53-1003723-03
14 September 2015

Brocade 5600 vRouter Policy-based Routing

Reference Guide

Supporting Brocade 5600 vRouter 3.5R6

BROCADE

Contents

Preface

Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Brocade technical documentation.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used in the flow of the text to highlight specific words or phrases.

Format	Description
bold text	Identifies command names
	Identifies keywords and operands
	Identifies the names of user-manipulated GUI elements
	Identifies text to enter at the GUI
italic text	Identifies emphasis
	Identifies variables
	Identifies document titles
`Courier font`	Identifies CLI output
	Identifies command syntax examples

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.
italic text	Identifies a variable.
value	In Fibre Channel products, a fixed value provided as input to a command option is printed in plain text, for example, **--show** WWN.

Convention	Description
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ **x** | **y** | **z** }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options. In Fibre Channel products, square brackets may be used instead for this purpose.
x | **y**	A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, *member*[*member*...].
\	Indicates a “soft” line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

NOTE
A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Brocade resources

Visit the Brocade website to locate related documentation for your product and additional Brocade resources.

You can download additional publications supporting your product at www.brocade.com. Select the Brocade Products tab to locate your product, then click the Brocade product name or image to open the individual product page. The user manuals are available in the resources module at the bottom of the page under the Documentation category.

To get up-to-the-minute information on Brocade products and resources, go to MyBrocade. You can register at no cost to obtain a user ID and password.

Release notes are available on MyBrocade under Product Downloads.

White papers, online demonstrations, and data sheets are available through the Brocade website.

Contacting Brocade Technical Support

As a Brocade customer, you can contact Brocade Technical Support 24x7 online, by telephone, or by e-mail. Brocade OEM customers contact their OEM/Solutions provider.

Brocade customers

For product support information and the latest information on contacting the Technical Assistance Center, go to http://www.brocade.com/services-support/index.html.

If you have purchased Brocade product support directly from Brocade, use one of the following methods to contact the Brocade Technical Assistance Center 24x7.

Online	Telephone	E-mail
Preferred method of contact for non-urgent issues: • My Cases through MyBrocade • Software downloads and licensing tools • Knowledge Base	Required for Sev 1-Critical and Sev 2-High issues: • Continental US: 1-800-752-8061 • Europe, Middle East, Africa, and Asia Pacific: +800-AT FIBREE (+800 28 34 27 33) • For areas unable to access toll free number: +1-408-333-6061 • Toll-free numbers are available in many countries.	support@brocade.com Please include: • Problem summary • Serial number • Installation details • Environment description

Brocade OEM customers

If you have purchased Brocade product support from a Brocade OEM/Solution Provider, contact your OEM/Solution Provider for all of your product support needs.

- OEM/Solution Providers are trained and certified by Brocade to support Brocade® products.
- Brocade provides backline support for issues that cannot be resolved by the OEM/Solution Provider.

- Brocade Supplemental Support augments your existing OEM support contract, providing direct access to Brocade expertise. For more information, contact Brocade or your OEM.
- For questions regarding service levels and response times, contact your OEM/Solution Provider.

Document feedback

To send feedback and report errors in the documentation you can use the feedback form posted with the document or you can e-mail the documentation team.

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. You can provide feedback in two ways:

- Through the online feedback form in the HTML documents posted on www.brocade.com.
- By sending your feedback to documentation@brocade.com.

Provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.

About This Guide

This guide describes how to define and configure routing policies on the Brocade 5600 vRouter (referred to as a virtual router, vRouter, or router in the guide).

Policy-based Routing

Introduction

Policy-based routing (PBR) enables you to use IP traffic rules to classify traffic based on its attributes and apply processing differentially according to the classification, and to selectively route IP packets, for example, to an alternate next hop. PBR on the Brocade vRouter is supported just on incoming Layer 3 and Layer 4 traffic.

All packets received on an interface are considered for policy-based routing provided that interface is assigned a routing policy.

When no routing policies are applied, routing decisions are made by using the default (main) routing table (Table 254) of the system.

PBR policies can be applied to dataplane interfaces for inbound traffic, but not to loopback, tunnel, bridge, OpenVPN, VTI, and IP unnumbered interfaces.

On the Brocade vRouter, you cannot apply policy based routing to locally generated packets.

Defining a routing policy

The routing policy classifies traffic and specifies the handling that should take place for different classes. This classification and handling are accomplished by using a set of policy rules.

Rules are configured with match criteria that include an extensive set of attributes—including protocol, source and destination addresses and ports, fragmentation, ICMP or ICMPv6 type, and TCP flags. You can also preconfigure groups of addresses, ports, and networks and refer to these groups in policy rules.

The routing policy must be applied to an interface for the policy to be effective.

To implement policy-based routing, perform the following steps:

1. Define the policy rules.
2. Attach the policy to an ingress interface.
3. Create a route in a PBR table other than Table 254.

NOTE
Table 254 is also known as the main table or default table.

Routing policy rules

Packets that match the PBR rule criteria do one of the following:

- They are dropped (if the **drop** action is set).
- They are routed by using a specific PBR routing table.

Packets that match the rule parameters are considered for policy-based routing. As many as 9,999 rules in a policy are supported. If no match criteria are specified, all packets are routed according to the default Table 254.

The packets that do not match any policy rule are routed according to the routes in the main table.

Routing policy rules are executed in numeric sequence, from lowest to highest. You can renumber rules by using the **rename** command in configuration mode (refer to *Brocade 5600 vRouter Basic System Reference Guide*).

NOTE
To avoid having to renumber routing policy rules, a good practice is to number rules in increments of 10. This increment allows room for the insertion of new rules within the policy.

PBR behavior

Routes that remain persistent in the controller. If the dataplane goes down, and up, the routes are automatically re-established without the need for reconfiguration.

PBR does not reassemble fragmented packets. PBR treats fragments as individual packets.

PBR rules can be changed dynamically and does not require the rebinding of the PBR policy to an interface.

Configuration for VLAN-based classification, virtual interface (vif), MAC address, packet mangling, and so on, are not supported.

The controller automatically continuously resyncs the route information to the dataplane.

Multiple PBR policies can be applied to an interface. For best results, we recommend that these policies are unique.

Packet forwarding path

When enabled, PBR processes incoming packets after packet validation and firewall action. Packets received by the dataplane ingress interfaces for transmission to the egress interface follow the forwarding path listed below. There is only a single Virtual Routing and Forwarding (VRF) instance for PBR.

1. Packet validation and reassembly
2. Firewall
3. PBR classification, route table ID determination
4. NAT
5. Firewall
6. QoS
7. Transmit out of an egress interface

Configuration Examples

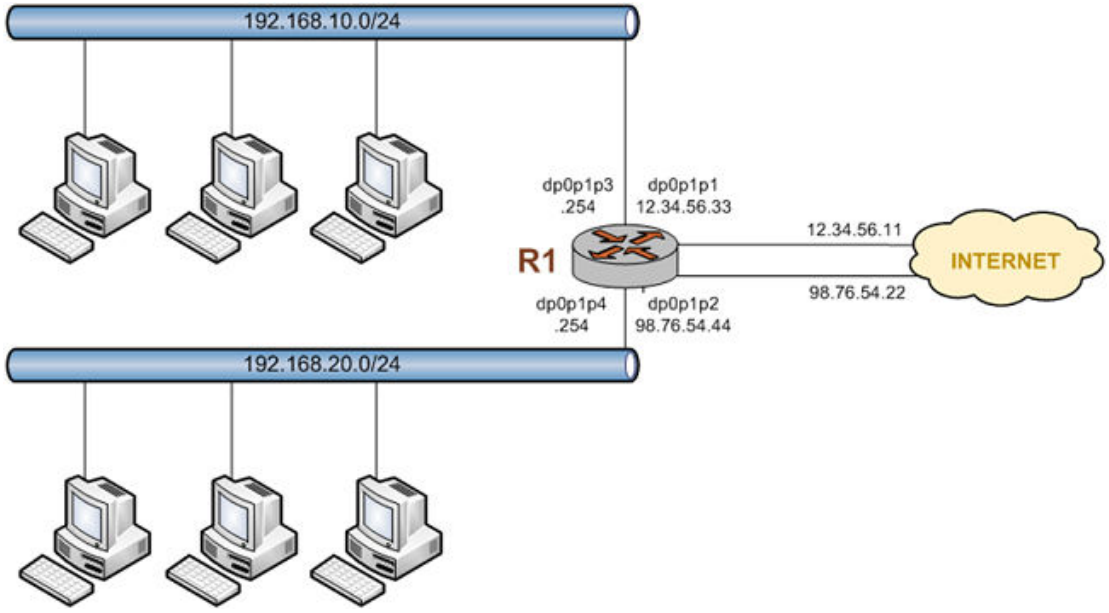
PBR routing example

The following figure shows a simple site that uses PBR on the Brocade vRouter (R1) to route traffic from two different internal subnets to two Internet links.

The following conditions apply to this scenario:

- All Internet-bound traffic from subnet 192.168.10.0/24 is routed out interface dp0p1p1.
- All Internet-bound traffic from subnet 192.168.20.0/24 is routed out interface dp0p1p2.

FIGURE 1 Routing using PBR



To configure the scenario, perform the following steps in configuration mode.

TABLE 1 Routing using PBR

Step	Command
Create Rule 10 and specify the destination address to match. In this case, any destination address is a match.	`vyatta@R1# set policy route pbr myroute rule 10 action accept` `vyatta@R1# set policy route pbr myroute rule 10 destination address 0.0.0.0/0`

TABLE 1 Routing using PBR (Continued)

Step	Command
Specify the source address to match. In this case, any address on subnet 192.168.10.0/24 is a match.	`vyatta@R1# set policy route pbr myroute rule 10 source address 192.168.10.0/24`
Specify that all matching packets use alternate routing table 1.	`vyatta@R1# set policy route pbr myroute rule 10 table 1`

Create rule 20 and specify the destination address to match. In this case, any destination address is a match.	`vyatta@R1# set policy route pbr myroute rule 20 address-family ipv4`
	`vyatta@R1# set policy route pbr myroute rule 20 action accept`
	`vyatta@R1# set policy route pbr myroute rule 20 destination address 0.0.0.0/0`

Specify the source address to match. In this case, any address on subnet 192.168.20.0/24 is a match.	`vyatta@R1# set policy route pbr myroute rule 20 source address 192.168.20.0/24`
Specify that all matching packets use alternate routing table 2.	`vyatta@R1# set policy route pbr myroute rule 20 table 2`
Commit the changes.	`vyatta@R1# commit`

Show the policy-based routing configuration.	`vyatta@R1# show policy route`
	`route {`
	`  pbr myroute {`
	`  rule 10 {`
	`  action accept`
	`  destination {`
	`  address 0.0.0.0/0`
	`  }`
	`  source {`
	`  address 192.168.10.0/24`
	`  }`
	`  table 1`
	`  }`
	`  rule 20 {`
	`  action accept`
	`  address-family ipv4`
	`  destination {`
	`  address 0.0.0.0/0`
	`  }`
	`  source {`
	`  address 192.168.20.0/24`
	`  }`
	`  table 2`
	`  }`
	`  }`
	`}`

Create the alternative routing table 1.	`vyatta@R1# set protocols static table 1 route 12.34.56.0/24 next-hop 12.34.56.11`
Create the alternative routing table 2.	`vyatta@R1# set protocols static table 2 route 12.34.56.0/24 next-hop 98.76.54.22`
Commit the change.	`vyatta@R1# commit`

TABLE 1 Routing using PBR (Continued)

Step	Command

Show the alternate routing table configuration.	`vyatta@R1# show protocols static`
	
	`static {`
	`  table 1 {`
	`  route 1.2.3.0/24 {`
	`  next-hop 12.34.56.11 {`
	`  }`
	`  }`
	`  }`
	`  table 2 {`
	`  route 2.3.4.0/24 {`
	`  next-hop 98.76.54.22 {`
	`  }`
	`  }`
	`  }`
	`}`

Apply the 12.34.56.33/24 address to dp0p1p1.	`vyatta@R1# set interfaces dataplane dp0p1p1 address 12.34.56.33/24`

Show the dataplane interface configuration.	`vyatta@R1# show interfaces dataplane`
	
	`dataplane dp0p1p1 {`
	`  address 12.34.56.33/24`
	`}`
	`dataplane dp0p1p2 {`
	`  address 98.76.54.44/24`
	`}`
	`dataplane dp0p1p3 {`
	`  address 192.168.10.254/24`
	`  policy {`
	`  route myroute`
	`  }`
	`}`
	`dataplane dp0p1p4 {`
	`  address 192.168.20.254/24`
	`  policy {`
	`  route myroute`
	`  }`
	`}`

Binding interfaces to PBR tables

To configure an interface-based static route in a policy route table, perform the following steps:

TABLE 2 Applying a policy route to an interface

Step	Command
Configure the interface route for the interface.	`vyatta@R1# set protocols static table 10 interface-route 192.168.20.254/24 nexthop-interface dp0p256p1 distance 25`

View the configuration.	`vyatta@vyatta:~$ show protocols`
	
	`protocols {`
	`  static {`
	`  table 10 {`
	`  interface-route 192.168.20.254/24 {`
	`  nexthop-interface dp0p256p1 {`
	`  distance 1`
	`  }`
	`  }`
	`  }`
	`  }`
	`}`

TABLE 2 Applying a policy route to an interface (Continued)

Step	Command
Apply the policy route to the interface.	`vyatta@R1# set interfaces dataplane dp0p16p1 policy route myroute`
Commit the change.	`vyatta@R1# commit`

Policy-based Routing Commands

clear policy

Clears the statistics for route policies.

Syntax **clear policy**

Modes Operational mode

Usage Guidelines Use this command to clear the statistics for policy-based routing.

interfaces <interface> policy route <name>

Applies an IP routing policy to inbound traffic on an interface.

Syntax

set interfaces *interface* **policy route** *name*

delete interfaces *interface* **policy route** [*name*]

show interfaces *interface* **policy route**

Parameters

interface

The type of interface. For detailed keywords and arguments that can be specified as interface types, refer to Supported Interface Types on page 47

NOTE
Policy-based routing policies can be applied to dataplane interfaces, but not on loopback, tunnel, bridge, OpenVPN, or VTI interfaces.

name

An IP routing policy.

Modes

Configuration mode

Configuration Statement

```
interfaces interface {
    policy {
        route name
    }
}
```

Usage Guidelines

A routing policy has no effect on traffic traversing the system until it has been applied to an interface.

To use the policy-based routing feature, you must define a routing policy by using the **set policy route pbr name** *name* **rule** *number* command, then apply the routing policy to interfaces by using a statement like this one. Once applied, the rule set acts as a packet filter.

Use the **set** form of this command to apply an IP routing policy to an interface.

Use the **delete** form of this command to remove an IP routing policy from an interface.

Use the **show** form of this command to display an IP routing policy configuration for an interface.

policy route pbr name <name> rule <number>

Defines an IP routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number*

delete policy route pbr name *name* **rule** [*number*]

show policy route pbr name *name* **rule**

Parameters

name

The name of an IP routing policy.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number
            }
        }
    }
}
```

Usage Guidelines

A policy identifies traffic that matches parameters and specifies which routing table to use. The table defines the route for a packet to take. A routing policy is a named collection of as many as 9,999 packet-classification rules. When applied to an interface, the policy rule classifies incoming traffic.

Use the **set** form of this command to create a rule.

Use the **delete** form of this command to delete an existing IP routing policy.

Use the **show** form of this command to display a rule.

policy route pbr name <name> rule <number> action <action>

Defines the action for an IP routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **action** { **drop** | **accept** }

delete policy route pbr name *name* **rule** *number* **action** [**drop** | **accept**]

show policy route pbr name *name* **rule** *number* **action**

Parameters

name

The name of an IP routing policy.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

accept

Accepts the packet.

drop

Drops the packet silently.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr
            name name {
                rule number {
                    action
                        accept
                        drop
                }
            }
        }
    }
}
```

Usage Guidelines

If a rule does not explicitly drop a packet in the action, the PBR action is to accept the packet, which causes it to be sent to the specified alternate routing table for lookup and forwarding.

An applied policy can only be deleted after first removing it from an assigned interface.

Use the **set** form of this command to set the action for a rule.

Use the **delete** form of this command to remove the action for a rule.

Use the **show** form of this command to display a rule within an IP routing policy.

policy route pbr name <name> rule <number> af <protocol>

Defines the address family and routing protocol for an IP routing policy rule.

Syntax

set policy route pbr name *name* **rule** *rule-number* **af** [**ipv4** | **ipv6**]

delete policy route pbr name *name* **rule** *rule-number* **af** [**ipv4** | **ipv6**]

show policy route pbr name *name* **rule** *rule-number* **af**

Parameters

name

The name of an IP routing policy. The policy name must be unique and must not be used with other PBR policy commands.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    address-family
                        ipv4
                        ipv6
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to define the address family and routing protocol for an IP routing policy rule.

Use the **delete** form of this command to remove the address family and routing protocol for an IP routing policy rule.

Use the **show** form of this command to view the address family and routing protocol for an IP routing policy rule.

policy route pbr name <name> rule <number> description <description>

Provides a brief description for an IP routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **description** *description*

delete policy route pbr name *name* **rule** *number* **description**

show policy route pbr name *name* **rule** *number* **description**

Parameters

name

The name of an IP routing policy.

number

The numeric identifier of the rule. The numbers range from 1 through 9999.

description

A brief description for the rule. If the description contains spaces, it must be enclosed in double quotation marks (”).

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    description description
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to provide a description for an IP routing policy rule.

Use the **delete** form of this command to remove a description for an IP routing policy rule.

Use the **show** form of this command to display a description for an IP routing policy rule.

policy route pbr name <name> rule <number> destination <destination>

Defines the destination address for an IP routing policy rule.

Syntax **set policy route pbr name** *name* **rule** *number* **destination** { **address** *address* | **port** [*port-name* | *1-65535* | *start-end* | *port-group-name*] }

delete policy route pbr name *name* **rule** *number* **destination** [**address** *address* | **port** [*name* | *1-65535* | *start-end* | *port-group-name*]]

show policy route pbr name *name* **rule** *number* **destination** [**address** | **port**]

Parameters *name*

The name of an IP routing policy.

number

The numeric identifier of a policy rule. Rule numbers determine the order in which rules are processed. Each rule must have a unique rule number. The number ranges from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

address

A destination address to match. Address formats are as follows:

ip-address: An IP address.

ip-address/prefix: An IPv4 network address, where 0.0.0.0/0 matches any network.

ip-address-ip-address: A range of contiguous IPv4 addresses; for example, 192.168.1.1-192.168.1.150.

! *ip-address*: All IPv4 addresses except the one specified.

! *ip-address/prefix*: All IPv4 network addresses except the one specified.

! *ip-address-ip-address*: All IP addresses except those in the specified range.

[*port-name* | *1-65535* | *start-end* | *port-group-name*]

Applicable only when the protocol is TCP or UDP. A destination port to match. The format of the port is any of the following:

port-name: The name of an IP service; for example, http. You can specify any service name in the /etc/services file.

1-65535: A port number. The numbers range from 1 through 65535.

start-end: A specified range of ports; for example, 1001-1005.

port-group-name: A port group. A packet is considered a match if it matches any port name or number specified in the group. Only one port group may be specified. The port group must already be defined.

This criterion specifies a group of addresses, ports, or networks for packet destination address.

A packet is considered a match for an address, a network, or a port group if it matches any host IP address, network address, or port name or number, respectively, in the group. However, if more than one group is specified, the

packet must be a match for both groups to be considered a match. For example, if both an address group and a port group are specified, the destination of the packet must match at least one item in the address group and at least one item in the port group.

An address group may be specified with a port group, and a network group may be specified with a port group. You cannot specify both an address and a network group.

You can use a combination of these formats in a list separated by commas. You can also negate the entire list by prefixing it with an exclamation mark (!); for example, !22,telnet,http,123,1001-1005.

If both an address and a port are specified, the packet is considered a match only if both the address and the port match.

Modes Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    destination {
                        address address
                        port port-name
                        port 1-65535
                        port start-end
                        port port-group-name
                    }
                }
            }
        }
    }
}
```

Usage Guidelines Use the **set** form of this command to create or modify a rule within an IP routing policy.

Use the **delete** form of this command to remove a rule from an IP routing policy.

Use the **show** form of this command to display a rule within an IP routing policy.

policy route pbr name <name> rule <number> disable

Disables a routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **disable**

delete policy route pbr name *name* **rule** *number* **disable**

show policy route pbr name *name* **rule** *number*

Command Default

The rule is enabled.

Parameters

name

The name of an IP routing policy.

number

The numeric identifier of the rule. The numbers range from 1 through 9999.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    disable
                }
            }
        }
    }
}
```

Usage Guidelines

Use this command to disable a routing policy rule. Disabling a rule is a useful way to test how the policy route performs without a specific rule and without having to delete and reconfigure the rule.

Use the **set** form of this command to disable a routing policy rule.

Use the **delete** form of this command to re-enable a rule.

Use the **show** form of this command to display a routing policy rule.

policy route pbr name <name> rule <number> icmp <icmp>

Creates a routing policy rule to match Internet Control Message Protocol (ICMP) packets.

Syntax

set policy route pbr name *name* **rule** *number* **icmp** { **code** *number* | **type** *number* | **type-name** [*name*] }

delete policy route pbr name *name* **rule** *number* **icmp** [**code** | **type** *type* | **type-name** [*name*]]

show policy route pbr name *name* **rule** *number* **icmp**

Command Default

The rule is enabled.

Parameters

name

The name of an IP routing policy. The policy name must be unique and must not be used with other PBR policy commands.

number

The numeric identifier of the rule. The identifier ranges from 1 through 9999.

code *number*

An IPv4 ICMP code (0 through 255).

type *number*

An IPv4 ICMP type (0 through 255).

A valid ICMPv6 type code from 0 through 255; for example, 128 (Echo Request), or a type and code pair (each from 0 to 255); for example, 1/4 for port unreachable. Alternatively, you can specify an ICMPv6 type code explicitly; for example, echo-request (Echo Request). For a list of ICMP codes and types, see ICMP Types on page 43.

type-name *name*

The name of an ICMP.

For type options, refer to ICMP Types on page 43.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    icmp {
                        code number
                        type number
                        type-name all
                        type-name name
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to create a rule to match ICMP packets.

Use the **delete** form of this command to delete a rule that matches ICMP packets.

Use the **show** form of this command to display a rule that matches ICMP packets.

policy route pbr name <name> rule <number> icmpv6 <icmpv6>

Creates a routing policy rule to match Internet Control Message Protocol (ICMP) IP packets for a routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **icmpv6** [**code** *number* | **type** *number* | **type-name** [**any** | *name*]]

delete policy route pbr name *name* **rule** *number* **icmpv6** [**code** *number* | **type** *number* | **type-name** [**any** | *name*]]

show policy route pbr name *name* **rule** *number* **icmpv6**

Command Default

The rule is enabled.

Parameters

name

The name of an IP routing policy. The policy name must be unique and must not be used with other PBR policy commands.

number

The numeric identifier of the rule. The identifier ranges from 1 through 9999.

code *number*

An IPv6 ICMP code (0 through 255).

type *number*

An IPv6 ICMP type (0 through 255).

A valid ICMPv6 type code from 0 through 255; for example, 128 (Echo Request), or a type and code pair (each from 0 to 255); for example, **1/4** for **port unreachable**. Alternatively, you can specify an ICMPv6 type code explicitly; for example, **echo-request** (Echo Request).

For a list of ICMP codes and types, see ICMPv6 Types on page 45.

type-name [**any** | *name*]

The name of an ICMPv6.

any: Any ICMPv6 type.

name: For type options, refer to ICMPv6 Types on page 45.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    icmpv6
                        code number
                        type number
                        type-name any
                        type-name name
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to create a rule to match ICMP packets.

Use the **delete** form of this command to delete a rule that matches ICMP packets.

Use the **show** form of this command to view a rule that matches ICMP packets.

policy route pbr name <name> rule <number> log

Enables logging for a routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **log**

delete policy route pbr name *name* **rule** *number* **log**

show policy route pbr name *name* **rule** *number*

Command Default

Logging is disabled.

Parameters

name

The name of an IP routing policy.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    log
                }
            }
        }
    }
}
```

Usage Guidelines

When logging is enabled, any action taken is logged.

Use the **set** form of this command to enable logging for a routing policy rule.

Use the **delete** form of this command to restore the default behavior for logging, that is, actions are not logged.

Use the **show** form of this command to display whether logging is enabled or disabled.

policy route pbr name <name> rule <number> port <port-name>

Defines the source port name, number, range, or port group for a routing policy rule.

Syntax **set policy route pbr name** *name* **rule** *number* { **port** [*name* | *1-65535* | *start-end* | *port-group-name*] }

delete policy route pbr name *name* **rule** *number* [**port** [*name* | *1-65535* | *start-end* | *port-group-name*]]

show policy route pbr name *name* **rule** *number* [**port**]

Parameters *name*

The name of an IP routing policy.

port [*name* | *1-65535* | *start-end* | *port-group-name*]

Applicable only when the protocol is TCP or UDP. A source port to match. The format of the port is any of the following:

port-name: The name of an IP service; for example, http. You can specify any service name in the /etc/services file.

1-65535: A port number. The numbers range from 1 through 65535.

start-end: A specified range of ports; for example, 1001-1005.

port-group-name: A port group. A packet is considered a match if it matches any port name or number specified in the group. Only one port group may be specified. The port group must already be defined.

This criterion specifies a group of addresses, ports, or networks for packet source address.

A packet is considered a match for an address, a network, or a port group if it matches any host IP address, network address, or port name or number, respectively, in the group. However, if more than one group is specified, the packet must be a match for both groups to be considered a match. For example, if both an address group and a port group are specified, the source of the packet must match at least one item in the address group and at least one item in the port group.

An address group may be specified with a port group, and a network group may be specified with a port group. You cannot specify both an address and a network group.

You can use a combination of these formats in a list separated by commas. You can also negate the entire list by prefixing it with an exclamation mark (!); for example, !22,telnet,http,123,1001-1005.

If both an address and a port are specified, the packet is considered a match only if both the address and the port match.

Modes Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    port name
                    port 1-65535
                    port start-end
                    port port-group-name
```

```
                    }
                }
            }
        }
}
```

Usage Guidelines This criterion specifies a port or a group of ports for packet source address for a routing policy rule.

A packet is considered a match for an address, a network, or a port group if it matches any host IP address, network address, or port name or number, respectively, in the group. However, if more than one group is specified, the packet must be a match for both groups in order to be considered a match. For example, if an address group and a port group are both specified, the packet's source must match at least one item in the address group and at least one item in the port group.

An address group can be specified together with a port group, and a network group can4 be specified together with a port group. You cannot specify both an address and a network group.

The address family must match the specified family by using the **set policy route pbr name** *name* **rule** *number* **address-family ipv4** command.

Use the **set** form of this command to define the source for a routing policy rule.

Use the **delete** form of this command to remove the source for a routing policy rule.

Use the **show** form of this command to view the source for a routing policy rule.

policy route pbr name <name> rule <number> protocol <protocol>

Defines the protocol of an IP routing policy rule.

Syntax **set policy route pbr name** *name* **rule** *number* **protocol** { *text* | *0-255* | **all** | *name* }

delete policy route pbr name *name* **rule** *number* **protocol** [*text* | *0-255* | **all** | *name*]

show policy route pbr name *name* **rule** *number* **protocol**

Parameters *name*

The name of an IP routing policy.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

protocol

The *protocol* is any of the following:

text: Matches packets by protocol type. Any protocol literals or numbers listed in the file /etc/protocols can be specified. The keywords **icmpv6** and **all** (for all protocols) are also supported.

0-255: An IP protocol number that ranges from 0 through 255.

all: All IP protocols.

! *protocol*: All IP protocols except for the specified name or number. Prefixing the protocol name with the negation operator (the exclamation mark) matches every protocol except the specified protocol. For example, !tcp matches all protocols except TCP.

This parameter matches the last, next-header field in the IP header chain. This match means that if the packet has no extension headers, it matches the next-header field in the main header. If the packet does have extension headers, the parameter matches the next-header field of the last extension header in the chain. In other words, the parameter always matches the ID of the transport-layer packet that is being carried.

Exercise care when employing more than one rule that uses the negation. Routing policy rules are evaluated sequentially, and a sequence of negated rules could result in unexpected behavior.

Modes Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    protocol
                        text
                        0-255
                        all
                        name
                }
            }
        }
```

```
        }
}
```

Usage Guidelines Use the **set** form of this command to define the protocol of an IP routing policy rule.

Use the **delete** form of this command to remove a protocol from a routing policy rule.

Use the **show** form of this command to view the protocol of a routing policy rule.

policy route pbr name <name> rule <number> set <match-criteria>

Defines the address family or routing table ID for an IP routing policy rule.

Syntax

set policy route pbr name *name* **rule** *rule-number* **set** { **af** *name* | **table** *number* }

delete policy route pbr name *name* **rule** *rule-number* **set** [**af** *name* | **table** *number*]

show policy route pbr name *name* **rule** *rule-number* **set**

Parameters

name

The name of an IP routing policy. The policy name must be unique and must not be used with other PBR policy commands.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

af *name*

To match IPv4 or IPv6 address family for this rule. Performs alternate processing on packets satisfying the match criteria.

table *number*

To match according to the PBR Table ID numbers 1 through 128. Performs alternate processing on packets satisfying the match criteria.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    set
                       af name
                       table number
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to define the address family or routing table ID for an IP routing policy rule.

Use the **delete** form of this command to remove the address family or routing table ID for a rule.

Use the **show** form of this command to view the address family or routing table ID for a rule.

The address family must match the specified family by using the **set policy route pbr name** *name* **rule** *number* **address-family ipv4** command.

Use the **set** form of this command to define the source for a routing policy rule.

Use the **delete** form of this command to remove the source for a routing policy rule.

Use the **show** form of this command to view the source for a routing policy rule.

policy route pbr name <name> rule <number> source address <address>

Defines the source address for a routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **source address** *address*

delete policy route pbr name *name* **rule** *number* **source address** *address*

show policy route pbr name *name* **rule** *number* **source address**

Parameters

name

The name of an IP routing policy.

address *address*

A source address to match. The match criteria are any of the following:

ipv6-address: An IPv6 address; for example, fe80::20c:29fe:fe47:f89.

ipv6-address/prefix: A network address, where ::/0 matches any network; for example, fe80::20c:29fe:fe47:f88/64.

ipv6-address-ipv6-address: A range of contiguous IP addresses; for example, fe80::20c:29fe:fe47:f00-fe80::20c:29fe:fe47:f89.

! *ipv6-address*: All IP addresses except the one specified.

! *ipv6-addressprefix*: All network addresses except the one specified.

! *ipv6-address-ipv6-address*: All IP addresses except those in the specified range.

If both an address and a port are specified, the packet is considered a match only if both the address and the port match.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    source {
                        address address
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

This criterion specifies a port or a group of ports for packet source address for a routing policy rule.

A packet is considered a match for an address, a network, or a port group if it matches any host IP address, network address, or port name or number, respectively, in the group. However, if more than one group is specified, the packet must be a match for both groups in order to be considered a match. For example, if an address group and a port group are both specified, the packet's source must match at least one item in the address group and at least one item in the port group.

An address group can be specified together with a port group, and a network group can4 be specified together with a port group. You cannot specify both an address and a network group.

The address family must match the specified family by using the **set policy route pbr name** *name* **rule** *number* **address-family ipv4** command.

Use the **set** form of this command to define the source for a routing policy rule.

Use the **delete** form of this command to remove the source for a routing policy rule.

Use the **show** form of this command to view the source for a routing policy rule.

policy route pbr name <name> rule <number> source port <port>

Defines the source port name, number, range, or port group for a routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **source port** [*name* | *1-65535* | *start-end* | *port-group-name*]

delete policy route pbr name *name* **rule** *number* **source port** [*name* | *1-65535* | *start-end* | *port-group-name*]

show policy route pbr name *name* **rule** *number* **source port**

Parameters

name

The name of an IP routing policy.

port [*name* | *1-65535* | *start-end* | *port-group-name*]

Applicable only when the protocol is TCP or UDP. A source port to match. The format of the port is any of the following:

name: The name of an IP service; for example, http. You can specify any service name in the /etc/services file.

1-65535: A port number. The numbers range from 1 through 65535.

start-end: A specified range of ports; for example, 1001-1005.

port-group-name: A port group. A packet is considered a match if it matches any port name or number specified in the group. Only one port group may be specified. The port group must already be defined.

This criterion specifies a group of addresses, ports, or networks for packet source address.

A packet is considered a match for an address, a network, or a port group if it matches any host IP address, network address, or port name or number, respectively, in the group. However, if more than one group is specified, the packet must be a match for both groups to be considered a match. For example, if both an address group and a port group are specified, the source of the packet must match at least one item in the address group and at least one item in the port group.

An address group may be specified with a port group, and a network group may be specified with a port group. You cannot specify both an address and a network group.

You can use a combination of these formats in a list separated by commas. You can also negate the entire list by prefixing it with an exclamation mark (!); for example, !22,telnet,http,123,1001-1005.

If both an address and a port are specified, the packet is considered a match only if both the address and the port match.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    source {
                        port name
                        port 1-65535
```

```
                            port start-end
                            port port-group-name
                        }
                    }
                }
            }
        }
    }
```

Usage Guidelines This criterion specifies a port or a group of ports for packet source address for a routing policy rule.

A packet is considered a match for an address, a network, or a port group if it matches any host IP address, network address, or port name or number, respectively, in the group. However, if more than one group is specified, the packet must be a match for both groups in order to be considered a match. For example, if an address group and a port group are both specified, the packet's source must match at least one item in the address group and at least one item in the port group.

Use the **set** form of this command to define the source for a routing policy rule.

Use the **delete** form of this command to remove the source for a routing policy rule.

Use the **show** form of this command to view the source for a routing policy rule.

policy route pbr name <name> rule <number> table

Defines the table number for an IP routing policy rule.

Syntax

set policy route pbr name *name* **rule** *rule-number* **table** *number*

delete policy route pbr name *name* **rule** *rule-number* **table** [*number*]

show policy route pbr name *name* **rule** *rule-number* **set**

Parameters

name

The name of an IP routing policy. The policy name must be unique and must not be used with other PBR policy commands.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

af *name*

To match IPv4 or IPv6 address family for this rule. Performs alternate processing on packets satisfying the match criteria.

table *number*

To match according to the PBR Table ID numbers 1 through 128. Performs alternate processing on packets satisfying the match criteria.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    set
                        af name
                        table number
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to define the address family or routing table ID for an IP routing policy rule.

Use the **delete** form of this command to remove the address family or routing table ID for a rule.

Use the **show** form of this command to view the address family or routing table ID for a rule.

The address family must match the specified family by using the **set policy route pbr name** *name* **rule** *number* **address-family ipv4** command.

Use the **set** form of this command to define the source for a routing policy rule.

Use the **delete** form of this command to remove the source for a routing policy rule.

Use the **show** form of this command to view the source for a routing policy rule.

policy route pbr name <name> rule <number> tcp flags <flags>

Defines the types of TCP flags to be matched for a routing policy rule.

Syntax

set policy route pbr name *name* **rule** *number* **tcp flags** *flags*

delete policy route pbr name *name* **rule** *number* **tcp flags** [*flags*]

show policy route pbr name *name* **rule** *number* **tcp flags**

Parameters

name

The name of an IP routing policy.

number

The numeric identifier of the rule. Rule numbers determine the order in which rules are executed. Each rule must have a unique rule number. The numbers range from 1 through 9999.

You can define multiple rules by creating more than one **rule** configuration node.

flags

The flags to be matched in a packet. The flags are any of SYN, ACK, FIN, RST, URG, and PSH. You can specify more than one flag in a list separated by commas.

Prefixing a flag name with the negation operator matches packets with that flag unset. You can also use **!** to match packets by not using a given TCP flag. For example, the list SYN, **!**ACK, **!**FIN, **!**RST matches only packets with the SYN flag set and the ACK, FIN, and RST flags unset.

Modes

Configuration mode

Configuration Statement

```
policy {
    route {
        pbr {
            name name {
                rule number {
                    tcp {
                        flags text
                    }
                }
            }
        }
    }
}
```

Usage Guidelines

Use the **set** form of this command to define the types of TCP flags to be matched for a routing policy rule.

Use the **delete** form of this command to remove the types of TCP flags to be matched for a routing policy rule.

Use the **show** form of this command to view the types of TCP flags to be matched for a routing policy rule.

show policy route <interface>

Displays routing policy configuration or statistics.

Syntax **show policy route** *interface*

Parameters *interface*

The name of an interface.

Modes Operational mode

Usage Guidelines A policy identifies traffic that matches parameters and specifies which table to use. The table defines the routes for a packet to take. A routing policy is a named collection of as many as 9,999 packet-classification rules. When applied to an interface, the policy rule classifies incoming traffic.

NOTE
The PBR rule counters count all of the matched packets regardless of the availability of the route.

Use this command in operational mode to display packet statistics for all PBR rules in all groups.

For example:

show policy route

```
vyatta@vyatta:~$ show policy route
----------------------------
Rulesets Information
----------------------------
------------------------------------------------------------------------------
PBR Group: "dp0p192p1-group-v6":
Active on (dp0p192p1, in)
rule        proto        packets            bytes
----        -----        -------            -----
1           icmpv6          0               0
  condition - proto icmpv6  from abcd::1 to 5555::1 tag 98

2           tcp             0               0
  condition - proto tcp from abcd::1 to 5555::1 tag 99

3           udp             0               0
  condition - proto udp from abcd::1 to 3333::1 tag 100
```

Related commands

The following table lists related commands that are documented elsewhere.

Related commands documented elsewhere	
protocols static table	The commands for creating alternate routing tables are described in *Brocade 5600 vRouter Basic Routing Reference Guide*
resources group address-group <group-name>	Defines a group of IP addresses that are referenced in firewall rules. (Refer to *Brocade 5600 vRouter Basic Routing Reference Guide*.)

Related commands documented elsewhere	
resources group port-group <group-name>	Defines a group of ports that are referenced in firewall rules. (Refer to *Brocade 5600 vRouter Basic Routing Reference Guide*.)
show ip route table	The command for displaying the contents of an alternate routing table is described in *Brocade 5600 vRouter Basic Routing Reference Guide*.
firewall group	Routing policy match criteria support references to predefined groups of addresses, ports, and networks. Commands for defining such groups are described in *Brocade 5600 vRouter Firewall Reference Guide*.

ICMP Types

This appendix lists the Internet Control Messaging Protocol (ICMP) types defined by the Internet Assigned Numbers Authority (IANA).

The IANA has developed a standard that maps a set of integers onto ICMP types. The following table lists the ICMP types and codes defined by the IANA and maps them to the literal strings that are available in the Brocade vRouter.

TABLE 3 ICMP types

ICMP Type	Code	Literal	Description
0 - Echo reply	0	echo-reply	Echo reply (pong)
3 - Destination unreachable		destination- unreachable	Destination is unreachable
	0	network-unreachable	Destination network is unreachable
	1	host-unreachable	Destination host is unreachable
	2	protocol-unreachable	Destination protocol is unreachable
	3	port-unreachable	Destination port is unreachable
	4	fragmentation-needed	Fragmentation is required
	5	source-route-failed	Source route has failed
	6	network-unknown	Destination network is unknown
	7	host-unknown	Destination host is unknown
	9	network-prohibited	Network is administratively prohibited
	10	host-prohibited	Host is administratively is prohibited
	11	ToS-network-unreachable	Network is unreachable for ToS
	12	ToS-host-unreachable	Host is unreachable for ToS
	13	communication-prohibited	Communication is administratively prohibited
	14	host-precedence-violation	Requested precedence is not permitted.
	15	precedence-cutoff	Precedence is lower than the required minimum.
4 - Source quench	0	source-quench	Source is quenched (congestion control)
5 - Redirect message		redirect	Redirected message

TABLE 3 ICMP types (Continued)

ICMP Type	Code	Literal	Description
	0	network-redirect	Datagram is redirected for the network
	1	host-redirect	Datagram is redirected for the host
	2	ToS-network-redirect	Datagram is redirected for the ToS and network
	3	ToS-host-redirect	Datagram is redirected for the ToS and host
8 - Echo request	0	echo-request	Echo request (ping)
9 - Router advertisement	0	router-advertisement	Router advertisement
10 - Router solicitation	0	router-solicitation	Router solicitation
11 - Time exceeded		time-exceeded	Time to live (TTL) has exceeded
	0	ttl-zero-during-transit	TTL has expired in transit
	1	ttl-zero-during-reassembly	Fragment reassembly time has exceeded
12 - Parameter problem: Bad IP header		parameter-problem	Bad IP header
	0	ip-header-bad	Pointer that indicates an error
	1	required-option-missing	Missing required option
13 - Timestamp	0	timestamp-request	Request for a timestamp
14 - Timestamp reply	0	timestamp-reply	Reply to a request for a timestamp
15 - Information request	0		Information request
16 - Information reply	0		Information reply
17 - Address mask request	0	address-mask-request	Address mask request
18 - Address mask reply	0	address-mask-reply	Address mask reply
19 - Ping		ping	A ping message
20 - Pong		pong	A pong message

ICMPv6 Types

This appendix lists the ICMPv6 types defined by the Internet Assigned Numbers Authority (IANA).

The Internet Assigned Numbers Authority (IANA) has developed a standard that maps a set of integers onto ICMPv6 types. The following table lists the ICMPv6 types and codes defined by the IANA and maps them to the strings literal strings available in the Brocade vRouter.

TABLE 4 ICMPv6 types

ICMPv6 Type	Code	Literal	Description
1 - Destination unreachable		destination- unreachable	
	0	no-route	No route to destination
	1	communication-prohibited	Communication with destination administratively prohibited
	2		Beyond scope of source address
	3	address-unreachable	Address unreachable
	4	port-unreachable	Port unreachable
	5		Source address failed ingress/egress policy
	6		Reject route to destination
2 - Packet too big	0	packet-too-big	
3 - Time exceeded		time-exceeded	
	0	ttl-zero-during-transit	Hop limit exceeded in transit
	1	ttl-zero-during-reassembly	Fragment reassembly time exceeded
4 - Parameter problem		parameter-problem	
	0	bad-header	Erroneous header field encountered
	1	unknown-header-type	Unrecognized Next Header type encountered
	2	unknown-option	Unrecognized IPv6 option encountered
128 - Echo request	0	echo-request (ping)	Echo request
129 - Echo reply	0	echo-reply (pong)	Echo reply
133 - Router solicitation	0	router-solicitation	Router solicitation

TABLE 4 ICMPv6 types (Continued)

ICMPv6 Type	Code	Literal	Description
134 - Router advertisement	0	router-advertisement	Router advertisement
135 - Neighbor solicitation	0	neighbor-solicitation (neighbour-solicitation)	Neighbor solicitation
136 - Neighbor advertisement	0	neighbor-advertisement (neighbour-advertisement)	Neighbor advertisement

Supported Interface Types

The following table shows the syntax and parameters of supported interface types. Depending on the command, some of these types may not apply.

Interface Type	Syntax	Parameters
Bridge	**bridge** *brx*	*brx*: The name of a bridge group. The name ranges from br0 through br999.
Data plane	**dataplane** *interface-name*	*interface-name*: The name of a data plane interface. Following are the supported formats of the interface name: • **dp***x***p***y***p***z***—The name of a data plane interface, where — **dp***x* specifies the data plane identifier (ID). Currently, only dp0 is supported. — p*y* specifies a physical or virtual PCI slot index (for example, p129). — **p***z* specifies a port index (for example, p1). For example, dp0p1p2, dp0p160p1, and dp0p192p1. • **dp***x***em***y* —The name of a data plane interface on a LAN-on-motherboard (LOM) device that does not have a PCI slot, where **em***y* specifies an embedded network interface number (typically, a small number). For example, dp0em3. • **dp***x***s***y* —The name of a data plane interface on a device that is installed on a virtual PCI slot, where *x***s***y* specifies an embedded network interface number (typically, a small number). For example, dp0s2. Currently, this format applies only when using the KVM or Hyper-V platforms. • **dp***x***P***n***p***y***p***z* —The name of a data plane interface on a device that is installed on a secondary PCI bus, where **P***n* specifies the bus number. You can use this format to name data plane interfaces on large physical devices with multiple PCI buses. For these devices, it is possible to have network interface cards installed on different buses with these cards having the same slot ID. The value of *n* must be an integer greater than 0. For example, dp0P1p162p1 and dp0P2p162p1.

Interface Type	Syntax	Parameters
Data plane vif	**dataplane** *interface-name* **vif** *vif-id* [**vlan** *vlan-id*]	*interface-name*: Refer to the preceding description. *vif-id*: A virtual interface ID. The ID ranges from 1 through 4094. *vlan-id*: The VLAN ID of a virtual interface. The ID ranges from 1 through 4094.
Loopback	**loopback lo** or **loopback lo***n*	*n*: The name of a loopback interface, where *n* ranges from 1 through 99999.
OpenVPN	**openvpn** *vtunx*	*vtunx*: The identifier of an OpenVPN interface. The identifier ranges from vtun0 through vtun*x*, where *x* is a nonnegative integer.
Tunnel	**tunnel** *tunx* or **tunnel** *tunx* **parameters**	*tunx*: The identifier of a tunnel interface you are defining. The identifier ranges from tun0 through tun*x*, where *x* is a nonnegative integer.
Virtual tunnel	**vti** *vtix*	*vtix*: The identifier of a virtual tunnel interface you are defining. The identifier ranges from vti0 through vti*x*, where *x* is a nonnegative integer. **Note:** This interface does not support IPv6.
VRRP	*parent-interface* **vrrp vrrp-group** *group*	*parent-interface*: The type and identifier of a parent interface; for example, data plane dp0p1p2 or bridge br999. *group*: A VRRP group identifier. The name of a VRRP interface is not specified. The system internally constructs the interface name from the parent interface identifier plus the VRRP group number; for example, dp0p1p2v99. Note that VRRP interfaces support the same feature set as does the parent interface.

List of Acronyms

Acronym	Description
ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
AH	Authentication Header
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
DMVPN	dynamic multipoint VPN
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

Acronym	Description
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
IGMP	Internet Group Management Protocol
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
ISAKMP	Internet Security Association and Key Management Protocol
ISM	Internet Standard Multicast
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
mGRE	multipoint GRE
MIB	Management Information Base
MLD	Multicast Listener Discovery
MLPPP	multilink PPP
MRRU	maximum received reconstructed unit
MTU	maximum transmission unit
NAT	Network Address Translation
NBMA	Non-Broadcast Multi-Access
ND	Neighbor Discovery

Acronym	Description
NHRP	Next Hop Resolution Protocol
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
PIM	Protocol Independent Multicast
PIM-DM	PIM Dense Mode
PIM-SM	PIM Sparse Mode
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
RP	Rendezvous Point
RPF	Reverse Path Forwarding
RSA	Rivest, Shamir, and Adleman
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
SPT	Shortest Path Tree

Acronym	Description
SSH	Secure Shell
SSID	Service Set Identifier
SSM	Source-Specific Multicast
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