

VYATTA, INC.

| Vyatta System

Basic Routing

REFERENCE GUIDE

Forwarding and Routing
Static Routes



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

COPYRIGHT

Copyright © 2005–2011 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.

PROPRIETARY NOTICES

Vyatta is a registered trademark of Vyatta, Inc.

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

XenServer, and XenCenter are trademarks of Citrix Systems, Inc.

All other trademarks are the property of their respective owners.

RELEASE DATE: July 2011

DOCUMENT REVISION: R6.3 v01

RELEASED WITH: R6.3.0

PART NO. A0-0215-10-0011

Table of Contents

Quick Reference to Commands	v
Quick List of Examplesvi
Prefaceviii
Intended Audience	ix
Organization of This Guide	ix
Document Conventions	x
Vyatta Publications	xi
Chapter 1 Forwarding and Routing	1
Forwarding and Routing Commands	2
clear ip prefix-list	4
clear ip route cache	5
clear ipv6 prefix-list	6
clear ipv6 route cache	7
show ip forwarding	8
show ip route	9
show ip route <ipv4net> longer-prefixes	11
show ip route cache	12
show ip route connected	14
show ip route forward	15
show ip route kernel	17
show ip route static	18
show ip route summary	19
show ip route supernets-only	20
show ipv6 route	21
show ipv6 route <ipv6net> longer-prefixes	22
show ipv6 route bgp	23
show ipv6 route cache	24
show ipv6 route connected	25

show ipv6 route forward	26
show ipv6 route kernel	27
show ipv6 route ripng	28
show ipv6 route static	29
show ipv6 route summary	30
show table	31
Chapter 2 Static Routes	32
Static Route Configuration	33
Static Routes Overview	33
Configuring Static Routes	33
Floating Static Routes	34
Showing Static Routes in the Routing Table	35
Static IPv6 Route Configuration	35
Verify That IPv6 Forwarding is Enabled	36
Add the Default IPv6 Route	37
Add a Static IPv6 Route	38
Confirm Connectivity	38
Static Route Commands	40
protocols static interface-route <subnet> next-hop-interface <interface>	41
protocols static interface-route6 <subnet> next-hop-interface <ethx>	43
protocols static route <subnet> blackhole	45
protocols static route <subnet> next-hop <address>	47
protocols static route6 <subnet> blackhole	49
protocols static route6 <subnet> next-hop <address>	51
Glossary of Acronyms	53

Quick Reference to Commands

Use this section to help you quickly locate a command.

clear ip prefix-list	4
clear ip route cache	5
clear ipv6 prefix-list	6
clear ipv6 route cache	7
protocols static interface-route <subnet> next-hop-interface <interface>	41
protocols static interface-route6 <subnet> next-hop-interface <ethx>	43
protocols static route <subnet> blackhole	45
protocols static route <subnet> next-hop <address>	47
protocols static route6 <subnet> blackhole	49
protocols static route6 <subnet> next-hop <address>	51
show ip forwarding	8
show ip route	9
show ip route <ipv4net> longer-prefixes	11
show ip route cache	12
show ip route connected	14
show ip route forward	15
show ip route kernel	17
show ip route static	18
show ip route summary	19
show ip route supernets-only	20
show ipv6 route	21
show ipv6 route <ipv6net> longer-prefixes	22
show ipv6 route bgp	23
show ipv6 route cache	24
show ipv6 route connected	25
show ipv6 route forward	26
show ipv6 route kernel	27
show ipv6 route ripng	28
show ipv6 route static	29
show ipv6 route summary	30
show table	31

Quick List of Examples

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

Example 1-1	Displaying IP forwarding status	8
Example 1-2	Displaying routes in the RIB and FIB	9
Example 1-3	Displaying routing information about a specific address	10
Example 1-4	Displaying routes with longer prefixes	11
Example 1-5	Listing routes in the kernel route cache	12
Example 1-6	Displaying information about a route in the kernel route cache	13
Example 1-7	Displaying connected routes	14
Example 1-8	Displaying routes in the FIB	15
Example 1-9	Displaying information about a route in the FIB	16
Example 1-10	Displaying kernel routes	17
Example 1-11	Displaying static routes	18
Example 1-12	Displaying a summary of routes	19
Example 1-13	Displaying supernet routes	20
Example 1-14	“show ipv6 route summary”: Displaying a summary of IPv6 routes	30
Example 1-15	Displaying the routing table	31
Example 2-1	Creating a static route	34
Example 2-3	Determine if forwarding is enabled on R1	36
Example 2-4	Enable forwarding on R1	36
Example 2-5	Add the default route on R4	37
Example 2-6	Add a static route on R2	38
Example 2-7	Confirm connectivity between R2 and R4	38
Example 2-8	Confirm connectivity between R2 and R4 via R1	39

Preface

This document presents information about forwarding and routing on the Vyatta system.

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 Reference to Commands](#)
Use this list to help you quickly locate commands.
- [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:

Chapter	Description	Page
Chapter 1: Forwarding and Routing	This chapter describes commands for forwarding and basic routing.	1
Chapter 2: Static Routes	This chapter explains how to set static routes using the Vyatta system.	32
Glossary of Acronyms		53

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.
<i>italics</i>	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].
<i>num1–numN</i>	A inclusive range of numbers. An example is 1–65535, which means 1 through 65535, inclusive.
<i>arg1..argN</i>	A range of enumerated values. An example is eth0..eth3, which means eth0, eth1, eth2, or eth3.
<i>arg[arg...]</i> <i>arg[,arg...]</i>	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: Forwarding and Routing

This chapter describes commands for forwarding and basic routing.

This chapter presents the following topics:

- [Forwarding and Routing Commands](#)

Forwarding and Routing Commands

This chapter contains the following commands.

Configuration Commands

None

Operational Commands

<code>clear ip prefix-list</code>	Clears prefix list statistics or status.
<code>clear ip route cache</code>	Flushes the kernel route cache.
<code>clear ipv6 prefix-list</code>	Clears prefix list statistics or status.
<code>clear ipv6 route cache</code>	Flushes the kernel IPv6 route cache.
<code>show ip forwarding</code>	Displays IP forwarding status.
<code>show ip route</code>	Displays routes stored in the RIB and FIB.
<code>show ip route <ipv4net> longer-prefixes</code>	Displays prefixes longer than a specified prefix.
<code>show ip route cache</code>	Displays the kernel route cache.
<code>show ip route connected</code>	Displays directly connected routes.
<code>show ip route forward</code>	Displays routes stored in the FIB.
<code>show ip route static</code>	Displays static routes.
<code>show ip route kernel</code>	Displays kernel routes.
<code>show ip route summary</code>	Displays routes summary.
<code>show ip route supernets-only</code>	Displays supernet routes.
<code>show ipv6 route</code>	Displays IPv6 routes stored in the RIB and FIB.
<code>show ipv6 route <ipv6net> longer-prefixes</code>	Displays IPv6 prefixes longer than a specified prefix.
<code>show ipv6 route bgp</code>	Displays IPv6 BGP routes.
<code>show ipv6 route cache</code>	Displays the kernel IPv6 route cache.
<code>show ipv6 route connected</code>	Displays IPv6 connected routes.
<code>show ipv6 route forward</code>	Displays IPv6 routes stored in the FIB.
<code>show ipv6 route kernel</code>	Displays IPv6 kernel routes.
<code>show ipv6 route ripng</code>	Displays IPv6 RIPng routes.

<code>show ipv6 route static</code>	Displays IPv6 static routes.
<code>show ipv6 route summary</code>	Displays IPv6 routes summary.
<code>show table</code>	Displays the system's routing table.

clear ip prefix-list

Clears prefix list statistics or status.

Syntax

```
clear ip prefix-list [list-name [ipv4net]]
```

Command Mode

Operational mode.

Parameters

<i>list-name</i>	Optional. Clears statistics for the specified prefix list.
<i>ipv4net</i>	Optional. Clears statistics for the specified network.

Default

Statistics for all prefix-lists are cleared.

Usage Guidelines

Use this command to clear prefix list statistics or status.

clear ip route cache

Flushes the kernel route cache.

Syntax

```
clear ip route cache [ipv4net]
```

Command Mode

Operational mode.

Parameters

<i>ipv4net</i>	Optional. Flushes the specified route from the kernel route cache.
----------------	--

Default

Flushes the entire route cache.

Usage Guidelines

Use this command to flush the kernel route cache or to flush a specific route from the cache.

clear ipv6 prefix-list

Clears prefix list statistics or status.

Syntax

```
clear ipv6 prefix-list [list-name [ipv6net]]
```

Command Mode

Operational mode.

Parameters

<i>list-name</i>	Optional. Clears statistics for the specified prefix list.
<i>ipv6net</i>	Optional. Clears statistics for the specified network.

Default

Statistics for all prefix-lists are cleared.

Usage Guidelines

Use this command to clear prefix list statistics or status.

clear ipv6 route cache

Flushes the kernel IPv6 route cache.

Syntax

```
clear ipv6 route cache [ipv6net]
```

Command Mode

Operational mode.

Parameters

<i>ipv6net</i>	Optional. Flushes the specified route from the kernel IPv6 route cache.
----------------	---

Default

Flushes the entire IPv6 route cache.

Usage Guidelines

Use this command to flush the kernel IPv6 route cache or a flush a specific route from the cache.

show ip forwarding

Displays IP forwarding status.

Syntax

```
show ip forwarding
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display the current IP forwarding status.

Examples

[Example 1-1](#) shows how to display the status of IP forwarding.

Example 1-1 Displaying IP forwarding status

```
vyatta@vyatta:~$ show ip forwarding
IP forwarding is on
vyatta@vyatta:~$
```

show ip route

Displays routes stored in the RIB and FIB.

Syntax

```
show ip route [ipv4 | ipv4net]
```

Command Mode

Operational mode.

Parameters

<i>ipv4</i>	Optional. Displays routing information for the specified address.
<i>ipv4net</i>	Optional. Displays routing information for the specified prefix.

Default

Lists all routes in the RIB and FIB.

Usage Guidelines

Use this command to display active prefixes stored in the Routing Information Base (RIB), as well as those stored in the Forwarding Information Base (FIB).

The routes shown in the FIB can also be seen using [show ip route forward command](#).

Examples

[Example 1-2](#) shows how to display routes in the RIB and FIB

Example 1-2 Displaying routes in the RIB and FIB

```
vyatta@vyatta:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

S>* 0.0.0.0/0 [1/0] via 10.1.0.1, eth0
O 10.1.0.0/24 [110/10] is directly connected, eth0, 05:35:15
C>* 10.1.0.0/24 is directly connected, eth0
```

```
O>* 10.192.32.0/24 [110/20] via 10.1.0.45, eth0, 05:35:15
O>* 10.192.128.0/24 [110/11] via 10.1.0.66, eth0, 05:35:15
O>* 10.192.128.1/32 [110/11] via 10.1.0.66, eth0, 05:35:15
O>* 10.192.129.0/24 [110/11] via 10.1.0.66, eth0, 05:35:15
O>* 10.192.130.0/24 [110/11] via 10.1.0.66, eth0, 05:35:15
O>* 10.192.131.0/24 [110/11] via 10.1.0.66, eth0, 05:35:15
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.0.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.1.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.2.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.3.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.4.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.5.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.6.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.7.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.8.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
O>* 172.16.9.0/24 [110/11] via 10.1.0.4, eth0, 05:35:15
C>* 172.16.234.0/25 is directly connected, eth1
S>* 192.94.202.0/24 [1/0] via 172.16.234.27, eth1
vyatta@vyatta:~$
```

[Example 1-3](#) shows information how to display information for the route to address 10.192.128.1.

Example 1-3 Displaying routing information about a specific address

```
vyatta@vyatta:~$ show ip route 10.192.128.1
Routing entry for 10.192.128.1/32
  Known via "ospf", distance 110, metric 11, best
  Last update 09:47:07 ago
  * 10.1.0.66, via eth0
vyatta@vyatta:~$
```

show ip route <ipv4net> longer-prefixes

Displays prefixes longer than a specified prefix.

Syntax

```
show ip route ipv4net longer-prefixes
```

Command Mode

Operational mode.

Parameters

<i>ipv4net</i>	Mandatory. Displays all prefixes longer than the specified prefix.
----------------	--

Default

None.

Usage Guidelines

Use this command to display all prefixes in the Routing Information Base (RIB) that are longer than a given IP address or prefix.

Examples

[Example 1-4](#) shows how to list prefixes longer than the prefix 10.192.128.0/24.

Example 1-4 Displaying routes with longer prefixes

```
vyatta@vyatta:~$ show ip route 10.192.128.0/24 longer-prefixes
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

O>* 10.192.128.0/24 [110/11] via 10.1.0.66, eth0, 09:36:20
O>* 10.192.128.1/32 [110/11] via 10.1.0.66, eth0, 09:36:20
vyatta@vyatta:~$
```

show ip route cache

Displays the kernel route cache.

Syntax

```
show ip route cache [ipv4net]
```

Command Mode

Operational mode.

Parameters

<i>ipv4net</i>	Optional. Displays kernel route cache information for the specified route.
----------------	--

Default

Lists routes in the kernel route cache.

Usage Guidelines

Use this command to display information about routes stored in the kernel route cache. The route cache contains all paths currently in use by the cache. Multiple equal-cost paths are necessary before equal-cost-multi-path (ECMP) routing can be performed.

Examples

[Example 1-5](#) shows how to list routes in the kernel route cache.

Example 1-5 Listing routes in the kernel route cache

```
vyatta@vyatta:~$ show ip route cache
local 10.1.0.62 from 10.1.0.1 dev lo  src 10.1.0.62
      cache <local,src-direct>  users 1 age 42sec iif eth0
multicast 224.0.0.5 from 10.1.0.45 dev lo  src 10.1.0.62
      cache <local,mc>  users 1 used 8 age 5sec iif eth0
local 10.1.0.62 from 69.59.150.131 dev lo  src 10.1.0.62
      cache <local>  users 1 used 3 age 47sec iif eth0
10.1.0.1 from 10.1.0.62 dev eth0
```

```

cache users 1 age 42sec mtu 1500 advmss 1460 hoplimit 64
10.0.0.30 from 10.1.0.62 tos lowdelay via 10.1.0.1 dev eth0
cache users 2 age 0sec mtu 1500 advmss 1460 hoplimit 64
multicast 224.0.0.5 from 10.1.0.56 dev lo src 10.1.0.62
cache <local,mc> users 1 used 8 age 8sec iif eth0
multicast 224.0.0.5 from 10.1.0.66 dev lo src 10.1.0.62
cache <local,mc> users 1 used 8 age 0sec iif eth0
multicast 224.0.0.6 dev eth0 src 10.1.0.62
cache <mc> users 1 age 21sec mtu 1500 advmss 1460 hoplimit 64
multicast 224.0.0.5 from 10.1.0.4 dev lo src 10.1.0.62
cache <local,mc> users 1 used 9 age 1sec iif eth0
69.59.150.131 via 10.1.0.1 dev eth0 src 10.1.0.62
cache users 1 age 47sec mtu 1500 advmss 1460 hoplimit 64
multicast 224.0.0.5 dev eth0 src 10.1.0.62
cache <local,mc> users 1 used 8 age 5sec mtu 1500 advmss 1460 hoplimit
64
69.59.150.131 from 10.1.0.62 via 10.1.0.1 dev eth0
cache users 1 used 1 age 47sec mtu 1500 advmss 1460 hoplimit 64
local 10.1.0.62 from 10.0.0.30 tos lowdelay dev lo src 10.1.0.62
cache <local> users 1 used 1 age 0sec iif eth0
vyatta@vyatta:~$

```

Example 1-6 shows how to display information about route 10.1.0.62 in the kernel route cache.

Example 1-6 Displaying information about a route in the kernel route cache

```

vyatta@vyatta:~$ show ip route cache 10.1.0.62
local 10.1.0.62 from 10.1.0.1 dev lo src 10.1.0.62
cache <local,src-direct> users 1 used 3 age 9sec iif eth0
local 10.1.0.62 from 69.59.150.131 dev lo src 10.1.0.62
cache <local> users 1 used 7 age 102sec iif eth0
local 10.1.0.62 from 10.0.0.30 tos lowdelay dev lo src 10.1.0.62
cache <local> users 1 used 33 iif eth0
vyatta@vyatta:~$

```

show ip route connected

Displays directly connected routes.

Syntax

```
show ip route connected
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display routes directly connected to the local system.

Examples

[Example 1-7](#) shows how to list directly connected routes.

Example 1-7 Displaying connected routes

```
vyatta@vyatta:~$ show ip route connected
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C>* 10.1.0.0/24 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
C>* 172.16.234.0/25 is directly connected, eth1
vyatta@vyatta:~$
```

show ip route forward

Displays routes stored in the FIB.

Syntax

```
show ip route forward [ipv4net]
```

Command Mode

Operational mode.

Parameters

<i>ipv4net</i>	Optional. Displays information from the kernel forwarding table for the specified route.
----------------	--

Default

Lists routes in the FIB.

Usage Guidelines

Use this command to display the FIB.

The FIB contains multiple equal-cost paths if existed. Multiple equal-cost paths are necessary before equal-cost multi-path (ECMP) routing or WAN load balancing can be performed.

Examples

[Example 1-8](#) shows how to display routes recorded in the FIB.

Example 1-8 Displaying routes in the FIB

```
vyatta@vyatta:~$ show ip route forward
default via 10.1.0.1 dev eth0 proto zebra
10.1.0.0/24 dev eth0 proto kernel scope link src 10.1.0.62
10.192.32.0/24 via 10.1.0.45 dev eth0 proto zebra metric 20
10.192.128.0/24 via 10.1.0.66 dev eth0 proto zebra metric 11
10.192.128.1 via 10.1.0.66 dev eth0 proto zebra metric 11
10.192.129.0/24 via 10.1.0.66 dev eth0 proto zebra metric 11
10.192.130.0/24 via 10.1.0.66 dev eth0 proto zebra metric 11
```

```
10.192.131.0/24 via 10.1.0.66 dev eth0 proto zebra metric 11
172.16.0.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.1.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.2.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.3.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.4.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.5.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.6.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.7.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.8.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.9.0/24 via 10.1.0.4 dev eth0 proto zebra metric 11
172.16.234.0/25 dev eth1 proto kernel scope link src 172.16.234.23
192.94.202.0/24 via 172.16.234.27 dev eth1 proto zebra
vyatta@vyatta:~$
```

[Example 1-9](#) shows how to display information from the FIB about route 10.1.0.0/24.

Example 1-9 Displaying information about a route in the FIB

```
vyatta@vyatta:~$ show ip route forward 10.1.0.0/24
10.1.0.0/24 dev eth0 proto kernel scope link src 10.1.0.62
vyatta@vyatta:~$
```

show ip route kernel

Displays kernel routes.

Syntax

```
show ip route kernel
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display kernel routes. Kernel routes are routes that have been added through means other than by using the Vyatta CLI; for example by using the operating system route command, as in the following:

```
route add -net 10.172.24.0 netmask 255.255.255.0 gw 10.1.0.1
```

Examples

[Example 1-10](#) shows how to display kernel routes.

Example 1-10 Displaying kernel routes

```
vyatta@vyatta:~$ show ip route kernel
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

K>* 10.172.24.0/24 via 10.1.0.1, eth0
vyatta@vyatta:~$
```

show ip route static

Displays static routes.

Syntax

```
show ip route static
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

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

Examples

[Example 1-11](#) shows how to list static routes.

Example 1-11 Displaying static routes

```
vyatta@vyatta:~$ show ip route static
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

S>* 0.0.0.0/0 [1/0] via 10.1.0.1, eth0
S>* 192.94.202.0/24 [1/0] via 172.16.234.27, eth1
vyatta@vyatta:~$
```

show ip route summary

Displays routes summary.

Syntax

```
show ip route summary
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display a summary of the various routes by route source.

Examples

[Example 1-12](#) shows how to display a summary of routes.

Example 1-12 Displaying a summary of routes

```
vyatta@vyatta:~$ show ip route summary
Route Source      Routes      FIB
connected         4           4
static            2           2
ospf              1           0
ebgp              0           0
ibgp             289016     289011
-----
Totals            289023     289017
vyatta@vyatta:~$
```

show ip route supernets-only

Displays supernet routes.

Syntax

```
show ip route supernets-only
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display supernet routes.

Supernet routes are routes that have a subnet mask that is less specific than the natural classful mask.

Examples

[Example 1-13](#) shows how to list supernet routes.

Example 1-13 Displaying supernet routes

```
vyatta@vyatta:~$ show ip route supernets-only
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

S>* 0.0.0.0/0 [1/0] via 10.1.0.1, eth0
vyatta@vyatta:~$
```

show ipv6 route

Displays IPv6 routes stored in the RIB and FIB.

Syntax

```
show ipv6 route [ipv6 | ipv6net]
```

Command Mode

Operational mode.

Parameters

<i>ipv6</i>	Optional. Displays routing information for the specified IPv6 address.
<i>ipv6net</i>	Optional. Displays routing information for the specified IPv6 prefix.

Default

Lists all IPv6 routes in the RIB and FIB.

Usage Guidelines

Use this command to display active IPv6 prefixes stored in the Routing Information Base (RIB), as well as those stored in the Forwarding Information Base (FIB).

The routes shown in the FIB can also be seen using [show ip route forward command](#).

show ipv6 route <ipv6net> longer-prefixes

Displays IPv6 prefixes longer than a specified prefix.

Syntax

```
show ipv6 route ipv6net longer-prefixes
```

Command Mode

Operational mode.

Parameters

<i>ipv6net</i>	Mandatory. Displays all prefixes longer than the specified IPv6 prefix.
----------------	---

Default

None.

Usage Guidelines

Use this command to display all prefixes in the Routing Information Base (RIB) that are longer than a given IPv6 address or prefix.

show ipv6 route bgp

Displays IPv6 BGP routes.

Syntax

```
show ipv6 route bgp
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display IPv6 BGP routes.

show ipv6 route cache

Displays the kernel IPv6 route cache.

Syntax

```
show ipv6 route cache [ipv6net]
```

Command Mode

Operational mode.

Parameters

<i>ipv6net</i>	Optional. Displays kernel IPv6 route cache information for the specified route.
----------------	---

Default

Lists routes in the kernel IPv6 route cache.

Usage Guidelines

Use this command to display information about routes stored in the kernel IPv6 route cache. The route cache contains all paths currently in use by the cache. Multiple equal-cost paths are necessary before equal-cost-multi-path (ECMP) routing can be performed.

show ipv6 route connected

Displays IPv6 connected routes.

Syntax

```
show ipv6 route connected
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display IPv6 routes directly connected to the local system.

show ipv6 route forward

Displays IPv6 routes stored in the FIB.

Syntax

```
show ipv6 route forward [ipv6net]
```

Command Mode

Operational mode.

Parameters

<i>ipv6net</i>	Optional. Displays information from the kernel forwarding table for the specified IPv6 route.
----------------	---

Default

Lists IPv6 routes in the FIB.

Usage Guidelines

Use this command to display the FIB.

The FIB contains multiple equal-cost paths if existed. Multiple equal-cost paths are necessary before equal-cost multi-path (ECMP) routing or WAN load balancing can be performed.

show ipv6 route kernel

Displays IPv6 kernel routes.

Syntax

```
show ipv6 route kernel
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display IPv6 kernel routes. Kernel routes are routes that have been added through means other than by using the Vyatta CLI.

show ipv6 route ripng

Displays IPv6 RIPng routes.

Syntax

```
show ipv6 route ripng
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display IPv6 RIPng routes.

show ipv6 route static

Displays IPv6 static routes.

Syntax

```
show ipv6 route static
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display IPv6 static routes in the Routing Information Base (RIB).

show ipv6 route summary

Displays IPv6 routes summary.

Syntax

```
show ipv6 route summary
```

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display a summary of the various IPv6 routes by route source.

Examples

[Example 1-14](#) shows a summary of IPv6 routes.

Example 1-14 “show ipv6 route summary”: Displaying a summary of IPv6 routes

```
vyatta@vyatta:~$ show ipv6 route summary
Route Source      Routes      FIB
connected         4           4
static            2           2
-----
Totals            6           6
vyatta@vyatta:~$
```

show table

Displays the system's routing table.

Syntax

show table

Command Mode

Operational mode.

Parameters

None.

Default

None.

Usage Guidelines

Use this command to display the system's routing table.

Examples

[Example 1-15](#) shows how to display the routing table.

Example 1-15 Displaying the routing table

```
vyatta@vyatta:~$ show table
table 0
vyatta@vyatta:~$
```

Chapter 2: Static Routes

This chapter explains how to set static routes using the Vyatta system.

This chapter presents the following topics:

- [Static Route Configuration](#)
- [Static IPv6 Route Configuration](#)
- [Static Route Commands](#)

Static Route Configuration

This section presents the following topics:

- [Static Routes Overview](#)
- [Configuring Static Routes](#)
- [Floating Static Routes](#)
- [Showing Static Routes in the Routing Table](#)

Static Routes Overview

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

In general, static routes should only be used for very simple network topologies, or to override the behavior of a dynamic routing protocol for a small number of routes.

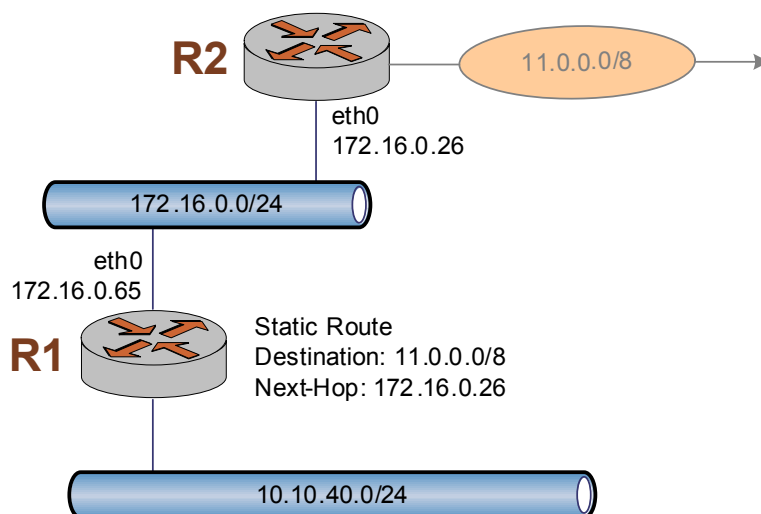
The collection of all routes the router has learned from its configuration or from its dynamic routing protocols is stored in its Routing Information Base (RIB).

Unicast routes are directly used to determine the forwarding table used for unicast packet forwarding.

Configuring Static Routes

In this example, a sample configurations are presented for basic static routes. When you are finished, the system will be configured as shown in [Figure 2-1](#). In this example, a static route is created that says, in effect, “any packets destined for the 11.0.0.0/8 network should be forwarded to 172.16.0.26”.

Figure 2-1 Static routes



This section includes the following examples:

- Example 2-1 Creating a static route

[Example 2-1](#) creates a static route to network 11.0.0.0/8 directed towards 172.16.0.26.

To create a static route, perform the following steps in configuration mode:

Example 2-1 Creating a static route

Step	Command
Create a static route to R2.	<code>vyatta@R1# set protocols static route 11.0.0.0/8 next-hop 172.16.0.26</code>
Commit the configuration.	<code>vyatta@R1# commit</code>

Floating Static Routes

Usually, static routes have a relatively short administrative distance—typically 1, and usually shorter than the administrative distances for dynamic (learned) routes. A “floating” static route is a static route with an administrative distance greater than that for dynamic routes.

You can configure a static route to be a floating route by setting the administrative distance higher than the distance applied to the routes in your dynamic routing protocol. This renders the static route less desirable than a dynamic route. At the same time, if the dynamic route is lost, the static route is available to take over traffic, which can be forwarded through the static route as an alternate path.

Showing Static Routes in the Routing Table

To display route information, use the **show ip route** command. To show just static routes, use the **show ip route static** filter, as shown in [Example 2-2](#).

Example 2-2 Showing static routes in the routing table

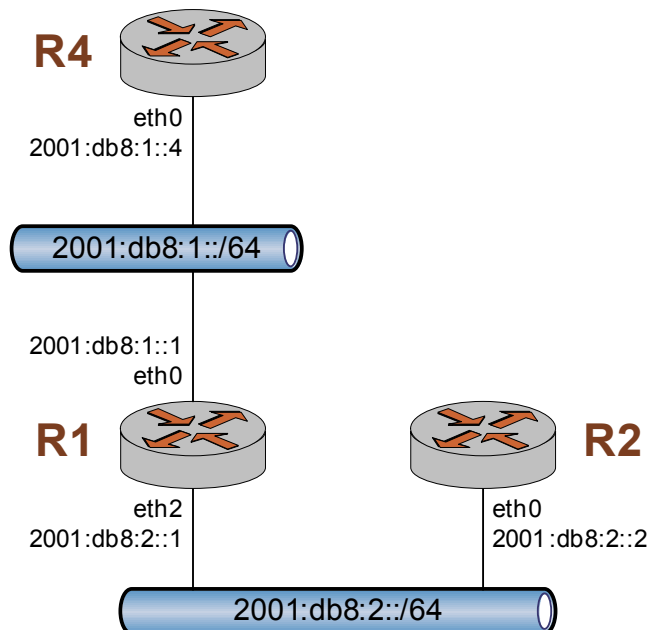
```
vyatta@R1:~$ show ip route static
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

S>* 11.0.0.0/8 [1/0] via 172.16.0.26, eth0
vyatta@R1:~$
```

Static IPv6 Route Configuration

[Figure 2-2](#) shows an IPv6 network with three nodes. In this example we will show configuration of the nodes using static routes to enable R2 and R4 to communicate via R1.

Figure 2-2 Static IPv6 routing example



Verify That IPv6 Forwarding is Enabled

In order for R1 to be able to pass data between interfaces eth0 and eth2 (i.e., between R4 and R2) it must be configured to enable forwarding. To determine if forwarding is enabled, perform the following step in operational mode.

Example 2-3 Determine if forwarding is enabled on R1

Step	Command
Display the state of IPv6 forwarding on R1.	vyatta@R1:~\$ show ipv6 forwarding ipv6 forwarding is off

If forwarding is not enabled, as is the case in [Example 2-3](#), the system must be configured to enable forwarding. To enable forwarding, perform the following steps in configuration mode.

Example 2-4 Enable forwarding on R1

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

Example 2-4 Enable forwarding on R1

Change to operational mode	vyatta@R1# exit exit vyatta@R1:~\$
----------------------------	---

Display the state of IPv6 forwarding on R1.	vyatta@R1:~\$ show ipv6 forwarding ipv6 forwarding is on
---	--

Add the Default IPv6 Route

On R4, all traffic that is not routed elsewhere will be sent to R1. To configure the default route, perform the following steps in configuration mode.

Example 2-5 Add the default route on R4

Step	Command
Add the default route on R4.	vyatta@R4# set protocols static route6 ::/0 next-hop 2001:db8:1::1
Commit the change.	vyatta@R4# commit
Change to operational mode.	vyatta@R4# exit exit vyatta@R4:~\$
Verify the default route in the routing table.	vyatta@R4:~\$ show ipv6 route Codes: K - kernel route, C - connected, S - static, R - RIPng, O - OSPFv3, I - ISIS, B - BGP, * - FIB route. S>* ::/0 [1/0] via 2001:db8:1::1, eth0 C>* ::1/128 is directly connected, lo C>* 2001:db8:1::/64 is directly connected, eth0 C * fe80::/64 is directly connected, eth1 C>* fe80::/64 is directly connected, eth0 K>* ff00::/8 is directly connected, eth0

Add a Static IPv6 Route

As an alternative to the default route we created on R4, we'll create a static route on R2. To configure a static route to the 2001:db8:1::/64 network, perform the following steps in configuration mode.

Example 2-6 Add a static route on R2

Step	Command
Add a static route on R2.	vyatta@R1# set protocols static route6 2001:db8:1::/64 next-hop 2001:db8:2::1
Commit the change.	vyatta@R1# commit
Change to operational mode.	vyatta@R1# exit exit vyatta@R2:~\$
Verify the static route in the routing table.	vyatta@R2:~\$ show ipv6 route Codes: K - kernel route, C - connected, S - static, R - RIPng, O - OSPFv3, I - ISIS, B - BGP, * - FIB route. C>* ::1/128 is directly connected, lo S>* 2001:db8:1::/64 [1/0] via 2001:db8:2::1, eth0 C>* 2001:db8:2::/64 is directly connected, eth0 C * fe80::/64 is directly connected, eth1 C>* fe80::/64 is directly connected, eth0 K>* ff00::/8 is directly connected, eth0

Confirm Connectivity

To confirm that R2 and R4 can communicate, use the **ping6** command. To confirm connectivity between R2 and R4, perform the following step in operational mode.

Example 2-7 Confirm connectivity between R2 and R4

Step	Command
Ping R4 from R2.	vyatta@R2:~\$ ping6 2001:db8:1::4 PING 2001:db8:1::4(2001:db8:1::4) 56 data bytes 64 bytes from 2001:db8:1::4: icmp_seq=1 ttl=63 time=5.65 ms 64 bytes from 2001:db8:1::4: icmp_seq=2 ttl=63 time=0.382 ms ^C --- 2001:db8:1::4 ping statistics --- 2 packets transmitted, 2 received, 0% packet loss, time 1011ms rtt min/avg/max/mdev = 0.382/3.016/5.650/2.634 ms

As an alternative, use **tracertoe6** to verify that the goes from R2 to R1 to R4. To confirm connectivity between R2 and R4 through R1 using **tracertoe6**, perform the following step in operational mode.

Example 2-8 Confirm connectivity between R2 and R4 via R1

Step	Command
Trace the route from R2 to R4.	<pre>vyatta@R2:~\$ tracertoe6 2001:db8:1::4 tracertoe to 2001:db8:1::4 (2001:db8:1::4), 30 hops max, 40 byte packets 1 (2001:db8:2::1) 4.448 ms 4.148 ms 4.092 ms 2 (2001:db8:1::4) 4.297 ms 4.306 ms 4.308 ms</pre>

Static Route Commands

This chapter contains the following commands.

Configuration Commands

<code>protocols static interface-route <subnet> next-hop-interface <interface></code>	Allows you to configure the next-hop interface for an interface-based static route.
<code>protocols static interface-route6 <subnet> next-hop-interface <ethx></code>	Allows you to configure the next hop interface for an interface-based IPv6 static route.
<code>protocols static route <subnet> blackhole</code>	Allows you to configure a “black-hole” static route.
<code>protocols static route <subnet> next-hop <address></code>	Allows you to configure the next hop for a static route.
<code>protocols static route6 <subnet> blackhole</code>	Allows you to configure a blackhole IPv6 static route.
<code>protocols static route6 <subnet> next-hop <address></code>	Allows you to configure the next hop for an IPv6 static route.

Operational Commands

<code>show ip route static</code>	Displays static routes. <i>See page 18.</i>
<code>show ipv6 route static</code>	Displays IPv6 static routes. <i>See page 18.</i>

protocols static interface-route <subnet> next-hop-interface <interface>

Allows you to configure the next-hop interface for an interface-based static route.

Syntax

```
set protocols static interface-route subnet next-hop-interface interface [disable |  
distance distance]
```

```
delete protocols static interface-route subnet next-hop-interface interface [disable |  
distance]
```

```
show protocols static interface-route subnet next-hop-interface interface [disable |  
distance]
```

Command Mode

Configuration mode.

Configuration Statement

```
protocols {  
  static {  
    interface-route subnet {  
      next-hop-interface interface {  
        disable  
        distance distance  
      }  
    }  
  }  
}
```

Parameters

<i>subnet</i>	Mandatory. Multi-node. Defines an interface-based static route. The format is a destination subnet of the form <i>address/prefix</i> . You can define multiple interface-based routes by creating multiple interface-route configuration nodes.
<i>interface</i>	Mandatory. The next-hop interface.
disable	Disables the interface-based static route.

<i>distance</i>	Optional. Sets the next-hop distance for this route. Routes with a smaller distance are selected before those with a larger distance. The range is 1 to 255. The default is 1.
-----------------	--

Default

None.

Usage Guidelines

Use this command to configure interface-based static routes on the router.

Use the **set** form of this command to specify the next-hop interface for the route.

Use the **delete** form of this command to remove the next-hop interface.

Use the **show** form of this command to view the next-hop interface for the route.

protocols static interface-route6 <subnet> next-hop-interface <ethx>

Allows you to configure the next hop interface for an interface-based IPv6 static route.

Syntax

```
set protocols static interface-route6 subnet next-hop-interface ethx [disable | distance distance]
```

```
delete protocols static interface-route6 subnet next-hop-interface ethx [disable | distance]
```

```
show protocols static interface-route6 subnet next-hop-interface ethx [disable | distance]
```

Command Mode

Configuration mode.

Configuration Statement

```
protocols {  
  static {  
    interface-route6 subnet {  
      next-hop-interface ethx {  
        disable  
        distance distance  
      }  
    }  
  }  
}
```

Parameters

<i>subnet</i>	Mandatory. Multi-node. Defines an interface-based static route. The format is a destination subnet of the form IPv6- <i>address/prefix</i> . You can define multiple interface-based routes by creating multiple interface-route6 configuration nodes.
<i>ethx</i>	Mandatory. The next hop Ethernet interface.

disable	Disables the interface-based IPv6 static route.
<i>distance</i>	Optional. Defines the next-hop distance for this route. Routes with a smaller distance are selected before those with a larger distance.

Default

None.

Usage Guidelines

- Use this command to configure interface-based IPv6 static routes on the system.
- Use the **set** form of this command to specify the next hop interface for the route.
- Use the **delete** form of this command to remove the next hop interface.
- Use the **show** form of this command to view the next hop interface for the route.

protocols static route <subnet> blackhole

Allows you to configure a “black-hole” static route.

Syntax

```
set protocols static route subnet blackhole [distance distance]  
delete protocols static route subnet blackhole [distance]  
show protocols static route subnet blackhole [distance]
```

Command Mode

Configuration mode.

Configuration Statement

```
protocols {  
  static {  
    route subnet {  
      blackhole {  
        distance distance  
      }  
    }  
  }  
}
```

Parameters

<i>subnet</i>	Mandatory. Multi-node. Defines a static route. The format is a destination subnet of the form <i>address/prefix</i> . You can define multiple static routes by creating multiple route configuration nodes.
<i>distance</i>	Optional. Defines the black-hole distance for this route. Routes with a smaller distance are selected before those with a larger distance. The range is 1 to 255. The default is 1.

Default

None.

Usage Guidelines

Use this command to configure a “black-hole” static route on the router. A black-hole route is a route for which the system silently discards packets that are matched.

Use the **set** form of this command to set a black-hole route.

Use the **delete** form of this command to remove a black-hole route.

Use the **show** form of this command to view black-hole route configuration.

protocols static route <subnet> next-hop <address>

Allows you to configure the next hop for a static route.

Syntax

```
set protocols static route subnet next-hop address [disable | distance distance]  
delete protocols static route subnet next-hop address [disable | distance]  
show protocols static route subnet next-hop address [disable | distance]
```

Command Mode

Configuration mode.

Configuration Statement

```
protocols {  
  static {  
    route subnet {  
      next-hop address {  
        disable  
        distance distance  
      }  
    }  
  }  
}
```

Parameters

<i>subnet</i>	Mandatory. Multi-node. Defines a static route. The format is a destination subnet of the form <i>address/prefix</i> . You can define multiple static routes by creating multiple route configuration nodes.
<i>address</i>	Mandatory. The address of the next-hop router.
disable	Disables the static route.

<i>distance</i>	Optional. Defines the next-hop distance for this route. Routes with a smaller distance are selected before those with a larger distance. The range is 1 to 255. The default is 1.
-----------------	---

Default

None.

Usage Guidelines

Use this command to configure static routes on the router.

Use the **set** form of this command to specify the next hop for the route.

Use the **delete** form of this command to remove the static route next hop.

Use the **show** form of this command to view static route next-hop configuration.

protocols static route6 <subnet> blackhole

Allows you to configure a blackhole IPv6 static route.

Syntax

```
set protocols static route6 subnet blackhole [distance distance]  
delete protocols static route6 subnet blackhole [distance]  
show protocols static route6 subnet blackhole [distance]
```

Command Mode

Configuration mode.

Configuration Statement

```
protocols {  
  static {  
    route6 subnet {  
      blackhole {  
        distance distance  
      }  
    }  
  }  
}
```

Parameters

<i>subnet</i>	Mandatory. Multi-node. Defines an IPv6 static route. The format is a destination subnet of the form IPv6- <i>address/prefix</i> . You can define multiple static routes by creating multiple route configuration nodes.
<i>distance</i>	Optional. Defines the blackhole distance for this route. Routes with a smaller distance will be selected before those with a larger distance.

Default

None.

Usage Guidelines

Use this command to configure a blackhole IPv6 static route. A blackhole route silently discards packets that are matched.

Use the **set** form of this command to specify a blackhole IPv6 static route.

Use the **delete** form of this command to remove a blackhole IPv6 static route.

Use the **show** form of this command to view blackhole IPv6 static route configuration.

protocols static route6 <subnet> next-hop <address>

Allows you to configure the next hop for an IPv6 static route.

Syntax

```
set protocols static route6 subnet next-hop address [disable | distance distance | interface interface]
```

```
delete protocols static route6 subnet next-hop address [disable | distance | interface]
```

```
show protocols static route6 subnet next-hop address [disable | distance | interface]
```

Command Mode

Configuration mode.

Configuration Statement

```
protocols {
  static {
    route6 subnet {
      next-hop address {
        disable
        distance distance
        interface interface
      }
    }
  }
}
```

Parameters

<i>subnet</i>	Mandatory. Multi-node. Defines an IPv6 static route. The format is a destination subnet of the form IPv6- <i>address/prefix</i> . You can define multiple static routes by creating multiple <code>route6</code> configuration nodes.
<i>address</i>	Mandatory. The IPv6 address of the next hop router.
<i>disable</i>	Disable the IPv6 static route.

<i>distance</i>	Optional. Defines the next-hop distance for this route. Routes with a smaller distance will be selected before those with a larger distance.
<i>interface</i>	Optional. The outgoing interface used to reach the next-hop address. This is necessary when the next-hop address is a link-local address (that is, it has a fe80::/64 prefix).

Default

None.

Usage Guidelines

Use this command to configure IPv6 static routes on the system.

Use the **set** form of this command to specify the next hop for the route.

Use the **delete** form of this command to remove the static route next hop.

Use the **show** form of this command to view static route next hop configuration.

Glossary of Acronyms

ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
API	Application Programming Interface
AS	autonomous system
ARP	Address Resolution Protocol
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
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
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

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
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
TCP	Transmission Control Protocol
TKIP	Temporal Key Integrity 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
WAP	wireless access point

WPA Wired Protected Access
