta# commit
[edit]
```

---

# show

Displays configuration information in configuration mode.

---

## Syntax

**show** [-all] *config-node*

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

---

<i>config-node</i>	The configuration node you want to view, including the path. The node must exist and the created node must have been committed.  The configuration node specification is interpreted relative to your current position in the configuration tree.
<b>-all</b>	Includes default information in the displayed information.

---

---

## Default

When used with no configuration node specification, this command displays all existing configuration nodes and sub-nodes starting from your current location in the configuration tree.

When used without the **-all** option, default information is not shown

---

## Usage Guidelines

Use this command in configuration mode to display the configured state of the system.

This command displays the specified configuration node and all sub-nodes. The node specification is interpreted relative to your current location in the configuration tree.

Unless the **-all** keyword is used, default information is not included in displayed information.

In addition to this command, there are a number of **show** commands available in operational mode. For a list of these commands, please see the Quick Reference to Commands, which begins on page ix.

---

## Examples

Example 1-12 shows the **service** node displayed using the **show** command in configuration mode.

Example 1-12 “show”: Displaying configuration information

---

```
vyatta@vyatta# show service
dhcp-server {
}
dns {
}
ssh {
}
telnet {
}
[edit]
vyatta@vyatta#
```

---

# show configuration

Displays system configuration from operational mode.

---

## Syntax

**show configuration** [all]

---

## Command Mode

Operational mode.

---

## Parameters

---

<b>all</b>	Displays all configuration, including default values that would not normally be displayed.
------------	--

---

---

## Default

Displays only the values that have been set explicitly (that is, non-default values).

---

## Usage Guidelines

Use this command to list configuration information while remaining in operational mode.

Using **show configuration** in operational mode is equivalent to using **show** in configuration mode.

---

## Examples

Example 1-13 displays the configuration from operational mode.

Example 1-13 “show configuration”: Displaying the first screen of the configuration tree in operational mode

---

```
vyatta@vyatta:~$ show configuration
interfaces {
  ethernet eth0 {
    address 192.168.1.77/24
    hw-id 00:0c:29:68:b3:9f
  }
  ethernet eth1 {
```

```
        hw-id 00:0c:29:68:b3:a9
    }
    loopback lo {
    }
}
service {
    ssh {
        allow-root false
    }
}
system {
    gateway-address 192.168.1.254
    host-name R1
    login {
        user root {
            authentication {
                encrypted-password *****
            }
        }
    }
}
```

---

# top

Exits to the top level of configuration mode.

---

## Syntax

**top**

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

None.

---

## Usage Guidelines

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

---

## Examples

Example 1-14 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 1-14 “top”: Navigating to the top of the configuration tree

---

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

---

# up

Navigates up one level in the configuration tree.

---

## Syntax

**up**

---

## Command Mode

Configuration mode.

---

## Configuration Statement

None.

---

## Parameters

None.

---

## Usage Guidelines

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

---

## Examples

Example 1-15 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 1-15 “up”: Navigating up through the configuration tree

---

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

```

---



## Chapter 2: System Management

This chapter describes Vyatta system features for basic system management tasks, such as setting host information, working with the ARP cache, and setting the system date and time.

This section presents the following topics:

- Basic System Configuration
- Monitoring System Information
- System Management Commands

# Basic System Configuration

The commands in this chapter allow you to change and view basic IP system information. This section presents the following topics:

- Configuring Host Information
- Configuring DNS
- Configuring Date and Time
- Monitoring System Information

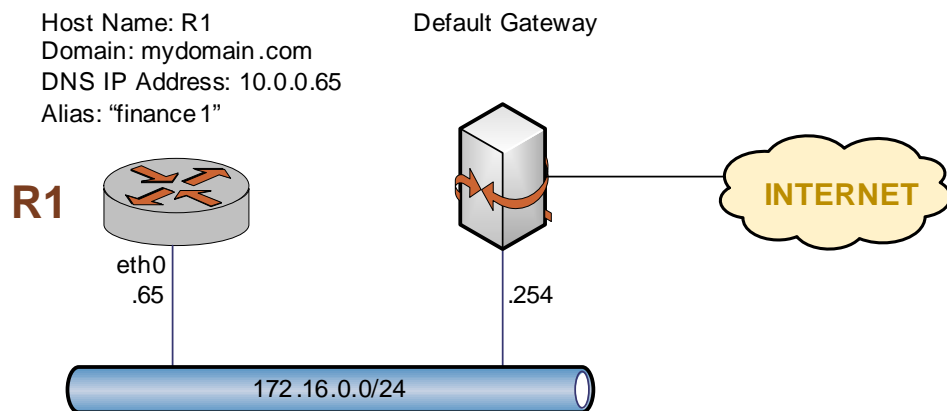
## Configuring Host Information

This section presents the following topics:

- Host Name
- Domain
- IP Address
- Default Gateway
- Aliases

In this section, sample configurations are presented for the system's host information. The sample configuration used is shown in Figure 2-1.

Figure 2-1 Host information



This section includes the following examples:

- Example 2-1 Setting the system's host name

- Example 2-2 Setting the system's domain
- Example 2-3 Mapping the system's IP address to its host name
- Example 2-4 Setting the default gateway
- Example 2-5 Creating an alias for the system

## Host Name

The Vyatta system's name is set using the **system host-name** command. System names can include letters, numbers, and hyphens ("-").

Example 2-1 sets the system's host name to R1. To set the system host name, perform the following steps in configuration mode:

Example 2-1 Setting the system's host name

Step	Command
Set the system's host name.	vyatta@vyatta# <b>set system host-name R1</b> [edit]
Commit the change. The command prompt changes to reflect the change	vyatta@vyatta# <b>commit</b> [edit] vyatta@R1#
Show the configuration.	vyatta@R1# <b>show system host-name</b> host-name R1 [edit]

## Domain

The system's domain is set using the **system domain-name** command. Domain names can include letters, numbers, hyphens, and periods.

**NOTE** **system domain-name** and **system domain-search** are mutually exclusive. Only one of the two can be configured at any one time.

Example 2-2 sets the system's domain to **mydomain.com**.

To set the system's domain, perform the following steps in configuration mode:

Example 2-2 Setting the system's domain

Step	Command
Set the domain name.	vyatta@R1# <b>set system domain-name mydomain.com</b> [edit]
Commit the change.	vyatta@R1# <b>commit</b> [edit]
Show the configuration.	vyatta@R1# <b>show system domain-name</b> domain-name mydomain.com [edit]

## IP Address

The system's IP address can be statically mapped to its host name for local DNS purposes, using the **system static-host-mapping** command.

IP networks are specified in CIDR format—that is, in *ip-address/prefix* notation such as 192.168.12.0/24. For single addresses, use dotted quad format, that is, *a.b.c.d*. For network prefixes, enter a decimal number from 1 through 32.

A good practice is to map the system's host name to the loopback address, as the loopback interface is the most reliable on the system. In this example, the loopback interface is given the address 10.0.0.65. This is the address configured for the loopback interface in the sample topology used in this guide.

Example 2-3 creates a static mapping between the host name R1 and IP address 10.0.0.65. This is the IP address the DNS server will use to resolve DNS requests for **R1.mydomain.com**.

To map the host name to the IP address, perform the following steps in configuration mode:

Example 2-3 Mapping the system's IP address to its host name

Step	Command
Map host name R1 to IP address 10.0.0.65.	<pre>vyatta@R1# set system static-host-mapping host-name R1 inet 10.0.0.65 [edit]</pre>
Commit the change.	<pre>vyatta@R1# commit [edit]</pre>
Show the configuration.	<pre>vyatta@R1# show system static-host-mapping   host-name R1 {     inet 10.0.0.65   } [edit]</pre>

## Default Gateway

Example 2-4 specifies a default gateway for the system at 172.16.0.254.

To specify the default gateway, perform the following steps in configuration mode:

Example 2-4 Setting the default gateway

Step	Command
Specify the default gateway.	<pre>vyatta@R1# set system gateway-address 172.16.0.254 [edit]</pre>
Commit the change.	<pre>vyatta@R1# commit [edit]</pre>
Show the configuration.	<pre>vyatta@R1# show system gateway-address   gateway-address 172.16.0.254 [edit]</pre>

## Aliases

You can define one or more aliases for the system by mapping the system's IP address to more than one host name.

Example 2-5 creates the alias **finance1** for the system.

To create an alias for the system, perform the following steps in configuration mode:

Example 2-5 Creating an alias for the system

Step	Command
Define an alias.	<pre>vyatta@R1# set system static-host-mapping host-name R1 alias finance1 [edit]</pre>
Commit the change.	<pre>vyatta@R1# commit [edit]</pre>
Show the configuration.	<pre>vyatta@R1# show system static-host-mapping   host-name R1 {     alias finance1     inet 10.0.0.65   } [edit]</pre>

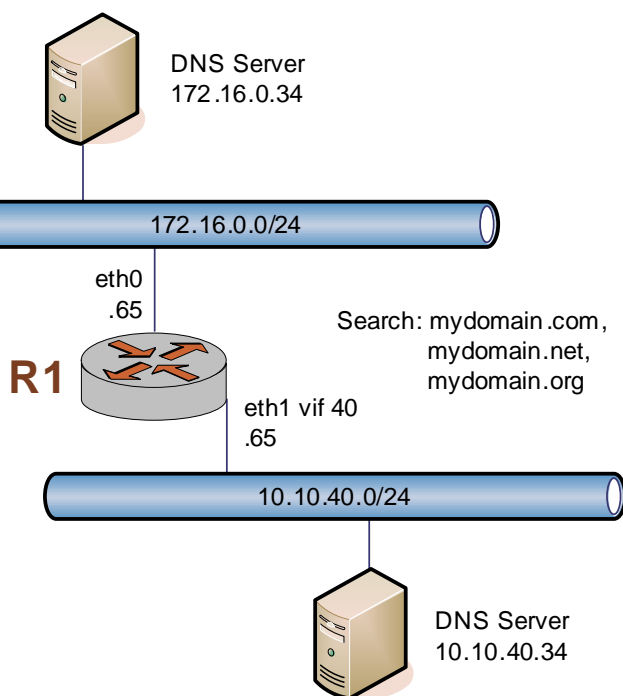
## Configuring DNS

This section presents the following topics:

- DNS Name Servers
- Domain Search Order

In this section, sample configurations are presented for DNS information. The DNS configuration used is shown in Figure 2-2.

Figure 2-2 DNS



This section includes the following examples:

- Example 2-6 Specifying DNS name servers
- Example 2-7 Setting search order for domain completion

## DNS Name Servers

DNS name servers are specified using the **system name-server** command.

Example 2-6 specifies two DNS servers for the system: one at 172.16.0.34, and the other at 10.10.40.34.

To specify DNS servers, perform the following steps in configuration mode:

### Example 2-6 Specifying DNS name servers

Step	Command
Specify the first DNS server.	<code>vyatta@R1# set system name-server 172.16.0.34</code> <code>[edit]</code>
Specify the second DNS server.	<code>vyatta@R1# set system name-server 10.10.40.34</code> <code>[edit]</code>

**Example 2-6** Specifying DNS name servers

Commit the change.	vyatta@R1# <b>commit</b> [edit]
Show configuration.	vyatta@R1# <b>show system name-server</b> name-server 172.16.0.34 name-server 10.10.40.34 [edit]

## Domain Search Order

You can specify a list of domains for the system to use to complete an unqualified host name. To define this list, specify the order in which domains are searched using the **system domain-search** command.

**NOTE** **system domain-name** and **system domain-search** are mutually exclusive. Only one of the two can be configured at any one time.

The **system domain-search** command requires you to enter each domain name separately, specified in the order you want them searched. A domain name can include letters, numbers, hyphens (“-”), and periods (“.”).

Example 2-7 directs the system to attempt domain completion in the following order: first, mydomain.com; second, mydomain.net; and last mydomain.org.

To specify domain search order, perform the following steps in configuration mode:

**Example 2-7** Setting search order for domain completion

Step	Command
Specify the first domain name.	vyatta@R1# <b>set system domain-search domain mydomain.com</b> [edit]
Specify the second domain name.	vyatta@R1# <b>set system domain-search domain mydomain.net</b> [edit]
Specify the third domain name.	vyatta@R1# <b>set system domain-search domain mydomain.org</b> [edit]
Commit the change.	vyatta@R1# <b>commit</b> [edit]
Show the configuration.	vyatta@R1# <b>show system domain-search</b> domain mydomain.com domain mydomain.net domain mydomain.org [edit]



## Configuring Date and Time

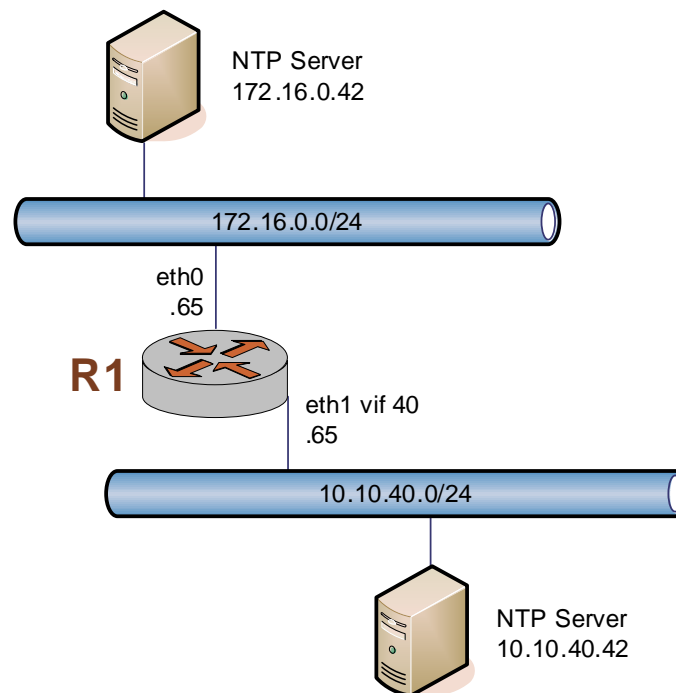
This section presents the following topics:

- Setting the Date
- Manually Synchronizing with an NTP Server
- Setting the Time Zone
- Using NTP for Automatic Synchronization

Date and time can either be set manually, or obtained by manually or automatically synchronizing the system with one or more Network Time Protocol (NTP) servers. Time zone must be manually set, and may be specified as an offset from Universal Coordinated Time (UTC) or as one of a number of supported literal time zones.

In this section, sample configurations are presented for maintaining date and time information. The sample configuration used is shown in Figure 2-3.

Figure 2-3 Date and time



This section includes the following examples:

- Example 2-8 Setting the date and time manually
- Example 2-9 Manually synchronizing the system with an NTP server
- Example 2-10 Setting the time zone as an offset from UTC
- Example 2-11 Setting the time zone as a time zone name

- Example 2-12 Using NTP for automatic synchronization

## Setting the Date

Example 2-8 manually sets the date to 1:15 PM exactly on April 24, 2007. The format is *MMDDhhmmCCYY*. Alternate formats are *MMDDhhmm*, *MMDDhhmmYY*, and *MMDDhhmmCCYY.ss*.

To manually set the date, perform the following steps in operational mode:

Example 2-8 Setting the date and time manually

Step	Command
Specify the date. The format is <i>MMDDhhmmCCYY</i> .	<pre>vyatta@R1:~\$ set date 042413152007 Tue Apr 24 13:15:00 GMT 2007 vyatta@R1:~\$</pre>

## Manually Synchronizing with an NTP Server

Example 2-9 manually synchronizes the system clock with the NTP server at 172.16.0.42.

Note that this merely performs a one-time synchronization. It does not set up an ongoing association with the NTP server. For information about setting up automatic synchronization, please see “Using NTP for Automatic Synchronization” on page 59.

To perform a one-time synchronization with an NTP server, perform the following steps in operational mode:

Example 2-9 Manually synchronizing the system with an NTP server

Step	Command
Specify the location of the NTP server.	<pre>vyatta@R1:~\$ set date ntp 172.16.0.42 Tue Apr 24 13:15:00 UTC 2007 vyatta@R1:~\$</pre>

## Setting the Time Zone

Time zone must be configured, using **system time-zone** command. To do this, you specify the amount by which your time zone is offset from UTC (coordinated universal time). UTC has the same time as the Greenwich time zone. The string giving the offset is enclosed in quotes.

The offset you specify is added to UTC to produce the local time. You can also use one of the support time zone name to indicate time zone. Again, the string supplying the time zone name must be enclosed in quotes.

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

Example 2-10 sets the time zone to 8 hours west of Greenwich, which is Pacific Standard Time.

To set the time zone using an offset from UTC, perform the following steps in configuration mode:

Example 2-10 Setting the time zone as an offset from UTC

Step	Command
Set the time zone.	<pre>vyatta@R1# set system time-zone "GMT+8" [edit] vyatta@R1#</pre>
Commit the information.	<pre>vyatta@R1# commit [edit]</pre>
Show the configuration.	<pre>vyatta@R1# show system time-zone time-zone GMT+8  [edit]</pre>

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.

Example 2-11 sets the time zone to Los Angeles time.

To set the time zone using a supported time zone name, perform the following steps in configuration mode:

Example 2-11 Setting the time zone as a time zone name

Step	Command
Specify the time zone as a time zone name.	vyatta@R1# <b>set system time-zone "Los Angeles"</b> [edit] vyatta@R1#
Commit the information.	vyatta@R1# <b>commit</b> [edit]
Show the configuration.	vyatta@R1# <b>show system time-zone</b> time-zone "Los Angeles"  [edit]

## Using NTP for Automatic Synchronization

To use NTP for automatic synchronization, you must create associations with the NTP servers. To create an association with an NTP server, use the **system ntp-server** command and specify the IP address of the server.

Example 2-12 configures two NTP servers: one at 172.16.0.42, and one at 10.10.40.42.

To specify NTP servers, perform the following steps in configuration mode:

Example 2-12 Using NTP for automatic synchronization

Step	Command
Specify a server at 176.16.0.42.	vyatta@R1# <b>set system ntp-server 172.16.0.42</b> [edit]
Specify a server at 10.10.40.42.	vyatta@R1# <b>set system ntp-server 10.10.40.42</b> [edit]
Commit the information.	vyatta@R1# <b>commit</b> [edit]

---

**Example 2-12 Using NTP for automatic synchronization**

---

Show the configuration. (Output is abbreviated here.)

```
vyatta@R1# show system
host-name R1
domain-name mydomain.com
domain-search {
    domain mydomain.com
    domain mydomain.net
    domain mydomain.org
}
name-server 172.16.0.34
name-server 10.10.40.34
time-zone "Los Angeles"
ntp-server 172.16.0.42
ntp-server 10.10.40.42
[edit]
```

---

## Monitoring System Information

---

This section presents the following topics:

- Showing Host Information
- Showing the Date and Time

This section includes the following examples:

- Example 2-13 Showing the system host name
- Example 2-14 Showing the system date and time

### Showing Host Information

To view the configured host name, use the **show host name** command in operational mode, as shown in Example 2-13:

---

**Example 2-13 Showing the system host name**

---

```
vyatta@R1:~$ show host name
R1
vyatta@R1:~$
```

---

## Showing the Date and Time

To view the time according to the system clock, use the **show host date** command in operational mode, as shown in Example 2-14:

Example 2-14 Showing the system date and time

---

```
vyatta@R1:~$ show host date
Tue Apr 24 22:23:07 GMT+8 2007
vyatta@R1:~$
```

---

# System Management Commands

This section presents the following commands.

## Configuration Commands

system domain-name <domain>	Sets the system's domain.
system domain-search domain <domain>	Defines a set of domains for domain completion.
system gateway-address <address>	Specifies the default gateway for the system.
system host-name <name>	Sets the host name for the system.
system name-server <address>	Specifies the DNS name servers available to the system.
system ntp-server <name>	Specifies the NTP servers to use when synchronizing the system's clock.
system options reboot-on-panic <value>	Allows you set system behavior on system panic.
system static-host-mapping host-name <name>	Defines a static mapping between a host name and an IP address.
system time-zone <zone>	Sets the time zone for the local system clock.

## Operational Commands

clear arp address <ipv4>	Clears the system's ARP cache for the specified IP address.
clear arp interface <ethx>	Clears the system's ARP cache for the specified interface.
clear console	Clears the user's console.
clear interfaces counters	Clears interface counters for all interfaces.
init-floppy	Formats a floppy diskette and prepares it to receive a configuration file.
reboot	Reboots the system.
set date	Sets the system date and time directly or specifies an NTP server to acquire it from.
show arp	Displays the system's ARP cache.
show date	Displays the system date and time.
show files	Displays file information.
show hardware cpu	Displays information about the system's processor.
show hardware dmi	Displays information about the system's DMI.
show hardware mem	Displays information about the system's memory.

show hardware pci	Displays information about the system's PCI bus.
show host	Displays host information for hosts reachable by the system.
show interfaces	Displays information about system interfaces.
show license	Displays Vyatta license information.
show ntp	Shows the status of configured NTP servers.
show system boot-messages	Displays boot messages generated by the kernel.
show system connections	Displays active network connections on the system.
show system kernel-messages	Displays messages in the kernel ring buffer.
show system memory	Displays system memory usage.
show system processes	Displays active system processes.
show system storage	Displays system file system usage and available storage space.
show system uptime	Displays information on how long the system has been running.
show system usb	Displays information about peripherals connected to the USB bus.
show tech-support	Provides a consolidated report of system information.
show version	Displays information about the version of system software.
terminal	Controls behaviors of the system terminal.

Some commands related to certain features of system management are located in other locations:

#### Related Commands Documented Elsewhere

full-upgrade	User management commands are described in "Chapter 6: Installation and Upgrade."
install-system	User management commands are described in "Chapter 6: Installation and Upgrade."
system login	User management commands are described in "Chapter 3: User Management."
system package	Software upgrade commands are described in "Chapter 6: Installation and Upgrade."
system syslog	System logging commands are described in "Chapter 4: Logging."



## clear arp address <ipv4>

Clears the system's ARP cache for the specified IP address.

---

### Syntax

**clear arp address** *ipv4*

---

### Command Mode

Operational mode.

---

### Parameters

<i>ipv4</i>	Removes the ARP entry for the specified IP address from the ARP cache.
-------------	--

---

### Default

None.

---

### Usage Guidelines

Use this command to remove ARP entries associated with a specific IP address from the ARP cache.

## clear arp interface <ethx>

Clears the system's ARP cache for the specified interface.

---

### Syntax

```
clear arp interface eth0..eth23
```

---

### Command Mode

Operational mode.

---

### Parameters

---

<i>eth0..eth23</i>	Clears the entire ARP cache for the specified Ethernet interface. The range of values is <b>eth0</b> to <b>eth23</b> .
--------------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to remove ARP entries associated with an Ethernet interface from the ARP cache.

# clear console

Clears the user's console.

---

## Syntax

**clear console**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to clear the screen of the console.

## clear interfaces counters

Clears interface counters for all interfaces.

---

### Syntax

**clear interfaces counters**

---

### Command Mode

Operational mode.

---

### Parameters

None.

---

### Default

None.

---

### Usage Guidelines

Use this command to clear the counters for all interfaces of all types, including ADSL, bridge, Ethernet, loopback, multilink, serial, and tunnel.

# init-floppy

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

---

## Syntax

**init-floppy**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

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 **/media/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 “y” in response to the question “Continue (y/n)? [y]” or pressing <Ctrl>+c.

Once the floppy disk has been formatted, the **config.boot** file is automatically saved to it. You can also save the **config.boot** configuration file to disk using the **save** command (see page 37).

---

## Examples

Example 2-15 prepares a floppy disk for receiving a configuration file and saves the running configuration to **/media/floppy/config/config.boot**.

Example 2-15 “init-floppy”: Initializing a floppy diskette for saving configuration files

---

```
vyatta@R1:~$ init-floppy
This will erase all data on floppy /dev/fd0.
Your configuration was saved in:
/media/floppy/config/config.boot
vyatta@R1:~$
```

---

# reboot

Reboots the system.

---

## Syntax

**reboot**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to reboot the system.

Before the system reboots, a message from the root user is broadcast to all logged on users warning them of the reboot.

---

## Examples

Example 2-16 reboots the system.

Example 2-16 “reboot”: Rebooting the system

---

```
vyatta@R1:~$ reboot
Proceed with reboot? [confirm]y

Broadcast message from root@R1 (tty1) (Mon Jan 21 17:52:37 2008):

The system is going down for reboot NOW!
```

---

## set date

Sets the system date and time directly or specifies an NTP server to acquire it from.

---

### Syntax

```
set date { datetime | ntp ntpserver }
```

---

### Command Mode

Operational mode.

---

### Parameters

---

<i>datetime</i>	Set the date and time directly using one of the following formats:  MMDDhhmm MMDDhhmmYY MMDDhhmmCCYY MMDDhhmmCCYY.ss  Note that the hour field (hh) uses the 24 hour clock (e.g. 3:00 pm would be represented as 15 in the hour field).
<i>ntpserver</i>	Specifies a Network Time Protocol (NTP) to acquire the current time from. You can specify either an IPv4 address or a hostname to identify the NTP server.

---

---

### Default

None.

---

### Usage Guidelines

Use this command to set the system date and time either directly or by specifying a Network Time Protocol (NTP) server to acquire the date and time from. If a timezone has not been configured then GMT is assumed. The timezone is set using the **system time-zone <zone>** command (see page 124).



---

## Examples

Example 2-17 sets the system date and time to May 15, 2008 at 10:55 pm (assuming that the timezone is set to Pacific Daylight Time).

Example 2-17 Set the date and time directly

---

```
vyatta@R1:~$ set date 051522552008
Thu May 15 22:55:00 PDT 2008
vyatta@R1:~$
```

---

Example 2-18 sets the system date and time using an NTP server.

Example 2-18 Set the date and time using an NTP server

---

```
vyatta@R1:~$ set date ntp 69.59.150.135
15 May 23:00:00 ntpdate[7038]: step time server 69.59.150.135
offset 425.819267 sec
vyatta@R1:~$
```

---

# show arp

Displays the system's ARP cache.

---

## Syntax

**show arp**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to display the system's ARP cache.

Table 2-1 shows possible ARP states.

Table 2-1 ARP states

State	Description
incomplete	Address resolution is currently being performed 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 system 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 2-19 shows the ARP cache of systemR1.

Example 2-19 “show arp”: Displaying the ARP cache

---

```
vyatta@R1:~$ show arp
Address      HWtype  HWaddress      Flags Mask  Iface
172.16.215.1 ether    00:12:D9:74:BE:91  C          eth0
10.1.0.1     ether    00:04:23:09:0F:79  C          eth0
vyatta@R1:~$
```

---

# show date

Displays the system date and time.

---

## Syntax

**show date [utc]**

---

## Command Mode

Operational mode.

---

## Parameters

---

<b>utc</b>	Shows the date and time in Coordinated Universal Time.
------------	--

---

---

## Default

None.

---

## Usage Guidelines

Use this command to display the system date and time in either local time or UTC time.

---

## Examples

Example 2-20 shows the system date and time on R1.

Example 2-20 “show date”: Displaying the system date and time

---

```
vyatta@R1:~$ show date
Tue May 20 17:27:07 PDT 2008
vyatta@R1:~$
```

---

# show files

Displays file information.

---

## Syntax

**show files** *directory*

---

## Command Mode

Operational mode.

---

## Parameters

---

<i>directory</i>	Mandatory. The absolute path to the file to be shown. Note that the root directory (“/”) itself cannot be shown.
------------------	--

---

---

## Default

None.

---

## Usage Guidelines

Use this command to display information about files in the specified directory.

---

## Examples

Example 2-21 shows information about the files in **/opt/vyatta/etc/config** on R1.

Example 2-21 “show files <directory>”: Displaying file information

---

```
vyatta@R1:~$ show files /opt/vyatta/etc/config
total 8.0K
-rw-rw---- 1 root vyattacfg 777 May 20 10:13 config.boot
-rw-r----- 1 root root      712 May 20 10:13
config.boot.2008-05-20-1713.pre-migration
vyatta@R1:~$
```

---

# show hardware cpu

Displays information about the system's processor.

---

## Syntax

**show hardware cpu**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

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

---

## Examples

Example 2-22 shows CPU information on R1.

Example 2-22 "show hardware cpu": Showing CPU information

---

```
vyatta@R1:~$ show hardware cpu
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 15
model name    : Intel(R) Xeon(R) CPU           E5310  @ 1.60GHz
stepping      : 8
cpu MHz       : 1595.101
cache size    : 4096 KB
fdiv_bug      : no
hlt_bug       : no
f00f_bug      : no
coma_bug      : no
fpu           : yes
```

```
fpu_exception    : yes
cpuid level      : 10
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 nx constant_tsc up arch_perfmon pebs bts pni ds_cpl ssse3 dca
bogomips         : 3213.51
clflush size     : 64
power management:

vyatta@R1:~$
```

---

# show hardware dmi

Displays information about the system's DMI.

---

## Syntax

**show hardware dmi**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to view information about the system's desktop management interface (DMI). The DMI provides a standard framework for managing resources in the device.

---

## Examples

Example 2-23 shows DMI information on R1.

Example 2-23 "show hardware dmi": Showing DMI information

---

```
vyatta@R1:~$ show hardware dmi
bios_date: 04/17/2006
bios_vendor: Phoenix Technologies LTD
bios_version: 6.00
board_asset_tag:
board_name: 440BX Desktop Reference Platform
board_vendor: Intel Corporation
board_version: None
chassis_asset_tag: No Asset Tag
chassis_type: 1
chassis_vendor: No Enclosure
chassis_version: N/A
product_name: VMware Virtual Platform
product_version: None
```



```
sys_vendor: VMware, Inc.  
vyatta@R1:~$
```

---

## show hardware mem

Displays information about the system's memory.

---

### Syntax

**show hardware mem**

---

### Command Mode

Operational mode.

---

### Parameters

None.

---

### Default

None.

---

### Usage Guidelines

Use this command to view information about the system memory.

---

### Examples

Example 2-24 shows memory information on R1.

Example 2-24 “show hardware mem”: Showing memory information

---

```
vyatta@R1:~$ show hardware mem
MemTotal:      515972 kB
MemFree:       341468 kB
Buffers:       28772 kB
Cached:        116712 kB
SwapCached:    0 kB
Active:        35912 kB
Inactive:      117272 kB
HighTotal:     0 kB
HighFree:      0 kB
LowTotal:      515972 kB
LowFree:       341468 kB
SwapTotal:     0 kB
SwapFree:      0 kB
Dirty:         0 kB
```

```
Writeback:          0 kB
AnonPages:         7700 kB
Mapped:            4048 kB
Slab:              14644 kB
SReclaimable:      9440 kB
SUnreclaim:        5204 kB
PageTables:        288 kB
NFS_Unstable:       0 kB
Bounce:            0 kB
CommitLimit:      257984 kB
Committed_AS:     21636 kB
VmallocTotal:     507896 kB
VmallocUsed:       3896 kB
VmallocChunk:     503932 kB
vyatta@R1:~$
```

---

## show hardware pci

Displays information about the system's PCI bus.

---

### Syntax

**show hardware pci [detailed]**

---

### Command Mode

Operational mode.

---

### Parameters

---

<b>detailed</b>	Shows detailed information about the PCI bus.
-----------------	---

---

---

### Default

None.

---

### Usage Guidelines

Use this command to view information about the peripheral component interconnect (PCI) bus. The PCI provides communication among the system's peripheral components and the processor.

---

### Examples

Example 2-25 shows PCI information on R1.

Example 2-25 “show hardware pci”: Showing PCI bus information

---

```
vyatta@R1:~$ show hardware pci
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX -
82443BX/ZX/DX Host bridge (rev 01)
00:01.0 PCI bridge: Intel Corporation 440BX/ZX/DX -
82443BX/ZX/DX AGP bridge (rev 01)
00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA
(rev 08)
00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE
(rev 01)
00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev
08)
```

```
00:0f.0 VGA compatible controller: VMware Inc Abstract SVGA II
Adapter
00:10.0 SCSI storage controller: LSI Logic / Symbios Logic
53c1030 PCI-X Fusion-MPT Dual Ultra320 SCSI (rev 01)
00:11.0 Ethernet controller: Advanced Micro Devices [AMD] 79c970
[PCnet32 LANCE] (rev 10)
vyatta@R1:~$
```

---

# show host

Displays host information for hosts reachable by the system.

---

## Syntax

```
show host {lookup hostname / lookup ipv4 / name | date | os}
```

---

## Command Mode

Operational mode.

---

## Parameters

<b>lookup</b> <i>hostname</i>	Shows the canonical name and IP address plus any configured aliases recorded in the name server for the host with the specified name.
<b>lookup</b> <i>ipv4</i>	Shows the canonical name and IP address plus any configured aliases recorded in the name server for the host with the specified IP address.
<b>date</b>	Shows the date and time according to the system clock.
<b>name</b>	Shows the name configured for this system.
<b>os</b>	Shows details about the system's operating system.

---

## Default

None.

---

## Usage Guidelines

Use this command to view information configured for the host.

---

## Examples

Example 2-26 shows host information for R2.

Example 2-26 “show host lookup”: Looking up network hosts

```
vyatta@R1:~$ show host lookup R2
```

```
R2.vyatta.com      A      10.1.0.3
vyatta@R1:~$
```

---

Example 2-27 shows the name configured for R1.

Example 2-27 “show host name”: Showing network host names

---

```
vyatta@R1:~$ show host name
R1
vyatta@R1:~$
```

---

Example 2-28 shows the date and time according to the system clock.

Example 2-28 “show host date”: Showing the system date and time

---

```
vyatta@R1:~$ show host date
Mon Jan 21 17:28:47 PST 2008
vyatta@R1:~$
```

---

Example 2-29 shows information about the operating system.

Example 2-29 “show host os”: Showing operating system information

---

```
vyatta@R1:~$ show host os
Linux R1 2.6.23-1-486-vyatta #1 SMP Tue Jan 15 02:00:31 PST 2008
i686 GNU/Linux
vyatta@R1:~$
```

---

# show interfaces

Displays information about system interfaces.

---

## Syntax

```
show interfaces [detailed | system [enabled]]
```

---

## Command Mode

Operational mode.

---

## Parameters

<b>detailed</b>	Displays detailed information about all the interfaces available on your system.
<b>system</b>	Displays all the physical interfaces available on your system.
<b>enabled</b>	Shows only enabled system interfaces known to the operating system kernel.

---

## Default

Displays information for all interfaces configured on the system.

---

## 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 system. You can see specific information by using other versions of this command:

To see all the physical interfaces known to the operating system kernel, use the **system** option. This option differs from the other versions of this command: the other versions show interfaces that have been configured on the system, while the **system** option shows all the physical interfaces available on your system (that is, the physical interfaces known to the operating system kernel).

The physical interfaces available to you determine which interfaces you will be able to configure and view, because you cannot configure or view an interface that does not physically exist on the system.



---

## Examples

Example 2-30 shows the first screen of output for **show interfaces system enabled**.

Example 2-30 “show interfaces”: Displaying interface information

---

```
vyatta@R1:~$ show interfaces system enabled
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast qlen 100
    link/ether 00:30:48:82:e2:0c brd ff:ff:ff:ff:ff:ff
    inet 10.1.0.54/24 brd 10.1.0.255 scope global eth0
    inet6 fe80::230:48ff:fe82:e20c/64 scope link
        valid_lft forever preferred_lft forever

    RX:  bytes    packets    errors    dropped    overrun    mcast
          348646         4144          0          0          0          0
    TX:  bytes    packets    errors    dropped    carrier collisions
          168294         1594          0          0          0          0

eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast qlen 10
    link/ether 00:30:48:82:e2:0d brd ff:ff:ff:ff:ff:ff
    inet 172.16.215.2/24 brd 172.16.215.255 scope global eth1
    inet6 fe80::230:48ff:fe82:e20d/64 scope link
        valid_lft forever preferred_lft forever

    RX:  bytes    packets    errors    dropped    overrun    mcast
          1384          11          0          0          0          0
    TX:  bytes    packets    errors    dropped    carrier collisions
          1990          18          0          0          0          0

eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
lines 1-23
```

---

## show license

Displays Vyatta license information.

---

### Syntax

**show license**

---

### Command Mode

Operational mode.

---

### Parameters

None.

---

### Default

None.

---

### Usage Guidelines

Use this command to view Vyatta license information.

---

## Examples

Example 2-31 shows the first screen of output for **show license**.

Example 2-31 “show license”: Displaying license information

---

```
GNU GENERAL PUBLIC LICENSE
                        Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
                        51 Franklin St, Fifth Floor, Boston, MA
02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

                                Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General
Public
License is intended to guarantee your freedom to share and change
free
software--to make sure the software is free for all its users.
This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors
commit to
using it. (Some other Free Software Foundation software is
covered by
the GNU Library General Public License instead.) You can apply
it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure
that you
have the freedom to distribute copies of free software (and
charge for
this service if you wish), that you receive source code or can
get it
if you want it, that you can change the software or use pieces
of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the
rights.
These restrictions translate to certain responsibilities for you
```

if you  
distribute copies of the software, or if you modify it.

---

## show ntp

Shows the status of configured NTP servers.

---

### Syntax

```
show ntp {host | ipv4 | 69.59.150.135 }
```

---

### Command Mode

Operational mode.

---

### Parameters

<i>host</i>	Shows the status of the connection to the NTP server with the specified host name.
<i>ipv4</i>	Shows the status of the connection to the NTP server at the specified IPv4 address.
<b>69.59.150.135</b>	Shows the status of the connection to the Vyatta NTP server, specified by IP address.

---

### Default

None.

---

### 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 system 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.

NTP server connections are configured using the **system ntp-server <name>** command (see page 118).

---

### Examples

Example 2-32 shows the NTP server at IP address 69.59.150.135.

Example 2-32 “show ntp”: Showing configured NTP servers

---

```
vyatta@R1:~$ show ntp 69.59.150.135
```

```
server 69.59.150.135, stratum 3, offset 46.614524, delay 0.03207
22 Jan 12:20:36 ntpdate[10192]: step time server 69.59.150.135
offset 46.614524 sec
vyatta@R1:~$
```

---

# show system boot-messages

Displays boot messages generated by the kernel.

---

## Syntax

**show system boot-messages** [all]

---

## Command Mode

Operational mode.

---

## Parameters

<b>all</b>	Displays all kernel boot messages.
------------	------------------------------------

---

## Default

A subset of the full list of kernel boot messages is displayed.

---

## Usage Guidelines

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

---

## Examples

Example 2-33 shows the first screen of output for **show system boot-messages**.

Example 2-33 “show system boot-messages”: Displaying startup messages

```
vyatta@R1:~$ show system boot-messages
Linux version 2.6.23-1-486-vyatta (autobuild@sydney) (gcc
version 4.2.3 20071123 (prerelease) (Debian 4.2.2-4)) #1 SMP Fri
Jan 18 07:17:50 PST 2008
BIOS-provided physical RAM map:
  BIOS-e820: 0000000000000000 - 000000000009f800 (usable)
  BIOS-e820: 000000000009f800 - 00000000000a0000 (reserved)
  BIOS-e820: 00000000000f0000 - 0000000000100000 (reserved)
  BIOS-e820: 0000000000100000 - 0000000001fee0000 (usable)
  BIOS-e820: 0000000001fee0000 - 0000000001fee3000 (ACPI NVS)
  BIOS-e820: 0000000001fee3000 - 0000000001fef0000 (ACPI data)
  BIOS-e820: 0000000001fef0000 - 0000000001fff0000 (reserved)
  BIOS-e820: 00000000fec00000 - 00000000100000000 (reserved)
```

```
0MB HIGHMEM available.
510MB LOWMEM available.
found SMP MP-table at 000f5a20
Entering add_active_range(0, 0, 130784) 0 entries of 256 used
Zone PFN ranges:
  DMA             0 ->      4096
  Normal          4096 ->   130784
  HighMem        130784 ->   130784
Movable zone start PFN for each node
early_node_map[1] active PFN ranges
    0:             0 ->   130784
On node 0 totalpages: 130784
:
```

---



# show system connections

Displays active network connections on the system.

## Syntax

**show system connections**

## Command Mode

Operational mode.

## Parameters

None.

## Default

None:

## Usage Guidelines

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

## Examples

Example 2-34 shows the first screen of output for **show system connections**.

Example 2-34 “show system connections”: Displaying active connections

```
vyatta@R1:~$ show system connections
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:179             0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN
tcp        0      0 192.168.1.77:22        192.168.1.102:2449      ESTABLISHED
tcp6       0      0 :::2606                :::*                    LISTEN
tcp6       0      0 :::80                  :::*                    LISTEN
tcp6       0      0 :::179                  :::*                    LISTEN
tcp6       0      0 :::22                  :::*                    LISTEN
udp        0      0 192.168.1.77:123       0.0.0.0:*               *
udp        0      0 127.0.0.1:123          0.0.0.0:*               *
udp        0      0 0.0.0.0:123            0.0.0.0:*               *
udp6       0      0 fe80::20c:29ff:fe68:123 :::*                    *
udp6       0      0 ::1:123                :::*                    *
```

```
udp6      0      0 :::123          :::*
raw6      0      0 :::58          :::*          7
raw6      0      0 :::89          :::*          7
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags      Type       State      I-Node    Path
unix  12      [ ]       DGRAM          10203    /dev/log
unix   2      [ ACC ]    STREAM    LISTENING   10657    /var/run/vyatta/quagga/zserv.api
unix   2      [ ACC ]    STREAM    LISTENING   10665    /var/run/vyatta/quagg
:
```

# show system kernel-messages

Displays messages in the kernel ring buffer.

---

## Syntax

**show system kernel-messages**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

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

---

## Examples

Example 2-35 shows the first screen of output for **show system kernel-messages**.

Example 2-35 “show system kernel-messages”: Displaying messages from the kernel

---

```
vyatta@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 - 0000000000fee0000 (usable)
  BIOS-e820: 0000000000fee0000 - 0000000000fee3000 (ACPI NVS)
  BIOS-e820: 0000000000fee3000 - 0000000000fef0000 (ACPI data)
  BIOS-e820: 0000000000fef0000 - 0000000000ff00000 (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
:
```

---

## show system memory

Displays system memory usage.

---

### Syntax

```
show system memory [quagga]
```

---

### Command Mode

Operational mode.

---

### Parameters

---

<b>quagga</b>	Displays memory usage by the Quagga subsystem.
---------------	--

---

---

### Default

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 2-36 shows information about memory usage on R1.

Example 2-36 “show system memory”: Displaying information about memory usage

```
vyatta@R1:~$ show system memory
      total      used      free      shared    buffers    cached
Mem:    515484    286708    228776         0      48224    197228
Swap:        0         0         0
Total:   515484    286708    228776
vyatta@R1:~$
```

# show system processes

Displays active system processes.

---

## Syntax

**show system processes [summary]**

---

## Command Mode

Operational mode.

---

## Parameters

---

<b>summary</b>	Shows a summary of system usage.
----------------	----------------------------------

---

---

## Default

Lists all processes currently running on the system.

---

## Usage Guidelines

Use this command to see information about processes currently running on the system.

---

## Examples

Example 2-37 shows the first screen of output for **show system processes**.

Example 2-37 “show system processes”: Displaying process information

---

```
vyatta@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
:			

---

# show system storage

Displays system file system usage and available storage space.

---

## Syntax

**show system storage**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

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 2-38 shows file system usage information for R1.

Example 2-38 “show system storage”: Displaying file system and storage information

---

```
vyatta@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
vyatta@R1:~$
```

---



# show system uptime

Displays information on how long the system has been running.

---

## Syntax

**show system uptime**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to see how long the system has been running, the number of users currently logged in, and the average system load.

---

## Examples

Example 2-39 shows file system usage information for R1.

Example 2-39 “show system storage”: Displaying file system and storage information

---

```
vyatta@R1:~$ show system uptime
20:45:59 up 3:04, 2 users, load average: 0.00, 0.00, 0.00
vyatta@R1:~$
```

---

# show system usb

Displays information about peripherals connected to the USB bus.

---

## Syntax

**show system usb**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to see which peripherals are connected to the USB bus.

---

## Examples

Example 2-40 shows system USB information for R1.

Example 2-40 “show system usb”: Displaying USB peripheral information

---

```
vyatta@R1:~$ show system usb
Bus 001 Device 002: ID 0d49:7212 Maxtor
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
vyatta@R1:~$
```

---

# show tech-support

Provides a consolidated report of system information.

---

## Syntax

```
show tech-support [save [filename]]
```

---

## Command Mode

Operational mode.

---

## Parameters

<b>save</b>	<p>Saves the support information to a file in the <b>/opt/vyatta/etc/config/support</b> directory. The file name takes the format <i>hostname.tech-support.timestamp</i>, where <i>hostname</i> is the host name configured for the Vyatta device and <i>timestamp</i> is the time the file was saved in the format <i>YYYY-MM-DD-hhmmss</i>.</p> <p>A rotation mechanism is used to limit the number of output files to 10; that is, creating an eleventh file causes the oldest file to be deleted.</p>
<i>filename</i>	<p>Saves the support information to the file <i>filename.hostname.tech-support.timestamp</i>, where <i>hostname</i> is the host name configured for the Vyatta device and <i>timestamp</i> is the time the file was saved. If an absolute path is prepended to filename, the file is saved in that location. Otherwise, the file is saved to a location relative to the default path, which is <b>/opt/vyatta/etc/config/support</b> directory.</p>

---

## Default

Information is sent to the console.

---

## 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 2-41 shows the first screen of a technical report.

Example 2-41 “show tech-support” Displaying consolidated system information

---

```
vyatta@R1:~$ show tech-support
-----
Current time
-----
Wed Oct 29 01:28:07 GMT 2008

-----
OFR Version and Package Changes
-----
Version   : 888.islavista
Copyright: 2006-2008 Vyatta, Inc.
Built by  : root@vyatta.com
Built on  : Tue Oct 28 11:25:54 UTC 2008
Build ID  : 2008-10-28-0749-f64e188
Boot via  : livedcd
Uptime    : 01:28:07 up 1:28, 2 users, load average: 0.00,
0.00, 0.00

-----
Installed Packages
-----
Desired=Unknown/Install/Remove/Purge/Hold
|
Status=Not/Inst/Cfg-files/Unpacked/Failed-cfg/Half-inst/trig-aW
ait/Trig-pend
|/ Err?=(none)/Hold/Reinst-required/X=both-problems
(Status,Err: uppercase=bad)
:
```

---

# show version

Displays information about the version of system software.

---

## Syntax

```
show version [all | added | deleted | downgraded | quagga | upgraded]
```

---

## Command Mode

Operational mode.

---

## Parameters

<b>all</b>	Show all packages that have been added, deleted, downgraded, or upgraded since the last baseline version upgrade.
<b>added</b>	Show all packages that have been upgraded since the last baseline version upgrade.
<b>deleted</b>	Show all packages that have been deleted since the last baseline version upgrade.
<b>downgraded</b>	Show all packages that have been downgraded since the last baseline version upgrade.
<b>quagga</b>	Shows the version of quagga code used in the system.
<b>upgraded</b>	Show all packages that have been upgraded since the last baseline version upgrade.

---

## Default

A brief summary of version information is shown. Detailed information about constituent packages is not shown.

---

## Usage Guidelines

Use this command to see what package changes have occurred since the last time a full version upgrade was performed.

The information shown is always with respect to the last full version upgrade. Therefore, for example:

- Immediately following a full version upgrade, issuing a **show version all** command will show no changes.
- If a package is added after upgrading, issuing a **show version all** will show the added package.
- However, if the added package is then deleted again, issuing a **show version all** will show no change, since the system is now in the same state as immediately after the full version upgrade.

Keep in mind that if you delete a package, and packages depending on the deleted package are also removed.

Example 2-42 shows sample output for the **show version** command used with no option.

Example 2-42 “show version”: Displaying a summary of version information

---

```
vyatta@vyatta:~$ show version
Version :      888.islavista
Copyright:    2006-2008 Vyatta, Inc.
Built by :    root@vyatta.com
Built on :    Tue Oct 28 11:25:54 UTC 2008
Build ID :    2008-10-28-0749-f64e188
Boot via :    lived
Uptime  :    01:29:58 up  1:30,  2 users,  load average: 0.00,
0.00, 0.00
vyatta@vyatta:~$
```

---

Example 2-43 shows the first page of sample output for the **show version all** command.

Example 2-43 “show version all”: Displaying software package version information

---

```
vyatta@vyatta:~$ show version all
Version :      888.islavista
Copyright:    2006-2008 Vyatta, Inc.
Built by :    root@vyatta.com
Built on :    Tue Oct 28 11:25:54 UTC 2008
Build ID :    2008-10-28-0749-f64e188
Boot via :    lived
Uptime  :    01:29:58 up  1:30,  2 users,  load average: 0.00,
0.00, 0.00

ADDED:
Aii aptitude 0.4.4-4
Aii libc6 2.3.6.ds1-13
Aii libdb4.4 4.4.20-8
```

---

```
Aii libexpat1 1.95.8-3.4
Aii libncurses5 5.5-5
Aii libnetaddr-ip-perl 3.14-2
Aii libpam0g 0.79-4
Aii libsasl2 2.1.22.dfsg1-8
Aii libtasn1-3 0.3.6-2
Aii libwrap0 7.6.dbs-13
Aii snmp 5.2.3-7
Aii supported-version 2.2
:
```

---

Example 2-44 shows sample output for the **show version added** command.

Example 2-44 “show version added”: Displaying information about added software packages

---

```
vyatta@vyatta:~$ show version added
Version :      888.islavista
Copyright:    2006-2008 Vyatta, Inc.
Built by :    root@vyatta.com
Built on :    Tue Oct 28 11:25:54 UTC 2008
Build ID :    2008-10-28-0749-f64e188
Boot via :    livedd
Uptime  :    01:29:58 up 1:30, 2 users, load average: 0.00,
0.00, 0.00
```

ADDED:

```
Aii aptitude 0.4.4-4
Aii libc6 2.3.6.ds1-13
Aii libdb4.4 4.4.20-8
Aii libexpat1 1.95.8-3.4
Aii libncurses5 5.5-5
Aii libnetaddr-ip-perl 3.14-2
Aii libpam0g 0.79-4
Aii libsasl2 2.1.22.dfsg1-8
Aii libtasn1-3 0.3.6-2
Aii libwrap0 7.6.dbs-13
Aii snmp 5.2.3-7
Aii supported-version 2.2
Aii sysvinit 2.86.ds1-38
Aii tasksel 2.66
Aii vyatta-bgp 1.4-9
Aii vyatta-cli 2.1.1-9
Aii vyatta-config-migrate 2.1.1-4
Aii vyatta-dhcp-support 2.1.1-4
Aii vyatta-firewall 1.4-9
```

```
Aii vyatta-nat 2.1.1-5
Aii vyatta-nat-cli 2.1.1-4
Aii vyatta-nat-xorp 2.1.1-3
Aii vyatta-ospf 1.4-9
Aii vyatta-rip 1.4-9
Aii vyatta-xg 1.4-9
Aii zliblg 1.2.3-13
:
```

---



## system domain-name <domain>

Sets the system's domain.

---

### Syntax

```
set system domain-name domain
delete system domain-name
set system domain-name
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

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

---

### Parameters

<i>domain</i>	Mandatory. The domain where the system resides; for example, "vyatta.com". The format is a string containing letters, numbers, hyphens ("-") and one period.
---------------	--

---

### Default

None.

---

### Usage Guidelines

Use this command to set the system's domain.

Note that both **domain-name** and **domain-search** cannot be configured simultaneously - they are mutually exclusive.

Use the **set** form of this command to specify the domain name to be used by the system.

Use the **delete** form of this command to remove the domain name.

Use the **show** form of this command to view domain name configuration.

## system domain-search domain <domain>

Defines a set of domains for domain completion.

---

### Syntax

```
set system domain-search domain domain
delete system domain-search domain domain
show system domain-search domain
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  domain-search {
    domain text
  }
}
```

---

### Parameters

---

<i>domain</i>	Mandatory. Multi-node. The 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 <b>vyatta.com</b> . Letters, numbers, hyphens (“-”) and one period (“.”) are allowed.  You can specify up to 6 domains by creating up to 6 <b>domain-search</b> multi-nodes.
---------------	--

---

---

### Default

None.

---

## Usage Guidelines

Use this command to list up to 6 domains to be searched in DNS lookup requests.

When the system receives an unqualified host name, it attempts to form a Fully Qualified Domain Name (FQDN) by appending the domains in this list to the host name. The system tries each domain name in turn, in the order in which they were configured. If none of the resulting FQDNs succeeds, the name is not resolved and an error is reported.

Note that both **domain-name** and **domain-search** cannot be configured simultaneously - they are mutually exclusive.

Use the **set** form of this command to add a domain to the search list. Note that you cannot use **set** to change a domain name in the list. To replace an incorrect domain, delete it and replace it with a new one.

Use the **delete** form of this command to remove a domain name from the list.

Use the **show** form of this command to view the list of domain names.

## system gateway-address <address>

Specifies the default gateway for the system.

---

### Syntax

```
set system gateway-address ipv4
delete system gateway-address
show system gateway-address
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
    gateway-address ipv4
}
```

---

### Parameters

<i>address</i>	Mandatory. The IPv4 address of the default gateway.
----------------	---

---

### Default

None.

---

### Usage Guidelines

Use this command to set the location of the default gateway.

The default gateway is the location where packets are routed when the destination does not match any specific routing entries. Only one default gateway can be set per system.

Use the **set** form of this command to specify the address of default gateway.

Use the **delete** form of this command to remove the default gateway. Note that, in most cases, traffic cannot be routed correctly if a default gateway is not specified.

Use the **show** form of this command to view the address of the default gateway.

## system host-name <name>

Sets the host name for the system.

---

### Syntax

**set system host-name** *name*

**delete system host-name**

**show system host-name**

---

### Command Mode

Configuration mode.

---

### Configuration Statement

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

---

### Parameters

---

<i>name</i>	The name you want to give the system. Letters, numbers, and hyphens (“-”) only are allowed.  The default is “vyatta”. If you delete the host name, or if you try to delete the <b>system</b> node, the host name reverts to the default.
-------------	--

---

---

### Default

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.

---

### Usage Guidelines

Use the **set** form of this command to specify a host name for the system. 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 system shell and log back in again.

Use the **delete** form of this command to remove the host name. The host name “vyatta” will be used by default.

Use the **show** form of this command to view the host name.

## system name-server <address>

Specifies the DNS name servers available to the system.

---

### Syntax

```
set system name-server address
delete system name-server address
show system name-server
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

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

---

### Parameters

<i>ipv4</i>	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 <b>name-server</b> configuration node.
-------------	---

---

### Default

None.

---

### Usage Guidelines

Use this command to specify domain name servers (DNS) for the system.

Use the **set** form of this command to specify a name server for the system. Note that you cannot modify a DNS name server entry using the **set** command. To replace a name server entry, delete the entry and create a new one.

Use the **delete** form of this command to remove a name server.

Use the **show** form of this command to view the name servers that have been defined.

## system ntp-server <name>

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

---

### Syntax

```
set system ntp-server server
delete system ntp-server server
show system ntp-server
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
    ntp-server [ipv4/text] {}
}
```

---

### Parameters

<i>server</i>	Multi-node. The IP address or host name of an NTP server. The system will automatically obtain the system date and time from the specified server(s).  You can specify multiple NTP servers by creating multiple instances of the <b>name-server</b> configuration node.
---------------	--

---

### Default

By default, the system uses the NTP server at **ntp.vyatta.com**.

---

## Usage Guidelines

Use this command to specify NTP servers for the system.

Use the **set** form of this command to specify an NTP server for the system. Note that you cannot modify an NTP server entry using the **set** command. To replace an NTP server entry, delete the entry and create a new one.

Use the **delete** form of this command to remove an NTP server.

Use the **show** form of this command to view the NTP servers that have been defined.



## system options reboot-on-panic <value>

Allows you set system behavior on system panic.

---

### Syntax

**set system options reboot-on-panic** *value*

**delete system options reboot-on-panic**

**show system options reboot-on-panic**

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {  
    options {  
        reboot-on-panic [true|false]  
    }  
}
```

---

### Parameters

---

<i>value</i>	Mandatory. Indicates whether or not the system should automatically reboot if a kernel panic occurs. Supported values are as follows:  <b>true:</b> The system reboots if a kernel panic occurs.  <b>false:</b> The system does not reboot if a kernel panic occurs.
--------------	--

---

---

### Default

The default is **true**.

---

## Usage Guidelines

Configuring the system not to reboot on kernel panic allows you to examine information that might help you determine the cause of the panic.

Use the **set** form of this command to specify whether or not to reboot on kernel panic.

Use the **delete** form of this command to restore this option to its default value.

Use the **show** form of this command to view configuration for this option.

## system static-host-mapping host-name <name>

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

---

### Syntax

**set system static-host-mapping host-name** *name* [**inet** *address* | **alias** *alias*]

**delete system static-host-mapping host-name** *name* [**inet** | **alias**]

**show system static-host-mapping host-name** *name* [**inet** | **alias**]

---

### Command Mode

Configuration mode.

---

### Configuration Statement

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

---

### Parameters

<i>name</i>	<p>Multi-node. The Fully Qualified Domain Name (FQDN) name being statically mapped to an IP address (for example, <b>router1@mydomain.com</b>). Letters, numbers, periods (“.”) and hyphens (“-”) only are allowed.</p> <p>You can define multiple mappings by creating multiple <b>host-name</b> configuration nodes.</p>
<i>address</i>	<p>Mandatory. The IPv4 address of the interface being statically mapped to the host name.</p>
<i>alias</i>	<p>Optional. Multi-node. An alias for the interface. Letters, numbers, and hyphens are allowed.</p> <p>You can define multiple aliases for a host by creating multiple <b>alias</b> configuration nodes.</p>

---

## Default

None.

---

## Usage Guidelines

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

Use the **set** form of this command to create a new static mapping between a host name and an IP address, assign an address, or specify an alias. Note that you cannot use **set** to change the host name. To change the host name, delete the mapping entry and create a new one with the correct host name.

Use the **delete** form of this command to remove a static mapping, an address, or an alias.

Use the **show** form of this command to view a static mapping, an address, or an alias.

## system time-zone <zone>

Sets the time zone for the local system clock.

---

### Syntax

**set system time-zone** *zone*

**delete system time-zone**

**show system time-zone**

---

### Command Mode

Configuration mode.

---

### Configuration Statement

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

---

### Parameters

---

<i>zone</i>	<p>A string representing the time-zone and offset from UTC, enclosed in double quotes.</p> <p>The format is “<b>GMT</b> [{+   -} <i>h</i>]”, where <i>h</i> is a number from 1 to 12 representing the hours offset from GMT (see “Usage Guidelines,” below). The string must be enclosed in double quotes. The following time zone names, enclosed in double quotes, are also accepted:</p> <p>“<b>Los Angeles</b>”: Sets the time zone to Los Angeles time.</p> <p>“<b>New York</b>”: Sets the time zone to New York time.</p> <p>“<b>Denver</b>”: Sets the time zone to Denver time.</p> <p>“<b>Chicago</b>”: Sets the time zone to Chicago time.</p> <p>“<b>Anchorage</b>”: Sets the time zone to Anchorage time.</p> <p>“<b>Honolulu</b>”: Sets the time zone to Honolulu time.</p> <p>“<b>Phoenix</b>”: Sets the time zone to Phoenix time.</p>
-------------	--

---

---

### Default

The default is **GMT** (Greenwich Mean Time).

---

## Usage Guidelines

Use this command 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 system 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).

Use the **set** form of this command to set the time zone for the first time, or to change the time zone setting.

Use the **delete** form of this command to remove the time zone setting. This restores the time zone to the default (GMT).

Use the **show** form of this command to view the time zone setting.

# terminal

Controls behaviors of the system terminal.

---

## Syntax

**terminal** {**key query-help** {**enable**|**disable**} | **length** *length* | **pager** [*pager*] | **width** *width*}

---

## Command Mode

Operational mode.

---

## Parameters

<b>key query-help</b>	Set whether or not you can get help using a question mark. The options are <b>enable</b> and <b>disable</b> .The default is <b>enable</b> .
<i>length</i>	Sets the terminal screen length to a given number of rows.
<i>pager</i>	The program to use as the terminal pager. If none is specified, the default (less) is used.
<i>width</i>	Sets the terminal screen width to a given number of columns.

---

## Default

None.

---

## Usage Guidelines

Use this command to set the terminal behavior.

## Chapter 3: User Management

This chapter explains how to set up user accounts and user authentication.

This chapter presents the following topics:

- User Management Configuration
- Viewing Authentication Information
- User Management Commands



# User Management Configuration

---

This section presents the following topics:

- User Management Overview
- Creating “Login” User Accounts
- Configuring for a RADIUS Server
- Viewing Authentication Information

## User Management Overview

The Vyatta system supports both of the following:

- Role-based user account management through a local user database (“login” authentication)
- Authentication using a Remote Authentication Dial In User Service (RADIUS) authentication server. A local user database (“login” authentication).

## Login Authentication

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**.

If no RADIUS server has been set, the system authenticates users using the password configured in the **system login user <user> authentication** command (see page 140).

**NOTE** *You can change user account information using operating system commands, but the changes will not persist across reboots. For persistent changes to user account information, use the Vyatta CLI.*

The **login** configuration node is a mandatory node. It is created automatically with default information when the system is first started. If this node is subsequently deleted, the system recreates it with default information when restarted.

Login user passwords are supplied 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 is displayed.

## RADIUS Authentication

RADIUS servers are used only to authenticate user passwords. Using RADIUS authentication does not affect a user's configured privilege level.

The RADIUS secret is specified in plain text. RADIUS secrets are stored in plain text on the system, and used as part of a cryptographic operation for transferring authentication information securely over the network. When you view RADIUS secrets, they are displayed in plain text.

Where RADIUS authentication is used, some delay can be expected; this will depend on the cumulative timeout values configured for all RADIUS servers.

If you are using RADIUS authentication, the users must still be configured in the Vyatta login database, using the `system login user <user> authentication` command (see page 140); otherwise, the user will not even be able to query the RADIUS server because he will not be able to access the Vyatta system.

## Order of Authentication

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 system queries the RADIUS server using the supplied RADIUS secret. After the query is validated, the server authenticates the user from information in its database.

RADIUS servers are queried in the order of their configuration: if a query times out, the next server in the list is queried. If all queries fail, the system attempts to authenticate the user through the local Vyatta authentication database. If local authentication fails, the access attempt is rejected.

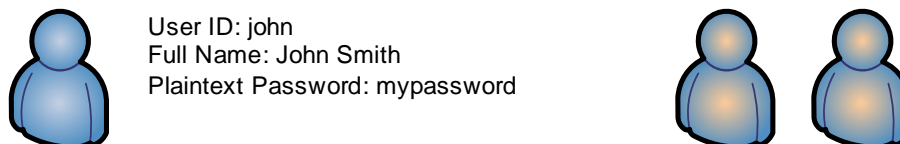
The specific authentication behavior is as follows:

- If a user only has a system login password P2 configured, she can use P2 to log on.
- If a user has both a RADIUS password P1 and a local password P2 configured, she can use either P1 or P2 to log on.

## Creating “Login” User Accounts

In this section, a sample configuration is presented for a user account that will be validated using the local user database. The sample configuration used is shown in Figure 3-1.

Figure 3-1 “Login” User Account



This section includes the following example:

- Example 3-1 Creating a “login” user account

Example 3-1 creates a user account for **John Smith**. John has a user ID of **john** and will use a plain text password of **mypassword**. Note that once configuration has been committed, only the encrypted version of the password displays when configuration is shown.

**NOTE** User information can be changed through the UNIX shell (providing you have sufficient permissions). However, any changes to Vyatta router user accounts or authentication through the UNIX shell will be overwritten the next time you commit Vyatta router CLI configuration.

To create a login user account, perform the following steps in configuration mode:

### Example 3-1 Creating a “login” user account

Step	Command
Create the user configuration node, define the user ID, and give the user’s full name.	vyatta@R1# <b>set system login user john full-name “John Smith”</b> [edit]
Specify the user’s password in plain text.	vyatta@R1# <b>set system login user john authentication plaintext-password mypassword</b> [edit]

**Example 3-1 Creating a “login” user account**

---

Commit the change. After a password has been committed, it can be displayed only in encrypted form, as the value of the **encrypted-password** attribute.

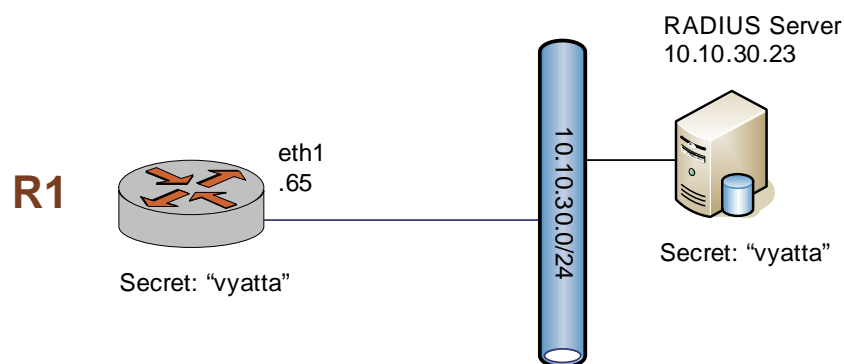
---

```
vyatta@R1# commit
[edit]
```

## Configuring for a RADIUS Server

In this section, a sample configuration is presented for a user account that will be validated using the RADIUS authentication server. The sample configuration used is shown in Figure 3-2.

Figure 3-2 RADIUS User Account



This section includes the following example:

- Example 3-2 Configuring for a RADIUS server

Example 3-2 defines a RADIUS authentication server at 10.10.30.23. The system will access the RADIUS server using a secret of **vyatta**. In this example, the port used for RADIUS traffic is left at the default 1812, which is the well-known port for RADIUS. The timeout after which the next RADIUS server should be queried is left at 2 seconds.

To define a RADIUS server, perform the following steps in configuration mode:

Example 3-2 Configuring for a RADIUS server

Step	Command
Provide the location of the server, and the secret to be used to access it.	<pre>vyatta@R1# set system login radius-server 10.10.30.23 secret vyatta [edit]</pre>
Commit the change. After the secret has been committed, it can be displayed only in encrypted form.	<pre>vyatta@R1# commit [edit]</pre>

## Viewing Authentication Information

You can always view the information in configuration nodes by using the **show** command in configuration mode. In this case, you can view authentication configuration by using the **show system login** command or the **show system radius-server** command, as shown in Example 3-3.

To show user authentication configuration, perform the following step in configuration mode:

Example 3-3 Viewing the “system login” and “system radius-server” configuration nodes

Step	Command
Show the contents of the <b>system login</b> configuration node.	<pre>vyatta@R1# show system login user root {     authentication {         encrypted-password \$1\$A.EPAdNj\$/2bTvc433VZ.VV5YWAbd1     } } user vyatta {     authentication {         encrypted-password \$1\$\$ZbzUPUD24iyfRwCKITl6q0     } } user john {     authentication         encrypted-password \$1\$\$Ht7gBYnxIlxCdO/JOnodh.         plaintext-password ""     }     full-name "John Smith" }</pre>
Show the contents of the <b>system radius-server</b> configuration node.	<pre>vyatta@R1# show system radius-server radius-server 10.10.30.23 {     port 1812     secret vyatta     timeout 2 }</pre>

# User Management Commands

This section presents the following commands.

## Configuration Commands

system login	Creates the configuration node for user management and authentication.
system login radius-server <address>	Defines a RADIUS server for user authentication.
system login user <user>	Creates a user account.
system login user <user> authentication	Sets an authentication password for a user.
system login user <user> full-name <name>	Allows you to specify a user's full name.
system login user <user> level <level>	Specifies a user's privilege level and system access.

## Operational Commands

show users	Shows which users are currently logged on to the system.
------------	--

# system login

Creates the configuration node for user management and authentication.

---

## Syntax

```
set system login
delete system login
show system login
```

---

## Command Mode

Configuration mode.

---

## Configuration Statement

```
system {
    login {}
}
```

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command, and its sub-commands, to manage user accounts and authentication.

The **login** configuration node is a mandatory node. It is created automatically with default information when the system is first started. If this node is subsequently deleted, the system recreates it with default information.

Use the **set** form of this command to create the **login** configuration node.

Use the **delete** form of this command to restore default user information and authentication information.

Use the **show** form of this command to view user and authentication configuration.



## system login radius-server <address>

Defines a RADIUS server for user authentication.

---

### Syntax

**set system login radius-server** *address* [**port** *port* / **secret** *secret* / **timeout** *timeout*]

**delete system login radius-server** *address* [**port** / **secret**/ **timeout**]

**show system login radius-server** *address* [**port** / **secret**/ **timeout**]

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {  
  login {  
    radius-server ipv4 {  
      port 1-65534  
      secret text  
      timeout 1-4294967296  
    }  
  }  
}
```

---

### Parameters

---

<i>address</i>	<p>Mandatory. Multi-node. The IP address of a remote authentication server running the RADIUS protocol. This server can be used to authenticate multiple users.</p> <p>You can define multiple RADIUS servers by creating multiple radius-server configuration nodes.</p>
<i>port</i>	<p>Optional. The port to be used for RADIUS traffic. The default is 1812.</p>
<i>secret</i>	<p>Mandatory. The password for the RADIUS server. This must be the same as that recorded on the RADIUS server.</p> <p>Supported characters are alphanumeric, space, and special characters. Strings containing spaces must be enclosed in double quotes.</p>

---

---

<i>timeout</i>	Optional. The interval, in seconds, after which, if the RADIUS server has not responded, the next configured RADIUS server should be queried. The default is 2.
----------------	---

---

---

## Default

None.

---

## Usage Guidelines

Use this command to define a Remote Authentication Dial In User Service (RADIUS) server and specify the information necessary to log on to it.

The RADIUS secret is specified in plain text. RADIUS secrets are stored in plain text on the system, and used as part of a cryptographic operation for transferring authentication information securely over the network. When you view RADIUS secrets, they are displayed in plain text.

Use the **set** form of this command to define a RADIUS server.

Use the **delete** form of this command to remove a RADIUS server.

Use the **show** form of this command to view RADIUS server configuration.

## system login user <user>

Creates a user account.

---

### Syntax

```
set system login user user
delete system login user user
show system login user user
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  login {
    user text {}
  }
}
```

---

### Parameters

<i>user</i>	Mandatory. Multi-node. A unique user ID of up to 32 characters, including alphanumeric characters or hyphens.  You can define multiple user accounts by creating multiple <b>user</b> configuration nodes.
-------------	--

---

### Default

None.

---

### Usage Guidelines

Use this command to define a user that will be authenticated using the system's internal mechanism: "login" authentication.

The system first attempts to authenticate users in the local user database ("login" authentication). If this fails, the system proceeds to RADIUS authentication.

The system automatically creates two user accounts on startup:

- User **root**. User **root** is a user with admin-level privileges plus a full view of the system: command completion and CLI help show all commands and files, including bash commands and files.
- User **vyatta**. User **vyatta** is a user with admin-level privileges but a streamlined view of the system: command completion and CLI help show only Vyatta-specific commands.

Note that, although user account and authentication information can be changed using the operating system shell, the system will overwrite these changes the next time you commit configuration in the Vyatta shell. For persistent changes to user or authentication information, use Vyatta CLI commands.

Use the **set** form of this command to create a **user** configuration node.

Use the **delete** form of this command to remove a **user** configuration node. Note that you can delete the **vyatta** user account, but the **root** user account cannot be deleted.

Use the **show** form of this command to view **user** configuration.

## system login user <user> authentication

Sets an authentication password for a user.

---

### Syntax

```
set system login user user authentication { encrypted-password epwd |  
plaintext-password ppwd }  
  
delete system login user user authentication [ encrypted-password |  
plaintext-password ]  
  
show system login user user authentication [ encrypted-password |  
plaintext-password ]
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {  
  login {  
    user text {  
      authentication {  
        encrypted-password text  
        plaintext-password text  
      }  
    }  
  }  
}
```

---

### Parameters

<i>user</i>	Mandatory. A unique user ID of up to 32 characters, including alphanumeric characters or hyphens.
<i>epwd</i>	The encrypted password. This value is system generated and should not be altered.
<i>ppwd</i>	The user's password, specified in plain text.

---

---

## Default

None.

---

## Usage Guidelines

Use this command to set a password to authenticate a user. Passwords are set using the **plaintext-password** command and are automatically encrypted by the system using Message Digest 5 (MD5) encryption. The encrypted version is stored internally and used. When displayed the encrypted value is shown. The plaintext-password appears as double quotes in the configuration.

To disable a user account without deleting it simply set the **encrypted-password** to “\*”.

Use the **set** form of this command to set a user’s password.

Use the **delete** form of this command to remove a user’s password.

Use the **show** form of this command to view user password configuration.

## system login user <user> full-name <name>

Allows you to specify a user's full name.

---

### Syntax

**set system login user** *user* **full-name** *name*

**delete system login user** *user* **full-name**

**show system login user** *user* **full-name**

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {  
  login {  
    user text {  
      full-name text  
    }  
  }  
}
```

---

### Parameters

---

<i>user</i>	Mandatory. A unique user ID of up to 32 characters, including alphanumeric characters or hyphens.
<i>name</i>	Mandatory. A string representing the user's name, including alphanumeric characters, space, and hyphens. Strings that include spaces must be enclosed in double quotes.

---

---

### Default

None.

---

## Usage Guidelines

Use **this** command to record a user's full name.

Use the **set** form of this command to specify the user's name.

Use the **delete** form of this command to remove the user's name.

Use the **show** form of this command to view a user's name.



## system login user <user> level <level>

Specifies a user's privilege level and system access.

---

### Syntax

**set system login user** *user* **level** *level*

**delete system login user** *user* **level**

**show system login user** *user* **level**

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {  
  login {  
    user text {  
      level [admin | operator]  
    }  
  }  
}
```

---

### Parameters

---

<i>user</i>	Mandatory. A unique user ID of up to 32 characters, including alphanumeric characters or hyphens.
<i>level</i>	Determines the user's level of privilege.  <b>admin:</b> Assigns the user administrative privileges. The user can execute any command in the Vyatta CLI or the underlying operating system.  <b>operator:</b> Assigns the user restricted privileges. The user can execute operational commands in the Vyatta CLI, plus a restricted form of <b>ping</b> and <b>traceroute</b> . The user cannot enter configuration mode or execute configuration commands.

---

---

### Default

Users are assigned administrative privileges by default.

---

## Usage Guidelines

Use this command to assign role-based system access to a user.

The system supports three system roles:

- **Root user.** The root user role has full access to all Vyatta commands plus all operating system shell commands. Access to operating system shell commands is direct: the root user need not exit another shell or mode before executing these commands. Command completion and CLI help show all commands and files, including operating system shell commands and files. Currently, there is exactly one user with a root user role: user **root**. User **root** cannot be deleted, and you cannot create another user with root privileges. You cannot change the level of access for user **root**.
- **Admin user.** Users assigned a role of admin have full access to all Vyatta-specific commands plus all operating system shell commands. Access to operating system shell commands is direct: the user need not exit another shell or mode before executing these commands. A user with a role of admin has a more streamlined view of the system than the **root** user does: although admin users can execute any command implemented in the system, command completion and CLI help show only Vyatta commands.
- **Operator user.** Users assigned a role of operator have access to the Vyatta operational command set, but no access to configuration commands. They also have limited access to operating system commands. At this time, command completion and CLI help show all Vyatta commands for users with the operator role.

Use the **set** form of this command to set a user's privilege level.

Use the **delete** form of this command to restore a user's privilege level to the default.

Use the **show** form of this command to view user privilege configuration.

## show users

Shows which users are currently logged on to the system.

---

### Syntax

**show users**

---

### Command Mode

Operational mode.

---

### Parameters

None.

---

### Default

None.

---

### Usage Guidelines

Use this command to see which users are currently logged on to the system.

## Chapter 4: Logging

This chapter describes the Vyatta system logging mechanism.

This chapter presents the following topics:

- Logging Configuration
- Logging Commands

# Logging Configuration

This section presents the following topics:

- Logging Overview
- Logging Configuration Example
- Enabling and Disabling Logging for Specific Features

## Logging Overview

Significant system events are captured in log messages (also called syslog messages), which you can view on the console, save to a file, or forward to an external server such as a syslog server, or direct to the terminal session of one or more specific users.

Depending on the level of message severity you choose to log, system log messages can include notices of ordinary and routine operations, as well as warnings, failure, and error messages.

The Vyatta system's logging function makes use of the UNIX **syslogd** process. Logging configuration performed within the system's CLI is stored in the **/etc/syslogd.conf** file.

By default, local logging is enabled, and sends messages to **/var/log/messages**.

## Logging Facilities

The Vyatta system supports standard syslog facilities. These are as follows:

Table 4-1 Syslog facilities

Facility	Description
auth	Authentication and authorization
authpriv	Non-system authorization
cron	Cron daemon
daemon	System daemons
kern	Kernel
lpr	Line printer spooler
mail	Mail subsystem
mark	Timestamp
news	USENET subsystem

Table 4-1 Syslog facilities

security	Security subsystem
syslog	System logging
user	Application processes
uucp	UUCP subsystem
local0	Local facility 0
local1	Local facility 1
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	All facilities excluding “mark”

In addition, logging can be selectively enabled for some specific routing components. For this information, please see the section ““Enabling and Disabling Logging for Specific Features” on page 152.

## Log Destinations

When logging is enabled, system log messages are always written to the **messages** file in the **/var/log** directory of the local file system. In addition, system logs can be sent to the console, to a named file in the local file system, to a server running the **syslogd** utility (that is, a syslog server), or to the terminal session of one or more specific users.

- To direct syslog messages to the console, use the **system syslog console** command.
- To direct syslog messages to a named file in the local file system, use the **system syslog file** command.
- To direct syslog messages to a remote machine running the **syslogd** utility, use the **system syslog host** command.
- To direct syslog messages to the terminal of a specific user, to multiple users, or to all users logged into the routing platform, use the **system syslog user** command.

## Log File Locations and Archiving

Messages are written either to the main log file (the default) or to a file that you specify. User-defined log files are written to the **/var/log/user** directory, under the user-specified file name.

The system 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 main log file or a name you have assigned to a user-defined 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, the maximum number of archived files is 10 (that is, up to *logfile.9.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 each archived log file.
- Use the **files** parameter to specify the maximum number of archive files to be maintained.

## Log Severities

System events generate log messages at different severities, which represent their level of importance for the system.

When you configure severity level for syslog, the system captures log messages at that severity and above. The lower the level of severity specified, the more detail is captured in the logs. For example, if you configure a log severity level of **crit**, the system captures log messages that have severity **crit**, **alert**, and **emerg**.

Currently, log severity is configurable *for user-defined log files only*. The main log file in **/var/log/messages** captures log messages of severity **warning** and above.

Log messages generated by the Vyatta system will be associated with one of the following levels of severity.

Table 4-2 Syslog message severities

Severity	Meaning
emerg	Emergency. A general system failure or other serious failure has occurred, such that the system is unusable.

Table 4-2 Syslog message severities

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.
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.



**CAUTION** Risk of service degradation. Debug severity is resource-intensive. Setting logging levels to Debug can affect performance.

## Logging Configuration Example

Example 4-1 creates a log file that captures kernel-related alerts of critical and higher severity.

To create a log file to capture kernel-related critical alerts, perform the following steps in configuration mode:

Example 4-1 Configuring a log to capture kernel-related alerts of critical and higher severity

Step	Command
Create a logfile called "kernel-log" and log kernel-related messages of "critical" and higher severity.	<pre>vyatta@R1# set system syslog file kernel-log facility kern level crit [edit]</pre>



---

**Example 4-1** Configuring a log to capture kernel-related alerts of critical and higher severity

---

```
Commit the configuration.      vyatta@R1# commit
                               Restarting system log daemon....
                               [edit]
                               vyatta@R1#
```

---

The command “**show log file** *kernel-log*” can then be used in operational mode to display the contents of the *kernel-log* logfile.

## Enabling and Disabling Logging for Specific Features

Some features of the Vyatta router—for example, BGP, OSPF, and IPsec VPN—produce feature-specific log messages that can be enabled and disabled within the configuration node for that feature. When you enable logging for a system feature, the log messages are sent to whatever destinations are configured for syslog.

By default, log messages are sent to the main log file. You can configure syslog to send log messages to a file you specify in **/var/user**.

# Logging Commands

This section presents the following commands.

## Configuration Commands

system syslog	Configures the system's syslog utility.
system syslog console facility <facility> level <level>	Specifies which messages are sent to the console.
system syslog file <filename> archive	Specifies the settings for log file archiving of the user-defined log file.
system syslog file <filename> facility <facility> level <level>	Specifies which messages are sent to the user-defined log file.
system syslog global archive	Specifies the settings for log file archiving of the main system log file.
system syslog global facility <facility> level <level>	Specifies which messages are sent to the main system log file.
system syslog host <hostname> facility <facility> level <level>	Specifies which messages are sent to the remote syslog server.
system syslog user <userid> facility <facility> level <level>	Specifies which messages are sent to the specified user's terminal.

## Operational Commands

delete log file	Deletes the specified log file, including all its archive files.
show log	Displays the contents of the specified log file.
show log directory	Displays a list of files in the logging directory.
show log tail	Displays the last lines of the messages file.

## delete log file

Deletes the specified log file, including all its archive files.

---

### Syntax

**delete log file** *file-name*

---

### Command Mode

Operational mode.

---

### Parameters

---

<i>file-name</i>	Deletes the specified user-defined file in the <b>/var/log</b> 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 **show log tail** command (see page 157), and then delete it.

## show log

Displays the contents of the specified log file.

---

### Syntax

```
show log [all | file file-name]
```

---

### Command Mode

Operational mode.

---

### Parameters

<b>all</b>	Displays the contents of all master log files.
<b>file</b> <i>file-name</i>	Displays the contents of the specified log file directory.

---

### Usage Guidelines

Use this command to view the contents of a log file or files.

When used with no option, this command displays the contents of the main system log, which is the default log to which the system writes syslog messages.

When the **file** *file-name* is specified, this command displays the contents of the specified user-defined log file.

## show log directory

Displays a list of files in the logging directory.

---

### Syntax

**show log directory**

---

### Command Mode

Operational mode.

---

### Parameters

None.

---

### Usage Guidelines

Use this command to list the log files that have been defined by system users.

The directory displayed is the directory where user-defined log files are stored. Syslog messages can be written to these or to the main system log file. User-specified log files are defined using the **system syslog file <filename> facility <facility> level <level>** command (see page 166).

# show log tail

Displays the last lines of the messages file.

---

## Syntax

```
show log tail [lines]
```

---

## Command Mode

Operational mode.

---

## Parameters

<i>lines</i>	The number of lines to display.
--------------	---------------------------------

---

## Usage Guidelines

Use this command to display the last lines of the messages file..

When used with no option, the last ten lines are displayed and then will continue to display the messages as they are added to the file.

When the *lines* is specified, the last *lines* lines of the messages file are displayed.

# system syslog

Configures the system's syslog utility.

---

## Syntax

```
set system syslog  
delete system syslog  
show system syslog
```

---

## Command Mode

Configuration mode.

---

## Configuration Statement

```
system {  
    syslog {  
    }  
}
```

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to configure the system's syslog utility.

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 system will be associated with one of the following levels of severity.

Table 4-3 Syslog message severities

Severity	Meaning
emerg	Emergency. A general system failure or other serious failure has occurred, such that the system 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.
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 system supports standard syslog facilities. These are as follows:

Table 4-4 Syslog facilities

Facility	Description
auth	Authentication and authorization
authpriv	Non-system authorization
cron	Cron daemon
daemon	System daemons
kern	Kernel
lpr	Line printer spooler



Table 4-4 Syslog facilities

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
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	All facilities excluding "mark"

Messages are written either to the main log file (the default) or to a file that you specify. User-defined log files are written to the **/var/log/user** directory, under the user-specified file name.

The system 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 main log file or a name you have assigned to a user-defined 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, the maximum number of archived files is 10 (that is, up to *logfile.9.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 each archived log file.
- Use the **files** parameter to specify the maximum number of archive files to be maintained.

Use the **set** form of this command to create the syslog configuration.

Use the **delete** form of this command to remove the syslog configuration.

Use the **show** form of this command to view the syslog configuration.

## system syslog console facility <facility> level <level>

Specifies which messages are sent to the console.

---

### Syntax

```
set system syslog console facility facility level level
delete system syslog console facility [facility [level]]
show system syslog console facility [facility [level]]
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  syslog {
    console {
      facility text {
        level text
      }
    }
  }
}
```

---

### Parameters

<i>facility</i>	<p>Multi-node. The kinds of messages that will be sent to the console. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for supported facilities.</p> <p>You can send the log messages of multiple facilities to the console by creating multiple <b>facility</b> configuration nodes within the <b>console</b> node.</p>
<i>level</i>	<p>The minimum severity of log message that will be reported to the console. Supported values are <b>emerg</b>, <b>alert</b>, <b>crit</b>, <b>err</b>, <b>warning</b>, <b>notice</b>, <b>info</b>, and <b>debug</b>. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for the meanings of these levels.</p> <p>By default, messages of <b>err</b> severity are logged to the console.</p>

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify which messages are sent to the console.

Use the **set** form of this command to specify which messages are sent to the console.

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

Use the **show** form of this command to view the console message configuration.

## system syslog file <filename> archive

Specifies the settings for log file archiving of the user-defined log file.

---

### Syntax

```
set system syslog file filename archive {files files / size size}
delete system syslog file filename archive {files / size}
show system syslog file filename archive {files / size}
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  syslog {
    file text{
      archive {
        files u32
        size u32
      }
    }
  }
}
```

---

### Parameters

<i>filename</i>	Multi-node. Defines a file to which the specified log messages will be written. File names can include numbers, letters, and hyphens.  You can send log messages to multiple files by creating multiple <b>file</b> configuration nodes.
<i>files</i>	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 10.
<i>size</i>	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 1 MB.

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify the settings for log file archiving of the user-defined log file.

Use the **set** form of this command to specify the settings for log file archiving of the user-defined log file.

Use the **delete** form of this command to restore the default user-defined log file archiving configuration.

Use the **show** form of this command to view the user-defined log file archiving configuration.

## system syslog file <filename> facility <facility> level <level>

Specifies which messages are sent to the user-defined log file.

---

### Syntax

**set system syslog file** *filename* **facility** *facility* **level** *level*

**delete system syslog file** *filename* **facility** [*facility* [*level*]]

**show system syslog file** *filename* **facility** [*facility* [*level*]]

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  syslog {
    file text {
      facility text {
        level text
      }
    }
  }
}
```

---

### Parameters

---

<i>filename</i>	<p>Multi-node. Defines a file to which the specified log messages will be written. File names can include numbers, letters, and hyphens.</p> <p>You can send log messages to multiple files by creating multiple <b>file</b> configuration nodes.</p>
<i>facility</i>	<p>Multi-node. The kinds of messages that will be sent to the user-defined log file. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for supported logging facilities.</p> <p>You can send the log messages of multiple facilities to this log file by creating multiple <b>facility</b> configuration nodes within the <b>file</b> configuration node.</p>

---

---

<i>level</i>	The minimum severity of log message that will be reported. Supported values are <b>emerg</b> , <b>alert</b> , <b>crit</b> , <b>err</b> , <b>warning</b> , <b>notice</b> , <b>info</b> , <b>debug</b> . Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for the meanings of these levels.
--------------	---

---

By default, messages of **warning** severity are logged to file.

---

The Vyatta system supports sending log messages to the main system log file, to the console, to a remote host, to a user-specified file, or to a user account.

---

## Default

None.

---

## Usage Guidelines

Use this command to specify which messages are sent to the user-defined log file.

Use the **set** form of this command to specify which messages are sent to the user-defined log file.

Use the **delete** form of this command to restore the default user-defined log file message configuration.

Use the **show** form of this command to view the user-defined log file message configuration.



## system syslog global archive

Specifies the settings for log file archiving of the main system log file.

---

### Syntax

```
set system syslog global archive {files files / size size}  
delete system syslog global archive {files / size}  
show system syslog global archive {files / size}
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {  
  syslog {  
    global {  
      archive {  
        files u32  
        size u32  
      }  
    }  
  }  
}
```

---

### Parameters

<i>files</i>	Sets the maximum number of archive files that will be maintained for the main system log file. After the maximum has been reached, logs will be rotated with the oldest file overwritten. The default is 10.
<i>size</i>	Sets the maximum size in bytes of archive files for the main system log file. After the maximum has been reached, the file will be closed and archived in compressed format. The default is 1 MB.

---

### Default

None.

---

## Usage Guidelines

Use this command to specify the settings for log file archiving of the main system log file.

Use the **set** form of this command to specify the settings for log file archiving of the main system log file.

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

Use the **show** form of this command to view the log file archiving configuration.

## system syslog global facility <facility> level <level>

Specifies which messages are sent to the main system log file.

---

### Syntax

```
set system syslog global facility facility level level
delete system syslog global facility [facility [level]]
show system syslog global facility [facility [level]]
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  syslog {
    global {
      facility text {
        level text
      }
    }
  }
}
```

---

### Parameters

<i>facility</i>	<p>Multi-node. The kinds of messages that will be sent to the main system log file. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for supported facilities.</p> <p>You can send the log messages of multiple facilities to the main system log file by creating multiple <b>facility</b> configuration nodes within the <b>global</b> node.</p>
<i>level</i>	<p>The minimum severity of log message that will be reported. Supported values are <b>emerg</b>, <b>alert</b>, <b>crit</b>, <b>err</b>, <b>warning</b>, <b>notice</b>, <b>info</b>, <b>debug</b>. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for the meanings of these levels.</p> <p>By default, messages of <b>warning</b> severity are logged to the main system log file.</p>

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify which messages are sent to the main system log file.

Use the **set** form of this command to specify which messages are sent to the main system log file.

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

Use the **show** form of this command to view the log file message configuration.

## system syslog host <hostname> facility <facility> level <level>

Specifies which messages are sent to the remote syslog server.

---

### Syntax

**set system syslog host** *hostname* **facility** *facility* **level** *level*

**delete system syslog file** *hostname* **facility** [*facility* [**level**]]

**show system syslog file** *hostname* **facility** [*facility* [**level**]]

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  syslog {
    host text {
      facility text {
        level text
      }
    }
  }
}
```

---

### Parameters

<i>hostname</i>	<p>Multi-node. Sends the specified log messages to a host. The host must be running the syslog protocol. The <i>hostname</i> can be an IP address or a host name. Host names can include numbers, letters, and hyphens (“-”).</p> <p>You can send log messages to multiple hosts by creating multiple <b>host</b> configuration nodes.</p>
<i>facility</i>	<p>Multi-node. The kinds of messages that will be sent to the host. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for supported logging facilities.</p> <p>You can send the log messages of multiple facilities to a host by creating multiple <b>facility</b> configuration nodes within the <b>host</b> configuration node.</p>

---

---

<i>level</i>	<p>The minimum severity of log message that will be reported. Supported values are <b>emerg</b>, <b>alert</b>, <b>crit</b>, <b>err</b>, <b>warning</b>, <b>notice</b>, <b>info</b>, <b>debug</b>. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for the meanings of these levels.</p> <p>By default, messages of <b>err</b> severity are logged to hosts.</p>
--------------	--

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify which messages are sent to the remote syslog server.

Use the **set** form of this command to specify which messages are sent to the remote syslog server.

Use the **delete** form of this command to restore the default remote syslog server log file message configuration.

Use the **show** form of this command to view the remote syslog server log file message configuration.

## system syslog user <userid> facility <facility> level <level>

Specifies which messages are sent to the specified user's terminal.

---

### Syntax

**set system syslog user** *userid* **facility** *facility* **level** *level*

**delete system syslog user** *userid* **facility** [*facility* [*level*]]

**show system syslog user** *userid* **facility** [*facility* [*level*]]

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  syslog {
    user text {
      facility text {
        level text
      }
    }
  }
}
```

---

### Parameters

---

<i>userid</i>	<p>Multi-node. Sends the specified log messages to the specified user's terminal.</p> <p>You can send log messages to multiple users by creating multiple <b>user</b> configuration nodes.</p>
<i>facility</i>	<p>Multi-node. The kinds of messages that will be sent to the user. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for supported logging facilities.</p> <p>You can send the log messages of multiple facilities to a user account by creating multiple <b>facility</b> configuration nodes within the <b>user</b> configuration node.</p>

---

---

<i>level</i>	<p>The minimum severity of log message that will be reported to the user. Supported values are <b>emerg</b>, <b>alert</b>, <b>crit</b>, <b>err</b>, <b>warning</b>, <b>notice</b>, <b>info</b>, <b>debug</b>. Please see the Usage Guidelines in the <b>system syslog</b> command (see page 158) for the meanings of these levels.</p> <p>By default, messages of <b>err</b> severity are logged to specified user's.</p>
--------------	---

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify which messages are sent to the specified user's terminal.

Use the **set** form of this command to specify which messages are sent to the specified user's terminal.

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

Use the **show** form of this command to view the user terminal message configuration.



## Chapter 5: SNMP

This chapter describes the Vyatta system's support for SNMP.

This chapter presents the following topics:

- SNMP Configuration
- SNMP Commands

# SNMP Configuration

---

This section presents the following topics:

- SNMP Overview
- SNMP Configuration Examples

## SNMP Overview

This section presents the following topics:

- MIB Objects
- Traps
- SNMP Commands
- SNMP Versions
- SNMP MIBs

SNMP (Simple Network Management Protocol) is a mechanism for managing network and computer devices.

SNMP uses a manager/agent model for managing the devices. The agent resides in the device, and provides the interface to the physical device being managed. The manager resides on the management system and provides the interface between the user and the SNMP agent. The interface between the SNMP manager and the SNMP agent uses a Management Information Base (MIB) and a small set of commands to exchange information.

## MIB Objects

A MIB contains the set of variables/objects that are managed (for example, MTU on a network interface). Those objects are organized in a tree structure where each object is a leaf node. Each object has its unique Object Identifier (OID).

There are two types of objects: *scalar* and *tabular*. A scalar object defines a single object instance. A tabular object defines multiple related object instances that are grouped in MIB tables. For example, the uptime on a device is a scalar object, but the routing table in a system is a tabular object.

## Traps

In addition to MIB objects, the SNMP agent on a device can formulate alarms and notifications into SNMP *traps*. The device will asynchronously send the traps to the SNMP managers that are configured as trap destinations or *targets*. This keeps the network manager informed of the status and health of the device.

## SNMP Commands

SNMP commands can be used to read or change configuration, or to perform actions on a device, such as resetting it. The set of commands used in SNMP are: **GET**, **GET-NEXT**, **GET-RESPONSE**, **SET**, and **TRAP**.

- **GET** and **GET-NEXT** are used by the manager to request information about an object. These commands are used to view configuration or status, or to poll information such as statistics.
- **SET** is used by the manager to change the value of a specific object. Setting a configuration object changes the device's configuration. Setting an executable object performs an action, such as a file operation or a reset.
- **GET-RESPONSE** is used by the SNMP agent on the device to return the requested information by **GET** or **GET-NEXT**, or the status of the **SET** operation.
- The **TRAP** command is used by the agent to asynchronously inform the manager about events important to the manager.

## SNMP Versions

Currently there are three versions of SNMP:

- SNMP v1. This is the first version of the protocol. It is described in RFC 1157.
- SNMP v2. This is an evolution of the first version, and it adds a number of improvements to SNMPv1.
- SNMP v3. This version improves the security model in SNMPv2, and adds support for proxies.

The Vyatta system supports SNMP v2 with community string (SNMP v2c)

## SNMP MIBs

MIBs are typically located in the `/usr/share/snmp/mibs` directory.

The MIBs supported by the Vyatta system are listed in “Appendix A: SNMP MIB Support.” The Vyatta system does not currently have its own enterprise MIB.

## SNMP Configuration Examples

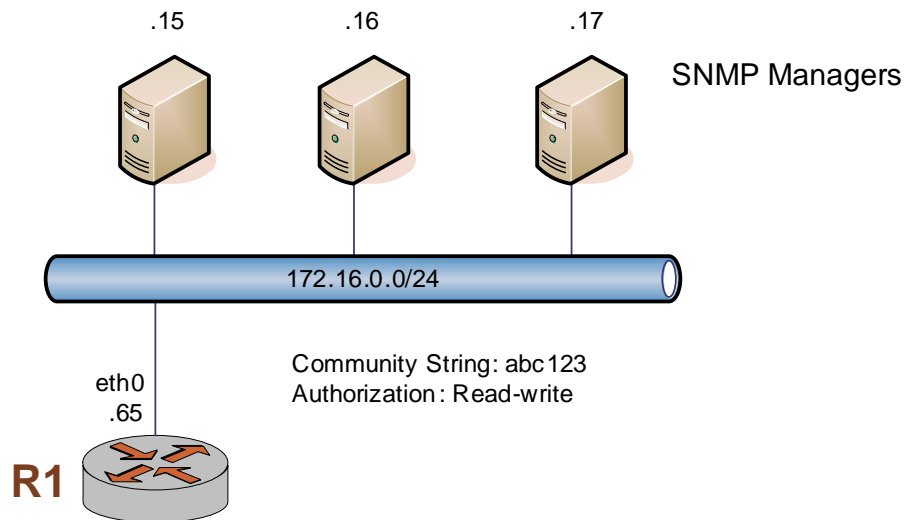
This section presents the following topics:

- Defining the SNMP Community
- Specifying Trap Destinations

To configure SNMP, there must be at least one user created, and the Vyatta MIB model must be loaded.

This sequence sets up an SNMP community that includes three hosts, which will serve as SNMP managers, and configures the system to send traps to all three managers. When you have finished, the system will be configured as shown in Figure 5-1.

Figure 5-1 Configuring SNMP communities and traps



This section includes the following examples:

- Example 5-1 Defining an SNMP community
- Example 5-2 Specifying SNMP trap destinations

## Defining the SNMP Community

The SNMP community of a system is the list of SNMP clients authorized to make requests of the system. Authorization for the community is in the form of a community string. The community string acts as a password, providing basic security and protecting the system against spurious SNMP requests.

- If no SNMP clients are explicitly defined, then any client presenting the correct community string is granted read-only access to the system.

- If any client is defined, then only explicitly listed clients are granted access to the system. Those clients will have the access privilege specified by the **authorization** option. (The default is read-only.)

Example 5-1 sets the SNMP community string to abc123 and specifies three clients for the community: 176.16.0.15, 176.16.0.16, and 176.16.0.17. Read-write access is provided for this community.

To define an SNMP community, perform the following steps in configuration mode:

Example 5-1 Defining an SNMP community

Step	Command
Create the <b>snmp</b> configuration node and the <b>community</b> configuration node. Set the community string. Navigate to the configuration node of the community.	vyatta@R1# <b>set protocols snmp community abc123</b> [edit] vyatta@R1# <b>edit protocols snmp community abc123</b> [edit protocols snmp community abc123]
List the SNMP clients making up this community.	vyatta@R1# <b>set client 176.16.0.15</b> [edit protocols snmp community abc123] vyatta@R1# <b>set client 176.16.0.16</b> [edit protocols snmp community abc123] vyatta@R1# <b>set client 176.16.0.17</b> [edit protocols snmp community abc123]
Set the privilege level for this community to read-write.	vyatta@R1# <b>set authorization rw</b> [edit protocols snmp community abc123]
Commit the change, and return to the top of the configuration tree.	vyatta@R1# <b>commit</b> [edit protocols snmp community abc123] vyatta@R1# <b>top</b> [edit]

## Specifying Trap Destinations

Example 5-1 directs the system to send SNMP traps to the configured network managers at 176.16.0.15, 176.16.0.16, and 176.16.0.17.

To specify trap destinations, perform the following steps in configuration mode:

**Example 5-2** Specifying SNMP trap destinations

Step	Command
Define the trap destinations, one at a time.	<pre>vyatta@R1# set protocols snmp trap-target 176.16.0.15 [edit] vyatta@R1# set protocols snmp trap-target 176.16.0.16 [edit] vyatta@R1# set protocols snmp trap-target 176.16.0.17 [edit]</pre>
Commit the change.	<pre>vyatta@R1# commit [edit]</pre>

## SNMP Commands

This section presents the following commands.

### Configuration Commands

protocols snmp	Defines SNMP community and trap information for the Vyatta system.
protocols snmp community <community>	Defines an SNMP community.
protocols snmp community <community> authorization <auth>	Specifies the privileges this community will have.
protocols snmp community <community> client <ipv4>	Specifies the SNMP clients in this community that are authorized to access the system.
protocols snmp community <community> network <ipv4net>	Specifies the network of SNMP clients in this community that are authorized to access the server.
protocols snmp contact <contact>	Records contact information for the system.
protocols snmp description <desc>	Records a brief description of the system.
protocols snmp location <location>	Records the location of the system.
protocols snmp trap-target <ipv4>	Specifies the IP address of the destination for SNMP traps.

### Operational Commands

None

## protocols snmp

Defines SNMP community and trap information for the Vyatta system.

---

### Syntax

```
set protocols snmp
delete protocols snmp
show protocols snmp
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
    snmp {
    }
}
```

---

### Parameters

None.

---

### Default

None.

---

### Usage Guidelines

Use this command to specify information about which SNMP communities this system should respond to, about the system's location and contact information, and about destinations for SNMP traps.

Use the **set** form of this command to define SNMP settings.

Use the **delete** form of this command to remove all SNMP configuration.

Use the **show** form of this command to view SNMP configuration.



## protocols snmp community <community>

Defines an SNMP community.

---

### Syntax

```
set protocols snmp community community
delete protocols snmp community community
show protocols snmp community community
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
  snmp {
    community text
  }
}
```

---

### Parameters

---

<i>community</i>	Optional. Multi-node. Defines an SNMP community. The argument is the community string to be used to authorize SNMP managers making requests of this system. Letters, numbers, and hyphens are supported.  You can define more than one community by creating multiple <b>community</b> configuration nodes.
------------------	---

---

---

### Default

By default, no community string is defined.

---

### Usage Guidelines

Use this command to specify an SNMP community.

Use the **set** form of this command to specify an SNMP community.

Use the **delete** form of this command to remove an SNMP community configuration.

Use the **show** form of this command to view an SNMP community configuration.

## protocols snmp community <community> authorization <auth>

Specifies the privileges this community will have.

---

### Syntax

```
set protocols snmp community community authorization auth
delete protocols snmp community community authorization
show protocols snmp community community authorization
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
  snmp {
    community text
    authorization [ro|rw]
  }
}
```

---

### Parameters

---

<i>community</i>	Optional. Multi-node. Defines an SNMP community. The argument is the community string to be used to authorize SNMP managers making requests of this system. Letters, numbers, and hyphens are supported.  You can define more than one community by creating multiple <b>community</b> configuration nodes.
<i>auth</i>	Optional. Specifies the privileges this community will have. Supported values are as follows:  <b>ro</b> : This community can view system information, but not change it. <b>rw</b> : This community has read-write privileges.  Deleting the <b>authorization</b> statement resets the privilege level to the default ( <b>ro</b> ).

---

---

## Default

The default authorization privilege is **ro**.

---

## Usage Guidelines

Use this command to specify the privileges this community will have.

Use the **set** form of this command to specify SNMP community privileges.

Use the **delete** form of this command to restore default SNMP community privileges.

Use the **show** form of this command to view SNMP community privilege configuration.

## protocols snmp community <community> client <ipv4>

Specifies the SNMP clients in this community that are authorized to access the system.

### Syntax

```
set protocols snmp community community client ipv4
delete protocols snmp community community client ipv4
show protocols snmp community community client
```

### Command Mode

Configuration mode.

### Configuration Statement

```
protocols {
  snmp {
    community text
    client ipv4
  }
}
```

### Parameters

<i>community</i>	<p>Optional. Multi-node. Defines an SNMP community. The argument is the community string to be used to authorize SNMP managers making requests of this system. Letters, numbers, and hyphens are supported.</p> <p>You can define more than one community by creating multiple <b>community</b> configuration nodes.</p>
<i>ipv4</i>	<p>Optional. Multi-node. The SNMP clients in this community that are authorized to access the system.</p> <p>You can define more than one client by creating the <b>client</b> configuration node multiple times.</p> <p>If no client or network is defined, then any client presenting the correct community string will have read-only access to the system. If any client or network is defined then only explicitly listed clients and/or networks will have access to the system.</p>

---

## Default

None.

---

## Usage Guidelines

Use this command to specify the SNMP clients in this community that are authorized to access the system.

Use the **set** form of this command to specify the SNMP clients in this community that are authorized to access the system.

Use the **delete** form of this command to remove SNMP clients in this community that are authorized to access the system.

Use the **show** form of this command to view SNMP clients in this community that are authorized to access the system.

## protocols snmp community <community> network <ipv4net>

Specifies the network of SNMP clients in this community that are authorized to access the server.

---

### Syntax

```
set protocols snmp community community network ipv4net
delete protocols snmp community community network ipv4net
show protocols snmp community community network
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
  snmp {
    community text
    network ipv4net
  }
}
```

---

### Parameters

<i>community</i>	<p>Optional. Multi-node. Defines an SNMP community. The argument is the community string to be used to authorize SNMP managers making requests of this system. Letters, numbers, and hyphens are supported.</p> <p>You can define more than one community by creating multiple <b>community</b> configuration nodes.</p>
<i>ipv4net</i>	<p>Optional. Multi-node. The network of SNMP clients in this community that are authorized to access the server.</p> <p>You can define more than one network by creating the <b>network</b> configuration node multiple times.</p> <p>If no client or network is defined, then any client presenting the correct community string will have read-only access to the system. If any client or network is defined then only explicitly listed clients and/or networks will have access to the system.</p>

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify a network of SNMP clients in this community that are authorized to access the server.

Use the **set** form of this command to specify a network of SNMP clients in this community that are authorized to access the server.

Use the **delete** form of this command to remove a network of SNMP clients in this community that are authorized to access the server.

Use the **show** form of this command to view a network of SNMP clients in this community that are authorized to access the server.

## protocols snmp contact <contact>

Records contact information for the system.

---

### Syntax

```
set protocols snmp contact contact
delete protocols snmp contact
show protocols snmp contact
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
  snmp {
    contact text
  }
}
```

---

### Parameters

---

<i>contact</i>	Optional. Records contact information for the system. This is stored as MIB-2 system information in the <b>snmpd.conf</b> configuration file. Letters, numbers, and hyphens are supported.
----------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to specify contact information for the system.

Use the **set** form of this command to specify contact information for the system.

Use the **delete** form of this command to remove contact information for the system.

Use the **show** form of this command to view contact information for the system.



## protocols snmp description <desc>

Records a brief description of the system.

---

### Syntax

```
set protocols snmp description desc
delete protocols snmp description
show protocols snmp description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
  snmp {
    description text
  }
}
```

---

### Parameters

---

<i>desc</i>	Optional. Records a brief description of the system. This is stored as MIB-2 system information in the <b>snmpd.conf</b> configuration file. Letters, numbers, and hyphens are supported.
-------------	---

---

---

### Default

None.

---

### Usage Guidelines

Use this command to specify a brief description of the system.

Use the **set** form of this command to specify a brief description of the system.

Use the **delete** form of this command to remove the system description.

Use the **show** form of this command to view the system description

## protocols snmp location <location>

Records the location of the system.

---

### Syntax

**set protocols snmp location** *location*

**delete protocols snmp location**

**show protocols snmp location**

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {  
    snmp {  
        location text  
    }  
}
```

---

### Parameters

---

<i>location</i>	Optional. Records the location of the system. This is stored as MIB-2 system information in the <b>snmpd.conf</b> configuration file. Letters, numbers, and hyphens are supported.
-----------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to specify the location of the system.

Use the **set** form of this command to specify the location of the system.

Use the **delete** form of this command to remove the system location.

Use the **show** form of this command to view the system location.

## protocols snmp trap-target <ipv4>

Specifies the IP address of the destination for SNMP traps.

---

### Syntax

```
set protocols snmp trap-target ipv4
delete protocols snmp trap-target ipv4
show protocols snmp trap-target
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
protocols {
  snmp {
    trap-target ipv4
  }
}
```

---

### Parameters

---

<i>ipv4</i>	Optional. Multi-node. The IP address of the destination for SNMP traps. You can specify multiple destinations for SNMP traps by creating multiple <b>trap-target</b> configuration nodes. Or, you can enter a space-separated list of IP addresses.
-------------	--

---

---

### Default

None.

---

### Usage Guidelines

Use this command to specify the IP address of the destination for SNMP traps.

Use the **set** form of this command to specify the IP address of the destination for SNMP traps.

Use the **delete** form of this command to remove a trap-target address.

Use the **show** form of this command to view the trap-target addresses.

## Chapter 6: Installation and Upgrade

This chapter describes the commands for installing and upgrading Vyatta software.

**NOTE** *This chapter provides reference information for software installation and upgrade commands. For detailed upgrade procedures, see the Vyatta System Installation and Upgrade Guide. For detailed installation procedures, see the Vyatta System Quick Start Guide.*

This chapter presents the following topics:

- Installation and Upgrade Overview
- Installation and Upgrade Commands

# Installation and Upgrade Overview

---

Software components updates are stored in *packages*. A package is a precompiled software component that can be installed or upgraded independently of other software components on your machine. Each package contains software and any scripts required to install, configure, or remove the package.

The Vyatta system keeps a record of which of all available packages have been installed on the system and which have not. The system also keeps track of differences between packages, including differences between versions of packages, as necessary, as well as dependencies between packages. When you direct the system to install a package (logged in as **root** using the **full-upgrade** command), the system also installs any packages on which the specified package depends and requires a reboot.

Installation of packages is atomic: all packages, and the packages on which they depend must install successfully, or else the installation will be rolled back, leaving the system as it was before.

Some updates (for example, a kernel upgrade) may require a system reboot. In general, though, updates do not require a reboot.

A detailed explanation of how to install Vyatta software is provided in the *Vyatta System Quick Start Guide*.

A detailed explanation of how to install and how to upgrade Vyatta software is provided in the *Vyatta System Installation and Upgrade Guide*.

# Installation and Upgrade Commands

This section presents the following commands.

## Configuration Commands

system package	Specifies the information needed for automatic software updates.
system package auto-sync <days>	Specifies how often the repository cache should be updated.
system package repository <repository>	Specifies information about the software repository.
system package repository <repository> components <component>	Specifies a repository component name.
system package repository <repository> description <desc>	Specifies a brief description for the repository.
system package repository <repository> distribution <dist>	Specifies a distribution name.
system package repository <repository> password <password>	Specifies a password for access to the repository.
system package repository <repository> url <url>	Specifies the URL of the server hosting the repository.
system package repository <repository> username <username>	Specifies a user name for access to the repository.

## Operational Commands

full-upgrade	Upgrades Vyatta system software.
install-system	Installs Vyatta system software to a persistent device.

# full-upgrade

Upgrades Vyatta system software.

---

## Syntax

**full-upgrade** [-h | -i | -k ]

---

## Command Mode

Operational mode.

---

## Parameters

---

-h	Help - Displays help for the full-upgrade command.
-i	Interactive upgrade - Provides prompts generated by the various packages during an upgrade. It should only be run on the local console (i.e. not via a telnet or ssh session) and is for expert users only.
-k	Keep all non-Vyatta packages - Where possible, this option keeps all non-Vyatta packages during the upgrade. There may be cases where, due to package dependencies, it is not possible to keep all non-Vyatta packages. Some of the non-Vyatta packages may be removed or changed to a different version. After the upgrade is complete, the system will consist of all the packages from the latest Vyatta release plus all the non-Vyatta packages that the system was able to keep. This option is recommended for users who have manually installed non-Vyatta software on their system and do not want to remove it.

---

---

## Default

When run with no parameters the command will display the same information as provided by the -h option.

---

## Usage Guidelines

Use this command to upgrade system software. You must be logged in as root to use this command. Prior to its use the appropriate repository must be configured. The repository information can be changed using the **system package repository <repository>** command (see page 1518) and related **system package repository** commands.

---

## Examples

Example 6-1 confirms that repository information is configured correctly and then proceeds to upgrade using full-upgrade.

### Example 6-1 “Upgrading Vyatta software with full-upgrade”

---

```
R1:~# configure
[edit]
root@R1# show system package
auto-sync 1
repository community {
    components main
    distribution stable
    url http://packages.vyatta.com/vyatta
}
[edit]
root@R1# exit
exit
R1:~# full-upgrade -k
Reading package lists...
Building dependency tree...
Reading state information...
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
Hit http://packages.vyatta.com testing Release.gpg
Hit http://packages.vyatta.com testing Release
Ign http://packages.vyatta.com testing/main Packages/DiffIndex
Hit http://packages.vyatta.com testing/main Packages
Reading package lists...
Reading package lists...
Building dependency tree...
Reading state information...
vyatta-base is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
R1:~#
```

---



# install-system

Installs Vyatta system software to a persistent device.

---

## Syntax

**install-system**

---

## Command Mode

Operational mode.

---

## Parameters

None.

---

## Default

None.

---

## Usage Guidelines

Use this command to install the Vyatta system software from the LiveCD onto a persistent device such as a hard disk.

For instructions on installing the Vyatta system software on other persistent devices, see the *Vyatta System Quick Start Guide*.

## system package

Specifies the information needed for automatic software updates.

---

### Syntax

```
set system package
delete system package
show system package
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
  }
}
```

---

### Parameters

None.

---

### Default

None.

---

### Usage Guidelines

Use this command to specify the information needed to obtain software updates from the Vyatta software archive.

Vyatta system packages are stored in the Vyatta software repositories. All customers have access to the community repository. Only customers with support contracts have access to the subscription repository.

Use the **set** form of this command to specify the information needed for automatic software updates.

Use the **delete** form of this command to remove the automatic software update configuration.

Use the **show** form of this command to view the automatic software update configuration.

## system package auto-sync <days>

Specifies how often the repository cache should be updated.

---

### Syntax

```
set system package auto-sync days
delete system package auto-sync
show system package auto-sync
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    auto-sync 1-4294967296
  }
}
```

---

### Parameters

---

<i>days</i>	Optional. Sets the system to update the repository cache every specified number of days. The range is 1 to 4294967296.
-------------	---

---

---

### Default

The default is 1.

---

### Usage Guidelines

Use this command to specify how often the repository cache should be updated.

Use the **set** form of this command to specify how often the repository cache should be updated.

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

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

## system package repository <repository>

Specifies information about the software repository.

---

### Syntax

```
set system package repository repository
delete system package repository repository
show system package repository repository
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
    }
  }
}
```

---

### Parameters

---

<i>repository</i>	Multi-node. A label for the repository. For example, “community”.  You can define more than one software repository by creating multiple <b>repository</b> nodes.
-------------------	---

---

---

### Default

None.

---

## Usage Guidelines

Use this command to create a configuration node to specify information about the repository. Note that a repository specified can be a Vyatta repository or some other debian package repository. When the **full-upgrade** command is run, all configured repositories are accessed.

Use the **set** form of this command to create the repository configuration node.

Use the **delete** form of this command to remove the repository configuration node.

Use the **show** form of this command to view the repository information.

## system package repository <repository> components <component>

Specifies a repository component name.

---

### Syntax

```
set system package repository repository components component
delete system package repository repository components component
show system package repository repository components
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
      components text
    }
  }
}
```

---

### Parameters

<i>repository</i>	<p>Multi-node. A label for the repository. For example, “community”.</p> <p>You can define more than one software repository by creating multiple <b>repository</b> nodes.</p>
<i>component</i>	<p>Multi-node. The repository component names.</p> <p>You can configure more than one component within a repository by creating multiple <b>components</b> nodes. The stock components is <b>main</b>.</p>

---

### Default

None.

---

## Usage Guidelines

Use **this** command to specify a repository component name.

Use the **set** form of this command to specify a repository component name.

Use the **delete** form of this command to remove the component name.

Use the **show** form of this command to view the components.

## system package repository <repository> description <desc>

Specifies a brief description for the repository.

---

### Syntax

```
set system package repository repository description desc
delete system package repository repository description
show system package repository repository description
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
      description text
    }
  }
}
```

---

### Parameters

<i>repository</i>	Multi-node. A label for the repository. For example, “community”.  You can define more than one software repository by creating multiple <b>repository</b> nodes.
<i>desc</i>	A brief description for the repository.

---

### Default

None.



---

## Usage Guidelines

Use this command to specify a repository description.

Use the **set** form of this command to specify a brief description for the repository.

Use the **delete** form of this command to remove the description.

Use the **show** form of this command to view the description.

## system package repository <repository> distribution <dist>

Specifies a distribution name.

---

### Syntax

```
set system package repository repository distribution dist
delete system package repository repository distribution
show system package repository repository distribution
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
      distribution text
    }
  }
}
```

---

### Parameters

---

<i>repository</i>	<p>Multi-node. A label for the repository. For example, “community”.</p> <p>You can define more than one software repository by creating multiple <b>repository</b> nodes.</p>
<i>dist</i>	<p>The name of the distribution. Valid values include:</p> <p><b>stable</b>: The latest stable release of software.</p> <p><b>testing</b>: The latest test version of software. This is typically newer software than the <b>stable</b> version but is currently being tested. The advantage of the <b>testing</b> distribution is that it contains the latest software features.</p>

---

---

## Default

None.

---

## Usage Guidelines

Use this command to specify a distribution name.

Use the **set** form of this command to specify a distribution name.

Use the **delete** form of this command to remove the distribution name.

Use the **show** form of this command to view the distribution name.

## system package repository <repository> password <password>

Specifies a password for access to the repository.

---

### Syntax

```
set system package repository repository password password
delete system package repository repository password
show system package repository repository password
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
      password text
    }
  }
}
```

---

### Parameters

<i>repository</i>	Multi-node. A label for the repository. For example, “community”.  You can define more than one software repository by creating multiple <b>repository</b> nodes.
<i>password</i>	The password required for access to the repository. Some repositories require a username and password and others do not. For example, the <b>packages.vyatta.com/vyatta-supported</b> repository requires a username and password but the <b>packages.vyatta.com/vyatta</b> repository does not.

---

### Default

None.

---

## Usage Guidelines

Use this command to specify a password for access to the repository.

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

Use the **delete** form of this command to remove the password.

Use the **show** form of this command to view the password.

## system package repository <repository> url <url>

Specifies the URL of the server hosting the repository.

---

### Syntax

```
set system package repository repository url url
delete system package repository repository url
show system package repository repository url
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
      url text
    }
  }
}
```

---

### Parameters

<i>repository</i>	Multi-node. A label for the repository. For example, “community”.  You can define more than one software repository by creating multiple <b>repository</b> nodes.
<i>url</i>	Mandatory. The full URL of the server hosting the software repository, including the path if required.

---

### Default

None.

---

## Usage Guidelines

Use this command to specify the URL of the server hosting the repository.

Use the **set** form of this command to specify the URL of the server hosting the repository.

Use the **delete** form of this command to remove the URL.

Use the **show** form of this command to view the URL.

## system package repository <repository> username <username>

Specifies a user name for access to the repository.

---

### Syntax

```
set system package repository repository username username
delete system package repository repository username
show system package repository repository username
```

---

### Command Mode

Configuration mode.

---

### Configuration Statement

```
system {
  package {
    repository text {
      username text
    }
  }
}
```

---

### Parameters

<i>repository</i>	Multi-node. A label for the repository. For example, “community”.  You can define more than one software repository by creating multiple <b>repository</b> nodes.
<i>username</i>	The user name required for access to the repository. Some repositories require a username and password and others do not. For example, the <b>packages.vyatta.com/vyatta-supported</b> repository requires a username and password but the <b>packages.vyatta.com/vyatta</b> repository does not.

---

### Default

None.



---

## Usage Guidelines

Use this command to specify a user name for access to the repository.

Use the **set** form of this command to specify a user name.

Use the **delete** form of this command to remove the user name.

Use the **show** form of this command to view the user name.

## Appendix A: SNMP MIB Support

This appendix lists the standard MIBs and traps supported by the Vyatta system.

Table A-1 Supported Standard MIBs

MIB Name	Document Title	Notes
BGP4-MIB	RFC 1657, <i>Definitions of Managed Objects for the Fourth Version of Border Gateway Protocol (BGP-4)</i>	Protocol MIB supported plus the following traps: <ul style="list-style-type: none"> <li>• BGP peer established</li> <li>• BGP peer backwards transition</li> </ul>
IF-MIB	RFC 2863, <i>The Interfaces Group MIB</i>	The following traps are supported: <ul style="list-style-type: none"> <li>• linkUp</li> <li>• linkDown</li> </ul>
OSPF2-MIB	RFC 1850, <i>OSPF Version 2 Management Information Base</i>	
OSPF Trap	MIB Module from RFC 1850, <i>OSPF Version 2 Management Information Base</i>	The following traps are supported: <ul style="list-style-type: none"> <li>• ospfVirtIfStateChange</li> <li>• ospfTxRetransmit</li> <li>• ospfVirtIfTxRetransmit</li> <li>• ospfOriginateLsa</li> <li>• ospfMaxAgeLsa</li> <li>• ospfLsdbOverflow</li> <li>• ospfLsdbApproachingOverflow</li> <li>• ospfIfStateChange</li> <li>• ospfIfStateChange</li> <li>• ospfVirtNbrStateChange</li> <li>• ospfIfConfigError</li> <li>• ospfVirtIfConfigError</li> <li>• ospfIfAuthFailure</li> <li>• ospfVirtIfAuthFailure</li> <li>• ospfIfRxBadPacket</li> <li>• ospfVirtIfRxBadPacket</li> </ul>
RIP	RFC 1724, <i>RIP Version 2 MIB Extension</i>	
SNMPv2-MIB	RFC 3418, <i>Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)</i>	The following traps are supported: <ul style="list-style-type: none"> <li>• coldStart</li> <li>• warmStart</li> </ul>

## Glossary of Acronyms

ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
AS	autonomous system
ARP	Address Resolution Protocol
BGP	Border Gateway Protocol
BIOS	Basic Input Output System
BPDU	Bridge Protocol Data Unit
CA	certificate authority
CHAP	Challenge Handshake Authentication Protocol
CLI	command-line interface
DDNS	dynamic DNS
DHCP	Dynamic Host Configuration Protocol
DLCI	data-link connection identifier
DMI	desktop management interface
DMZ	demilitarized zone
DNS	Domain Name System
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
eBGP	external BGP
EGP	Exterior Gateway Protocol

---

ECMP	equal-cost multipath
ESP	Encapsulating Security Payload
FIB	Forwarding Information Base
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HDLC	High-Level Data Link Control
I/O	Input/Output
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IGP	Interior Gateway Protocol
IPS	Intrusion Protection System
IKE	Internet Key Exchange
IP	Internet Protocol
IPOA	IP over ATM
IPsec	IP security
IPv4	IP Version 4
IPv6	IP Version 6
ISP	Internet Service Provider
L2TP	Layer 2 Tunneling Protocol
LACP	Link Aggregation Control Protocol
LAN	local area network
MAC	medium access control
MIB	Management Information Base
MLPPP	multilink PPP
MRRU	maximum received reconstructed unit
MTU	maximum transmission unit

---

---

NAT	Network Address Translation
ND	Neighbor Discovery
NIC	network interface card
NTP	Network Time Protocol
OSPF	Open Shortest Path First
OSPFv2	OSPF Version 2
OSPFv3	OSPF Version 3
PAM	Pluggable Authentication Module
PAP	Password Authentication Protocol
PCI	peripheral component interconnect
PKI	Public Key Infrastructure
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PVC	permanent virtual circuit
QoS	quality of service
RADIUS	Remote Authentication Dial-In User Service
RIB	Routing Information Base
RIP	Routing Information Protocol
RIPng	RIP next generation
Rx	receive
SNMP	Simple Network Management Protocol
SONET	Synchronous Optical Network
SSH	Secure Shell
STP	Spanning Tree Protocol
TACACS+	Terminal Access Controller Access Control System Plus

---

---

TCP	Transmission Control Protocol
ToS	Type of Service
Tx	transmit
UDP	User Datagram Protocol
vif	virtual interface
VLAN	virtual LAN
VPN	Virtual Private Network
VRRP	Virtual Router Redundancy Protocol
WAN	wide area network