VYATTA, INC. | Vyatta System

Remote Management

REFERENCE GUIDE

SSH
Telnet
Web GUI Access
SNMP

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

COPYRIGHT

PROPRIETARY NOTICES

RELEASE DATE: October 2012

DOCUMENT REVISION. 6.5R1 v01

RELEASED WITH: 6.5.0R1

PART NO. A0-0246-10-0004

Contents

Quick List of Commands v

List of Examples vi

Preface vii

Intended Audience viii
Organization of This Guide viii
Document Conventions ix
Vyatta Publications ix

Chapter 1 SSH 1

SSH Configuration 2
SSH Commands 3
service ssh 4
service ssh allow-root 5
service ssh disable-host-validation 6
service ssh disable-password-authentication 7
service ssh listen-address <ipv4> 9
service ssh port <port> 11
service ssh protocol-version <version> 12

Chapter 2 Telnet 14

Telnet Configuration 15
Telnet Commands 16
service telnet 17
service telnet allow-root 18
service telnet listen-address <ipv4> 19
service telnet port <port> 21
telnet <address> 22

Chapter 3 Web GUI Access (https) 24

Web GUI Access Configuration 25
Web GUI Access Commands 26
service https 27
service https listen-address <ipv4> 28
restart https 30

Chapter 4 SNMP 31
SNMP Overview 32
MIB Objects 32
Traps 32
SNMP Commands 32
SNMP Versions 33
Default Object IDs 33
Supported MIBs 33
SNMP Configuration Examples 35
Defining the SNMP Community 36
Specifying Trap Destinations 37
SNMP over IPv6 38
SNMP Commands 40
service snmp 41
service snmp community <community> 42
service snmp contact <contact> 44
service snmp description <desc> 45
service snmp listen-address <addr> 47
service snmp location <location> 49
service snmp trap-source <addr> 50
service snmp trap-target <addr> 52
show snmp 54

Glossary of Acronyms 55

Quick List of Commands

Use this list to help you quickly locate commands.

restart https 30
service https listen-address <ipv4> 28
service https 27
service snmp community <community> 42
service snmp contact <contact> 44
service snmp description <desc> 45
service snmp listen-address <addr> 47
service snmp location <location> 49
service snmp trap-source <addr> 50
service snmp trap-target <addr> 52
service snmp 41
service ssh allow-root 5
service ssh disable-host-validation 6
service ssh disable-password-authentication 7
service ssh listen-address <ipv4> 9
service ssh port <port> 11
service ssh protocol-version <version> 12
service ssh 4
service telnet allow-root 18
service telnet listen-address <ipv4> 19
service telnet port <port> 21
service telnet 17
show snmp 54
telnet <address> 22

List of Examples

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

Example 2-2 "telnet 192.168.1.77": Displaying the Telnet session being established 22
Example 3-2 "restart https" sample output 30
Example 4-4 "show snmp": Displaying output for SNMP 54

Preface

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

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

- Intended Audience
- 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
- IP services

Organization of This Guide

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

- Quick List of Commands

  Use this list to help you quickly locate commands.

- List of Examples

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

This guide has the following chapters:

Chapter	Description	Page
Chapter 1: SSH	This chapter explains how to set up Secure Shell (SSH) access on the Vyatta system.	1
Chapter 2: Telnet	This chapter explains how to set up Telnet access on the Vyatta system.	14
Chapter 3: Web GUI Access (https)	This chapter explains how to set up web GUI access on the Vyatta system.	24
Chapter 4: SNMP	This chapter describes the Vyatta system's support for SNMP.	31
Glossary of Acronyms		55

Document Conventions

This guide uses the following advisory paragraphs, as follows.

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

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

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

This document uses the following typographic conventions.

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

Vyatta Publications

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

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

Chapter 1: SSH

This chapter explains how to set up Secure Shell (SSH) access on the Vyatta system.

This chapter presents the following topics:

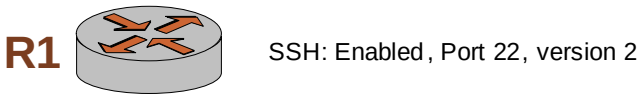
- SSH Configuration
- SSH Commands

SSH Configuration

Secure Shell (SSH) provides a secure mechanism to log on to the Vyatta system and access the Command Line Interface (CLI). Configuring SSH is optional, but is recommended to provide secure remote access to the Vyatta system. In addition to the standard password authentication provided by SSH, shared public key authentication is also available.

Example 1-1 enables SSH for password authentication on the default port (port 22), as shown in Figure 1-1. By default, only SSH version 2 is enabled.

Figure 1-1 Enabling SSH access



To enable the SSH service on the Vyatta system, perform the following steps in configuration mode.

Example 1-1 Enabling SSH access

Step	Command
Create the configuration node for the SSH service.	`vyatta@R1# set service ssh`
Commit the information	`vyatta@R1# commit` `Restarting OpenBSD Secure Shell server: sshd.`

Show the configuration.	`vyatta@R1# show service`
	`ssh {`
	`}`

SSH Commands

This chapter contains the following commands.

Configuration Commands	
service ssh	Enables SSH as an access protocol on the Vyatta system.
service ssh allow-root	Specifies that root logins are to be allowed on SSH connections.
service ssh disable-host-validation	Specifies that SSH should not validate clients via reverse DNS lookup.
service ssh disable-password-authentication	Specifies that SSH users are not to be authenticated using passords.
service ssh listen-address <ipv4>	Configures access to SSH on a specific address.
service ssh port <port>	Specifies the port the system will use for the SSH service.
service ssh protocol-version <version>	Specifies which versions of SSH are enabled.
Operational Commands	
None	

service ssh

Enables SSH as an access protocol on the Vyatta system.

Syntax

set service ssh

delete service ssh

show service ssh

Command Mode

Configuration mode.

Configuration Statement

```
service {
   ssh {
   }
}
```

Parameters

None.

Default

None.

Usage Guidelines

Use this command to configure the system to allow SSH requests from remote systems to the local system.

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

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

Use the **delete** form of this command to remove the SSH configuration. If you delete the SSH configuration node you will disable SSH access to the system.

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

service ssh allow-root

Specifies that root logins are to be allowed on SSH connections.

Syntax

set service ssh allow-root

delete service ssh allow-root

show service ssh

Command Mode

Configuration mode.

Configuration Statement

```
service {
    ssh {
        allow-root
    }
}
```

Parameters

None

Default

Root logins are not allowed on SSH connections.

Usage Guidelines

Use this command to specify that root logins are to be allowed on SSH connections.

NOTE *The **root** account is often the target of external attacks so its use is discouraged. The **vyatta** account provides sufficient privileges to administer the system.*

Use the **set** form of this command to specify that root logins are to be allowed on SSH connections.

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

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

service ssh disable-host-validation

Specifies that SSH should not validate clients via reverse DNS lookup.

Syntax

set service ssh disable-host-validation

delete service ssh disable-host-validation

show service ssh

Command Mode

Configuration mode.

Configuration Statement

```
service {
    ssh {
        disable-host-validation
    }
}
```

Parameters

None

Default

Client PTR/reverse-DNS records are resolved via DNS.

Usage Guidelines

Use this command to specify that SSH should not resolve client PTR/reverse-DNS records via a reverse DNS (PTR) lookup. This process can be time consuming and cause long delatys for clients trying to connect.

Use the **set** form of this command to specify that SSH should not resolve client PTR/reverse-DNS records via a reverse DNS (PTR) lookup.

Use the **delete** form of this command to restore the default configuration and allow reverse DNS lookups.

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

service ssh disable-password-authentication

Specifies that SSH users are not to be authenticated using passords.

Syntax

set service ssh disable-password-authentication

delete service ssh disable-password-authentication

show service ssh

Command Mode

Configuration mode.

Configuration Statement

```
service {
    ssh {
        disable-password-authentication
    }
}
```

Parameters

None

Default

Users are authenticated using passwords.

Usage Guidelines

Use this command to specify that SSH users are not to be authenticated using passwords. This is typically done in order for SSH users to be authenticated using shared public keys instead. Shared public key authentication is less susceptible to brute force guessing of common passwords. If password authentication is disabled then shared public keys must be configured for user authentication. For information on configuring public keys for user authentication see the *Vyatta Basic System Reference Guide*.

Use the **set** form of this command to specify that users are not to be authenticated by using passwords.

Use the **delete** form of this command to restore the default configuration and allow authentication by passwords.

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

service ssh listen-address <ipv4>

Configures access to SSH on a specific address.

Syntax

set service ssh listen-address *ipv4*

delete service ssh listen-address *ipv4*

show service ssh listen-address

Command Mode

Configuration mode.

Configuration Statement

```
service {
   ssh {
      listen-address ipv4
   }
}
```

Parameters

ipv4	Multi-node. An IP address that the ssh service listens for connection requests on. The address must be assigned to an interface. You can define more than one **listen-address** by creating multiple **listen-address** configuration nodes.

Default

Requests to access SSH will be accepted on any system IP address.

Usage Guidelines

Use this command to configure the system to accept requests for SSH access on specific addresses. This provides a way to limit access to the system.

Use the **set** form of this command to configure the system to accept requests for SSH access on specific addresses.

Use the **delete** form of this command to remove a listen-address.

Use the **show** form of this command to view the listen-address configuration.

service ssh port <port>

Specifies the port the system will use for the SSH service.

Syntax

set service ssh port *port*

delete service ssh port

show service ssh port

Command Mode

Configuration mode.

Configuration Statement

```
service {
    ssh {
        port port
    }
}
```

Parameters

port	The port the system will use for the SSH service. The range is 1 to 65534. The default is 22

Default

The SSH service runs on port 22.

Usage Guidelines

Use this command to specify the port the system will use for the SSH service.

Use the **set** form of this command to specify the port the system will use for the SSH service.

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

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

service ssh protocol-version <version>

Specifies which versions of SSH are enabled.

Syntax

set service ssh protocol-version *version*

delete service ssh protocol-version

show service ssh protocol-version

Command Mode

Configuration mode.

Configuration Statement

```
service {
    ssh {
        protocol-version version
    }
}
```

Parameters

version	Specifies which versions of SSH are enabled. Supported values are as follows: **v1**: SSH version 1 alone is enabled. **v2**: SSH version 2 alone is enabled. This is the recommended setting as **v1** is considered insecure. **all**: Both SSH version 1 and SSH version 2 are both enabled. The default value is **v2**.

Default

SSH version 2 alone is enabled.

Usage Guidelines

Use this command to specify which versions of SSH are enabled.

Use the **set** form of this command to specify which versions of SSH are enabled.

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

Use the **show** form of this command to view the protocol-version configuration.

Chapter 2: Telnet

This chapter explains how to set up Telnet access on the Vyatta system.

This chapter presents the following topics:

- Telnet Configuration
- Telnet Commands

Telnet Configuration

Configuring Telnet is optional, but creating the Telnet service will allow you to access the Vyatta system remotely. Example 2-1 enables Telnet on the default port (port 23), as shown in Figure 2-1.

Figure 2-1 Enabling Telnet access

To enable the Telnet service on the Vyatta system, perform the following steps in configuration mode.

Example 2-1 Enabling Telnet access

Step	Command
Create the configuration node for the Telnet service.	`vyatta@R1# set service telnet`
Commit the information.	`vyatta@R1# commit` `OK`

Show the configuration.	`vyatta@R1# show service`
	`  telnet {`
	`  }`

Telnet Commands

This chapter contains the following commands.

Configuration Commands	
service telnet	Configures Telnet as an access protocol on the system.
service telnet allow-root	Specifies that root logins are allowed on Telnet connections.
service telnet listen-address <ipv4>	Configures access to Telnet on a specific address.
service telnet port <port>	Specifies the port the system will use for the Telnet service.
Operational Commands	
telnet <address>	Creates a terminal session to a Telnet server.

service telnet

Configures Telnet as an access protocol on the system.

Syntax

set service telnet

delete service telnet

show service telnet

Command Mode

Configuration mode.

Configuration Statement

```
service {
   telnet {
   }
}
```

Parameters

None.

Default

None.

Usage Guidelines

Use this command to configure the system to accept Telnet as an access service to the system.

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

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

Use the **delete** form of this command to remove the Telnet configuration. If you delete the Telnet configuration node you will disable Telnet access to the system.

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

service telnet allow-root

Specifies that root logins are allowed on Telnet connections.

Syntax

set service telnet allow-root

delete service telnet allow-root

show service telnet

Command Mode

Configuration mode.

Configuration Statement

```
service {
   telnet {
      allow-root
   }
}
```

Parameters

None.

Default

Root logins are not allowed on Telnet connections.

Usage Guidelines

Use this command to specify that root logins are to be allowed on Telnet connections.

Use the **set** form of this command to specify that root logins are to be allowed on Telnet connections.

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

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

service telnet listen-address <ipv4>

Configures access to Telnet on a specific address.

Syntax

set service telnet listen-address *ipv4*

delete service telnet listen-address *ipv4*

show service telnet listen-address

Command Mode

Configuration mode.

Configuration Statement

```
service {
   telnet {
      listen-address ipv4
   }
}
```

Parameters

ipv4	Multi-node. An IP address that the telnet service listens for connection requests on. The address must be assigned to an interface. You can define more than one **listen-address** by creating multiple **listen-address** configuration nodes.

Default

Requests to access Telnet will be accepted on any system IP address.

Usage Guidelines

Use this command to configure the system to accept requests for Telnet access on specific addresses. This provides a way to limit access to the system.

Use the **set** form of this command to configure the system to accept requests for Telnet access on specific addresses.

Use the **delete** form of this command to remove a listen-address.

Use the **show** form of this command to view the listen-address configuration.

service telnet port <port>

Specifies the port the system will use for the Telnet service.

Syntax

set service telnet port *port*

delete service telnet port

show service telnet port

Command Mode

Configuration mode.

Configuration Statement

```
service {
   telnet {
      port port
   }
}
```

Parameters

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

Default

The default is port 23.

Usage Guidelines

Use this command to specify the port the system will use for the Telnet service.

Use the **set** form of this command to specify the port the system will use for the Telnet service.

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

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

telnet <address>

Creates a terminal session to a Telnet server.

Syntax

telnet *address*

Command Mode

Operational mode.

Parameters

address	Mandatory. The IP address or hostname of the Telnet server to connect to. The system connects through port 23 (the well-known port for the Telnet service).

Default

None

Usage Guidelines

Use this command to create a terminal session to a remote machine running a Telnet service.

Examples

Example 2-2 shows a telnet session being established to 192.168.1.77.

Example 2-2 "telnet 192.168.1.77": Displaying the Telnet session being established

```
vyatta@R1:~$ telnet 192.168.1.77

Entering character mode
Escape character is '^]'.


Welcome to Vyatta
vyatta login:
```

Chapter 3: Web GUI Access (https)

This chapter explains how to set up web GUI access on the Vyatta system.

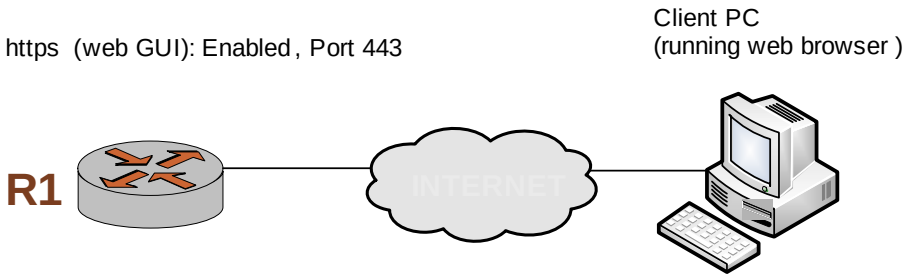
This chapter presents the following topics:

- Web GUI Access Configuration
- Web GUI Access Commands

Web GUI Access Configuration

Configuring web GUI access is optional, but creating the https service will allow you to access the web GUI on the Vyatta system remotely via a web browser. Example 3-1 enables https on the default port (port 443), as shown in Figure 3-1.

Figure 3-1 Enabling web GUI access



To enable the https service on the Vyatta system to provide access to the web GUI, perform the following steps in configuration mode.

Example 3-1 Enabling web GUI access

Step	Command
Create the configuration node for the https service.	`vyatta@R1# set service https`
Commit the information.	`vyatta@R1# commit`

Show the configuration.	`vyatta@R1# show service`
	` https {`
	` }`

Web GUI Access Commands

This chapter contains the following commands.

Configuration Commands	
service https	Configures access to the web GUI.
service https listen-address <ipv4>	Configures access to the web GUI on a specific address.
Operational Commands	
restart https	Restarts the https server.

service https

Configures access to the web GUI.

Syntax

set service https

delete service https

show service https

Command Mode

Configuration mode.

Configuration Statement

```
service {
   https
}
```

Parameters

None.

Default

None.

Usage Guidelines

Use this command to configure access to the web GUI via https (port 443). Once configured, the web GUI can be accessed by specifying one of the system IP addresses from a web browser.

Use the **set** form of this command to create the https configuration and enable access to the web GUI.

Use the **delete** form of this command to remove the https configuration. If you delete the https configuration node you will disable web GUI access to the system.

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

service https listen-address <ipv4>

Configures access to the web GUI on a specific address.

Syntax

set service https listen-address *ipv4*

delete service https listen-address *ipv4*

show service https listen-address

Command Mode

Configuration mode.

Configuration Statement

```
service {
   https {
      listen-address ipv4
   }
}
```

Parameters

ipv4	Multi-node. An IP address that the https service listens for connection requests on. The address must be assigned to an interface. You can define more than one **listen-address** by creating multiple **listen-address** configuration nodes.

Default

Requests to access the web GUI will be accepted on any system IP address.

Usage Guidelines

Use this command to configure the system to accept requests for web GUI access on specific addresses. This provides a way to limit access to the system.

Use the **set** form of this command to configure the system to accept requests for web GUI access on specific addresses.

Use the **delete** form of this command to remove a listen-address.

Use the **show** form of this command to view the listen-address configuration.

restart https

Restarts the https server.

Syntax

restart https

Command Mode

Operational mode.

Parameters

None.

Usage Guidelines

Use this command to restart the https server.

Examples

Example 3-2 shows the output resulting from the **restart https** command.

Example 3-2 "restart https" sample output

```
vyatta@R1> restart https
Stopping web server: lighttpd.
Starting web server: lighttpd.
Stopping PAGER server
Starting PAGER server
Stopping API PAGER server
Starting API PAGER server
spawn-fcgi: child spawned successfully: PID: 4219
vyatta@R1>
```

Chapter 4: SNMP

This chapter describes the Vyatta system's support for SNMP.

This chapter presents the following topics:

- SNMP Overview
- Supported MIBs
- SNMP Configuration Examples
- SNMP Commands

SNMP Overview

This section presents the following topics:

- MIB Objects
- Traps
- SNMP Commands
- SNMP Versions

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.

The Vyatta system supports SNMP over both IPv4 and IPv6 networks.

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)

Default Object IDs

Two default object IDs set by Vyatta are as follows:

- sysObjectID = 1.3.6.1.4.1.30803
- sysDescr = Vyatta VSE6.4-2012.07.28

The **sysDescr** object ID is updated automatically with each new release. It can also can be changed using the "service snmp description <desc>" on page 45.

Supported MIBs

MIBs are typically located in the **/usr/share/snmp/mibs** directory.

Table 4-1 lists the standard MIBs and traps supported by the Vyatta system. RFCs can be found at http://tools.ietf.org

Table 4-1 Supported Standard MIBs

MIB Name	Document Title	Notes
BGP4-MIB	RFC 1657, *Definitions of Managed Objects for the Fourth Version of Border Gateway Protocol (BGP-4)*	Protocol MIB supported plus the following traps: • bgpEstablished • bgpBackwardTransition
HOST-RESOURCES-MIB	RFC 2790, *Host Resources MIB*	
IF-MIB	RFC 2863, *The Interfaces Group MIB*	The following traps are supported: • linkUp • linkDown
IP-MIB	RFC 2011, *SNMPv2 Management Information Base for the Internet Protocol using SMIv2*	Only packets where the Vyatta system is an endpoint are accounted. Forwarded traffic is not accounted.
IPV6-TC	RFC2465, *Management Information Base for IP Version 6: Textual Conventions and General Group*	Only packets where the Vyatta system is an endpoint are accounted. Forwarded traffic is not accounted.
IPV6-UDP-MIB	RFC 2454, *IP Version 6 Management Information Base for the User Datagram Protocol*	Only packets where the Vyatta system is an endpoint are accounted. Forwarded traffic is not accounted.
KEEPALIVED-MIB	Authored by Vincent Bernat. Extends the keepalived daemon to support the Net-SNMP agentx protocol. Provides additional information specific to the Vyatta implementation, such as state information, sync group state information, and so on.	
OSPF-MIB	RFC 1850, *OSPF Version 2 Management Information Base*	
RFC1213-MIB	RFC 1213, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II*	
RFC 2787 VRRP-MIB	RFC 2787 *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*	Only the vrrpTrapNewMaster object is supported.
RIPv2-MIB	RFC 1724, *RIP Version 2 MIB Extension*	

Table 4-1 Supported Standard MIBs

MIB Name	Document Title	Notes
SNMPv2-MIB	RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*	The following traps are supported: • coldStart • warmStart
TCP-MIB	RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*	Only packets where the Vyatta system is an endpoint are accounted. Forwarded traffic is not accounted.
UDP-MIB	RFC 4113, *Management Information Base for the User Datagram Protocol (UDP)*	Only packets where the Vyatta system is an endpoint are accounted. Forwarded traffic is not accounted.

SNMP Configuration Examples

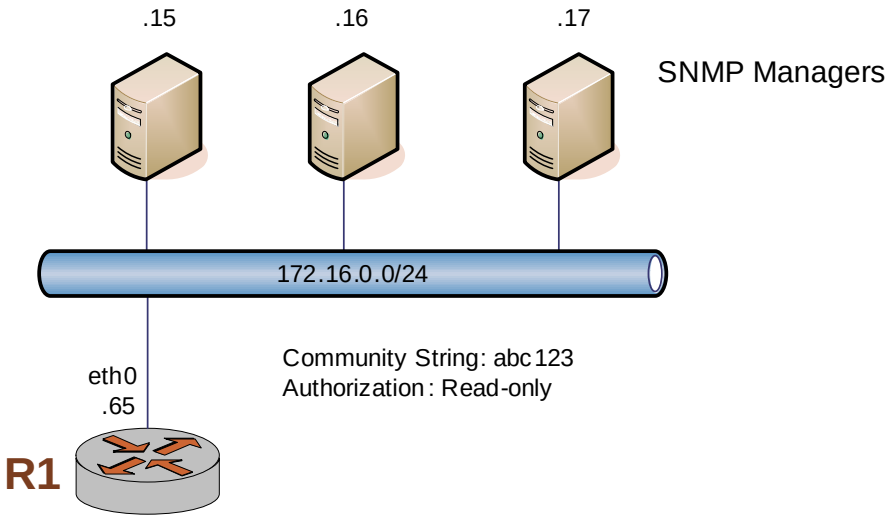
This section presents the following topics:

- Defining the SNMP Community
- Specifying Trap Destinations
- SNMP over IPv6

To configure SNMP, 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 4-1.

Figure 4-1 Configuring SNMP communities and traps



This section includes the following examples:

- Example 4-1 Defining an SNMP community
- Example 4-2 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 or networks are explicitly defined, then any client presenting the correct community string is granted the access privilege specified in the **authorization** option.
- If any client or network is defined, then only explicitly listed clients or networks are granted access to the system. Those clients will have the access privilege specified by the **authorization** option. (The default is read-only.)

Example 4-1 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-only access is provided for this community.

To define an SNMP community, perform the following steps in configuration mode.

Example 4-1 Defining an SNMP community

Step	Command
Create the **snmp** configuration node and the **community** configuration node. Set the community string. Note that using the **edit** command will create the community if it does not already exist. Navigate to the configuration node of the community for easier configuration.	`vyatta@R1# edit service snmp community abc123` `[edit service snmp community abc123]`

List the SNMP clients making up this community.	`vyatta@R1# set client 176.16.0.15`
	`vyatta@R1# set client 176.16.0.16`
	`vyatta@R1# set client 176.16.0.17`

Set the privilege level for this community to read-only.	`vyatta@R1# set authorization ro`
Commit the change.	`vyatta@R1# commit`

Verify the configuration.	`vyatta@R1# show`
	` authorization ro`
	` client 176.16.0.15`
	` client 176.16.0.16`
	` client 176.16.0.17`

Return to the top of the configuration tree.	`vyatta@R1# top`

Specifying Trap Destinations

Example 4-2 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 4-2 Specifying SNMP trap destinations

Step	Command

Define the trap destinations, one at a time.	`vyatta@R1# set service snmp trap-target 176.16.0.15`
	`vyatta@R1# set service snmp trap-target 176.16.0.16`
	`vyatta@R1# set service snmp trap-target 176.16.0.17`

Commit the change.	`vyatta@R1# commit`

Example 4-2 Specifying SNMP trap destinations

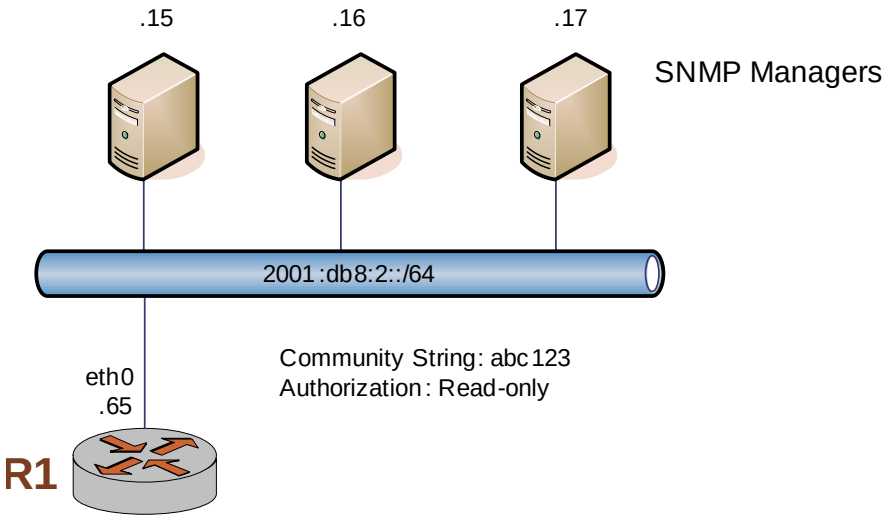
Step	Command

Verify the configuration.	`vyatta@R1# show service snmp trap-target`
	`trap-target 176.16.0.15 {`
	`}`
	`trap-target 176.16.0.16 {`
	`}`
	`trap-target 176.16.0.17 {`
	`}`

SNMP over IPv6

This sequence is the same as the previous example but uses IPv6 addresses. When you have finished, the system will be configured as shown in Figure 4-2.

Figure 4-2 Configuring SNMP communities and traps - IPv6



To define the SNMP configuration, perform the following steps in configuration mode.

Example 4-3 Defining the SNMP configurationy

Step	Command
Create the **snmp** configuration node and the **community** configuration node. Set the community string.	`vyatta@R1# set service snmp community abc123`

List the SNMP clients making up this community.	`vyatta@R1# set service snmp community abc123 client 2001:db8:2::15`
	`vyatta@R1# set service snmp community abc123 client 2001:db8:2::16`
	`vyatta@R1# set service snmp community abc123 client 2001:db8:2::17`

Set the privilege level for this community to read-only.	`vyatta@R1# set service snmp community abc123 authorization ro`

Define the trap destinations, one at a time.	`vyatta@R1# set service snmp trap-target 2001:db8:2::15`
	`vyatta@R1# set service snmp trap-target 2001:db8:2::16`
	`vyatta@R1# set service snmp trap-target 2001:db8:2::17`

Commit the change.	`vyatta@R1# commit`

Verify the configuration.	`vyatta@R1# show service snmp`
	` community abc123 {`
	`  authorization ro`
	`  client 176.16.0.15`
	`  client 176.16.0.16`
	`  client 176.16.0.17`
	`  client 2001:db8:2::15`
	`  client 2001:db8:2::16`
	`  client 2001:db8:2::17`
	` }`
	` trap-target 176.16.0.15 {`
	` }`
	` trap-target 176.16.0.16 {`
	` }`
	` trap-target 176.16.0.17`
	` }`
	` trap-target 2001:db8:2::15 {`
	` }`
	` trap-target 2001:db8:2::16`
	` }`
	` trap-target 2001:db8:2::17 {`
	` }`

SNMP Commands

This section presents the following commands.

Configuration Commands	
service snmp	Defines SNMP information for the Vyatta system.
service snmp community <community>	Defines an SNMP community.
service snmp contact <contact>	Records contact information for the system.
service snmp description <desc>	Records a brief description of the system.
service snmp listen-address <addr>	Specifies the IP address the SNMP agent will listen for requests on.
service snmp location <location>	Records the location of the system.
service snmp trap-source <addr>	Specifies the IP address of the source of SNMP traps.
service snmp trap-target <addr>	Specifies the IP address of a destination for SNMP traps.
Operational Commands	
show snmp	Displays SNMP statistics.

service snmp

Defines SNMP information for the Vyatta system.

Syntax

set service snmp

delete service snmp

show service snmp

Command Mode

Configuration mode.

Configuration Statement

```
service {
    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.

service snmp community <community>

Defines an SNMP community.

Syntax

set service snmp community *community* [**authorization** *auth* | **client** *addr* | **network** *net*]

delete service snmp community *community* [**authorization** | **client** | **network**]

show service snmp community *community* [**authorization** | **client** | **network**]

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        community community {
            authorization auth
            client addr
            network net
        }
    }
}
```

Parameters

community	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 **community** configuration nodes.

auth	Optional. Specifies the privileges this community will have. Supported values are as follows: **ro**: This community can view system information, but not change it. **rw**: This community has read-write privileges. The default is **ro**.
addr	Optional. Multi-node. The IPv4 or IPv6 address of an SNMP client in this community that is authorized to access the system. You can define more than one client by creating the **client** configuration node multiple times.
net	Optional. Multi-node. The IPv4 or IPv6 network of SNMP networks in this community that are authorized to access the server. You can define more than one network by creating the **network** configuration node multiple times.

Default

By default, no community string is defined.

Usage Guidelines

Use this command to specify an SNMP community.

If no SNMP clients or networks are explicitly defined, then any client presenting the correct community string is granted the access privilege specified by the authorization option. If any client or network is defined, then only explicitly listed clients or networks are granted access to the system.

Use the **set** form of this command to specify an SNMP community.

Use the **delete** form of this command to remove SNMP community configuration or to restore the default value of an option.

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

service snmp contact <contact>

Records contact information for the system.

Syntax

set service snmp contact *contact*

delete service snmp contact

show service snmp contact

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        contact contact
    }
}
```

Parameters

contact	Optional. Records contact information for the system. This is stored as MIB-2 system information. 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.

service snmp description <desc>

Records a brief description of the system.

Syntax

set service snmp description *desc*

delete service snmp description

show service snmp description

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        description desc
    }
}
```

Parameters

desc	Optional. Records a brief description of the system. This is stored as MIB-2 system information. Letters, numbers, and hyphens are supported. When set, this text is stored as the object ID **sysDescr**. By default **sysDescr** is set to **Vyatta**[*version-string*], where *version-string* is the version of Vyatta software.

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 descriptionof 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

service snmp listen-address <addr>

Specifies the IP address the SNMP agent will listen for requests on.

Syntax

set service snmp listen-address *addr* [**port** *port*]

delete service snmp listen-address *addr* [**port**]

show service snmp listen-address *ipv4* [**port**]

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        listen-address addr {
            port port
        }
    }
}
```

Parameters

addr	Optional. Multi-node. The IPv4 or IPv6 address the SNMP agent will listen for requests on. You can specify multiple listen addresses for SNMP by creating multiple *listen-address* configuration nodes.
port	The UDP port used for listening . The default value is 161.

Default

The SNMP agent will listen on all addresses on port 161.

Usage Guidelines

Use this command to specify the IPv4 or IPv6 address and port the SNMP agent will listen for requests on.

Use the **set** form of this command to specify the listen-address parameters.

Use the **delete** form of this command to remove listen-address parameters.

Use the **show** form of this command to view the listen-address configuration.

service snmp location <location>

Records the location of the system.

Syntax

set service snmp location *location*

delete service snmp location

show service snmp location

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        location location
    }
}
```

Parameters

location	Optional. Records the location of the system. This is stored as MIB-2 system information. 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.

service snmp trap-source <addr>

Specifies the IP address of the source of SNMP traps.

Syntax

set service snmp trap-source *addr*

delete service snmp trap-source *addr*

show service snmp trap-source

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        trap-source addr
    }
}
```

Parameters

addr	The IPv4 or IPv6 address of the source of SNMP traps. This address will be included source of SNMP traps in SNMP messages sent to an SNMP server. The address must an address configured on one of the system interfaces.

Default

By default, the system automatically selects the primary IP address address of the interface facing the trap target.

Usage Guidelines

Use this command to specify the IPv4 or IPv6 address of the source of SNMP traps.

Use the **set** form of this command to specify the IP address of the source of SNMP traps.

Use the **delete** form of this command to remove a trap-source address and have the system select the source address automatically.

Use the **show** form of this command to view the trap-source addresses.

service snmp trap-target <addr>

Specifies the IP address of a destination for SNMP traps.

Syntax

set service snmp trap-target *addr* [**community** *community* | **port** *port*]

delete service snmp trap-target *addr* [**community** | **port**]

show service snmp trap-target *ipv4* [**community** | **port**]

Command Mode

Configuration mode.

Configuration Statement

```
service {
    snmp {
        trap-target addr {
            community community
            port port
        }
    }
}
```

Parameters

addr	Optional. Multi-node. The IPv4 or IPv6 address of the destination for SNMP traps. You can specify multiple destinations for SNMP traps by creating multiple **trap-target** configuration nodes. Or, you can enter a space-separated list of IP addresses.
community	The community used when sending trap information. The default value is **public**.
port	The destination UDP port used for trap notification. The default value is 162.

Default

None.

Usage Guidelines

Use this command to specify the IPv4 or IPv6 address and port of the destination for SNMP traps as well as the community used when sending trap information.

Use the **set** form of this command to specify the trap-target parameters.

Use the **delete** form of this command to remove trap-target parameters.

Use the **show** form of this command to view the trap-target configuration.

show snmp

Displays SNMP statistics.

Syntax

show snmp

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display SNMP statistics.

Examples

Example 4-4 shows the output for **show snmp**.

Example 4-4 "show snmp": Displaying output for SNMP

```
vyatta@R1:~$ show snmp
[UDP: [127.0.0.1]:161->[0.0.0.0]]=>[Vyatta 999.larkspurse.06200031] Up:
0:02:40.80
Interfaces: 5, Recv/Trans packets: 545097/179020 | IP: 202587/89811
vyatta@R1:~$
```

Glossary of Acronyms

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

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

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

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

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