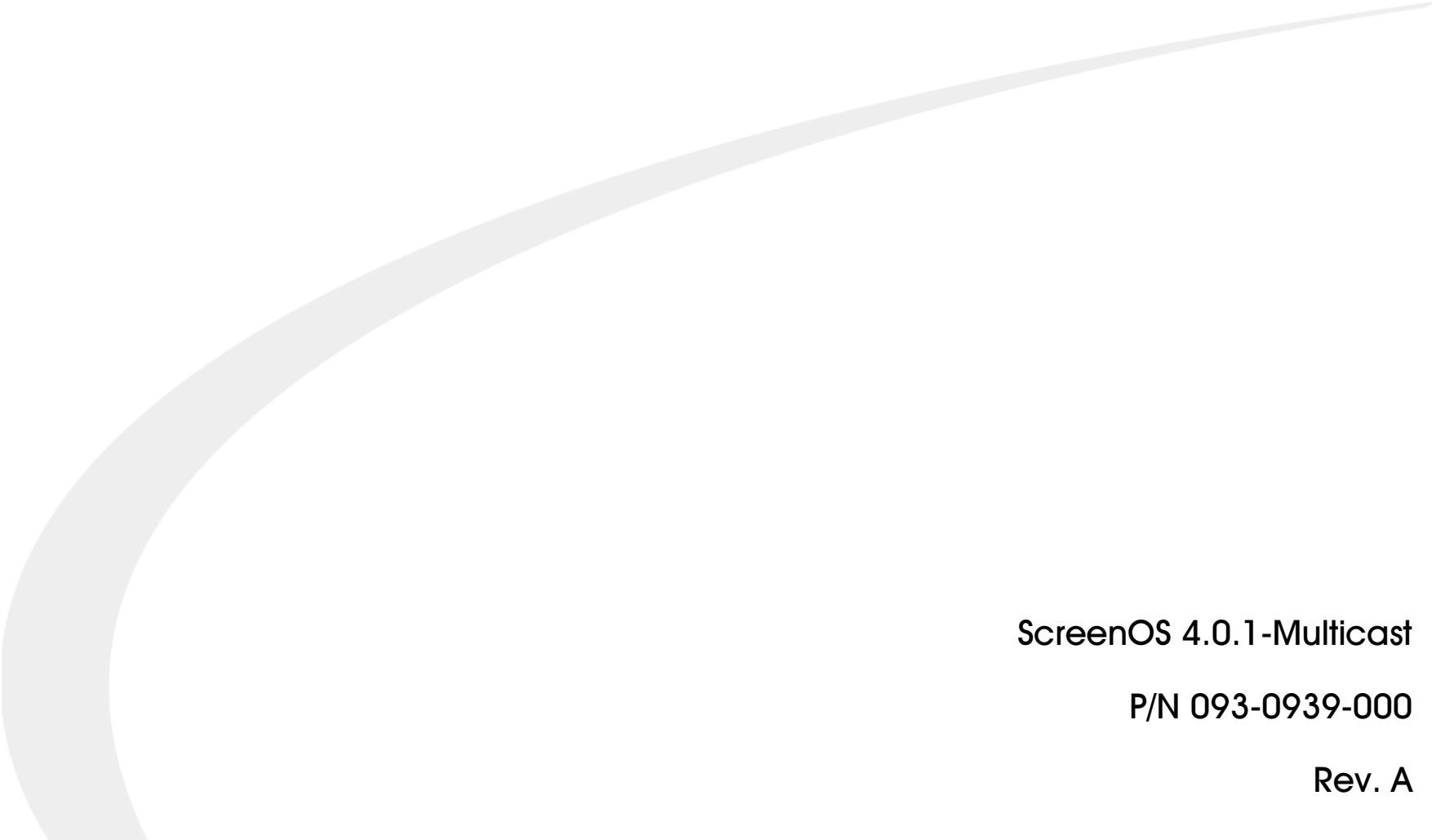


NetScreen New Features Guide



ScreenOS 4.0.1-Multicast

P/N 093-0939-000

Rev. A

Copyright Notice

Copyright © 2003 NetScreen Technologies, Inc. All rights reserved.

NetScreen, NetScreen Technologies, GigaScreen, and the NetScreen logo are registered trademarks of NetScreen Technologies, Inc. NetScreen-5GT, NetScreen-5XP, NetScreen-5XT, NetScreen-25, NetScreen-50, NetScreen-100, NetScreen-204, NetScreen-208, NetScreen-500, NetScreen-5200, NetScreen-5400, NetScreen-Global PRO, NetScreen-Global PRO Express, NetScreen-Remote Security Client, NetScreen-Remote VPN Client, NetScreen-IDP 10, NetScreen-IDP 100, NetScreen-IDP 500, GigaScreen ASIC, GigaScreen-II ASIC, and NetScreen ScreenOS are trademarks of NetScreen Technologies, Inc. All other trademarks and registered trademarks are the property of their respective companies.

Information in this document is subject to change without notice.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without receiving written permission from:

NetScreen Technologies, Inc.
Building #3
805 11th Avenue
Sunnyvale, CA 94089
www.netscreen.com

FCC Statement

The following information is for FCC compliance of Class A devices: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. The equipment generates, uses, and can radiate radio-frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case users will be required to correct the interference at their own expense.

The following information is for FCC compliance of Class B devices: The equipment described in this manual generates and may radiate radio-frequency energy. If it is not installed in accordance with NetScreen's installation instructions, it may cause interference with radio and television reception. This equipment has been tested and found to comply with the limits for a Class B digital device in accordance with the specifications in part 15 of the FCC rules. These specifications are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Consult the dealer or an experienced radio/TV technician for help.
- Connect the equipment to an outlet on a circuit different from that to which the receiver is connected.

Caution: Changes or modifications to this product could void the user's warranty and authority to operate this device.

Disclaimer

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR NETSCREEN REPRESENTATIVE FOR A COPY.

Contents

- Preface vii
 - Conventions viii
 - WebUI Navigation Conventions viii
 - Example: Objects > Addresses > List > New viii
 - CLI Conventions ix
 - CLI Syntax X
 - Variable Notation X
 - Common CLI Variable Names xi
 - NetScreen Documentation xiii
- Section 1 - Routing Information Protocol 1
- Chapter 1 Routing Information Protocol (RIP) 3
 - Overview of RIP 4
 - Basic RIP Configuration 5
 - Creating a RIP Routing Instance in a Virtual Router 5
 - Example: Creating a RIP Routing Instance 6
 - Enabling the RIP Instance 6
 - Example: Enabling a RIP Routing Instance 7
 - Example: Removing a RIP Routing Instance 8
 - Redistributing Routes 9
 - Example: Redistributing Routes into RIP 9
 - Global Parameters 11
 - Example: Advertising the Default Route
to RIP Neighbors 12
 - Interface Parameters 13
 - Example: Setting RIP Interface Parameters 14

- Security Configuration 15
 - Authenticating Neighbors 15
 - Example: Configuring Neighbor Authentication .. 15
 - Filtering RIP Neighbors 16
 - Example: Configuring Trusted Neighbors 16
 - Rejecting Default Routes 17
 - Example: Rejecting Default Routes 17
 - Protecting Against Flooding 18
 - Example: Configuring an Update Threshold 18
 - Example: Configuring RIP for the Trust and Untrust
Zones 19
- Chapter 2 New and Modified CLI Commands - RIP ...21
 - interface 22
 - Syntax 22
 - Keywords and Variables 23
 - RIP Context Commands 26
 - advertise-def-route 29
 - Syntax 29
 - Keywords and Variables 30
 - config 31
 - Syntax 31
 - Keywords and Variables 31
 - default-metric 32
 - Syntax 32
 - Keywords and Variables 32
 - enable 33
 - Syntax 33
 - Keywords and Variables 33

- flush-timer 34
 - Syntax 34
 - Keywords and Variables 34
- interface 35
 - Syntax 35
 - Keywords and Variables 35
- invalid-timer 36
 - Syntax 36
 - Keywords and Variables 36
- max-neighbor-count 37
 - Syntax 37
 - Keywords and Variables 37
- neighbors 38
 - Syntax 38
 - Keywords and Variables 38
- no-source-validation 39
 - Syntax 39
 - Keywords and Variables 39
- redistribute 40
 - Syntax 40
 - Keywords and Variables 41
- reject-default-route 42
 - Syntax 42
 - Keywords and Variables 42
- route-map 43
 - Syntax 43
 - Keywords and Variables 43
- routes-redistribute 45
 - Syntax 45
 - Keywords and Variables 45
- rules-redistribute 46
 - Syntax 46
 - Keywords and Variables 46

- threshold-update 47
 - Syntax 47
 - Keywords and Variables 47
- timer 48
 - Syntax 48
 - Keywords and Variables 48
- trusted-neighbors 49
 - Syntax 49
 - Keywords and Variables 49
- update-timer 50
 - Syntax 50
 - Keywords and Variables 50
- update-threshold 51
 - Syntax 51
 - Keywords and Variables 51
- vrouter 52
 - Commands 52
 - Arguments 53
- Chapter 3 New Messages - RIP 55**
 - RIP 56
 - Critical (00204) 56
- Section 2 - Multicast Routing 59**
- Chapter 4 Multicast Routing 61**
 - Multicast Routing Overview 62
 - Multicast Addresses 62
 - Multicast Sources and Receivers 63
 - Multicast Distribution Trees 63
 - Reverse Path Forwarding 63
 - Forwarding State 64

Configuring Multicast Routing on NetScreen Devices	65
Multicast Policies	65
Access Lists	66
Generic Routing Encapsulation (GRE)	66
Multicast Configurations	66
Chapter 5 IGMP	69
IGMP Overview	70
Maintaining Group Membership	70
Joining a Group	71
Leaving a Group	72
Configuring IGMP on NetScreen Devices	73
Enabling IGMP on Interfaces	73
Example: Enabling IGMP on an Interface	73
Example: Disabling IGMP on an Interface	74
Security Considerations	75
Example: Configuring an Access List for Accepted Groups	75
Basic IGMP Configuration	76
Example: Basic IGMP Configuration	76
Verifying Your IGMP Configuration	79
IGMP Operational Parameters	82
IGMP Proxy	84
Multicast Routing Using IGMP Proxy	86
Sending Membership Reports Upstream to the Source	86
Sending Multicast Data Downstream to the Receivers	87
Configuring IGMP Proxy	88
Enabling IGMP Proxy on Interfaces	88
Example: Enabling IGMP on Interfaces	88
Example: Disabling IGMP on Interfaces	89
Creating a Multicast Policy	89
Example: Creating a Multicast Policy for IGMP Messages	89
Example: Basic IGMP Proxy Configuration	90
Chapter 6 PIM-SM	97
Overview of PIM	98
Designated Router	99
Mapping RPs to Groups	99
Static RP Mapping	99
Dynamic RP Mapping	99
Forwarding Traffic on the Distribution Tree	100
Source Sends Data to a Group	100
Host Joins a Group	102
Configuring PIM on NetScreen Devices	104
Creating and Enabling a PIM Instance in a Virtual Router	104
Example: Creating and Enabling a PIM Instance in a Virtual Router	105
Example: Removing a PIM Instance	105
Enabling PIM on Interfaces	105
Example: Enabling PIM on an Interface	106
Example: Disabling PIM on an Interface	106
Creating a Multicast Group Policy	106
Example: Creating a Multicast Group Policy	106
Basic PIM Configuration	107
Example: Basic PIM Configuration	108
Verifying the Configuration	111
Configuring RP to Group Mappings	114
Example: Creating a Static RP	114
Example: Creating a Candidate RP	114

Security Considerations	116	Keywords and Variables	153
Restricting Multicast Groups.....	116	PIM Context Commands.....	156
Example: Restricting Multicast Groups.....	117	accept-group	158
Restricting Multicast Sources.....	118	Syntax	158
Example: Restricting Multicast Sources.....	118	Keywords and Variables	158
Restricting RPs	118	bsr	159
Example: Restricting RPs	119	Syntax	159
PIM Interface Parameters	120	Keywords and Variables	159
Example: Changing the DR Priority.....	121	enable	160
Neighbor Policy	121	Syntax	160
Example: Defining a Neighbor Policy.....	121	Keywords and Variables	160
Bootstrap Border	122	interface	161
Example: Defining a Bootstrap Border.....	122	Syntax	161
Proxy RP	123	Keywords and Variables	161
Configuring a Proxy RP	123	join-prune	162
Example: Creating a Proxy RP	124	Syntax	162
Example: Basic Proxy RP Configuration	125	Keywords and Variables	162
Chapter 7 New and Modified CLI Commands -		mgroup	163
Multicast	131	Syntax	163
igmp.....	132	Keywords and Variables	163
Syntax.....	132	mroute	165
Keywords and Variables.....	133	Syntax	165
interface.....	136	Keywords and Variables	165
Syntax.....	136	neighbor	167
Keywords and Variables (IGMP).....	141	Syntax	167
Keywords and Variables (PIM)	148	Keywords and Variables	167
Keywords and Variables (tunnel interface).....	150	rp.....	168
Defaults	151	Syntax	168
multicast-group-policy	152	Keywords and Variables	168
Syntax.....	152	rpf	170
		Syntax	170
		Keywords and Variables	170

spt-threshold 171
 Syntax..... 171
 Keywords and Variables..... 171
zone 172
 Syntax..... 172
 Keywords and Variables..... 173
vrouter 176
 Syntax..... 176
 Keywords and Variables..... 178

Chapter 8 New Messages - Multicast 181
 IGMP 182
 Warning 182
 Notification (00045)..... 183
 Multicast 190
 Warning (01001)..... 190
 Notification (00048)..... 192
 PIM..... 195
 Notification (00046)..... 195

Preface

This document presents the new features in this release of NetScreen ScreenOS software. It is organized into the following sections:

- “Section 1 - Routing Information Protocol” on page 1
 - Chapter 1, “Routing Information Protocol (RIP)” on page 3
 - Chapter 2, “New and Modified CLI Commands - RIP” on page 21
 - Chapter 3, “New Messages - RIP” on page 55
- “Section 2 - Multicast Routing” on page 59
 - Chapter 4, “Multicast Routing” on page 61
 - Chapter 5, “IGMP” on page 69
 - Chapter 6, “PIM-SM” on page 97
 - Chapter 7, “New and Modified CLI Commands - Multicast” on page 131
 - Chapter 8, “New Messages - Multicast” on page 181

For more information about ScreenOS features, CLI commands, and messages refer to the following documents:

- *NetScreen Concepts & Examples ScreenOS Reference Guide*
- *NetScreen CLI Reference Guide*
- *NetScreen Message Log Reference Guide*

CONVENTIONS

This book presents two management methods for configuring a NetScreen device: the Web user interface (WebUI) and the command line interface (CLI). The conventions used for both are introduced below.

WebUI Navigation Conventions

Throughout this book, a single chevron (>) is used to indicate navigation through the WebUI by clicking menu options and links.

Example: **Objects > Addresses > List > New**

To access the new address configuration dialog box, do the following:

1. Click **Objects** in the menu column.
The Objects menu option expands to reveal a subset of options for Objects.
2. (Applet menu) Hover the mouse over **Addresses**.
(DHTML menu) Click **Addresses**.
The Addresses option expands to reveal a subset of options for Addresses.
3. Click **List**.
The address book table appears.
4. Click the **New** link in the upper right corner.
The new address configuration dialog box appears.

CLI Conventions

The following conventions are used when presenting the syntax of a command line interface (CLI) command:

- Anything inside square brackets [] is optional.
- Anything inside braces { } is required.
- If there is more than one choice, each choice is separated by a pipe (|). For example,
set interface { ethernet1 | ethernet2 | ethernet3 } manage
means “set the management options for the ethernet1, ethernet2, or ethernet3 interface”.
- Variables appear in *italic*. For example:

```
set admin user name password
```

When a CLI command appears within the context of a sentence, it is in **bold** (except for variables, which are always in *italic*). For example: “Use the **get system** command to display the serial number of a NetScreen device.”

Note: When typing a keyword, you only have to type enough letters to identify the word uniquely. For example, typing **set adm u joe j12fmt54** is enough to enter the command **set admin user joe j12fmt54**. Although you can use this shortcut when entering commands, all the commands documented here are presented in their entirety.

CLI SYNTAX

Most NetScreen CLI commands have changeable parameters that affect the outcome of command execution. NetScreen documentation represents these parameters as variables. Such variables may include names, identification numbers, IP addresses, subnet masks, numbers, dates, and other values.

Variable Notation

The variable notation used in this manual consists of italicized parameter identifiers. For example, the **set arp** command uses four identifiers, as shown here:

```
set arp
{
  ip_addr mac_addr interface
  age number |
  always-on-dest |
  no-cache
}
```

where

- *ip_addr* represents an IP address.
- *mac_addr* represents a MAC address.
- *interface* represents a physical or logical interface.
- *number* represents a numerical value.

Thus, the command might take the following form:

```
ns-> set arp 172.16.10.11 00e02c000080 ethernet2
```

where **172.16.10.11** is an IP address, **00e02c000080** is a MAC address, and **ethernet2** is a physical interface.

Common CLI Variable Names

The following list shows the CLI variable names used in NetScreen documents.

<i>comm_name</i>	The community name of a host or other device.
<i>date</i>	A date value.
<i>dev_name</i>	A device name, as with flash card memory.
<i>dom_name</i>	A domain name, such as “acme” in www.acme.com .
<i>dst_addr</i>	A destination address, as with a policy definition that defines a source and destination IP address.
<i>filename</i>	The name of a file.
<i>fqdn</i>	Fully-qualified domain name, such as www.acme.com .
<i>grp_name</i>	The name of a group, such as an address group or service group.
<i>interface</i>	A physical or logical interface.
<i>id_num</i>	An identification number.
<i>ip_addr</i>	An IPv4 address.
<i>ipv6_addr</i>	An IPv6 address.
<i>key_str</i>	A key, such as a session key, a private key, or a public key.
<i>key_hex</i>	A key expressed as a hexadecimal number.
<i>loc_str</i>	A location of a file or other resource.
<i>mac_addr</i>	A MAC address.
<i>mbr_name</i>	The name of a member in a group, such as an address group or a service group.
<i>mask</i>	A subnet mask, such as 255.255.255.224 or /24 .
<i>mcst_addr</i>	A multicast address.
<i>name_str</i>	The name of an item, such as an address book entry.
<i>number</i>	A numeric value, usually an integer, such as a threshold or a maximum.

<i>pol_num</i>	A policy number.
<i>port_num</i>	A number identifying a logical port.
<i>pref_len</i>	A number identifying the prefix length for an IPv6 address.
<i>pswd_str</i>	A password.
<i>ptcl_num</i>	A number uniquely identifying a protocol, such as TCP, IP, or UDP.
<i>serv_name</i>	The name of a server.
<i>shar_secret</i>	A shared secret value.
<i>spi_num</i>	A Security Parameters Index (SPI) number.
<i>src_addr</i>	A source address, as with a policy definition that defines a source and destination IP address.
<i>string</i>	A character string, such as a comment.
<i>svc_name</i>	The name of a service, such as HTTP or MAIL.
<i>time</i>	A time value.
<i>tunn_str</i>	The name of a tunnel, such as an L2TP tunnel.
<i>url_str</i>	A URL, such as www.acme.com .
<i>usr_str</i>	A user, usually an external entity such as a dialup user.
<i>vrouter</i>	A local virtual router, such as trust-vr or untrust-vr.
<i>zone</i>	The name of a security zone.

Some commands contain multiple variables of the same type. The names of such variables may be numbered to identify each individually. For example, the **set dip** command contains two *id_num* variables, each numbered for easy identification:

```
set dip group id_num1 [ member id_num2 ]
```

NETSCREEN DOCUMENTATION

To obtain technical documentation for any NetScreen product, visit www.netscreen.com/resources/manuals/.

To obtain the latest software version, visit www.netscreen.com/services/download_soft. Select a category of software product from the dropdown list, then follow the displayed instructions. (You must be a registered user to download NetScreen software.)

If you find any errors or omissions in the following content, please contact us at the e-mail address below:

techpubs@netscreen.com

SECTION 1 - ROUTING INFORMATION PROTOCOL

Routing Information Protocol (RIP)

This chapter describes the Routing Information Protocol version 2 (RIPv2) routing protocol on NetScreen devices. It contains the following sections:

- [“Overview of RIP” on page 4](#)
- [“Basic RIP Configuration” on page 5](#)
 - [“Creating a RIP Routing Instance in a Virtual Router” on page 5](#)
 - [“Enabling the RIP Instance” on page 6](#)
 - [“Redistributing Routes” on page 9](#)
- [“Global Parameters” on page 11](#)
- [“Interface Parameters” on page 13](#)
- [“Security Configuration” on page 15](#)
 - [“Authenticating Neighbors” on page 15](#)
 - [“Filtering RIP Neighbors” on page 16](#)
 - [“Rejecting Default Routes” on page 17](#)
 - [“Protecting Against Flooding” on page 18](#)

OVERVIEW OF RIP

Routing information protocol (RIP) is a distance vector protocol used as an Interior Gateway Protocol (IGP) in moderate-sized autonomous systems (ASs). ScreenOS supports RIP version 2 (RIPv2), as defined by RFC 2453. While RIPv2 supports only simple password (plain text) authentication, NetScreen's RIP implementation also supports MD5 authentication extensions, as defined by RFC 2082.

As mentioned previously, RIP is intended for moderate-sized networks. It can also be used to manage route information within a small, homogeneous, network such as a corporate LAN. The longest path allowed in a RIP network is 15 hops. A metric value of 16 indicates an invalid or unreachable destination (this value is also referred to as "infinity" since it is larger than the 15-hop maximum allowed in RIP networks).

RIP is not intended for large networks or networks where routes are chosen based on real-time parameters such as measured delay, reliability, or load. RIP supports both point-to-point networks (used with VPNs) and broadcast/multicast Ethernet networks. RIP does not support point-to-multipoint interfaces.

RIP sends out messages that contain the complete routing table to every neighboring router every 30 seconds. These messages are normally sent as multicasts to address 224.0.0.9 from the RIP port.

The RIP routing database contains one entry for every destination that is reachable through the RIP routing instance. The RIP routing database includes the following information:

- IPv4 address of a destination. Note that RIP does not distinguish between networks and hosts.
- IP address of the first router along the route to the destination (the next hop).
- Network interface used to reach the first router.
- Metric that indicates the distance, or cost, of getting to the destination. Most RIP implementations use a metric of 1 for each network.
- A timer that indicates the time that has elapsed since the database entry was last updated.

BASIC RIP CONFIGURATION

Like OSPF and BGP, you create RIP on a per-Virtual Router basis on a NetScreen device. If you have multiple virtual routers (VRs) in a system, you can enable multiple instances of RIP, one instance for each VR.

This section describes the following basic steps to configure RIP on a NetScreen device:

1. Create the RIP routing instance in a Virtual Router.
2. Enable the RIP instance.
3. Redistribute routes learned from different routing protocols (such as OSPF, BGP, or statically configured routes) into the RIP instance.

This section describes how to perform each of these tasks using either the CLI or the WebUI.

You can also optionally configure other RIP parameters such as the following:

- Global parameters, such as timers and trusted RIP neighbors, that are set at the VR level for the RIP protocol (see [“Global Parameters” on page 11](#))
- Interface parameters, such as neighbor authentication, that are set on a per-interface basis for the RIP protocol (see [“Interface Parameters” on page 13](#))
- Security-related RIP parameters, that are set at either the VR level or on a per-interface basis (see [“Security Configuration” on page 15](#))

Creating a RIP Routing Instance in a Virtual Router

As described previously, you create a RIP routing instance on a specific virtual router on a NetScreen device. Deleting a RIP routing instance in a VR removes the corresponding RIP configurations for all interfaces that are in the VR. For more information about virtual routers and configuring a virtual router on NetScreen devices, see Chapter 3, “Routing and Virtual Routers,” in Volume 2, “Fundamentals,” in the *NetScreen Concepts and Examples ScreenOS Reference Guide*.

Example: Creating a RIP Routing Instance

In this example, you create a RIP routing instance on the virtual router trust-vr.

WebUI

Network > Routing > Virtual Router (trust-vr) > Edit: Select **Create RIP Instance**, and then click **OK**.

CLI

```
ns-> set vrouter trust-vr
ns(trust-vr)-> set protocol rip
```

Enabling the RIP Instance

You enable (and disable) RIP functions at two different levels:

- Enabling and disabling RIP at the VR level affects the RIP instance in the VR. When you enable RIP at the VR level, RIP can transmit and process RIP packets received on all RIP-enabled interfaces in the VR. When you disable RIP at the VR level, RIP stops transmitting and processing RIP packets on *all* RIP-enabled interfaces in the VR.
- Enabling and disabling RIP on an interface affects RIP on only a *specific* interface. By default, RIP is disabled on all interfaces in the VR and you must explicitly enable it on an interface. When you disable RIP at the interface level, RIP does not transmit or receive packets on the specified interface. Interface configuration parameters are preserved when you disable RIP on an interface.

Example: Enabling a RIP Routing Instance

In this example, you enable the RIP routing instance on the virtual router trust-vr and enable RIP on the trust interface.

WebUI

RIP Routing Instance

1. Network > Routing > Virtual Router (trust-vr) > Edit > Edit RIP Instance: Select **Enable**, and then click **OK**.

RIP Interface

2. Network > Interface > Edit (for Trust interface) > RIP: Select **Enable**, and then click **Apply**.

CLI

RIP Routing Instance

```
ns-> set vrouter trust-vr protocol rip enable
```

RIP Interface

```
ns-> set interface trust protocol rip enable
```

Example: Removing a RIP Routing Instance

As mentioned previously, you can delete RIP for the VR or for a specific interface. In this example, you delete the RIP routing instance on the virtual router trust-vr and disable RIP on the trust interface.

WebUI

RIP Routing Instance

1. Network > Routing > Virtual Router (trust-vr) > Edit: Select **Delete RIP Instance**, and then click **OK** at the confirmation prompt.

RIP Interface

2. Network > Interface (for Trust interface) > RIP: Clear **Enable**, and then click **Apply**.

CLI

RIP Routing Instance

```
ns-> unset vrouter trust-vr protocol rip
```

RIP Interface

```
ns-> unset interface trust protocol rip
```

Redistributing Routes

Route redistribution is the exchange of route information between routing protocols. For example, you can redistribute the following types of routes into the RIP routing instance in the same virtual router:

- Routes learned from BGP
- Routes learned from OSPF
- Directly connected routes
- Imported routes
- Statically configured routes

You need to configure a route map to filter the routes that are redistributed. For more information about creating route maps for route redistribution, see Chapter 3, “Routing and Virtual Routers” in Volume 2, “Fundamentals,” of the *NetScreen Concepts and Examples ScreenOS Reference Guide*.

Routes imported into RIP from other protocols have a default metric of 1. You can change the default metric (see [“Global Parameters” on page 11](#)).

Example: Redistributing Routes into RIP

In this example, you redistribute static routes that are in the subnetwork 20.1.0.0/16 to RIP neighbors in the trust-vr virtual router. To do this, you first create an access list to permit addresses in the 20.1.0.0/16 subnetwork. Then, configure a route map that permits addresses that match the access list you configured. Use the route map to specify the redistribution of static routes into the RIP routing instance.

WebUI

1. Network > Routing > Virtual Router (trust-vr) > Access List > New: Enter the following, and then click **OK**:
 - Access List ID: 20
 - Sequence No.: 1
 - IP/Netmask: 20.1.0.0/16
 - Action: Permit (select)

2. Network > Routing > Virtual Router (trust-vr) > Route Map > New: Enter the following, and then click **OK**:
 - Map Name: rmap1
 - Action: Permit (select)
 - Sequence No.: 1
 - Match Properties:
 - Access List: 20 (select)
3. Network > Routing > Virtual Router (trust-vr) > Edit > Edit RIP Instance > Redistributable Rules: Enter the following, and then click **Add**:
 - Route Map: rmap1 (select)
 - Protocol: Static (select)

CLI

```
set vrouter trust-vr acc-list 20 permit ip 20.1.0.0/16 1
set vrouter trust-vr route-map name rmap1 permit 1
set vrouter trust-vr route-map rmap1 1 match ip 20
set vrouter trust-vr protocol rip redistribute route-map rmap1 protocol static
```

GLOBAL PARAMETERS

This section describes RIP global parameters that you can configure at the VR level. When you configure a RIP parameter at the VR level, the parameter setting affects operations on all RIP-enabled interfaces. You can modify global parameter settings through the RIP routing protocol context in the CLI or by using the WebUI.

The following table describes the RIP global parameters and their default values.

RIP Global Parameter	Description	Default Value
Default metric	Default metric value for routes imported into RIP from other protocols, such as OSPF and BGP.	10
Update timer	Specifies, in seconds, when to issue updates of RIP routes to neighbors.	30 seconds
Maximum packets per update	Specifies the maximum number of packets received per update.	No maximum
Invalid timer	Specifies, in seconds, when a route becomes invalid from the time a neighbor stops advertising the route.	180 seconds
Flush timer	Specifies, in seconds, when a route is removed from the time the route is invalidated.	120 seconds
Maximum neighbors	The maximum number of RIP neighbors allowed.	16
Trusted neighbors	Specifies an access list that defines RIP neighbors. If no neighbors are specified, RIP uses multicasting or broadcasting to detect neighbors on an interface.	No neighbors are configured
Allow neighbors on different subnet	Specifies that RIP neighbors on different subnets are allowed.	Disabled
Advertise default route	Specifies whether the default route (0.0.0.0/0) is advertised.	Disabled
Reject default route	Specifies whether RIP rejects a default route learned from another protocol.	Disabled
Incoming route map	Specifies the filter for routes to be learned by RIP.	None
Outgoing route map	Specifies the filter for routes to be advertised by RIP.	None

Example: Advertising the Default Route to RIP Neighbors

By default, the default route (0.0.0.0/0) is not advertised to RIP neighbors. The following command advertises the default route to RIP neighbors in the trust-vr virtual router with a metric of 5 (you must enter a metric value). The default route must exist in the routing table.

WebUI

Network > Routing > Virtual Router (trust-vr) > Edit > Edit RIP Instance: Enter the following, and then click **OK**.

Advertising Default Route: (select)

Metric: 5

CLI

```
set vrouter trust-vr protocol rip adv-default-route always metric number 5
```

See [Chapter 2 “New and Modified CLI Commands” on page 65](#) for more information about global parameters that you can configure in the RIP routing protocol context.

INTERFACE PARAMETERS

This section describes RIP parameters that you configure at the interface level. When you configure a RIP parameter at the interface level, the parameter setting affects the RIP operation only on the specific interface. You can modify interface parameter settings with **interface** commands in the CLI or by using the WebUI.

The following table describes the RIP interface parameters and their default values.

RIP Interface Parameter	Description	Default Value
Split-horizon	Specifies whether to enable split-horizon (do not advertise routes learned from a neighbor back to the same neighbor). If this is disabled, routes that are learned from a neighbor are advertised back to the same neighbor with a metric of 16.	Disabled
RIP metric	Specifies the RIP metric for the interface.	1
Authentication	Specifies either clear text password or MD5 authentication.	No authentication used.
Passive mode	Specifies that the interface is to receive but not transmit RIP packets.	No
Incoming route map	Specifies the filter for routes to be learned by RIP.	None
Outgoing route map	Specifies the filter for routes to be advertised by RIP.	None

You can define incoming and outgoing route map filters at the VR level or at the interface level. A route map filter you define at the interface level takes precedence over a route map filter defined at the VR level. For example, if you define an incoming route map at the VR level and a different incoming route map at the interface level, the incoming route map defined at the interface level takes precedence.

Example: Setting RIP Interface Parameters

In this example, you configure the following RIP parameters for the trust interface:

- Set MD5 authentication, with the key 1234567898765432 and the key ID 215.
- Enable split horizon for the interface.

WebUI

Network > Interfaces(Edit) > RIP: Enter the following, and then click **OK**:

Authentication: MD5 (select)

Key: 1234567898765432

Key ID: 215

Split Horizon: (select)

CLI

```
set interface trust protocol rip authentication md5 1234567898765432 key -id
  215
set interface trust protocol rip split-horizon
```

SECURITY CONFIGURATION

This section describes possible security problems in the RIP routing domain and methods of preventing attacks.

Note: *To make RIP more secure, you should configure all routers in the RIP domain to be at the same security level. Otherwise, a compromised RIP router can bring down the entire RIP routing domain.*

Authenticating Neighbors

A RIP router can be easily spoofed, since RIP packets are not encrypted and most protocol analyzers provide decapsulation of RIP packets. Authenticating RIP neighbors is the best way to fend off these types of attacks.

RIP provides both simple password and MD5 authentication to validate RIP packets received from neighbors. All RIP packets received on the interface that are not authenticated are discarded. By default, there is no authentication enabled on any RIP interface.

Example: Configuring Neighbor Authentication

In this example, you configure MD5 authentication, with the key 1234567898765432 and the key ID 215, for the trust interface.

WebUI

Network > Interfaces (Edit) > RIP: Enter the following, and then click **OK**:

Authentication: MD5 (select)

Key: 1234567898765432

Key ID: 215

CLI

```
set interface trust protocol rip authentication md5 1234567898765432 key -id
215
```

Filtering RIP Neighbors

Multi-access environments can allow devices, including routers, to be connected into a network relatively easily. This can cause stability or performance issues if the connected device is not reliable. To prevent this problem, you can use an access list to filter the devices that are allowed to become RIP neighbors. By default, RIP neighbors are limited to devices that are on the same subnet as the NetScreen virtual router.

Example: Configuring Trusted Neighbors

In this example, you configure the following global parameters for the RIP routing instance running in the trust-vr virtual router:

- Maximum number of RIP neighbors is 1.
- The IP address of the trusted neighbor, 10.1.1.1, is specified in an access-list.

WebUI

1. Network > Routing > Virtual Router (trust-vr) > Access List > New: Enter the following, and then click **OK**:
 - Access List ID: 10
 - Sequence No.: 1
 - IP/Netmask: 10.1.1.1/32
2. Network > Routing > Virtual Router (trust-vr) > Edit > Edit RIP Instance: Enter the following, and then click **OK**:
 - Maximum Neighbors: 1
 - Trusted Neighbors: 10

CLI

```
ns-> set vrouter trust-vr
ns(trust-vr)-> set access-list 10 permit ip 10.1.1.1/32 1
ns(trust-vr)-> set protocol rip
ns(trust-vr/rip)-> set max-neighbor-count 1
ns(trust-vr/rip)-> set trusted-neighbors 10
```

Rejecting Default Routes

In a Route Detour Attack, a router injects a default route (0.0.0.0/0) into the routing domain in order to detour packets to itself. The router can then either drop the packets, causing service disruption, or it can obtain sensitive information in the packets before forwarding them. On NetScreen devices, RIP by default accepts any default routes that are learned in RIP and adds the default route to the routing table.

Example: Rejecting Default Routes

In this example, you configure the RIP routing instance running in the trust-vr virtual router to reject any default routes that are learned in RIP.

WebUI

Network > Routing > Virtual Router (trust-vr) > Edit > Edit RIP Instance: Enter the following, and then click **OK**:

Reject Default Route Learnt by RIP: (select)

CLI

```
ns-> set vrouter trust-vr
ns(trust-vr) -> set protocol rip
ns(trust-vr)-> set reject-default-route
```

Protecting Against Flooding

A malfunctioning or compromised router can flood its neighbors with RIP routing update packets. On NetScreen virtual routers, you can configure the maximum number of update packets that can be received on a RIP interface within a certain interval to avoid flooding of update packets. All update packets that exceed the configured update threshold are dropped. If you do not set an update threshold, all update packets are accepted.

You need to exercise care when configuring an update threshold when neighbors have large routing tables, as the number of routing updates can be quite high within a given duration because of flash updates. Update packets that exceed the threshold are dropped and valid routes may not be learned.

Example: Configuring an Update Threshold

In this example, you set the maximum number of routing update packets that RIP can receive on an interface to 4.

WebUI

Network > Routing > Virtual Router (trust-vr) > Edit > Edit RIP Instance: Enter the following, and then click **OK**:

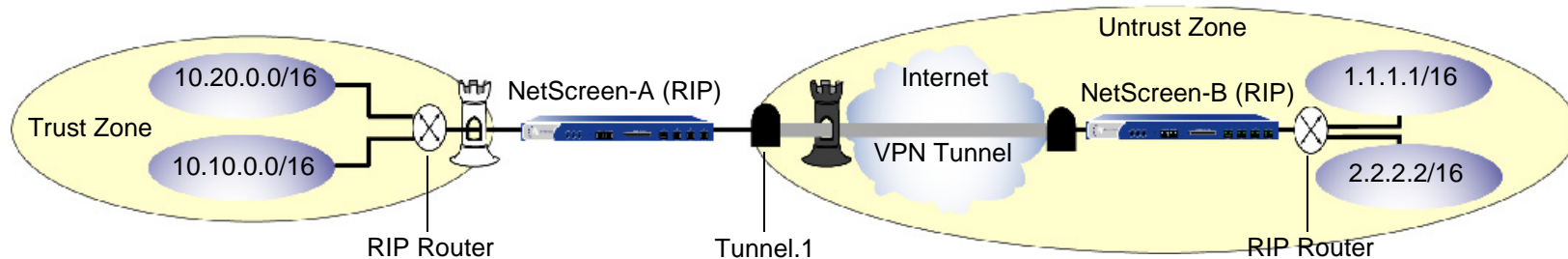
Maximum Number Packets per Update Time: 4

CLI

```
ns-> set vrouter trust-vr
ns(trust-vr)-> set protocol rip
ns(trust-vr/rip)-> set threshold-update 4
```

Example: Configuring RIP for the Trust and Untrust Zones

The following example creates and enables a RIP routing instance in the Trust-VR virtual router on the device NetScreen-A. You enable RIP on both the VPN tunnel interface and the Trust zone interface. Only routes that are in the subnet 10.10.0.0/16 are advertised to the RIP neighbor on NetScreen-B. This is done by first configuring an access list that permits only addresses in the subnet 10.10.0.0/16, then specifying a route map *abcd* that permits routes that match the access list. You then specify the route map to filter the routes that are advertised to RIP neighbors.



WebUI

1. Network > Routing > Virtual Router > Edit (for trust-vr) > Create RIP Instance: Select **Enable RIP**, and then click **OK**.
2. Network > Routing > Virtual Router > Access List (for trust-vr) > New: Enter the following, and then click **OK**:

Access List ID: 10

Sequence No.: 10

IP/Netmask: 10.10.0.0/16

Action: Permit

3. Network > Routing > Virtual Router > Route Map (for trust-vr) > New: Enter the following, and then click **OK**:
 - Map Name: abcd
 - Sequence No.: 10
 - Action: Permit
 - Match Properties:
 - Access List: (select), 10
4. Network > Routing > Virtual Router > Edit (for trust-vr) > Edit RIP Instance: Select the following, and then click **OK**:
 - Outgoing Route Map Filter: abcd
5. Network > Interfaces > Edit (for tunnel.1) > RIP: Enter the following, and then click **Apply**:
 - Enable RIP: (select)
6. Network > Interfaces > Edit (for trust) > RIP: Enter the following, and then click **Apply**:
 - Enable RIP: (select)

CLI

```
set vrouter trust-vr protocol rip
set vrouter trust-vr protocol rip enable
set interface tunnel.1 protocol rip enable
set interface trust protocol rip enable
set vrouter trust-vr access-list 10 permit ip 10.10.0.0/16 10
set vrouter trust-vr route-map name abcd permit 10
set vrouter trust-vr route-map abcd 10 match ip 10
set vrouter trust-vr protocol rip route-map abcd out
```

New and Modified CLI Commands - RIP

This chapter introduces the following new commands:

- [RIP Context Commands](#) on page 26
- [interface](#) on page 22
- [vrouter](#) on page 52

New command elements in the Syntax sections appear in **red**. For example, in the following command, **protocol rip** is new in this release:

```
get interface interface protocol rip
```

The following command descriptions focus only on the new elements added in this release. For more information about other command elements, refer to the *NetScreen CLI Reference Guide*.

interface

Description: Use the **interface** commands to configure RIP interface parameters

Note: This section only describes new keywords and variables for the **interface** commands. For more information on other keywords and variables for the **interface** commands, refer to the NetScreen CLI Reference Guide.

Syntax

get

```
get interface interface protocol rip
```

set (RIP)

```
set interface interface protocol rip  
  [  
    authentication { password pswd_str | md5 key_str key-id id_num }  
    enable |  
    metric number |  
    passive-mode |  
    route-map name_str |  
    split-horizon  
  ]
```

unset (RIP)

```
unset interface interface protocol rip
[
  authentication |
  enable |
  metric |
  passive-mode |
  route-map name_str |
  split-horizon
]
```

Keywords and Variables

Variable Parameter

```
get interface interface [ ... ]
set interface interface { ... } [ ... ]
unset interface interface { ... } [ ... ]
```

interface The interface on which RIP is enabled. By default,RIP are disabled.

protocol

```
get interface interface protocol rip
set interface interface protocol rip
  [
    authentication { password pswd_str | md5 key_str key-id id_num }
    enable |
    metric number number |
    passive-mode |
    route-map name_str { in | out } |
    split-horizon
  ]
unset interface interface protocol rip
  [
    authentication |
    enable |
    metric |
    passive-mode |
    route-map name_str { in | out } |
    split-horizon
  ]
```

- protocol rip** Sets, unsets, or displays the current routing protocol settings for the interface.
- **route-map** *name_str* Specifies the route-map on which to filter incoming routes (routes learned by RIP) or outgoing routes (routes advertised by RIP).
 - **in** Specifies the route map is to be used for incoming routes.
 - **out** Specifies the route map is to be used for outgoing routes.

- **authentication** { **password** *pswd_str* | **md5** *key_str* **key-id** *id_num* } Specifies the authentication method used to verify RIP neighbors.
 - **password** specifies a clear-text password used for verification. If you specify password authentication, you must also specify an 8-byte password.
 - **md5** directs the Netscreen device to use the Message Digest version 5 (MD5) authentication algorithm for verification. If you specify MD5 authentication, you must also specify a 16-byte key and key identifier.
- **enable** Enables RIP on the specified interface.
- **metric** *number* Configures the RIP metric for the specified interface. The default metric is 1.
- **passive-mode** Specifies that the interface is to receive but not transmit RIP packets.
- **split-horizon** Enables the split-horizon function on the specified interface. If split-horizon is enabled, RIP does not advertise routes learned from a neighbor back to the same neighbor. If split-horizon is disabled, RIP advertises routes learned from a neighbor back to the same neighbor with a metric of 16. By default, split-horizon is disabled.

RIP Context Commands

The commands described in the following pages are **rip** context commands. Use the **rip** context commands to configure the Routing Information Protocol (RIP) on a virtual router in a NetScreen device. You issue these commands within the context of a specific virtual router and the RIP protocol.

Initiating the **rip** context requires two steps:

1. Enter the virtual router context by executing the **set vrouter** command.

```
ns-> set vrouter trust-vr
```

2. Enter the RIP context by executing the **set protocol rip** command.

```
ns(trust-vr)-> set protocol rip
```

The following commands are executable in the **rip** context.

[advertise-def-route](#)

Use the **advertise-def-route** commands to advertise the default route (0.0.0.0/0) of the current virtual router in all areas.

Every virtual router has a default route entry, which matches every destination. (Any entry with a more specific prefix overrides the default route entry.)

Command options: **get, set, unset**

[config](#)

Use the **config** command to display all commands executed to configure the RIP routing instance.

Command options: **get**

[default-metric](#)

Use the **default-metric** commands to set the RIP metric for redistributed routes. The default value is 10.

Command options: **set, unset**

[enable](#)

Use the **enable** commands to enable or disable RIP in the virtual router.

Command options: **set, unset**

[flush-timer](#)

Use the **flush-timer** commands to configure the number of seconds that elapse before ScreenOS automatically removes an invalidated route. The default is 120 seconds.

Command options: **set, unset**

<u>interface</u>	Use the interface command to display all RIP interfaces in the virtual router. Command options: get
<u>invalid-timer</u>	Use the invalid-timer commands to configure the number of seconds that elapse after a neighbor stops advertising a route before the route becomes invalid. The default is 180 seconds. Command options: set, unset
<u>max-neighbor-count</u>	Use the max-neighbor-count commands to set the maximum number of RIP neighbors allowed. The default is 16. Command options: set, unset
<u>neighbors</u>	Use the neighbors command to display the status of RIP neighbors. Command options: get
<u>no-source-validation</u>	Use the no-source-validation commands to accept responses from RIP neighbors in other subnets or to reject such responses. Command options: set, unset
<u>redistribute</u>	Use the redistribute commands to import known routes from a router running a different protocol into the current routing instance. You can import the following types of routes: <ul style="list-style-type: none">• Manually created routes• BGP routes• OSPF routes• Routes sent by an external router that has at least one interface with an assigned IP address• Routes that have already been imported Command options: set, unset
<u>reject-default-route</u>	Use the reject-default-route commands to cause RIP to reject a default route learned from another protocol. Command options: get, set, unset
<u>route-map</u>	Use the route-map commands to filter and offset metric routes. Command options: get, set, unset

<u>routes-redistribute</u>	Use the routes-redistribute command to display redistributed routes. Command options: get
<u>rules-redistribute</u>	Use the rules-redistribute command to display redistribution rules. Command options: get
<u>threshold-update</u>	Use the threshold-update commands to set the maximum number of routing packets allowed per update interval. Command options: set, unset
<u>timer</u>	Use the timer command to display RIP timers. Command options: get
<u>trusted-neighbors</u>	Use the trusted-neighbors commands to set an access list that defines RIP neighbors. Command options: get, set, unset
<u>update-timer</u>	Use the update-timer commands to set the interval, in seconds, when route updates are issued to RIP neighbors. Command options: set, unset
<u>update-threshold</u>	Use the update-threshold command to display the number of routing packets per update interval. Command options: get

advertise-def-route

Description: Use the **advertise-def-route** commands to advertise the default route (0.0.0.0/0) of the current virtual router.

Every router has a default route entry, which matches every destination. (Any entry with a more specific prefix overrides the default route entry.)

Before you can execute the **advertise-def-route** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

get

```
get advertise-def-route
```

set

```
set advertise-def-route [ always ] [ metric number ]
```

unset

```
unset advertise-def-route
```

Keywords and Variables

always

```
set advertise-def-route always [ ... ]
```

always Directs the routing instance to advertise the default route under all conditions, even if there is no default route in the routing table.

metric

```
set advertise-def-route [always ] [ metric number ]
```

metric Specifies the metric (cost), which indicates the overhead associated with the default route.

config

Description: Use the **config** command to display all commands executed to configure the RIP local virtual router. Before you can execute the **config** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get config

Keywords and Variables

None.

default-metric

Description: Use the **default-metric** commands to set the RIP metric for redistributed routes.

Before you can execute the **default-metric** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

set default-metric *number*

unset

unset default-metric

Keywords and Variables

Variable Parameter

set default-metric *number*

number The metric for the routes redistributed into RIP. Enter a value between 1-16.

enable

Description: Use the **enable** commands to enable or disable RIP from the current virtual router.

Before you can execute the **enable** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

set enable

unset

unset enable

Keywords and Variables

None.

flush-timer

Description: Use the **flush-timer** commands to configure the time that elapses before an invalid route is removed.

Before you can execute the **flush-timer** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

set flush-timer *number*

unset

unset flush-timer

Keywords and Variables

Variable Parameter

set flush-timer *number*

number The number of seconds that elapses before an invalid route is removed. The default value is 120.

interface

Description: Use the **interface** command to display all RIP interfaces on the current virtual router.

Before you can execute the **interface** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get interface

Keywords and Variables

None.

invalid-timer

Description: Use the **invalid-timer** commands to configure the time that elapses after a neighbor stops advertising a route before the route becomes invalid.

Before you can execute the **invalid-timer** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

```
set invalid-timer number
```

unset

```
unset invalid-timer
```

Keywords and Variables

Variable Parameter

```
set invalid-timer number
```

number The number of seconds after a neighbor stops advertising a route that the route becomes invalid. The default value is 180.

max-neighbor-count

Description: Use the **max-neighbor-count** commands to set the maximum number of RIP neighbors allowed.

Before you can execute the **max-neighbor-count** commands, you must initiate the **rip** context. (See “[RIP Context Commands](#)” on page 26.)

Syntax

set

```
set max-neighbor-count number
```

unset

```
unset max-neighbor-count
```

Keywords and Variables

Variable Parameter

```
set max-neighbor-count number
```

number The maximum number of RIP neighbors allowed. The default is 16.

neighbors

Description: Use the **neighbors** command to display the status of all RIP neighbors.

Before you can execute the **neighbors** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get neighbors

Keywords and Variables

None.

no-source-validation

Description: Use the **no-source-validation** commands to accept or reject responses from RIP neighbors in different subnets.

Before you can execute the **no-source-validation** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

```
set no-source-validation
```

unset

```
unset no-source-validation
```

Keywords and Variables

None.

redistribute

Description: Use the **redistribute** commands to import known routes from a router running a different protocol into the current RIP routing instance.

You can import the following types of routes:

- Manually-created routes (**static**)
- BGP routes (**bgp**)
- OSPF routes (**ospf**)
- Directly-connected interface with an IP address assigned to it (**connected**)
- Routes that have already been imported (**imported**)

Before you can execute the **redistribute** commands, you must initiate the **rip** context. (See “[RIP Context Commands](#)” on page 26.)

Syntax

get

```
get routes-redistribute  
get rules-redistribute
```

set

```
set redistribute route-map name_str protocol  
{ bgp | connected | imported | ospf | static }
```

unset

```
unset redistribute route-map name_str protocol  
{ bgp | connected | imported | ospf | static }
```

Keywords and Variables

protocol

```
set redistribute route-map name_str protocol { ... }
```

- protocol** Specifies the routing protocol. The route map can use the protocol type to determine whether to forward or deny an incoming packet.
- **bgp** specifies that the route map performs an action only on BGP routes in the subnetwork.
 - **connected** specifies that the route map performs an action only on routes sent from an external router that has at least one interface with an IP address assigned to it.
 - **imported** specifies that the route map performs an action only on imported routes in the subnetwork.
 - **ospf** specifies that the route map performs an action only on OSPF routes in the subnetwork.
 - **static** specifies that the route map performs an action only on static routes in the subnetwork.

route-map

```
set redistribute route-map name_str protocol { ... }
```

route-map Identifies the route map that specifies the routes to be imported.

Example: The following command redistributes a route that originated from a BGP routing domain into the current RIP routing instance:

```
ns(trust-vr/rip)-> set redistribute route-map map1 protocol bgp
```

reject-default-route

Description: Use the **reject-default-route** commands to cause RIP to reject default routes learned from another protocol.

Before you can execute the **reject-default-route** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

get

`get reject-default-route`

set

`set reject-default-route`

unset

`unset reject-default-route`

Keywords and Variables

None.

route-map

Description: Use the **route-map** commands to filter incoming or outgoing routes.

Before you can execute the **route-map** commands, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

```
get route-map
```

set

```
set route-map name_str { in | out }
```

unset

```
set route-map name_str { in | out }
```

Keywords and Variables

Variable Parameter

```
set route-map name_str
```

name_str The name of the route map to filter routes.

in

set route-map *name_str* **in**

in Specifies the route map is applied to routes to be learned by RIP.

out

set route-map *name_str* **out**

out Specifies the route map is applied to routes to be advertised by RIP.

Example: The following command applies the route map map1 to routes to be advertised by RIP:

```
ns(trust-vr/rip)-> set route-map map1 out
```

routes-redistribute

Description: Use the **routes-redistribute** command to display details about routes imported from a protocol other than RIP.

Before you can execute the **routes-redistribute** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get routes-redistribute

Keywords and Variables

None.

rules-redistribute

Description: Use the **rules-redistribute** command to display conditions set for routes imported from a protocol other than RIP.

Before you can execute the **rules-redistribute** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get rules-redistribute

Keywords and Variables

None.

threshold-update

Description: Use the **threshold-update** commands to set the maximum number of routing packets allowed per update interval.

Before you can execute the **threshold-update** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

```
set threshold-update number
```

unset

```
unset threshold-update
```

Keywords and Variables

Variable Parameter

```
set threshold-update number
```

number The maximum number of routing packets allowed per update interval.

timer

Description: Use the **timer** command to display information about various RIP timers.

Before you can execute the **timer** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get timer

Keywords and Variables

None.

trusted-neighbors

Description: Use the **trusted-neighbors** commands to specify an access list that defines allowed RIP neighbors.

Before you can execute the **trusted-neighbors** commands, you must initiate the **rip** context. (See “[RIP Context Commands](#)” on page 26.)

Syntax

get

```
get trusted-neighbors
```

set

```
set trusted-neighbors id_num
```

unset

```
set trusted-neighbors id_num
```

Keywords and Variables

Variable Parameter

```
set trusted-neighbors id_num
```

id_num The number of the access list that defines the allowed RIP neighbors.

update-timer

Description: Use the **update-timer** commands to set the interval that RIP sends route updates to neighbors.

Before you can execute the **update-timer** commands, you must initiate the **rip** context. (See [“RIP Context Commands”](#) on page 26.)

Syntax

set

set update-timer *number*

unset

unset update-timer

Keywords and Variables

Variable Parameter

set update-timer *number*

number The interval, in seconds, that RIP sends route updates to neighbors. The default is 30.

update-threshold

Description: Use the **update-threshold** command to display the number of routing packets per update interval.

Before you can execute the **update-threshold** command, you must initiate the **rip** context. (See [“RIP Context Commands” on page 26.](#))

Syntax

get

get update-threshold

Keywords and Variables

None.

vrouter

Description: Use the **vrouter** commands to configure the virtual router on the NetScreen device.

Executing the **set vrouter *name_str*** command without specifying further options places the CLI in the virtual router context. For example, the following command places the CLI in the trust-vr virtual router context:

```
ns-> set vrouter trust-vr
```

Once you initiate the routing context, all subsequent command executions apply to the specified local virtual router (**trust-vr** in this example). You can then initiate the **rip** protocol context.

To enter the rip context, execute the set protocol rip command.

```
ns(trust-vr)-> set protocol rip
```

In the **rip** protocol context, all command executions apply to the protocol.

Commands

get

```
get vrouter name_str protocol rip1
```

set

```
set vrouter name_str protocol rip1
```

unset

```
unset vrouter name_str protocol rip1
```

1. For more information on the **protocol rip** options, refer to the **rip** command descriptions.

Arguments

protocol rip

Places the NetScreen device in the RIP context. (For information on this context, refer to [“RIP Context Commands” on page 26.](#))

New Messages - RIP

This chapter introduces all the new NetScreen messages for this release. Each message is presented, its meaning explained, and—where appropriate—an administrative action recommended. The messages are grouped by message type, and then within that type by severity level, from the most severe to the least.

- [“RIP” on page 56](#)

For a complete list of NetScreen log messages, refer to the *NetScreen Message Log Reference Guide*.

RIP

The following messages relate to the Routing Information Protocol (RIP) used for dynamic routing.

Critical (00204)

- Message** <vrouter> update packet flood on by neighbor <id_num> interface <interface> has dropped a packet
- Meaning** The NetScreen device detected a flood of update packets coming from the specified RIP neighbor arriving at the specified interface. The NetScreen device is dropping update packets it receives from that neighbor.
- Action** Remove the connection between the virtual router and the RIP neighbor.

Notification (00045)

- Message** RIP instance in vrouter <vrouter> created
- Meaning** An admin has configured a RIP instance in the specified virtual router.
- Action** No recommended action.
-
- Message** rip instance in vrouter <vrouter> deleted
- Meaning** An admin has removed a RIP instance in the specified virtual router.
- Action** No recommended action.

Information (00544)

- Message** RIP neighbor <id_num> in vrouter <vrouter> is added
- Meaning** The specified device was added as a RIP neighbor to the virtual router.
- Action** No recommended action.
-
- Message** RIP neighbor <id_num> in vrouter <vrouter> is removed
- Meaning** The specified device was removed as a RIP neighbor to the virtual router.
- Action** No recommended action.

SECTION 2 - MULTICAST ROUTING

Multicast Routing

This chapter introduces basic multicast routing concepts. It contains the following sections:

- “Multicast Routing Overview” on page 62
 - “Multicast Addresses” on page 62
 - “Multicast Sources and Receivers” on page 63
 - “Multicast Distribution Trees” on page 63
 - “Reverse Path Forwarding” on page 63
 - “Forwarding State” on page 64
- “Configuring Multicast Routing on NetScreen Devices” on page 65
 - “Multicast Policies” on page 65
 - “Access Lists” on page 66
 - “Generic Routing Encapsulation (GRE)” on page 66
 - “Multicast Configurations” on page 66

MULTICAST ROUTING OVERVIEW

Enterprises use multicast routing to transmit traffic, such as data or video streams, from one source to a group of receivers simultaneously. Any host can be a source, and the receivers can be anywhere on the Internet.

IP multicast routing provides an efficient method for forwarding traffic to multiple hosts, because multicast-enabled routers transmit multicast traffic only to hosts that want to receive the traffic. Hosts must signal their interest in receiving multicast data and they must join a multicast group in order to receive the data. Multicast-enabled routers forward multicast traffic only to receivers interested in receiving the traffic.

You can deploy NetScreen devices in networking environments that support multicast routing to secure both unicast and multicast traffic. You can apply security policies to control multicast traffic and restrict the flow of traffic to specific communities or multicast groups.

Multicast routing environments require the following to forward multicast information:

- A mechanism between hosts and routers to communicate group membership information. NetScreen devices support IGMP (Internet Group Management Protocol) version 1 and 2. Routers and hosts use IGMP to transmit membership information only, not to forward or route multicast traffic.
- A multicast routing protocol to populate the multicast route table and forward data to hosts throughout the network. NetScreen devices support PIM-SM (Protocol Independent Multicast - Sparse-Mode). You can also use IGMP Proxy to transmit multicast information between routers without running a multicast routing protocol.

The following sections introduce basic concepts used in multicast routing.

Multicast Addresses

When a source sends multicast traffic, the destination address is a multicast group address. Multicast group addresses are Class D addresses from 224.0.0.0 to 239.255.255.255.

Multicast Sources and Receivers

In multicast routing, the sender of the multicast traffic is the source, and the host or client that receives the multicast data is the receiver. To receive multicast traffic, the receiver must join a multicast group. Receivers send multicast control traffic upstream, towards the source. The source sends multicast control traffic and data traffic downstream towards the receiver.

Multicast Distribution Trees

Multicast routers forward multicast traffic downstream from the source to the receivers through a multicast distribution tree. There are two types of multicast distribution trees:

- Shortest-Path Tree (SPT) - The source is at the root of the tree and forwards the multicast data downstream to each receiver. This is also referred to as a source specific tree.
- Shared Distribution Tree - The source transmits the multicast data to a router at the core of the network. This router then forwards the traffic downstream to the receivers on the distribution tree.

When a receiver starts receiving multicast traffic for a particular multicast group, it *joins* the distribution tree. The receiver is *pruned* from the distribution tree when it stops receiving multicast traffic. Multicast routing protocols support either one or both types of distribution trees.

Reverse Path Forwarding

Regardless of which distribution tree is used to forward multicast traffic, multicast routers use a process called reverse path forwarding (RPF) to check the validity of a multicast packet. When a router receives a multicast packet, it checks if the interface on which it received the packet (incoming interface) is the same interface it would use to send packets back to the sender. If it is, the router checks the route table and forwards the packet to the next hop on the distribution tree. If it is not, the router drops the packet. Multicast routers always perform this RPF check before they forward multicast traffic.

Forwarding State

Devices store information about the outgoing and incoming interface for a group. This is called the forwarding state and is denoted as (S, G), where S is the source IP address and G is the multicast group. On shared distribution trees where the source of the multicast traffic is not known, the forwarding state is (*, G). Entries in the multicast routing table are denoted as either (*, G), usually called a “star comma G” entry, or (S, G) usually called an “S comma G” entry. In a (*, G) entry, the * indicates any source and G is a specific multicast group address. In an (S, G) entry, S is the source address and G is the multicast group address.

CONFIGURING MULTICAST ROUTING ON NETSCREEN DEVICES

NetScreen devices have two predefined virtual routers (VRs): a trust-vr and an untrust-vr. Each virtual router is a separate routing component with its own route table populated by static routes or routes learned through a unicast routing protocol such as the Border Gateway Protocol (BGP) routing protocol, Open Shortest Path First (OSPF) protocol, or Routing Information Protocol (RIP) routing protocol. (For information on RIP, see [“Section 1 - Routing Information Protocol” on page 1](#). For information on virtual routers and unicast routing protocols, refer to the *NetScreen Concepts and Examples Guide: Volume 5*.) Each VR maintains a separate multicast route table for each multicast routing protocol. Each entry in the multicast route table contains the following information:

- IP address of the source
- Multicast group address
- Incoming interface
- List of outgoing zones, including the outgoing interfaces in each zone

Multicast Policies

By default, NetScreen devices do not permit multicast control traffic, such as IGMP or PIM-SM messages, to cross NetScreen devices. To permit multicast control traffic, you must configure a multicast policy that specifies the following:

- Source – The zone from which traffic initiates
- Destination – The zone to which traffic is sent
- Multicast group – Either the multicast group or an access list that specifies the multicast groups for which control traffic will be allowed

Note: Multicast policies control the flow of multicast control traffic only. To allow data traffic (both unicast and multicast) to pass between zones, you must configure policies. (For information about policies, refer to the *NetScreen Concepts & Examples ScreenOS Reference Guide, Volume 2*.)

In addition to filtering multicast control traffic between zones, you can also configure multicast policies to translate multicast addresses.

Access Lists

On NetScreen devices, you can secure multicast traffic by using access lists. An access list is a sequential list of statements that specify the forwarding status (permit or deny) of a host, route, or multicast group. (For additional information on access lists, refer to the *NetScreen Concepts and Examples Guide: Volume 2*.) In multicast routing, you use access lists to control the flow of multicast traffic. You can create an access list to restrict the groups that hosts can join or the sources from which traffic is received. Each of the following chapters detail how you can use access lists with each multicast protocol.

Generic Routing Encapsulation (GRE)

NetScreen devices support Generic Routing Encapsulation (GRE) which is a mechanism that encapsulates packets within a specified protocol. You can use GRE to forward multicast packets through non-multicast aware routers and devices. For additional information on GRE, refer to *RFC 1701, Generic Routing Encapsulation (GRE)*.

Multicast Configurations

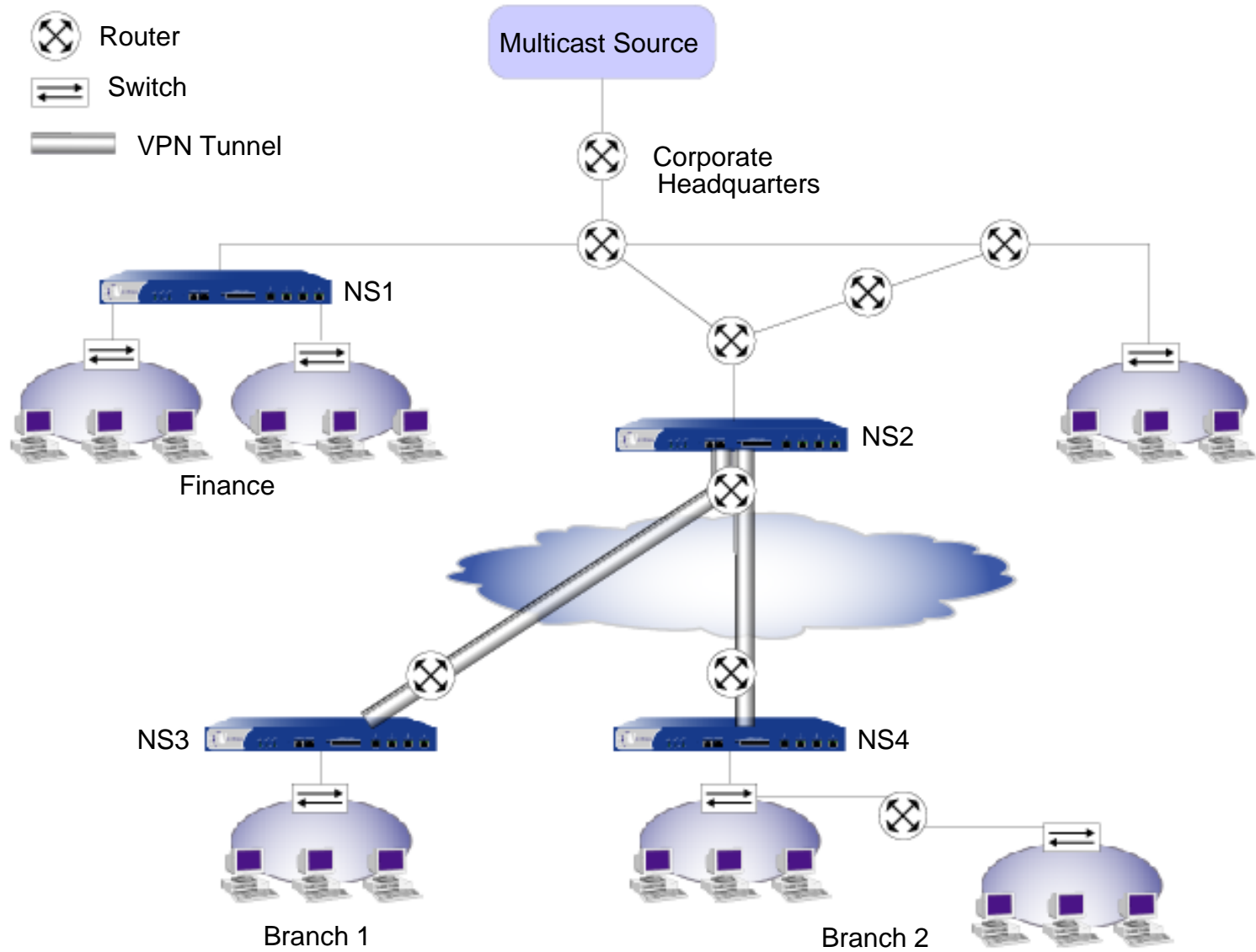
The following diagram illustrates a corporate network with the source at headquarters sending training videos to various departments and branch offices.

The NetScreen device NS1 is protecting the LAN of the Finance department which is receiving training videos from the source. The NetScreen device NS2 is protecting the corporate office and is connected to the two branch offices through VPN tunnels with NetScreen devices NS3 and NS4.

To configure multicast routing on a NetScreen device:

- Configure IGMP on interfaces connected to receivers.
- Configure PIM-SM on all NetScreen devices that pass multicast control traffic, or configure the IGMP Proxy feature.

The following chapters use this network diagram in the configuration examples.



IGMP

This chapter describes the Internet Group Management Protocol (IGMP) multicast protocol on NetScreen devices. It contains the following sections:

- “IGMP Overview” on page 70
 - “Maintaining Group Membership” on page 70
 - “Joining a Group” on page 71
 - “Leaving a Group” on page 72
- “Configuring IGMP on NetScreen Devices” on page 73
 - “Enabling IGMP on Interfaces” on page 73
 - “Security Considerations” on page 75
 - “Basic IGMP Configuration” on page 76
 - “Verifying Your IGMP Configuration” on page 79
 - “IGMP Operational Parameters” on page 82
- “IGMP Proxy” on page 84
 - “Multicast Routing Using IGMP Proxy” on page 86
 - “Configuring IGMP Proxy” on page 88
 - “Enabling IGMP Proxy on Interfaces” on page 88
 - “Creating a Multicast Policy” on page 89

IGMP OVERVIEW

The Internet Group Management Protocol (IGMP) multicast protocol is an intradomain protocol used between hosts and routers to establish and maintain multicast group memberships in a network. NetScreen devices support IGMPv1 as defined in *RFC 1112, Host Extensions for IP Multicasting*, and IGMPv2 as defined in *RFC 2236, Internet Group Management Protocol Version 2*. IGMPv2 expands on the functionality of IGMPv1 with the addition of a Querier selection process and Leave Group message (these are discussed in the following sections).

Hosts send IGMP messages to their local multicast routers when they wish to join a multicast group and begin receiving data for that group. If they support IGMPv2, they also send IGMP messages when they wish to leave the group and stop receiving data for that group.

Routers use IGMP to learn which groups have members on their local network. Routers listen for and send IGMP messages to their connected hosts only; they do not forward IGMP messages beyond their local network. Using the information obtained via IGMP, a router maintains a list of multicast group memberships on a per-interface basis. IGMP provides a mechanism for group membership only. Multicast routing protocols, such as PIM-SM, then process the membership information from IGMP to create entries in the multicast forwarding table and forward data to the hosts throughout the network.

The following sections describe how IGMP operates between hosts and routers.

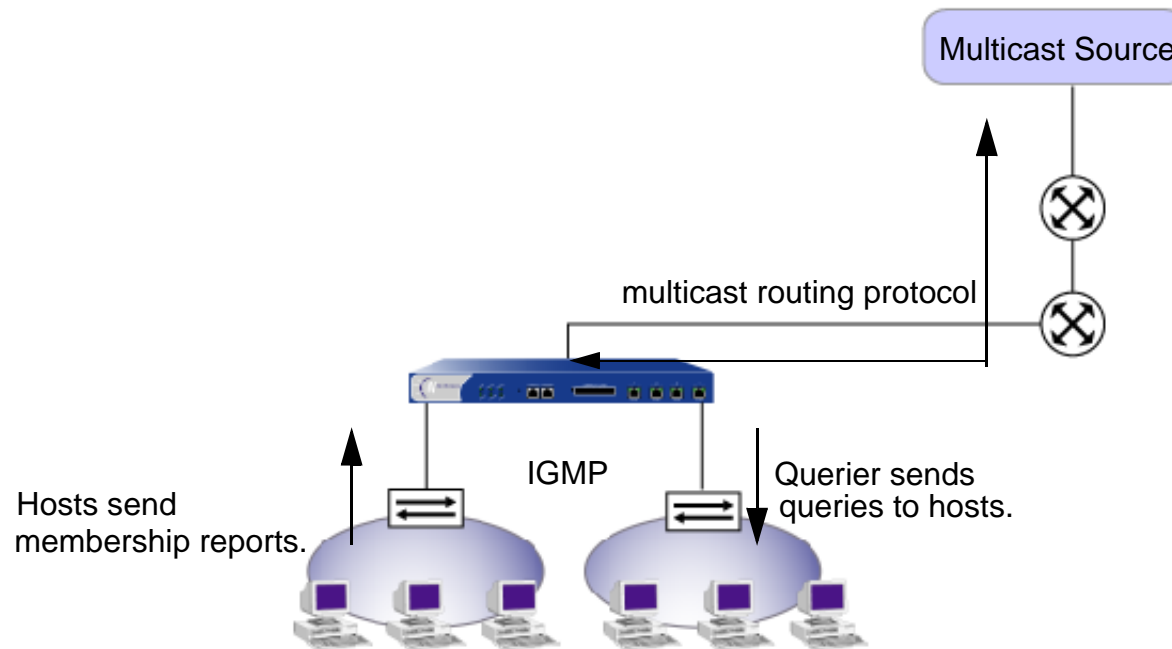
Maintaining Group Membership

Multicast routers maintain a list of multicast groups that have at least one member on each of their local networks. Each network selects a designated router, called the Querier¹. There is usually one Querier for each network. The Querier periodically transmits general queries to the “all hosts” group (224.0.0.1) in the network to solicit group membership information. If a host belongs to a group, it responds by periodically sending membership reports to the multicast group to which it belongs. Routers learn about the group memberships by listening to these IGMP messages on their local networks.

1. With IGMPv1, each multicast routing protocol determines the Querier for a network. With IGMPv2, the router interface with the lowest IP address in the network is the Querier.

Joining a Group

When a host wishes to join a multicast group, it sends a membership report to that group. When the multicast router on the local network receives the new membership report, it adds the group to its list of multicast group memberships. The router then uses a multicast routing protocol, such as PIM-SM, to join the group and begin forwarding multicast traffic to the host.



Leaving a Group

To leave a multicast group, a host running IGMPv1 simply stops sending membership reports to that group. When the router does not receive a Membership Report for a particular group within a specified time interval, the router assumes the group has no local members and stops forwarding multicast traffic for that group to its local network.

A host running IGMPv2 sends a Leave Group message to the “all routers group” (224.0.0.2) when it wishes to leave the multicast group. The Querier then sends a Group-Specific query to the multicast group that is being left to verify whether that particular group has any other members on its local network. If the Querier does not receive a response within a specified interval, then it assumes there are no more members for that group on its local network and stops forwarding multicast traffic for that group.

CONFIGURING IGMP ON NETSCREEN DEVICES

On NetScreen devices, you must explicitly enable IGMP and a multicast routing protocol. (On some routers, IGMP is automatically enabled when you enable a multicast routing protocol.) You must enable IGMP in router mode on the interfaces that are connected to hosts. When in router mode, the device runs IGMPv2 by default. (You can run IGMPv1 by specifying the **set interface *interface* protocol igmp version 1** command.) Then, you can define access lists to restrict multicast traffic to specific groups or hosts.

Enabling IGMP on Interfaces

IGMP is disabled by default on all interfaces. You must enable IGMP in router mode on all interfaces that are connected to hosts.

Example: Enabling IGMP on an Interface

In this example, you enable IGMP in router mode on the ethernet1 interface which is connected to a receiver.

CLI

```
set interface ethernet1 zone trust
set interface ethernet1 ip 10.1.1.1/24
set interface ethernet1 protocol igmp router
set interface ethernet1 protocol igmp enable
```

Example: Disabling IGMP on an Interface

In this example, you disable IGMP on the ethernet1 interface.

CLI

```
unset interface ethernet1 protocol igmp enable
```

Security Considerations

There are some security issues you must consider when running IGMP. Malicious users can forge IGMP queries, membership reports and leave messages. On NetScreen devices, you can use access lists to restrict multicast traffic to known hosts and multicast groups only. (For additional information on access lists, refer to the *NetScreen Concepts and Examples Guide: Volume 2*).

You can use access lists to do the following:

- restrict the groups that the hosts on the specified interface can join
- specify from which hosts the IGMP router interface can receive join and leave messages

Example: Configuring an Access List for Accepted Groups

In this example, you create an access list that specifies the multicast group 224.4.4.1/32. Then you specify that the hosts on ethernet1 can join only the multicast group specified in the access list.

CLI

```
set vrouter trust access-list 1 permit ip 224.4.4.1/32 1
set interface ethernet1 protocol igmp accept groups 1
```

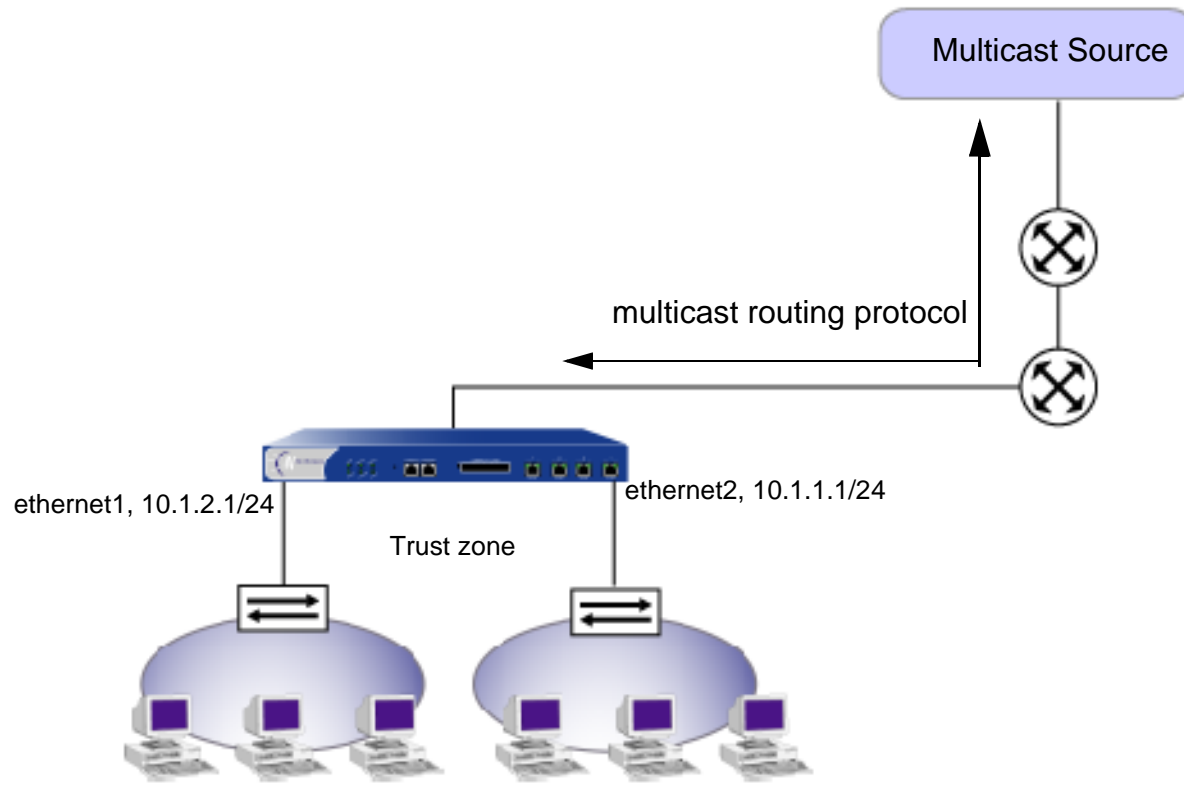
Basic IGMP Configuration

To run IGMP on a NetScreen device, you simply enable it in router mode on the interfaces that are directly connected to hosts. To ensure the security of your network, you can use access lists to limit multicast traffic to known multicast groups or hosts.

Example: Basic IGMP Configuration

In this example, the hosts that are connected to the NetScreen device NS1 are potential receivers of the multicast stream from the source in the corporate office. The multicast source is transmitting data to the multicast group 224.4.4.1. Perform the following steps to configure IGMP on the interfaces that are connected to the hosts:

1. Assign IP addresses to the interfaces and bind them to zones.
2. Enable IGMP in router mode.
3. Create an access list that specifies the multicast group 224.4.4.1/32.
4. Restrict the hosts to joining the multicast group 224.4.4.1/32 only.



CLI

Zones and Interfaces

```
set interface ethernet1 zone trust
set interface ethernet1 ip 10.1.2.1/24
set interface ethernet1 protocol igmp router
set interface ethernet1 protocol igmp enable
```

```
set interface ethernet2 zone trust
set interface ethernet2 ip 10.1.1.1/24
set interface ethernet2 protocol igmp router
set interface ethernet2 protocol igmp enable
```

Access List

```
set vrouter trust access-list 1 permit ip 224.4.4.1/32 1
set interface ethernet1 protocol igmp accept groups 1
set interface ethernet2 protocol igmp accept groups 1
save
```

After you configure IGMP on ethernet1 and ethernet2, you must configure a multicast routing protocol, such as PIM-SM, to forward multicast traffic. (For information on PIM-SM, see [“PIM-SM” on page 97.](#))

Verifying Your IGMP Configuration

To verify connectivity and ensure that IGMP is running properly, there are a number of **exec** and **get** commands that you can use. When you specify one of the following commands, the device sends an IGMP message as follows:

- To send either general queries or group-specific queries on a particular interface, use the **exec igmp interface *interface* query** command.

For example, to send a general query from ethernet2 enter:

```
exec igmp interface ethernet2 query
```

For example, to send a group-specific query from ethernet2 to the multicast group 224.4.4.1, enter:

```
exec igmp interface ethernet2 query 224.4.4.1
```

- To send a membership report on a particular interface, use the **exec igmp interface *interface* report** command. For example, to send a membership report from ethernet2, enter:

```
exec igmp interface ethernet2 report 224.4.4.1
```

You can review the IGMP parameters of an interface by entering the following command:

```
ns208(M)-> get igmp interface
```

```
Interface ethernet1:1 support IGMP version 2 router. It is enabled.  
It is not elected IGMP DRP interface within same subnet.  
IGMP proxy is disabled.  
IGMP packets without router alert IP option will not be dropped.  
Querier has not been found yet. I am the non-querier.  
There are 1 multicast groups active.  
  Inbound Router access list number: not set  
  Inbound Host access list number: not set  
  Inbound Group access list number: not set  
  query-interval: 125 seconds  
  query-max-response-time 10 seconds  
  leave-interval 1 seconds  
  last-member-query-interval 1 seconds
```

```
Interface ethernet1 support IGMP version 2 router. It is enabled.  
It is elected IGMP DRP interface within same subnet.  
IGMP proxy is disabled.  
IGMP packets without router alert IP option will not be dropped.  
Querier IP is 1.0.0.7, it has up 3035 seconds. I am the querier.  
There are 0 multicast groups active.  
  Inbound Router access list number: not set  
  Inbound Host access list number: not set  
  Inbound Group access list number: not set  
  query-interval: 125 seconds  
  query-max-response-time 10 seconds  
  leave-interval 1 seconds  
  last-member-query-interval 1 seconds
```

To display information about multicast groups, enter the following CLI command:

```
ns208(M)-> get igmp gr
total groups matched: 17
multicast group  interface  last reporter  expire ver
*224.2.156.189   ethernet1:1    0.0.0.0        ----- v2
224.5.5.1        ethernet2      2.0.0.1        260s v2
224.5.5.2        ethernet2      2.0.0.1        260s v2
224.5.5.3        ethernet2      2.0.0.1        260s v2
224.5.5.4        ethernet2      2.0.0.1        260s v2
224.5.5.5        ethernet2      2.0.0.1        260s v2
224.5.5.6        ethernet2      2.0.0.1        260s v2
224.5.5.7        ethernet2      2.0.0.1        260s v2
224.5.5.8        ethernet2      2.0.0.1        260s v2
224.5.5.9        ethernet2      2.0.0.1        260s v2
224.5.5.10       ethernet2      2.0.0.1        260s v2
224.5.5.11       ethernet2      2.0.0.1        260s v2
224.5.5.12       ethernet2      2.0.0.1        260s v2
224.5.5.13       ethernet2      2.0.0.1        260s v2
224.5.5.14       ethernet2      2.0.0.1        260s v2
224.5.5.15       ethernet2      2.0.0.1        260s v2
224.5.5.16       ethernet2      2.0.0.1        260s v2
```

IGMP Operational Parameters

When you enable IGMP in router mode on an interface in the NetScreen device, the interface starts up as a Querier. As the Querier, the interface uses certain defaults which you can change. When you set parameters on this level, it affects only the interface that you specified.

The following table describes the IGMP Querier interface parameters and their defaults.

IGMP Interface Parameters	Description	Default Value
General query interval	The interval at which the Querier interface sends general queries to the "all hosts" group (224.0.0.1).	125 seconds
Maximum response time	The maximum time between a general query and a response from the host.	10 seconds
Last Member Query Interval	The interval at which the interface sends a Group-Specific query. If it does not receive a response after the second Group-Specific query, then it assumes there are no more members for that group on its local network.	1 second

If the Querier hears a query message from another router with a lower IP address, then it becomes a non-Querier.

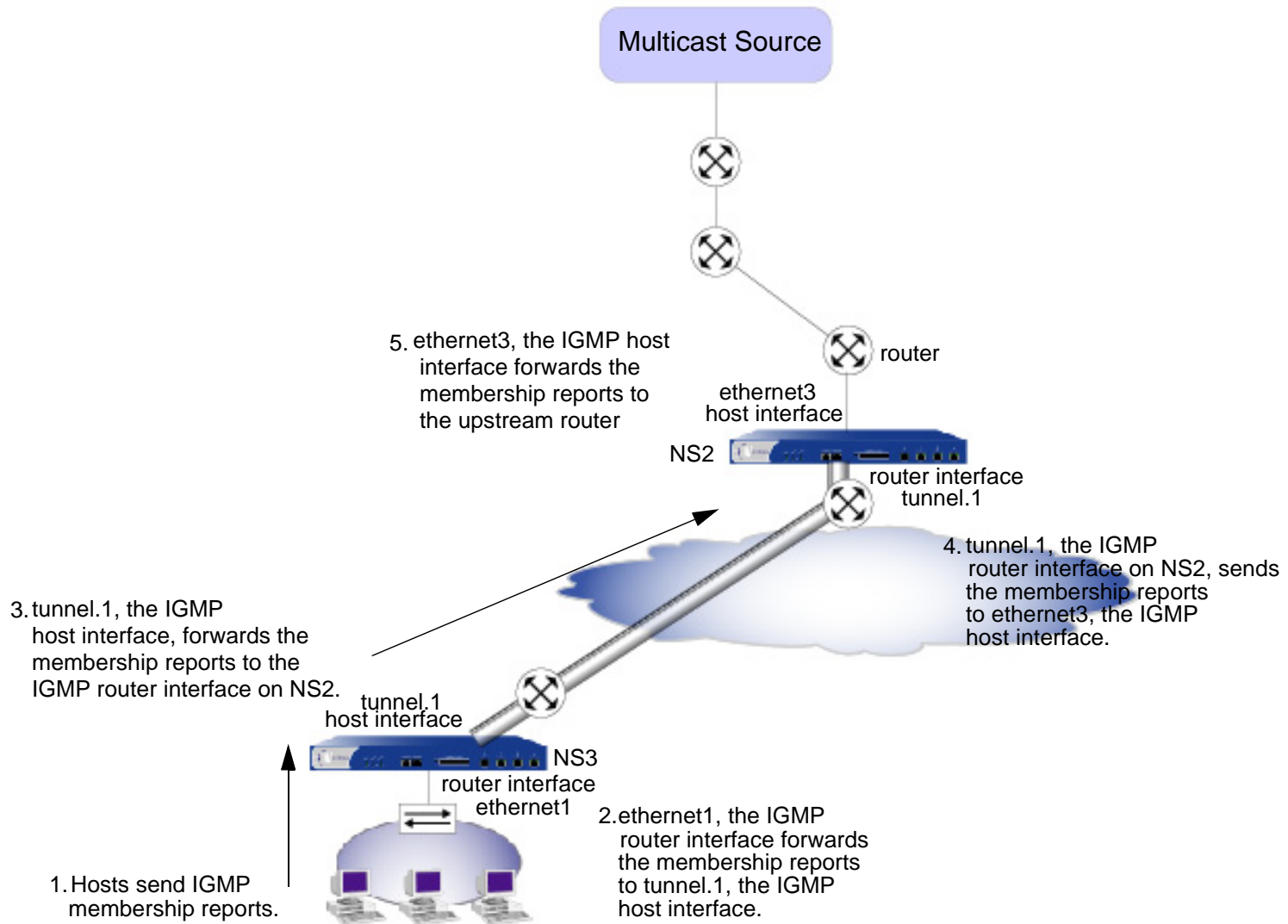
In addition, the IGMP protocol uses the following mechanisms:

- By default, an IGMP interface accepts IGMP messages only from its own subnet. It ignores IGMP messages from external sources. You can enable the NetScreen device to accept IGMP messages from all sources by entering the **set interface *interface* protocol igmp no-check-subnet** command. Use this option if the NetScreen device is running IGMP proxy or RP proxy. (For information on IGMP proxy, see [“IGMP Proxy” on page 84](#). For information on RP proxy, see [“Proxy RP” on page 123](#).)
- By default, an IGMPv2-enabled router accepts only IGMP packets with a router-alert IP option, and drops packets that do not have this option. Therefore, a NetScreen device running IGMPv2 drops IGMPv1 packets by default because IGMPv1 packets do not have this option. You can configure the NetScreen device to stop checking IGMP packets for the router-alert IP option and accept all IGMP packets, allowing backward compatibility with IGMPv1 routers. To enable this feature, use the **set interface *interface* protocol igmp no-check-router-alert** command.

IGMP PROXY

IGMP proxy allows a NetScreen device to extend the scope of a multicast domain by one hop without the CPU overhead of a multicast routing protocol. When you enable IGMP proxy on a device, the interface connected to the hosts (downstream interface) functions as a multicast router, and the interface connected to the upstream router functions as an IGMP host.

The following diagram illustrates how you can use IGMP proxy to forward multicast traffic. The NetScreen devices NS2 and NS3 are connected to each other through a VPN tunnel. On NS3, the downstream interface (ethernet1) is in router mode and the upstream interface (tunnel.1) is in IGMP host mode. On NS2, the interface on the other end of the tunnel, which is also the downstream interface, is in router mode; the upstream interface connected to the upstream router (ethernet 3) is in IGMP host mode.



Multicast Routing Using IGMP Proxy

This section describes how a NetScreen device creates a multicast distribution tree while running IGMP proxy. (For information about multicast distribution trees, see [“Multicast Distribution Trees” on page 63.](#))

Sending Membership Reports Upstream to the Source

When a host connected to a router interface on a NetScreen device joins a multicast group, it sends a membership report to the multicast group. When the router interface receives the membership report from the attached host, it checks if it has an entry for the multicast group.

- If the router interface has an entry for the multicast group, it ignores the membership report.
- If the router interface does not have an entry for the multicast group, it checks if there is a multicast policy for the group that specifies to which zone(s) the router interface should send the report.
 - If there is no multicast policy for the group, the router interface does not forward the report.
 - If there is a multicast policy for the group, the router interface creates an entry for the multicast group and forwards the membership report to the proxy host interfaces specified in the multicast policy.

When a proxy host interface receives the Membership Report, it checks if it has a (*, G) entry for the group.

- If it has a (*, G) entry for the group, the host interface adds the router interface to the list of outgoing interfaces for the entry.
- If it does not, it creates a (*, G) entry for that group; the incoming interface is the proxy host interface and the outgoing interface is the router interface. Then, the proxy host interface forwards the report to its upstream router.

Sending Multicast Data Downstream to the Receivers

When the router interface on the NetScreen device receives the multicast data, it checks if there is an existing session for the group.

- If there is a session for the group, the interface forwards the multicast data based on the session information.
- If there is no session for the group, the interface checks if the group has an (S, G) entry in the multicast route table.
 - If there is an (S, G) entry, the interface forwards the multicast data accordingly.
 - If there is no (S, G) entry, it checks if there is a (*, G) entry for the group.
 - If there is no (*, G) entry for the group, the interface drops the packet.
 - If there is a (*, G) entry for the group, the interface creates an (S, G) entry. When the interface receives subsequent multicast packets for that group, it forwards the traffic to the router interface (the outgoing interface), which in turn, forwards the traffic to its connected host.

Configuring IGMP Proxy

This section describes the basic steps required to configure IGMP proxy on a NetScreen device:

1. Enable IGMP in host mode on the upstream interfaces. IGMP proxy is enabled by default on host interfaces.
2. Enable IGMP in router mode on the downstream interfaces.
3. Enable IGMP proxy on the router interfaces.
4. Configure a multicast policy that allows multicast control traffic to pass between zones.

Enabling IGMP Proxy on Interfaces

When you run IGMP proxy on a NetScreen device, you configure the downstream interface in router mode and the upstream interface in host mode. (Note that an interface can either be in host mode or router mode, not both.) Additionally, in order for a router interface to forward multicast traffic, it must be the Querier in the local network. To allow a non-Querier interface to forward multicast traffic, you must specify the keyword **always** when you enable IGMP on a router interface.

Example: Enabling IGMP on Interfaces

In this example, the interface ethernet1 is connected to the upstream router and the interface ethernet3 is connected to the hosts. Therefore, you enable IGMP in host mode on ethernet1 and enable IGMP in router mode on ethernet3.

CLI

```
set interface ethernet1 protocol igmp host
set interface ethernet1 protocol igmp enable
set interface ethernet3 protocol igmp router
set interface ethernet3 protocol igmp proxy always
set interface ethernet3 protocol igmp enable
```

Example: Disabling IGMP on Interfaces

In this example, you disable IGMP on interface ethernet3.

CLI

```
unset interface ethernet3 protocol igmp router
unset interface ethernet3 protocol igmp proxy
unset interface ethernet3 protocol igmp enable
```

Creating a Multicast Policy

As stated in the previous section, NetScreen devices do not allow multicast control traffic, such as IGMP messages, to cross NetScreen devices. Therefore, if you bind the host and router interfaces to different zones, you must configure multicast policies that permit the multicast control traffic between zones. Note that multicast policies apply to multicast control messages only. To allow unicast and multicast data traffic to pass between zones, you must configure policies. (For information about policies, refer to the *NetScreen Concepts & Examples ScreenOS Reference Guide: Volume 2*.)

Example: Creating a Multicast Policy for IGMP Messages

In this example, you define a bi-directional multicast policy that allows IGMP messages between the Trust and Untrust zones

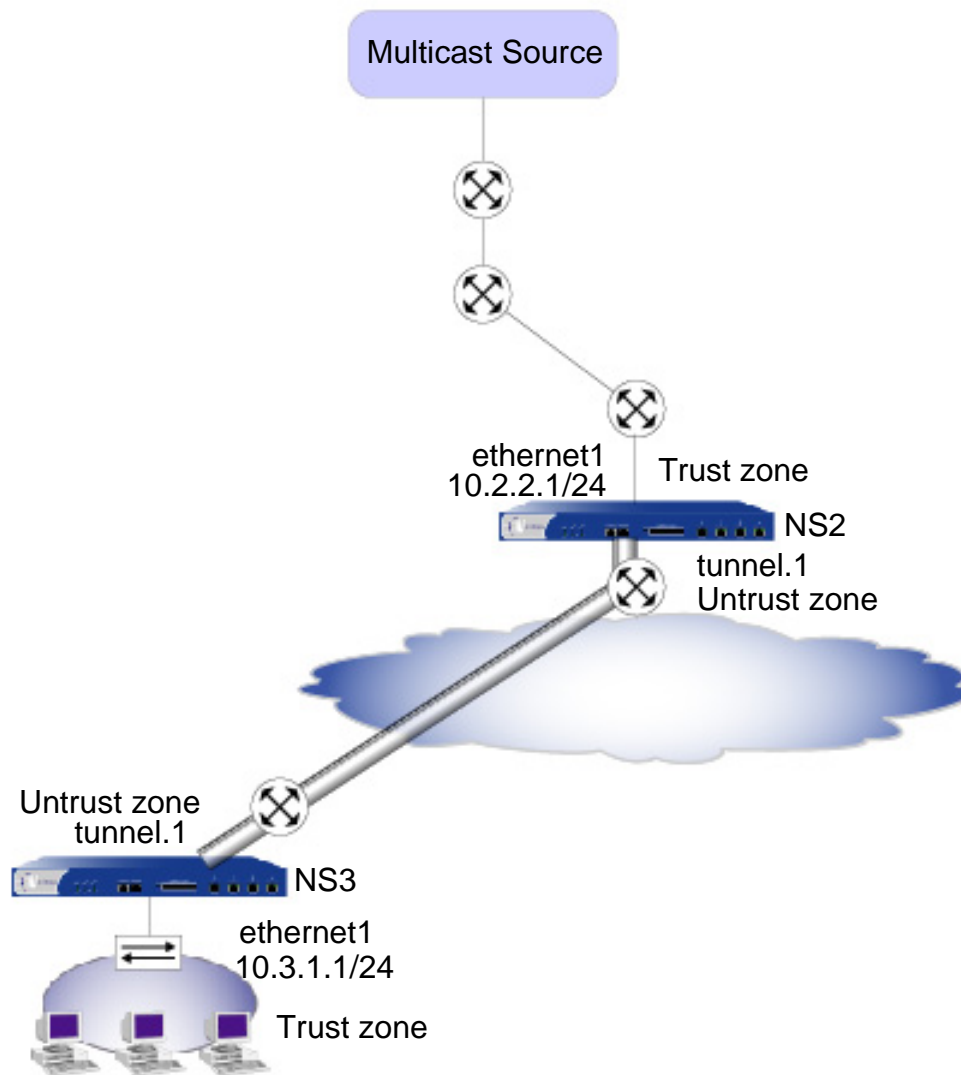
CLI

```
set multicast-group-policy from trust mgroup 224.2.202.99/32 to untrust
  igmp-message bi-directional
save
```

Example: Basic IGMP Proxy Configuration

In this example, you configure IGMP proxy on the NetScreen devices NS2 and NS3. They are connected to each other through a VPN tunnel. In addition, the devices are using GRE to encapsulate the packets. Perform the following steps on the NetScreen devices at both locations:

1. Assign IP addresses to the physical interfaces bound to the security zones.
2. Enable IGMP on the host and router interfaces.
3. Enable IGMP proxy on the router interface. (IGMP proxy is enabled by default on host interfaces.)
 - By default, an IGMP interface accepts IGMP packets from its own subnet only. In the example, the interfaces are on different subnets. When you enable IGMP, allow the interfaces to accept IGMP packets (queries, membership reports, and leave messages) from any subnet.
4. Set up policies for traffic to pass between each site.
 - Configure a policy to pass data traffic between zones.
 - Configure a multicast policy to pass multicast control traffic between zones. To pass IGMP messages between zones, you must use the **igmp-message** keyword. In this example, you restrict multicast traffic to one multicast group (224.4.4.1/32).



BRANCH - NS3

Interfaces – Zones

```
set interface ethernet1 zone trust
set interface ethernet1 ip 10.3.1.1/24

set interface ethernet3 zone untrust
set interface ethernet3 ip 3.1.1.1/24

set interface tunnel.1 zone untrust
set interface tunnel.1 ip unnumbered interface ethernet3
```

Addresses

```
set address trust mgroup1 224.4.4.1/32
set address untrust source-dr 10.2.2.1/24
```

IGMP

```
set interface ethernet1 protocol igmp router
set interface ethernet1 protocol igmp proxy always
set interface ethernet1 protocol igmp enable
set interface tunnel.1 protocol igmp host
set interface tunnel.1 protocol igmp enable
set interface tunnel.1 protocol igmp no-check-subnet
```

Routes

```
set route 10.2.2.0/24 interface tunnel.1
```

GRE Encapsulation

```
set interface tunnel.1 tunnel encap gre
set interface tunnel.1 tunnel local-if ethernet3 dst-ip 2.2.2.2
```

VPN Tunnel

```
set ike gateway To_Corp address 2.2.2.2 main outgoing-interface ethernet3
  preshare fg2g4h5j proposal pre-g2-3des-sha
set vpn Branch2_Corp gateway To-Corp sec-level compatible
set vpn Branch2_Corp bind interface tunnel.1
set vpn Branch2_Corp proxy-id local-ip 10.3.1.0/24 remote-ip 10.2.2.0/24 any
```

Policies

```
set policy from untrust to trust source-dr mgroup1 any permit
set multicast-group-policy from trust mgroup 224.4.4.1/32 to untrust
  igmp-message bi-directional
save
```

CORP - NS2

Interfaces – Zones

```
set interface ethernet1 zone trust
set interface ethernet1 ip 10.2.2.1/24

set interface ethernet3 zone untrust
set interface ethernet3 ip 2.2.2.2/24

set interface tunnel.1 zone untrust
set interface tunnel.1 ip unnumbered interface ethernet3
```

Addresses

```
set address untrust branch1 10.3.1.0/24
```

IGMP

```
set interface ethernet1 protocol igmp host
set interface ethernet1 protocol igmp enable
set interface tunnel.1 protocol igmp router
set interface tunnel.1 protocol igmp proxy always
set interface tunnel.1 protocol igmp enable
set interface tunnel.1 protocol igmp no-check-subnet
```

Routes

```
set route 10.3.1.0/24 interface tunnel.1
```

GRE Encapsulation

```
set interface tunnel.1 tunnel encap gre
set interface tunnel.1 tunnel local-if ethernet3 dst-ip 3.1.1.1
```

VPN Tunnel

```
set ike gateway To_Branch2 address 3.1.1.1 main outgoing-interface ethernet3
  preshare fg2g4h5j proposal pre-g2-3des-sha
set vpn Corp_Branch2 gateway To-Branch2 sec-level compatible
set vpn Corp_Branch2 bind interface tunnel.1
set vpn Corp_Branch2 proxy-id local-ip 10.2.2.0/24 remote-ip 10.3.1.0/24 any
```

Policies

```
set policy name To-Corp from untrust to trust branch2 any any permit
set multicast-group-policy from trust mgroup 224.4.4.1/32 to untrust
  igmp-message bi-directional
save
```


PIM-SM

This chapter describes the Protocol Independent Multicast-Sparse Mode (PIM-SM) protocol on NetScreen devices. It includes the following sections:

- [“Overview of PIM” on page 98](#)
 - [“Designated Router” on page 99](#)
 - [“Forwarding Traffic on the Distribution Tree” on page 100](#)
 - [“Mapping RPs to Groups” on page 99](#)
- [“Configuring PIM on NetScreen Devices” on page 104](#)
 - [“Creating and Enabling a PIM Instance in a Virtual Router” on page 104](#)
 - [“Enabling PIM on Interfaces” on page 105](#)
 - [“Creating a Multicast Group Policy” on page 106](#)
- [“Basic PIM Configuration” on page 107](#)
- [“Configuring RP to Group Mappings” on page 114](#)
- [“Security Considerations” on page 116](#)
- [“PIM Interface Parameters” on page 120](#)
- [“Proxy RP” on page 123](#)

OVERVIEW OF PIM

Protocol Independent Multicast (PIM) is an intradomain multicast routing protocol that runs between routers. Whereas the Internet Group Management Protocol (IGMP) runs between hosts and routers to exchange group membership information, PIM runs between routers to forward multicast traffic to the multicast group members throughout the network.

PIM is called protocol independent because it uses the route table of the underlying unicast routing protocol to perform its RPF (reverse path forwarding) checks, but does not depend on the functionality of the unicast routing protocol. (For information about RPF, see [“Reverse Path Forwarding” on page 63](#).)

PIM can operate in two modes: Dense-Mode or Sparse-Mode. PIM-Dense Mode (PIM-DM) floods multicast traffic throughout the network, then prunes routes to receivers that do not want to receive the multicast traffic. PIM-Sparse Mode (PIM-SM) forwards multicast traffic only to those receivers that request it. NetScreen devices support PIM-SM, as defined in <http://www.ietf.org/internet-drafts/draft-ietf-pim-sm-v2-new-08.txt>.

PIM-SM can use either a shared distribution tree or the shortest path tree (SPT) to forward multicast traffic throughout the network. (For information about multicast distribution trees, see [“Multicast Distribution Trees” on page 63](#).) By default, PIM-SM uses the shared distribution tree with an RP (Rendezvous Point) at the root of the tree. All sources in a group send their packets to the RP, and the RP sends data down the shared distribution tree to all receivers in a network. When a configured threshold is reached, the receivers can form an SPT to the source, decreasing the time it takes the receivers to get the multicast data. Regardless of which tree is used to distribute traffic, only receivers that explicitly join a multicast group can receive the traffic for that group.

Note: NetScreen devices support PIM-SM only, not PIM-DM. Therefore, in this chapter, the term PIM always refers to PIM-SM.

Designated Router

When there are multiple PIM routers in a multi-access LAN, the routers elect a designated router (DR). The DR on the LAN of the source is responsible for sending the multicast packets from the source to the RP and to the receivers that are on the source-specific distribution tree. The DR on the LAN of the receivers is responsible for forwarding join-prune messages from the hosts to the RP, and for sending multicast data traffic down to the hosts in the LAN.

The DR is selected through an election process. You can set the DR priority of each PIM router in a LAN. The PIM router advertises its DR priority in the Hello messages it periodically sends its neighbors. When the PIM routers receive the Hello messages, they select the router with the highest DR priority as the DR for the LAN. If multiple routers have the highest DR priority, then the router with the highest IP address becomes the DR of the LAN.

Mapping RPs to Groups

A PIM-SM domain is a group of PIM routers that have the same RP-to-group mappings. All sources in the domain send packets to the RP and the RP forwards the multicast traffic to all receivers in the multicast group. There are two ways to map groups to an RP: statically and dynamically.

Static RP Mapping

You can create a static mapping between an RP and a multicast group. To do so, you must configure the RP for the multicast group on each router in the network. Each time the address of the RP changes, you must re-configure the RP address.

Dynamic RP Mapping

PIM also provides a mechanism through which RPs are dynamically mapped to multicast groups. First, you configure routers as candidate rendezvous points (C-RPs) for each multicast group. Then the C-RPs send Candidate-RP advertisements to one router in the LAN, called the bootstrap router (BSR). The advertisements contain the multicast group(s) for which the router is to be an RP and the priority of the C-RP.

The BSR collects these C-RP advertisements and sends them out in a BSR message to all routers in the domain. The routers collect these BSR messages and use a well-known hash algorithm to select one active RP per multicast group. If the selected RP fails, then the router selects a new RP-group mapping from among the candidate RPs.

For information on the BSR selection process, refer to the RFC 2362.

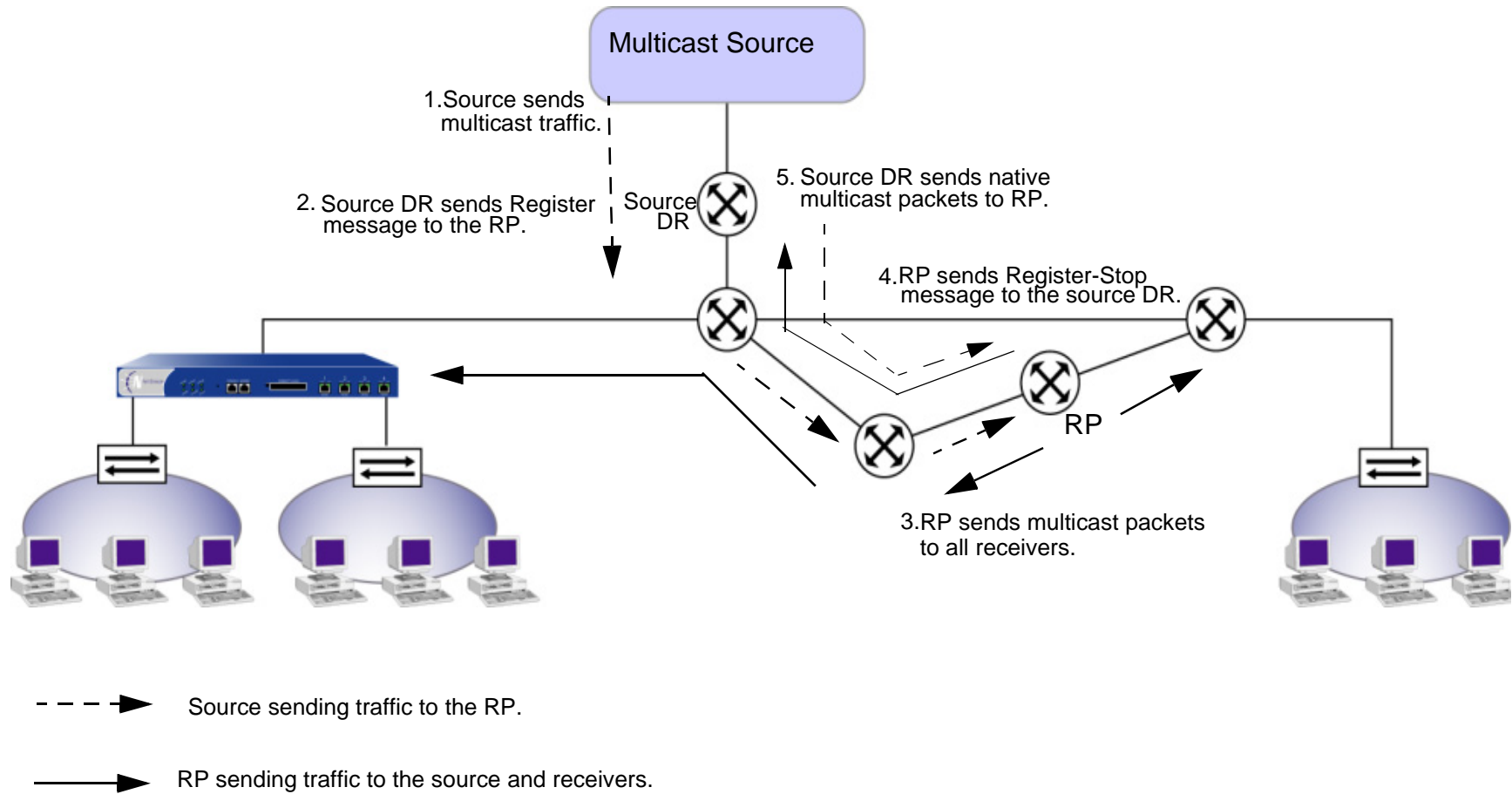
Forwarding Traffic on the Distribution Tree

This section describes how PIM routers send a join message towards the RP of a multicast group and how the RP sends multicast data to the receivers in the network.

Source Sends Data to a Group

When a source starts sending multicast packets, it sends the packets to the network. When the DR on that LAN receives the multicast packets, it encapsulates the multicast packets in unicast packets and sends the encapsulated packets to the RP. These encapsulated packets are called Register messages and indicate that there is a new source in the LAN. When the RP receives the register messages, it decapsulates the packets and sends the multicast packets down the distribution tree towards the receivers.

If the data rate from the source DR reaches a configured threshold, the RP sends a PIM join message towards the source DR so the RP can receive the native multicast data, instead of the encapsulated data. (Note that a NetScreen device functioning as an RP sends the PIM join message to the source DR after it receives the first data packet from the source.) When the source DR receives the join message, it sends multicast packets towards the RP along with the Register messages. When the RP receives the multicast packets from the DR, it sends the DR a Register-Stop message. When the DR receives the Register-Stop message, it stops sending the encapsulated messages and instead sends the native multicast traffic, which the RP then sends downstream to the receivers.

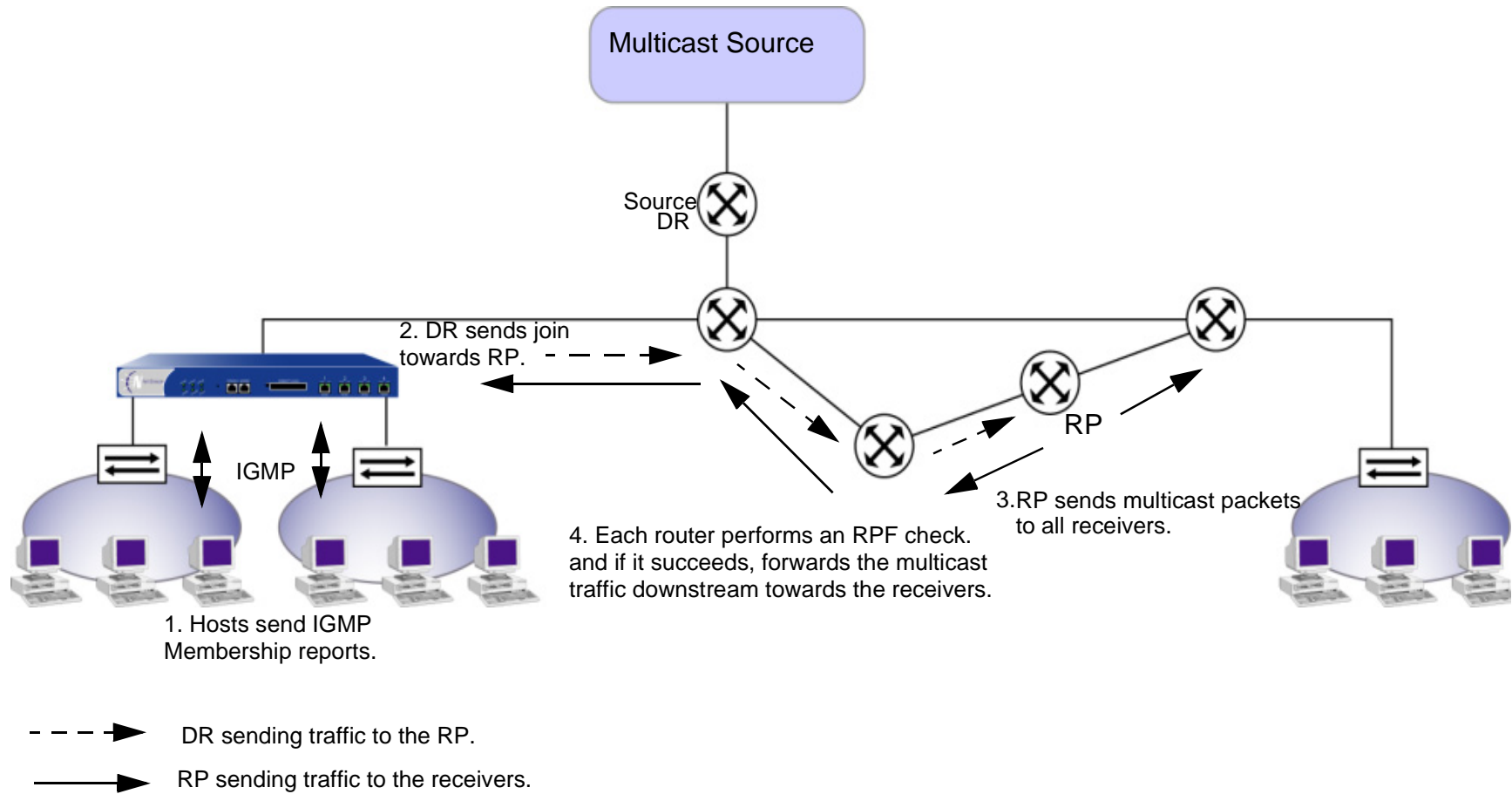


Host Joins a Group

When a host wants to join a multicast group, it sends an IGMP join message to that multicast group. When the DR on the LAN receives the IGMP join message, it looks up the RP for the group. It creates a (*,G) entry in its PIM route table and sends a PIM join message upstream towards the RP. When the upstream router receives the PIM join message, it creates a (*,G) in its forwarding table, if it does not have an entry for that group. The router then checks the unicast route table to determine the best path towards the RP and forwards the PIM join message towards the RP. This process continues until the PIM join message reaches the RP. When the RP receives the join message, it sends the multicast data downstream towards the receiver.

Each downstream router performs an RPF check when it receives the multicast data. Each router checks if it received the multicast packets from the same interface it uses to send traffic towards the RP. If the RPF check is successful, the router then looks for a matching (*,G) forwarding entry. If it finds the (*,G) entry, it places the source in the entry, which becomes an (S,G) entry, and forwards the multicast packets downstream. This process continues down the distribution tree until the host receives the multicast data.

When the traffic rate reaches a configured threshold, the DR on the LAN of the host can form a source-specific tree directly to the multicast source. When the DR starts receiving traffic directly from the source, it sends a source-specific prune message upstream towards the RP. Each intermediate router “prunes” the link to the host off the distribution tree, until the prune message reaches the RP which then stops sending the multicast traffic down that particular branch of the distribution tree.



CONFIGURING PIM ON NETSCREEN DEVICES

As stated in previous sections, PIM uses the unicast routes to perform its RPF checks. Therefore, in order to run PIM, you must first configure either static routes or a dynamic routing protocol such as OSPF, BGP, or RIP. (For information on RP, see [“Section 1 - Routing Information Protocol” on page 1](#). For information on unicast routing protocols, refer to the *NetScreen Concepts and Examples Guide: Volume 5*.)

This section describes the following basic steps that are required to configure PIM on a NetScreen device:

- Create and enable a PIM instance.
- Enable PIM on interfaces.
- Configure a multicast policy to allow PIM messages to cross the NetScreen device.

Creating and Enabling a PIM Instance in a Virtual Router

NetScreen devices have two predefined virtual routers (VRs): a trust-vr and an untrust-vr. Each virtual router is a separate routing component with its own route table which is populated by static routes or routes learned through a unicast routing protocol such as BGP, OSPF or RIP.

You can configure one PIM instance for each VR. The PIM instance on the VR uses the unicast route table to perform its RPF check. After you create and enable a PIM routing instance on a VR, you can then enable PIM on the interfaces in the VR.

Example: Creating and Enabling a PIM Instance in a Virtual Router

In this example, you first create a PIM instance on the trust-vr virtual router, and then enable that PIM instance.

CLI

```
set vrouter trust
set vrouter trust protocol pim
set vrouter trust protocol pim enable
save
```

Example: Removing a PIM Instance

In this example, you delete the PIM instance on the trust-vr virtual router. When you delete the PIM instance, PIM stops processing PIM packets on all PIM-enabled interfaces.

CLI

```
unset vrouter trust protocol pim enable
unset vrouter trust protocol pim

deleting PIM instance, are you sure? y/[n] y
save
```

Enabling PIM on Interfaces

PIM is disabled by default on all interfaces. After you create and enable PIM on a virtual router, you must enable PIM on the interfaces that transmit the multicast traffic. If an interface is connected to a receiver, you must also configure IGMP in router mode on that interface. (For information on IGMP, see [“Configuring IGMP on NetScreen Devices” on page 73.](#))

Example: Enabling PIM on an Interface

In this example, you enable PIM on the ethernet1 interface.

CLI

```
set interface ethernet1 protocol pim
set interface ethernet1 protocol pim enable
```

Example: Disabling PIM on an Interface

In this example, you disable PIM on the ethernet1 interface. Note that any other interfaces on which you have enabled PIM are still transmitting and processing PIM packets.

CLI

```
unset interface ethernet1 protocol pim enable
```

Creating a Multicast Group Policy

As stated in previous sections, a NetScreen device does not allow multicast control messages to cross zones. You must configure a multicast policy to allow PIM messages between zones.

Example: Creating a Multicast Group Policy

In this example, you define a bi-directional multicast group policy that allows BSR and Join-Prune messages between the Trust and Untrust zones for group 224.4.4.1.

CLI

```
set multicast-group-policy from trust mgroup 224.4.4.1/32 to untrust pim-message bsr
    join-prune bi-directional
save
```

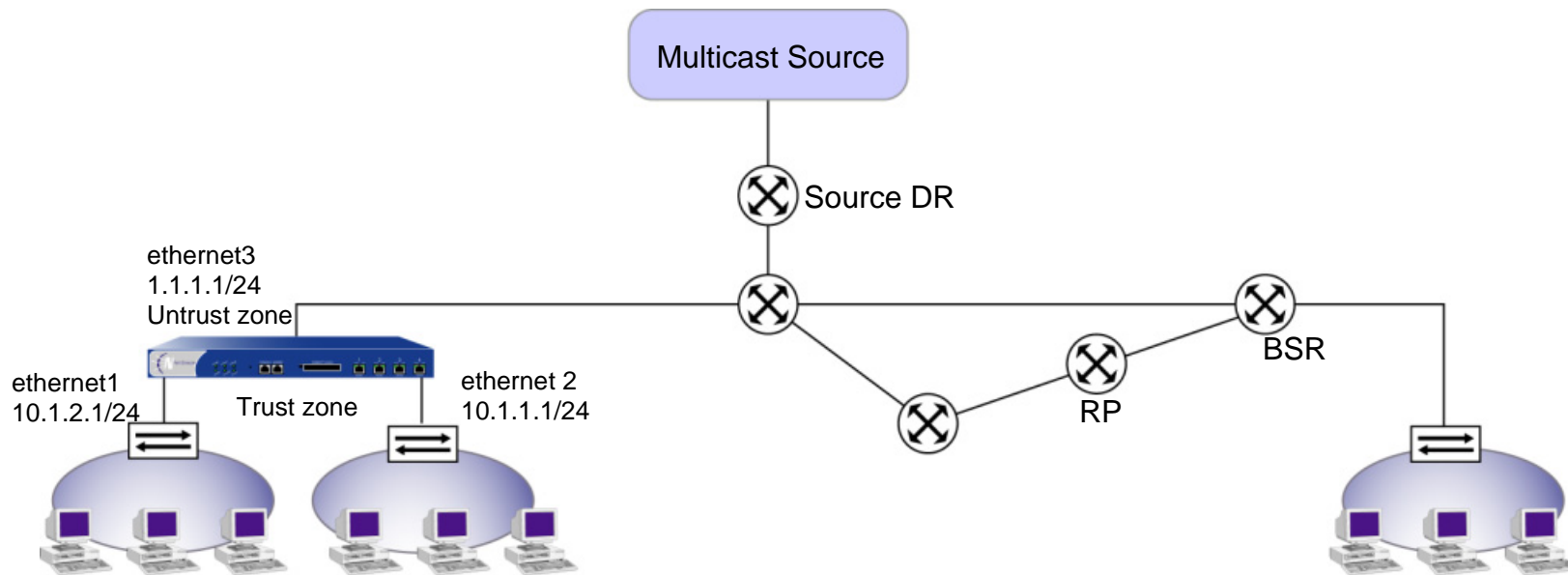
BASIC PIM CONFIGURATION

Perform the following steps to configure PIM on a NetScreen device:

1. Configure zones and interfaces.
2. Configure either a static route or a dynamic routing protocol such as RIP, BGP or OSPF.
3. Create a policy to pass unicast and multicast traffic between the zones.
4. Create and enable a PIM routing instance.
5. Enable PIM on the interfaces connected to the LAN and to the upstream router.
6. Configure a multicast policy to permit PIM messages between zones.

Example: Basic PIM Configuration

In this example, the hosts connected to the NetScreen device are to receive the multicast stream for the multicast group 224.4.4.1/32. You configure RIP as the unicast routing protocol and create a policy to pass data traffic between the trust and untrust zones. You create a PIM instance on the trust-vr and enable PIM on ethernet1 and ethernet2 in the Trust zone, and on ethernet3 in the Untrust zone. ethernet 1 and ethernet2 are connected to the potential receivers, therefore you also configure IGMP on these interfaces. You then create a multicast policy that permits BSR and join-prune messages between the zones.



CLI

Zones and Interfaces

```
set interface ethernet 1 zone trust
set interface ethernet 1 ip 10.1.2.1/24
set interface ethernet 1 protocol igmp router
set interface ethernet 1 protocol igmp enable

set interface ethernet 2 zone trust
set interface ethernet 2 ip 10.1.1.1/24
set interface ethernet 2 protocol igmp router
set interface ethernet 2 protocol igmp enable

set interface ethernet 3 zone untrust
set interface ethernet 3 ip 1.1.1.1/24
```

Access List

```
set vrouter trust access-list 1 permit ip 224.4.4.1/32 1
set interface ethernet1 protocol igmp accept groups 1
set interface ethernet2 protocol igmp accept groups 1
```

RIP

```
set vrouter trust router-id 10
set vrouter trust protocol rip
set vrouter trust protocol rip enable
set interface ethernet3 protocol rip enable
```

PIM-SM

```
set vrouter trust protocol pim
set vrouter trust protocol pim enable

set interface ethernet1 protocol pim
set interface ethernet1 protocol pim enable
set interface ethernet2 protocol pim
set interface ethernet2 protocol pim enable
set interface ethernet3 protocol pim
set interface ethernet3 protocol pim enable
```

Addresses

```
set address trust mgroup1 224.4.4.1/32
set address untrust source-dr 6.6.6.1/24
```

Policies

```
set policy from untrust to trust source-dr mgroup1 any permit
set multicast-group-policy from trust mgroup 224.4.4.1/32 any to untrust
    pim-message bsr-static-rp join bi-directional
save
```

VERIFYING THE CONFIGURATION

You can view the multicast route table entries of a VR by executing the following CLI command:

```
ns-> get vr trust protocol pim mroute
trust-vr - Multicast routing table
-----
Register - R, Connected members - C, Pruned - P
Forward - F, Null - N, Negative Cache - E, Local Receivers - L
SPT - T, Proxy-Register - X Imported - I
-----
(*, 230.1.1.5) RP: 172.16.1.1 00:00:35/-          Flags: ILF
Zone           : Untrust
Pkt Cnt        : 0                               Ext Receivers : 0
Upstream       : tunnel.1                       State         : Joined
RPF Neighbor   : 20.20.20.2                     Expires       : 00:00:0
Downstream
  ethernet1    230.1.1.5      00:00:35/-          Join          Flags: FC
(*, 239.1.1.1) RP: 10.10.10.21 00:00:39/-          Flags: F
Zone           : Trust
Pkt Cnt        : 0                               Ext Receivers : 0
Upstream       : ethernet2                       State         : Joined
RPF Neighbor   : local                          Expires       : -
Downstream
  tunnel.1     239.1.1.1      00:00:39/00:02:50  Join          Flags: F
(10.150.43.133/21, 239.1.1.1) 00:00:38/00:02:52  Flags: TF
Zone           : test
Pkt Cnt        : 0                               Ext Receivers : 0
Upstream       : ethernet1                       State         : Joined
RPF Neighbor   : local                          Expires       : -
Downstream
  tunnel.1     239.1.1.1      00:00:38/00:02:53  Join          Flags: F
```

You can verify the following in each route entry:

- The (S,G) state or (*,G) forwarding state
- If the forwarding state is (*,G), the IP address of the RP
- If the forwarding state is (S,G), the IP address of the source
- Zone that owns the route
- Number of packets forwarded by PIM
- The “join” status
- Incoming and outgoing interfaces
- Timer values

To view the active RPs in each zone, execute the following command:

```
ns-> get vrouter trust protocol pim rp active
```

Zone	Group	RP	Source
Trust	230.1.1.0/24	172.16.1.1	BSR
	239.1.1.1/32	20.1.1.1	Static
DMZ	230.1.1.0/24	172.16.1.1	BSR
	239.1.1.1/32	20.1.1.1	Static

To verify that there is an RPF neighbor, execute the following command:

```
ns-> get vrouter trust protocol pim rpf
```

Flags : RP address - R, Source address - S

Address	RPF Interface	RPF Neighbor	Flags
10.10.11.51	ethernet3	10.10.11.51	R
10.150.43.133	ethernet3	10.10.11.51	S

To view the status of the join-prune messages being sent to each neighbor in a virtual router, execute the following command:

```
ns-> get vrouter untrust protocol pim join
```

Neighbor	Interface	J/P	Group	Source
1.1.1.1	ethernet4:1	(S,G)	J 224.11.1.1	60.60.0.1
		(S,G)	J 224.11.1.1	60.60.0.1
20.20.0.1	ethernet2:1	(* ,G)	J 224.11.1.1	40.40.0.2
		(S,G,Rpt)	P 224.11.1.1	60.60.0.1
		(* ,G)	J 224.1.1.4	40.40.0.2

CONFIGURING RP TO GROUP MAPPINGS

On a NetScreen device, you can either configure a static RP for a particular zone, or you can use dynamic RP mappings and configure a virtual router as a C-RP. Note that you cannot configure a NetScreen device as a bootstrap router, although it can receive and process bootstrap messages. Therefore if there is not bootstrap router in the network, you can configure a static RP for a particular zone. When you do, you must define the multicast groups associated with the RP. A static RP should be configured on one zone only. All other zones import RP to group mappings from this zone as long as a multicast policy permits it.

Example: Creating a Static RP

In this example, you create an access list for the multicast group 224.4.4.1, and then create a static RP for that group. To ensure that the multicast groups in the access list always use the same RP, you include the keyword **always**.

CLI

```
set vrouter trust-vr access-list 2 permit ip 224.4.4.1/32 1
set vrouter trust-vr protocol pim zone trust rp address 1.1.1.5 mgroup 2 always
save
```

You can also configure a virtual router as a C-RP for a set of multicast groups. On NetScreen devices, you can create one C-RP for each zone. When you configure a virtual router as a C-RP, you specify the following:

- The zone in which the C-RP is configured
- IP address of the interface that is advertised as the C-RP
- An access list that defines the multicast groups of the C-RP
- The advertised C-RP priority

Example: Creating a Candidate RP

In this example, you create an access list that defines the multicast groups of the C-RP. Then you create a C-RP in the Trust zone of the trust-vr. You set the priority of the C-RP to 200, to ensure that it is selected the RP.

CLI

```
set vrouter trust-vr access-list 1 permit ip 224.2.2.1/32 1
set vrouter trust-vr access-list 1 permit ip 224.3.3.1/32 2
set vrouter trust-vr protocol pim zone trust rp candidate interface ethernet1
    mgroup-list 1 priority 200
save
```

SECURITY CONSIDERATIONS

Malicious users can send invalid PIM messages and cause problems in your network. One way to control multicast traffic is through the multicast group policies described in [“Creating a Multicast Group Policy” on page 106](#). In these policies, you can specify which multicast groups are allowed to cross the NetScreen device.

When you run PIM, there are also certain options that you can set at the VR level to control traffic to and from the VR. Settings defined at the VR level affect all PIM-enabled interfaces in the VR.

When an interface receives multicast control traffic (IGMP or PIM messages), the NetScreen device first checks if there is a multicast policy that allows the traffic. (Note that if the multicast control traffic does not pass between zones, then the device does not check for a multicast policy.)

- If the device does not find a multicast policy that allows the traffic, then it drops the traffic.
- If the device finds a multicast policy that allows the traffic, then it checks for any PIM options on the virtual router that apply to the traffic. For example, if you configured the virtual router to accept join-prune messages from multicast groups specified in an access list, the device checks if the traffic is from a multicast group that is in the access list. If it is, then it allows the traffic. If it is not, then it drops the traffic.

Note that if a NetScreen device is configured with multiple VRs, all VRs must have the same PIM options.

Restricting Multicast Groups

You can restrict the VR to forwarding PIM join-prune messages for a particular set of groups only. You specify the allowed multicast groups in an access list. When you use this feature, the VR drops join-prune messages from groups that are not in the access list.

Example: Restricting Multicast Groups

In this example, you create an access list that specifies the allowed multicast groups. Then you configure the trust-vr to accept join-prune messages from the multicast groups in the access list.

CLI

```
set vrouter trust access-list 1 permit ip 224.2.2.1/32 1
set vrouter trust access-list 1 permit ip 224.3.3.1/32 2
set vrouter trust protocol pim accept-group 1
save
```

Restricting Multicast Sources

You can also control from which sources a multicast group can receive data. You identify the allowed source(s) in an access list, then specify the multicast groups to which the sources can send traffic. This prevents unauthorized sources from sending data into your network. When you use this feature, the VR drops multicast data from sources not in the list. If the VR is the RP in the zone, it checks the access list before accepting a Register message from a source. It drops the Register packets if they are not from an allowed source.

Example: Restricting Multicast Sources

In this example, you first create an access list that specifies the allowed source. Then you configure the trust-vr to accept multicast data for the multicast group 224.4.4.1/32 from the source specified in the access list.

CLI

```
set vrouter trust-vr access-list 5 permit ip 1.1.1.1/32 1
set vrouter trust-vr protocol pim mgroup 224.4.4.1/32 accept-source 5
save
```

Restricting RPs

You can control which RPs are mapped to a multicast group. You identify the allowed RP(s) in an access list, then specify the multicast groups for which the RPs can send data. When the VR receives a bootstrap message for a particular group, it checks its list of accepted RPs for that group. If it does not find a match, then it does not select an RP for the multicast group.

Example: Restricting RPs

In this example, you first create an access list that specifies the allowed RPs. Then you configure the trust-vr to accept the RPs in the access list.

CLI

```
set vrouter trust-vr access-list 6 permit ip 2.1.1.1/32 1
set vrouter trust-vr protocol pim mgroup 224.4.4.1/32 accept-rp 6
save
```

PIM INTERFACE PARAMETERS

You can change certain defaults for each interface on which PIM is enabled. When you set parameters on this level, it affects only the interface that you specified.

The following table describes the PIM interface parameters and their defaults.

PIM Interface Parameters	Description	Default Value
Neighbor policy	Controls neighbor adjacencies. For additional information, see “Neighbor Policy” on page 121 .	Disabled
Hello interval	Specifies the interval at which the interface sends hello messages to its neighboring routers.	30 seconds
DR priority	Specifies the priority of the interface for the designated router election.	1
Join-Prune interval	Specifies the interval, in seconds, at which the interface sends join-prune messages.	60 seconds
Bootstrap border	Specifies that the interface is a bootstrap border. For additional information, see “Bootstrap Border” on page 122 .	Disabled

Example: Changing the DR Priority

In this example, you change the DR priority of interface ethernet1 from 1 to 10.

CLI

```
set interface ethernet1 protocol pim dr-priority 10
save
```

Neighbor Policy

You can control with which neighbors an interface can form an adjacency. PIM routers periodically send Hello messages to announce themselves as PIM routers. If you use this feature, the interface checks its list of allowed or disallowed neighbors and forms adjacencies with the allowed interfaces only.

Example: Defining a Neighbor Policy

In this example, you create an access list that specifies the permitted neighbors. Then you specify that ethernet 1 can form an adjacency with the neighbors in the access list.

CLI

```
set vrouter trust access-list 6 permit ip 2.1.1.1/24 1
set vrouter trust access-list 6 permit ip 2.1.1.3/24 2
set interface ethernet1 protocol pim neighbor-policy 6
save
```

Bootstrap Border

You can define an interface as a bootstrap border. An interface that is a bootstrap border receives and processes BSR messages, but does not forward these messages to other interfaces even if there is a multicast group policy that allows BSR messages between zones. This ensures that the RP-group mappings always stay within a zone.

Example: Defining a Bootstrap Border

In this example, you configure ethernet1 as a bootstrap border.

CLI

```
set interface ethernet1 protocol pim boot-strap-border
save
```

PROXY RP

You can run PIM-SM in Route mode or in NAT mode. If the zones on a device are operating in different modes, then each zone should be in a different domain. For example, if the Trust zone is in NAT mode and the Untrust zone is in route mode, then each zone should be in a different multicast domain. You can manage the multicast traffic between the two domains by configuring a proxy RP.

You can configure a VR to function as a proxy RP. A proxy RP acts as the RP for groups learned from other zones. It functions as the root of the shared tree for receivers in that domain and it can form the shortest path tree to the source to stop register messages. A proxy-RP functions as an RP, but forwards the register messages to the real RP.

On NetScreen devices, you can use proxy RP to do the following:

- Translate a BSR or RP address in one zone to another zone
- Translate a source address from one zone to another
- Advertise a multicast group address in one zone as a different group in another zone

Configuring a Proxy RP

To configure a proxy RP, you must first create a PIM instance on the virtual router, enable PIM on the appropriate interfaces, and then create the proxy RP.

Example: Creating a Proxy RP

In this example, you bind ethernet1 and ethernet3 to the Trust ZOne. Then you create a proxy RP in the Trust zone of the trust-vr.

CLI

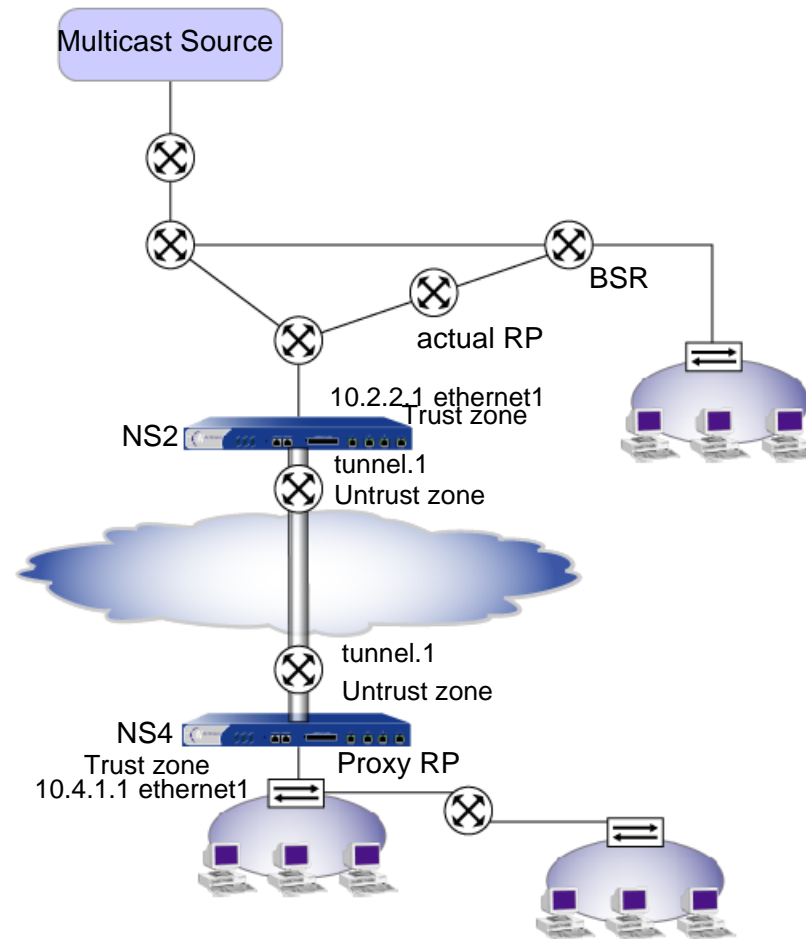
```
set interface ethernet 1 zone trust
set interface ethernet 1 ip 10.4.1.1/24

set interface ethernet 3 zone untrust
set interface ethernet 3 ip 4.1.1.1/24

set vrouter trust protocol pim
set interface ethernet1 protocol pim
set interface ethernet1 protocol pim enable
set interface ethernet3 protocol pim
set interface ethernet3 protocol pim enable
set vrouter trust protocol pim zone trust rp proxy
set vrouter trust protocol pim enable
save
```

Example: Basic Proxy RP Configuration

In this example, the NetScreen devices NS2 and NS4 are connected through a VPN tunnel. Both devices are running BGP as the unicast routing protocol. On NS2, you configure PIM on ethernet1 and tunnel.1. On NS4, you configure PIM on ethernet1 and tunnel.1. In addition, on NS4, you create a proxy RP in the Trust zone of the trust-vr.



NS2-Corp

Interfaces – Zones

```
set interface ethernet1 zone trust
set interface ethernet1 ip 10.2.2.1/24

set interface ethernet3 zone untrust
set interface ethernet3 ip 2.2.2.2/24

set interface tunnel.1 zone untrust
set interface tunnel.1 ip unnumbered interface ethernet3
```

Addresses

```
set address trust mgroup1 224.4.4.1/32
set address untrust branch3 10.4.1.0/24
```

PIM-SM

```
set vrouter trust
set vrouter trust protocol pim enable

set interface ethernet1 protocol pim
set interface ethernet1 protocol pim enable
set interface tunnel.1 protocol pim
set interface tunnel.1 protocol pim enable
```

VPN Tunnel

```
set ike gateway To_Branch3 address 4.1.1.1 main outgoing-interface ethernet3
  preshare fg2g4h5j proposal pre-g2-3des-sha
set vpn Corp_Branch3 gateway To-Branch3 sec-level compatible
set vpn Corp_Branch3 bind interface tunnel.1
set vpn Corp_Branch3 proxy-id local-ip 10.2.2.0/24 remote-ip 10.4.1.0/24
```

BGP

```
set vrouter trust router-id 10
set vrouter trust protocol bgp 6500
set vrouter trust protocol bgp enable
set vrouter trust protocol bgp neighbor 4.1.1.1
set vrouter trust protocol bgp network 2.2.2.0/24
set vrouter trust protocol bgp network 10.2.2.0/24
set interface ethernet3 protocol bgp enable
set interface ethernet3 protocol bgp neighbor 4.1.1.1
```

Policies

```
set policy name To-Branch3 from untrust to trust branch any any permit
set multicast-group-policy from trust mgroup 224.4.4.1/32 any to untrust
  pim-message bsr-static-rp join bi-directional
```

```
save
```

NS4-Branch

Zones and Interfaces

```
set interface ethernet 1 zone trust
set interface ethernet 1 ip 10.4.1.1/24
set interface ethernet 1 protocol igmp router
set interface ethernet 1 protocol igmp enable

set interface ethernet 3 zone untrust
set interface ethernet 3 ip 4.1.1.1/24

set interface tunnel.1 zone untrust
set interface tunnel.1 ip unnumbered interface ethernet3
```

Addresses

```
set address trust mgroup1 224.4.4.1/32
set address untrust corp 2.2.2.0/24
```

PIM-SM

```
set vrouter trust protocol pim
set interface ethernet1 protocol pim
set interface ethernet1 protocol pim enable
set interface tunnel.1 protocol pim
set interface tunnel.1 protocol pim enable
set vrouter trust protocol pim zone trust rp proxy
set vrouter trust protocol pim enable
```

VPN Tunnel

```
set ike gateway To_Corp address 2.2.2.2 main outgoing-interface ethernet3
  preshare fg2g4h5j proposal pre-g2-3des-sha
set vpn Branch3_Corp gateway To-Corp sec-level compatible
set vpn Branch3_Corp bind interface tunnel.1
set vpn Branch3_Corp proxy-id local-ip 10.4.1.0/24 remote-ip 10.2.2.0/24
```

BGP

```
set vrouter trust-vr router-id 10
set vrouter trust-vr protocol bgp 6500
set vrouter trust-vr protocol bgp enable
set vrouter trust-vr protocol bgp neighbor 2.2.2.2
set vrouter trust protocol bgp network 4.1.1.0/24
set vrouter trust protocol bgp network 10.4.1.0/24
set interface ethernet3 protocol bgp neighbor 2.2.2.2
```

Policies

```
set policy name To-Corp from untrust to trust corp any any permit
set multicast-group-policy from trust mgroup 224.4.4.1/32 any to untrust
  pim-message bsr-static-rp join bi-directional
save
```


New and Modified CLI Commands - Multicast

This chapter introduces the following new commands:

- [igmp](#) on page 132
- [multicast-group-policy](#) on page 152
- “PIM Context Commands” on page 156

In addition, it presents changes to the following commands:

- [interface](#) on page 136
- [vrouter](#) on page 176

New command elements in the Syntax sections appear in **red**. For example, in the following command, **protocol igmp** is new in this release:

```
get interface interface protocol igmp
```

The following command descriptions focus only on the new elements added in this release. For more information about other command elements, refer to the *NetScreen CLI Reference Guide*.

igmp

Description: Use the **igmp** command to send an IGMP message, to display IGMP settings and monitor IGMP states on a NetScreen device, or to clear IGMP information.

Syntax

exec

```
exec igmp interface interface
{
  query [ ip_addr ]
  report ip_addr
  leave ip_addr
}
```

get

```
get igmp
{
  group [ ip_addr | all ] |
  interface [ all ] |
  statistic [ all ]
}
```

clear

```
clear igmp interface interface
{
group ip_addr | all |
statistic
}
```

Keywords and Variables

group

```
get igmp group [ ip_addr | all ]
clear igmp interface interface group { mcast_addr | all }
```

group Displays or clears information for the multicast group specified. Specify **all** to display or clear information for all multicast groups.

interface

```
exec igmp interface interface { ... }
get igmp interface [ all ]
clear igmp interface interface group { mcast_addr | all }
```

interface Displays, clears or sends IGMP messages for the specified interface.

leave

```
exec igmp interface interface leave mcast_addr
```

leave Sends a leave message for the specified multicast group. You can execute this command if the interface is in host mode only.

query

```
exec igmp interface interface query [ mcast_addr ]
```

query Sends an IGMP query message. If you specify a multicast group address, the interface sends a group-specific query to the specified multicast group. If you do not specify a multicast group address, then the interface sends a general query to the “all hosts” group (224.0.0.1). Enter this command only if the interface is in router mode.

Example: The following command sends a general query to the “all hosts” group from interface **ethernet4**:

```
exec igmp interface ethernet4 query
```

report

```
exec igmp interface interface report mcast_addr
```

report Sends an IGMP membership report to the specified group. Enter this command if the interface is in host mode.

Example: The following command sends a membership report to the specified multicast group:

```
exec igmp interface ethernet4 report 224.2.1.1
```

statistic

```
get igmp statistic [ all ]
```

```
clear igmp interface interface statistic
```

statistic Displays or clears IGMP statistics. Enter this command if the interface is in router mode.

interface

Description: Use the **interface** commands to define or display interface settings, and to monitor the status of a NetScreen device. Interfaces are physical or logical connections that handle network, virtual private network (VPN), High Availability (HA), administrative, and multicast traffic.

Note: This section only describes new keywords and variables for the **interface** commands. For more information on other keywords and variables for the **interface** commands, refer to the NetScreen CLI Reference Guide.

Syntax

get

```
get interface interface
  protocol
  {
    igmp [group ip_addr | all ] | statistic [ all ]
    pim [ statistics ]
  }
```

set (IGMP Interfaces)

```
set interface interface
[
protocol igmp
{
    host
    {
        enable |
        no-check-subnet |
        no-check-router-alert
    }
} |
{
    router
    {
        enable |
        join-group ip_addr
        no-check-subnet |
        no-check-router-alert |
        version { 1 | 2 } |
        query-interval number
        query-max-response-time number
        leave-interval number
        last-member-query-interval number
        proxy [ always ]
    }
}
```

```
        accept routers id_num
        accept hosts id_num
        accept groups id_num
        static-group ip_addr
    }
]
}
```

unset (IGMP)

```
unset interface interface
{
protocol igmp
    host
    router
    enable |
    no-check-subnet |
    no-check-router-alert |
    version |
    query-interval
    query-max-response-time
    leave-interval
    last-member-query-interval
    proxy [ always ]
    accept routers
    accept hosts
    accept groups
    static-group [ ip_addr | all ]
    join-group [ ip_addr | all ]
}
```

set (PIM Interfaces)

```
set interface interface
[
protocol pim
    [
    boot-strap-border
    dr-priority number
    enable
    hello-interval number
    join-prune-interval number
    neighbor-policy number
    ]
]
```

unset (PIM Interfaces)

```
unset interface interface
[
protocol pim
    [
    boot-strap-border
    dr-priority
    enable
    hello-interval
    join-prune-interval
    neighbor-policy
    ]
]
```

set (Tunnel)

```
set interface tunnel.number
{
  protocol { igmp | pim }
  tunnel
  {
    local-if interface dst-ip ip_addr
    encap gre
    keep-alive [interval number | threshold number ]
  }
}
```

unset (Tunnel)

```
unset interface interface
[
  tunnel [ keep-alive ]
]
```

Keywords and Variables (IGMP)

accept groups

```
set interface interface protocol igmp accept groups id_num  
unset interface interface protocol igmp accept groups
```

accept groups Specifies the access list that identifies the multicast groups the hosts on the specified interface can join. Enter this command only if the interface is in router mode.

Example: The following command allows the hosts on interface **ethernet4** to join the multicast group(s) in the access list with ID number 1:

```
set interface ethernet4 protocol igmp accept groups 1
```

accept hosts

```
set interface interface protocol igmp accept hosts id_num  
unset interface interface protocol igmp accept hosts
```

accept hosts Specifies the access list that identifies from which hosts the interface can receive join and leave messages. After you have set this command, the interface accepts join and leave messages only from the hosts in the access list. Enter this command only if the interface is in router mode.

Example: The following command allows interface **ethernet4** to accept join and leave messages from the hosts in the access list with ID number 10:

```
set interface ethernet4 protocol igmp accept hosts 10
```

accept routers

```
set interface interface protocol igmp accept routers id_num  
unset interface interface protocol igmp accept routers
```

accept routers Specifies the access list that identifies the routers that are eligible for Querier selection. Only the routers in this list can be elected as Querier. Enter this command only if the interface is in router mode.

Example: The following command allows interface **ethernet4** to accept queries from the routers in the access list with ID number 5:

```
set interface ethernet4 protocol igmp accept routers 5
```

always

```
set interface interface protocol igmp proxy always  
unset interface interface protocol igmp proxy always
```

always Enables the interface to forward IGMP messages even if it is a non-Querier. Enter this command only if the interface is in router mode and IGMP proxy is enabled.

Example: The following command sets interface **ethernet4** to non-Querier IGMP proxy mode:

```
set interface ethernet4 protocol igmp proxy always
```

enable

```
set interface interface protocol igmp enable  
unset interface interface protocol igmp enable
```

enable Enables or disables the IGMP protocol on the interface.

host

```
set interface interface protocol igmp host
unset interface interface protocol igmp host
```

host Creates an IGMP host instance on the specified interface.

join-group

```
set interface interface protocol igmp join-group mcast_addr
unset interface interface protocol igmp join-group
```

join-group Enables the interface to join the specified multicast group. Enter this command only if the interface is in router mode.

last-member-query-interval

```
set interface interface protocol igmp last-member-query-interval number
unset interface interface protocol igmp last-member-query-interval
```

last-member-query-interval Sets the interval (in seconds) the Querier waits for a response to a group-specific query before it stops sending multicast traffic for that particular group on the specified interface (range 1-25 inclusive). Enter this command if the interface is in router mode and it is running IGMP version 2.

Example: The following command specifies the number of seconds the Querier will wait for a response to a group-specific query before it assumes there are no more group members on the interface:

```
set interface ethernet4 protocol igmp last-member-query-interval 5
```

leave-interval

```
set interface interface protocol igmp leave-interval number  
unset interface interface protocol igmp leave-interval
```

leave-interval Sets the interval (in seconds) between group specific-queries (range 1 - 255 inclusive). Enter this command if the interface is in router mode.

no-check-router-alert

```
set interface interface protocol igmp no-check-router-alert  
unset interface interface protocol igmp no-check-router-alert
```

no-check-router-alert IGMP packets contain a router-alert IP option. By default, an IGMP-enabled device checks IGMP packets for this option and drops packets that do not have this option. Enter this command to accept all IGMP packets without checking for the router-alert option.

no-check-subnet

```
set interface interface protocol igmp no-check-subnet  
unset interface interface protocol igmp no-check-subnet
```

no-check-subnet By default, an IGMP interface accepts IGMP packets from its own subnet only. Enter this command to allow the interface to accept IGMP packets (queries, membership reports, and leave messages) from any subnet.

proxy

```
set interface interface protocol igmp proxy  
unset interface interface protocol igmp proxy
```

proxy When the interface is in router mode, enables IGMP proxy mode.

query-interval

```
set interface interface protocol igmp query-interval number  
unset interface interface protocol igmp query-interval
```

query-interval Specifies the interval (in seconds) between General Queries (range 1 to 255, inclusive). Enter this command if the interface is set to router mode and it is the Querier for a multicast group.

Example: The following command sets the interface **ethernet4** to router mode and sets the interval between general queries to 150 seconds:

```
set interface ethernet4 protocol igmp router  
set interface ethernet4 protocol igmp query-interval 150
```

query-max-response-time

```
set interface interface protocol igmp query-max-response-time number  
unset interface interface protocol igmp query-max-response-time
```

query-max-response-time Sets the maximum number of seconds that elapses between the time a Querier sends a General Query and the time a host responds to it (range 1 to 25, inclusive). Enter this command if the interface is in router mode.

Example: The following command sets the interface **ethernet4** to router mode and sets the maximum response to 15 seconds:

```
set interface ethernet4 protocol igmp router  
set interface ethernet4 protocol igmp query-max-response-time 15
```

router

```
set interface interface protocol igmp router
unset interface interface protocol igmp router
```

router Sets the specified interface to router mode.

Example: The following command sets interface **ethernet4** to IGMP router mode:

```
set interface ethernet4 protocol igmp router
```

static-group

```
set interface interface protocol igmp static-group ip_addr
unset interface interface protocol igmp static-group ip_addr
```

static-group Manually adds the multicast group to the specified interface. Enter this command only if the interface is in router mode.

Example: The following command sets interface **ethernet4** to router mode and adds the multicast group 224.2.1.1 to the interface:

```
set interface ethernet4 protocol igmp router
set interface ethernet4 protocol igmp static-group 224.25.1.1
```

version

```
set interface interface protocol igmp version { 1 | 2 }  
unset interface interface protocol igmp version
```

version Specifies the IGMP version. When an interface is in host mode, the device automatically sets the IGMP version. When an interface is in router mode, it runs IGMP version 2 by default. Enter this command to change the IGMP version of a router interface.

Example: The following command sets interface **ethernet4** to router mode and sets it to IGMP version 1:

```
set interface ethernet4 protocol igmp router  
set interface ethernet4 protocol igmp version 1
```

Keywords and Variables (PIM)

boot-strap-border

```
set interface interface protocol pim boot-strap-border  
unset interface interface protocol pim boot-strap-border
```

boot-strap border Configures the interface as a border for bootstrap (BSR) messages. The interface receives and processes BSR messages, but does not forward these messages to other interfaces even if there is a multicast group policy that allows BSR messages between zones.

dr-priority

```
set interface interface protocol pim dr-priority num  
unset interface interface protocol pim dr-priority
```

dr-priority Configures the priority of the interface during the designated router election.

enable

```
set interface interface protocol pim enable  
unset interface interface protocol pim enable
```

enable Enables or disables the PIM-SM protocol on the interface.

hello-interval

```
set interface interface protocol pim hello-interval number  
unset interface interface protocol pim hello-interval
```

hello-interval Specifies the interval (in seconds) at which the interface sends hello messages to its neighbors.

join-prune-interval

```
set interface interface protocol pim join-prune-interval number  
unset interface interface protocol pim join-prune-interval
```

join-prune-interval Sets the interval, in seconds, at which the interface sends join-prune messages.

neighbor-policy

```
set interface interface protocol pim neighbor-policy number  
unset interface interface protocol pim neighbor-policy
```

neighbor-policy Identifies the access list that allows or disallows certain neighbor adjacencies.

Keywords and Variables (tunnel interface)

encap gre

```
set interface tunnel.number tunnel encap gre
unset interface tunnel.number
```

encap gre Specifies that all traffic in the tunnel is encapsulated using the GRE (Generic Routing Encapsulation) protocol.

keep-alive

```
set interface tunnel.number tunnel keep-alive interval number
set interface tunnel.number tunnel keep-alive threshold number
unset interface tunnel.number tunnel [ keep-alive ]
```

keep-alive The tunnel interface sends keep-alive messages to monitor the status of the connection. You can specify the interval (in seconds) between keep-alive messages, and the number of times the local tunnel interface sends keep-live messages without receiving a reply before it terminates the connection.

local-if

```
set interface tunnel.number tunnel local-if interface dst-ip ip_addr
unset interface tunnel.number tunnel
```

local-if Specifies the local interface and the destination IP address of a GRE tunnel.

Defaults

The default for query intervals is 125 seconds.

The default for the query maximum response time is 10 seconds.

The default last-member query interval is 1 second.

The default priority of an interface during the designated router election is 1.

The default hello interval is 30 seconds.

The default join-prune interval is 60 seconds.

multicast-group-policy

Description: Use the **multicast-group-policy** command to define access policies for multicast traffic.

Syntax

get

```
get multicast-group-policy
  between zone1 zone2
```

set

```
set multicast-group-policy
  from zone1
  { mgroup { mcast_addr1 | any } | mgroup-list id_num }
  to zone2
  { mgroup mcast_addr2 { igmp-message } | { pim-message { bsr-static-rp |
  join-prune } }
  [ bi-directional ]
```

unset

```
unset multicast-group-policy
  from zone1
  { mgroup { mcast_addr1 | any } | mgroup-list number }
  to zone2 { igmp-message } | { pim-message bsr-static-rp | join-prune }
  [ bi-directional ]
```

Keywords and Variables

between

```
get multicast-group policy between zone1 zone2
```

between Displays the multicast policy configured between the specified zones.

bi-directional

```
set multicast-group policy from { ... } to { ... } bi-directional  
unset multicast-group policy from { ... } to { ... } bi-directional
```

bi-directional Specifies that the policy applies to both directions of multicast traffic.

Example: The following command defines a bi-directional multicast group policy that allows PIM messages between the trust and untrust zones:

```
set multicast-group-policy from trust mgroup any to untrust pim-message  
bsr-static-rp join-prune bi-directional
```

from ... to

```
set multicast-group policy from zone1 mgroup mcast_addr1 to zone2 mgroup  
mcast_addr2 { ... }  
set multicast-group policy from zone1 mgroup any to zone2 { ... }  
set multicast-group policy from zone1 mgroup-list id_num to zone2  
unset multicast-group policy from zone1 mgroup mcast_addr1 to zone2 { ... }  
unset multicast-group policy from zone1 mgroup any to zone2  
unset multicast-group policy from zone1 mgroup-list id_num to zone2
```

from ... to

Specifies the two zones between which the policy applies.

- *zone1* is the name of the source security zone.
- *zone2* is the name of the destination security zone.
- *mcast_addr1* is the multicast IP address of the multicast group from which the zone accepts multicast packets
- *mcast_addr2* is the translated multicast group address, if you are translating a multicast group address from one zone to another
- *id_num* is the ID number of the access list that specifies the multicast groups from which the zone accepts multicast packets

Example: The following command creates a multicast policy from the trust zone to the untrust zone:

```
set multicast-group-policy from trust mgroup-list 12 to untrust
```

igmp-message

```
set multicast-group policy from { ... } to { ... } igmp-message  
unset multicast-group policy from { ... } to { ... } igmp-message
```

igmp-message Specifies a multicast group policy that allows IGMP messages between the specified zones.

pim-message

```
set multicast-group policy from { ... } to { ... } pim-message { bsr-static-rp |  
join-prune }  
unset multicast-group policy from { ... } to { ... } pim-message { bsr-static-rp |  
join-prune }
```

pim-message Specifies a multicast group policy that allows PIM BSR and/or join-prune messages between the specified zones.

PIM CONTEXT COMMANDS

The commands described in the following pages are **pim** context commands. Use the **pim** context commands to configure PIM-SM (Protocol-Independent Multicast - Sparse Mode) on a virtual router in a NetScreen device. You issue these commands within the context of a specific virtual router and the PIM-SM protocol.

Initiating the **pim** context requires two steps:

1. Enter the virtual router context by executing the **set vrouter** command.

```
ns-> set vrouter trust-vr
```

2. Enter the PIM context by executing the **set protocol pim** command.

```
ns(trust-vr)-> set protocol pim
```

To exit out of each context, enter **exit**.

The following commands are executable in the **pim** context.

[accept-group](#)

Use the **accept-group** command to specify the access list that identifies the multicast group(s) for which the virtual router processes PIM messages.

[bsr](#)

Use the **bsr** command to display information about the bootstrap router.

[enable](#)

Use the **enable** command to enable or disable the PIM-SM instance on the virtual router.

[interface](#)

Use the **interface** command to display all interfaces running PIM-SM.

[join-prune](#)

Use the **join-prune** command to display join-prune messages sent to each neighbor.

[mgroup](#)

Use the **mgroup** command to specify from which source(s) and/or RP the multicast group accepts traffic.

[mroute](#)

Use the **mroute** commands to display multicast route table entries.

[neighbor](#)

Use the **neighbor** command to display information about all neighbors discovered for each interface.

[rp](#)

Use the **rp** command to display the status of the RP (rendezvous point).

[rpf](#)

Use the **rpf** command to display RPF information for a particular source or RP.

[spt-threshold](#)

Use the **spt-threshold** command to specify the packet rate that triggers the device to switch from the shared distribution tree to the source-specific distribution tree.

[zone](#)

Configures the following:

- an RP candidate in the specified zone
- a static RP for the specified multicast groups in the named zone

accept-group

Description: Use the **accept-group** command to specify the access list that identifies the multicast group(s) for which the virtual router processes PIM messages.

Before you can execute the **accept-group** command, you must initiate the **pim** context. (See “[PIM Context Commands](#)” on page 156.)

Syntax

set

```
set accept-group number
```

unset

```
unset accept-group
```

Keywords and Variables

Variable Parameter

```
set accept-group number
```

number Specifies the access list that identifies the multicast group(s) for which the virtual router accepts PIM messages.

bsr

Description: Use the **bsr** command to display information about the elected bootstrap router.

Before you can execute the **bsr** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Syntax

get

get bsr

Keywords and Variables

None.

enable

Description: Use the **enable** command to enable or disable the PIM-SM instance on the virtual router.

Before you can execute the **enable** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Syntax

set

set enable

unset

unset enable

Keywords and Variables

None.

interface

Description: Use the **interface** command to display all interfaces running PIM-SM.

Before you can execute the **interface** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Syntax

get

get interface

Keywords and Variables

None.

join-prune

Description: Use the **join-prune** command to display join-prune messages sent to each neighbor.

Before you can execute the **join-prune** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Syntax

get

get join-prune

Keywords and Variables

None.

mgroup

Description: Use the **mgroup** command to specify from which source(s) and/or RP the multicast group accepts traffic.

Before you can execute the **mgroup** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Syntax

set

```
mgroup mcast_addr { accept-rp number | accept-source number }
```

Keywords and Variables

Variable Parameter

```
set mgroup mcast_addr
```

ip_addr Specifies the IP address of the multicast group.

accept-rp

```
set mgroup mcast_addr accept-rp number
```

accept-rp Specifies the access list that identifies the RP(s) from which the device forwards traffic to the multicast group. The device drops traffic for the multicast group if the traffic is from an RP that is not on the specified access list.

accept-source

```
set mgroup mcast_addr accept-source number
```

accept-source Specifies the access list that identifies the source(s) from which the device forwards traffic to the multicast group. The device drops traffic for the multicast group if the traffic is from a source that is not on the specified access list.

mroute

Description: Use the **mroute** commands to display multicast route table entries.

Syntax

get

```
get mroute  
[  
  brief |  
  mgroup mcast_addr [ detail ] [ brief ] [ source ip_addr [ detail ] [ brief ] ]  
]
```

Note: Before you can execute the **get mroute** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Keywords and Variables

brief

```
get mroute brief  
get mroute mgroup mcast_addr brief  
get mroute mgroup mcast_addr source ip_addr brief
```

brief Displays summary information about the multicast routes. Displays the source address, multicast group address, and the list of incoming and outgoing interfaces.

detail

```
get mroute mgroup mcast_addr detail
```

```
get mroute mgroup mcast_addr source ip_addr detail
```

brief Displays information about the multicast route, including the RPF and type of route. It also provides details on the input and output interfaces.

mgroup

```
get mroute mgroup mcast_addr brief
```

```
get mroute mgroup mcast_addr detail
```

```
get mroute mgroup mcast_addr source ip_addr [ brief | detail ]
```

mgroup Displays multicast route table entries for the specified multicast group or defines a multicast route for a particular multicast group.

source

```
get mroute mgroup ip_addr source ip_addr
```

source Specifies the IP address of the source of the multicast traffic.

neighbor

Description: Use the **neighbor** command to display information about all neighbors discovered for each interface.

Before you can execute the **neighbor** command, you must initiate the **pim** context. (See [“PIM Context Commands” on page 156.](#))

Syntax

get

get neighbor

Keywords and Variables

None.

rp

Description: Use the **rp** command to display the status of the RP (rendezvous point).

Before you can execute the **rp** command, you must initiate the **pim** context. (See “PIM Context Commands” on page 156.)

Syntax

get

```
get rp
{
  active | all | candidate | mgroup ip_addr [ active ] | proxy
}
```

Keywords and Variables

active

```
get rp active
```

active Displays the RP that is actively sending multicast traffic to the multicast groups.

all

```
get rp all
```

all Displays all RPs, including candidate RPs, for each multicast group.

candidate

`get rp candidate`

candidate Displays the status of the RP candidates that you configured for each zone on the virtual router.

mgroup

`get rp mgroup ip_addr [active]`

mgroup Displays information about the group-RP set for the specified multicast group. Specify **active** to display the RP for the specified multicast group.

proxy

`get rp proxy`

proxy Displays the proxy-RP status for each zone in the PIM instance of the virtual router.

rpf

Description: Use the **rpf** command to display RPF (reverse path forwarding) information for a particular source or RP.

Syntax

get

get rpf

Keywords and Variables

None.

spt-threshold

Description: Use the **spt-threshold** command to specify the threshold that triggers the virtual router to switch from the shared distribution tree to the source-based tree.

Syntax

set

```
set spt-threshold { number | infinity }
```

unset

```
unset spt-threshold
```

Keywords and Variables

Variable Parameter

```
set spt-threshold number
```

number Specifies the threshold (number of packets per second) that triggers the device to switch from the shared distribution tree to the source-specific distribution tree. If you specify **infinity**, the device never switches to a source-specific distribution tree.

zone

Description: Use the **zone** command to configure the following for the specified zone:

- an RP candidate
- a static RP for the specified multicast groups in the named zone
- a proxy-RP

Syntax

get

```
get zone
[ zone
  [
    bsr |
    rp { active | all | candidate | mgroup ip_addr [ active ] | proxy }
  ]
]
```

set

```
set zone zone rp
{
address ip_addr mgroup-list number [ always ] |
candidate interface interface mgroup-list number
  [ holdtime number | priority number ] |
proxy
}
```

unset

```
unset zone zone rp
{
address ip_addr | candidate | proxy
}
```

Keywords and Variables

address

```
set zone zone rp address ip_addr mgroup-list number [ always ]
unset zone zone rp address ip_addr
```

address

Configures a static RP for the multicast groups specified in the access list. If no group is specified, then this RP is used for any multicast group that has no RP.

- **zone** *zone* Specifies the zone of the RP.
- **address** *ip_addr* Specifies the IP address of the RP. This IP address can also be the IP address an interfaces on the device.
- **mgroup-list** *number* Specifies the access list that identifies the multicast group(s) mapped to the RP.
- **always** Specifies that this RP should always be used for the specifies multicast group even if there is a dynamic group-RP mapping for the same group.

bsr

```
get zone zone bsr
```

bsr Displays information about the bootstrap router in the zone.

candidate

```
set zone zone rp candidate interface interface mgroup-list number holdtime number  
set zone zone rp candidate interface interface mgroup-list number priority number  
unset zone zone rp candidate
```

candidate Configures an RP candidate in the specified zone.

- **zone** *zone* Specifies the zone of the RP.
- **interface** *interface* Specifies the interface that is advertised as the RP candidate.
- **mgroup-list** *number* Specifies the access list which identifies the multicast group(s) for which the interface is the RP candidate.
- **holdtime** *number* Specifies the holdtime advertised to the bootstrap router.
- **priority** *number* Specifies the priority of the interface as the RP candidate.

proxy

```
set zone zone rp proxy
```

proxy Configures a proxy RP.

rp

```
get zone zone rp {...}
```

rp

Displays information about the RP in the specified zone.

- **active** Displays information about the RP that is sending multicast traffic to the multicast group in the specified zone.
- **all** Displays all RPs, including candidate RPs, in the specified zone.
- **candidate** Displays the configured RP in the zone.
- **mgroup** *ip_addr* Displays the RP for the specifies multicast group.
- **proxy** Displays the proxy-RP for the specified zone.

vrouter

Description: Use the **vrouter** commands to configure the NetScreen device to perform as a local virtual router.

Syntax

get

```
get vrouter name_str
[
  mcore |
  mroute [ mgroup ip_addr1 ] [ source ip_addr2 [ iif interface ] ] |
  protocol pim |
]
```

set

```
set vrouter name_str
[
  max-routes number |
  mroute
  {
    mgroup ip_addr1 source ip_addr2 iif interface1 oif interface2
      [outgroup ip_addr3 ] |
    max-entries number |
    multiple-iif-enable
  }
  protocol pim
]
```

unset

```
unset vrouter name_str
[
| ospf | imported | static } |
  max-routes |
    mroute
    {
    mgroup ip_add1 source ip_addr2 iif interface1 oif interface2 |
    max-entries |
    multiple-iif-enable
    }
  protocol pim |
]
```

Keywords and Variables

max-entries

```
set mroute max-entries number
```

max-entries Specifies the maximum number of multicast route entries.

mcore

```
get vrouter name_str mcore
```

mcore Displays multicast routing information for each interface on which a multicast routing protocol is enabled.

mroute

```
get vrouter name_str mroute mgroup ip_addr1 [ source ip_addr2 ] [ iif interface1 ]
set vrouter name_str mroute max-entries number
set {...} mroute mgroup ip_addr1 source ip_addr2 iif interface1 oif interface2
set {...} mroute mgroup ip_addr1 {...} out-group ip_addr3
set vrouter name_str mroute multiple-iif-enable
unset vrouter name_str mroute max-entries
unset {...} mroute mgroup ip_addr1 source ip_addr2 iif interface1 oif interface2
unset vrouter name_str mroute multiple-iif-enable
```

- mroute** Configures a static multicast route in the specified virtual router.
- *ip_addr1* is the multicast group address of the route
 - *ip_addr2* is the source address of the multicast data
 - *interface1* is the incoming interface of the multicast data
 - *interface2* is the outgoing interface of the multicast data
 - *ip_addr3* is the multicast group address on the outgoing interface
- multiple-iif-enable** Permits multiple multicast routes for the same source and group.

multiple-iif-enable

set multiple-iif-enable

- multiple-iif-enable** Allows multiple multicast routes for the same source and multicast group.

New Messages - Multicast

This chapter introduces all the new NetScreen messages for this release. Each message is presented, its meaning explained, and—where appropriate—an administrative action recommended. The messages are grouped by message type, and then within that type by severity level, from the most severe to the least.

- “IGMP” on page 182
- “Multicast” on page 190
- “PIM” on page 195

For a complete list of NetScreen log messages, refer to the *NetScreen Message Log Reference Guide*.

IGMP

These messages relate to the Internet Group Management Protocol (IGMP).

Warning

Message IGMP: total group number <number> (or per interface number <number>) exceed limit

Meaning The number of groups joined by the interface exceeds the maximum allowed.

Action Use the **clear igmp interface** <interface> **group** command to delete multicast groups. For more information on this command, see “igmp” on page 132.

Message igmp received IGMPv2 query on IGMPv1 interface

Meaning An interface running IGMPv1 received an IGMPv2 query.

Action No recommended action.

Message igmp received IGMPv1 query on IGMPv2 interface

Meaning An interface running IGMPv2 received an IGMPv1 query.

Action No recommended action

Message igmp rcv_membership_report: group <mcast_addr> doesn't match dest ip <ip_addr>
Meaning The membership report that was received for the specified multicast group did not have the correct IP address.
Action No recommended action.

Message <interface> receive wrong group <mcast_addr> igmp membership report
Meaning The interface received a membership report for a group that it has not joined.
Action No recommended action

Notification (00045)

Message IGMP host instance was deleted on interface <interface>
Meaning An admin removed the IGMP host instance from the specified interface.
Action No recommended action

Message IGMP host instance was created on interface <interface>
Meaning An admin created the IGMP host instance on the specified interface.
Action No recommended action

- Message** IGMP router instance was deleted on interface <interface>
- Meaning** An admin removed the IGMP router instance from the specified interface.
- Action** No recommended action.
-
- Message** IGMP router instance was created on interface <interface>
- Meaning** An admin created an IGMP router instance on the specified interface.
- Action** No recommended action.
-
- Message** IGMP function was disabled on interface <interface>
- Meaning** An admin disabled IGMP on the specified interface.
- Action** No recommended action
-
- Message** IGMP function was enabled on interface <interface>
- Meaning** An admin enabled IGMP on the specified interface.
- Action** No recommended action

Message IGMP will not do same subnet check on interface <interface>

Meaning The specified interface accepts IGMP messages from all sources, regardless of their subnet.

Action No recommended action.

Message IGMP will do same subnet check on interface <interface>

Meaning The specified interface accepts IGMP messages only from its own subnet.

Action No recommended action.

Message IGMP will not do router alert ip option check on interface <interface>

Meaning The specified interface does not check whether an IGMP packet has the router-alert IP option before it accepts the packet.

Action No recommended action.

Message IGMP will do router alert ip option check on interface <interface>

Meaning The specified interface checks whether an IGMP packet has the router-alert IP option before it accepts the packet. The interface drops all packets that do not have this option.

Action No recommended action.

Message IGMP is changed to <number> on interface <interface>

Meaning An admin changed the IGMP version that was enabled on the interface.

Action No recommended action

Message IGMP query interval is changed to <number> seconds on interface <interface>

Meaning An admin changed the IGMP query interval on the specified interface.

Action No recommended action

Message IGMP query max response time is changed to <number> seconds on interface <interface>

Meaning An admin changed the maximum response time on the specified interface.

Action No recommended action

Message IGMP leave interval is changed to <number> seconds on interface <interface>

Meaning An admin changed the leave interval on the specified interface.

Action No recommended action

Message IGMP last member query interval is changed to <number> seconds on interface <interface>

Meaning An admin changed the last member query interval on the specified interface.

Action No recommended action

Message IGMP routers accept id is changed to <id_num> on interface <interface>

Meaning An admin changed the access list that identifies the routers that are eligible for Querier election. Only the routers in the specified access list can be elected as Querier.

Action No recommended action

Message IGMP hosts accept id is changed to <id_num> on interface <interface>

Meaning An admin changed the access list that identifies the hosts from which the interface can accept IGMP messages.

Action No recommended action

Message IGMP groups accept id is changed to <id_num> on interface <interface>

Meaning An admin changed the access list that identifies the multicast groups the hosts on the specified interface can join.

Action No recommended action

Message IGMP group <mcast_addr> static flag were removed on interface <interface>

Meaning An admin deleted the static mapping between the multicast group and the specified interface.

Action No recommended action

Message IGMP all groups static flag were removed on interface <interface>

Meaning An admin deleted the static mapping between the multicast groups and the specified interface.

Action No recommended action

Message IGMP static group <mcast_addr> were added on interface <interface>

Meaning An admin manually added the multicast group to the specified interface.

Action No recommended action

Message IGMP group <mcast_addr> static flag were added on interface <interface>

Meaning An admin manually added the multicast group to the specified interface.

Action No recommended action

Message IGMP proxy is disabled on interface <interface>

Meaning An admin disabled the IGMP proxy feature on the specified interface.

Action No recommended action

Message IGMP proxy is enabled on interface <interface>

Meaning An admin enabled the IGMP proxy feature on the specified interface.

Action No recommended action

Message IGMP proxy always is disabled on interface

Meaning An admin disabled the feature that allows the interface to forward GMP messages in Querier and non-Querier mode. The interface can forward IGMP messages only if it is in Querier mode.

Action No recommended action

Message IGMP proxy always is enabled on interface <interface>

Meaning An admin enabled the feature that allows the interface to forward IGMP messages in Querier and non-Querier mode.

Action No recommended action

MULTICAST

These messages relate to the multicast routing protocols.

Warning (01001)

Message Error in initializing multicast

Meaning An error occurred when the NetScreen device started up.

Action Contact NetScreen technical support by visiting www.netscreen.com/cso. (Note: You must be a registered NetScreen customer.)

Message Failure in intialising the RPH Task

Meaning An error occurred when the NetScreen device started up.

Action Contact NetScreen technical support by visiting www.netscreen.com/cso. (Note: You must be a registered NetScreen customer.)

Message Failure in intialising the MDH Task

Meaning An error occurred when the NetScreen device started up.

Action Contact NetScreen technical support by visiting www.netscreen.com/cso. (Note: You must be a registered NetScreen customer.)

- Message** Unable to Register with IP for mcast data pkts
- Meaning** An error occurred when the NetScreen device started up.
- Action** Contact NetScreen technical support by visiting www.netscreen.com/cso. (Note: You must be a registered NetScreen customer.)
-

Warning (01002{

- Message** System wide multicast route limit exceeded,mroute add failed
- Meaning** The NetScreen device did not add the new multicast route to the multicast route table because the number of multicast route entries exceeded the maximum allowed. The maximum number of entries allowed depends on the NetScreen device.
- Action** Clear any unused multicast routes.
-
- Message** System wide multicast route maximum routes not added from last exceed - <number>
- Meaning** The NetScreen device did not add the specified number of multicast routes to the multicast route table because the number of multicast route entries exceeded the maximum allowed. The maximum number of entries allowed depends on the NetScreen device.
- Action** Clear any unused multicast routes.

Warning (01003)

Message VR: virtual router multicast route limit exceeded, mroute add failed

Meaning The NetScreen device did not add the new multicast route to the multicast route table because the number of multicast route entries exceeded the maximum configured for the virtual router.

Action You can remove the configured maximum number of entries with the **unset vrouter <name_str> mroute max-entries** command.

Message <name_str>: virtual router multicast route maximum routes not added from last exceed - <number>

Meaning The NetScreen device did not add the specified number of multicast routes to the multicast route table because the number of multicast route entries exceeded the maximum configured for the named virtual router.

Action You can remove the configured maximum number of entries with the **unset vrouter <name_str> mroute max-entries** command.

Notification (00048)

Message VR: static multicast route src=<ip_addr>, grp=<mcast_addr> input ifp = <interface1> output ifp = <interface2> added

Meaning An admin added the specified static multicast route to the multicast route table of the virtual router.

Action No recommended action

- Message** VR: static multicast route src=<ip_addr>, grp=<mcast_addr> ifp = <interface> deleted
- Meaning** An admin removed the specified static multicast route from the multicast route table of the virtual router.
- Action** No recommended action
-
- Message** VR: maximum multicast routeslimit removed
- Meaning** An admin removed the configured limit on the number of multicast routes allowed for the virtual router.
- Action** No recommended action
-
- Message** VR: maximum multicast routes limit configured to <number> max_mroute_entries
- Meaning** An admin set the maximum number of allowed multicast routes for the virtual router.
- Action** No recommended action
-
- Message** Remove multicast policy from src-zone src-ip to dst-zone dst-ip of message type
- Meaning** An admin removed the specified multicast policy.
- Action** No recommended action

Message Add multicast policy from src-zone src-ip to dst-zone dst-ip of message type

Meaning An admin created the specified multicast policy.

Action No recommended action

PIM

These messages relate to the Protocol Independent Multicast-Sparse Mode (PIM-SM) protocol.

Notification 00046

Message PIM protocol configured on interface <interface>

Meaning An admin configured the PIM protocol on the specified interface.

Action No recommended action

Message PIM protocol enabled on interface <interface>

Meaning An admin enabled PIM on the specified interface.

Action No recommended action

Message PIM interface <interface>'s DR priority set to <number>

Meaning An admin set the designated router (DR) priority of the interface.

Action No recommended action

Message PIM interface <interface>'s Join-Prune Interval set to <number>

Meaning An admin set the interval at which the interface sends join-prune messages to its upstream routers.

Action No recommended action

Message PIM interface <interface>'s Hello Interval set to <number>

Meaning An admin set the interval at which the interface sends hello messages to its neighboring routers.

Action No recommended action

Message PIM interface <interface> set pim accept neighbors <number>

Meaning An admin set the feature that restricts the interface to forming adjacencies with the routers in the specified access list.

Action No recommended action

- Message** PIM interface <interface> configured as boot-strap border
- Meaning** An admin configured the specified interface as a bootstrap border. A bootstrap border processes bootstrap messages but does not forward them to any other interface.
- Action** No recommended action
-
- Message** PIM interface <interface>'s hello holdtime set to <number>
- Meaning** An admin set the hello holdtime on the specified interface.
- Action** No recommended action
-
- Message** PIM protocol unconfigured on interface <interface>
- Meaning** An admin unset the PIM protocol on the specified interface.
- Action** No recommended action
-
- Message** PIM protocol disabled on interface <interface>
- Meaning** An admin disabled PIM on the specified interface.
- Action** No recommended action

Message PIM interface <interface>'s DR priority set to default

Meaning An admin unset the configured DR priority to the default DR priority, which is 1.

Action No recommended action

Message PIM interface <interface> unset pim accept neighbors

Meaning An admin disabled the feature that restricts the interface to forming adjacencies with the routers in the specified access list.

Action No recommended action

Message PIM interface <interface> boot-strap border unconfigured

Meaning An admin unset the interface as a bootstrap border.

Action No recommended action

Message <name_str>: PIM protocol disabled

Meaning The PIM protocol was disabled on the specified virtual router.

Action No recommended action

Message <interface>: PIM SPT threshold reset

Meaning An admin reset the shortest-path tree (SPT) threshold of the specified interface.

Action No recommended action

Message vr <vrouter> unset pim mgroup <mcast_addr> rp accept lists

Meaning An admin removed the restriction that limits the multicast group to accepting traffic only from the RPs specified in an access list.

Action No recommended action

Message vr <vrouter> unset pim mgroup <mcast_addr> accept sources

Meaning An admin removed the restriction that limits the multicast group to accepting traffic only from the sources specified in an access list.

Action No recommended action

Message vr <vrouter> unset pim accept groups

Meaning An admin removed the restriction that limits the virtual router to accepting traffic from the multicast groups specified in an access list

Action No recommended action

Message <vrouter>: PIM RP address <ip_addr> removed

Meaning An admin removed the specified static RP from the virtual router.

Action No recommended action

Message <vrouter>: PIM RP candidate removed

Meaning An admin removed the RP candidate from the specified virtual router.

Action No recommended action

Message vr <vrouter>, zone <zone> pim remove rp proxy

Meaning An admin deleted the proxy RP instance from the specified zone in the named virtual router.

Action No recommended action

Message <vrouter>: PIM protocol enabled

Meaning An admin enabled PIM on the specified virtual router.

Action No recommended action

Message <vrouter>: PIM SM SPT threshold configured to infinity

Meaning An admin set the SPT threshold to infinity; therefore the virtual router never joins the SPT.

Action No recommended action

Message <vrouter>: PIM SM SPT threshold configured at <number>

Meaning An admin set the SPT threshold to the specified value.

Action No recommended action

Message vr <vrouter> set pim mgroup <mcast_addr> accept sources <id_num>

Meaning The named multicast group can accept multicast traffic only from the sources in the specified access list.

Action No recommended action

Message vr <vrouter> set pim mgroup <mcast_addr> rp accept list id <id_num>

Meaning The named multicast group can accept multicast traffic only from the RPs in the specified access list.

Action No recommended action

- Message** vr <vrouter> configure pim accept group list <id_num>
- Meaning** The named virtual router can process PIM messages from the multicast groups in the specified access list.
- Action** No recommended action
-
- Message** <vrouter>: set zone <zone> proxy rp
- Meaning** An admin configured the a proxy RP on the specified zone in the named virtual router.
- Action** No recommended action
-
- Message** <vrouter>: PIM RP address <ip_addr> configured for access list <id_num>
- Meaning** An admin added the RP address to the specified access list.
- Action** No recommended action
-
- Message** <vrouter>: PIM RP candidate interface <interface> configured for access list
- Meaning** An admin added the RP candidate to the specified access list.
- Action** No recommended action

Notification (00546)

- Message** <vrouter> PIMSM address <mcast_addr> in access-list not a multicast address
- Meaning** The IP address in the specified access list is not a valid multicast group address.
- Action** Replace the invalid IP address with a valid multicast group address.

