JUNIPER NETWORKS

Network Address Translation

Published: 2013-02-15

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Network Address Translation

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at http://www.juniper.net/support/eula.html. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at http://www.juniper.net/techpubs/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at http://www.juniper.net/books.

Supported Platforms

For the features described in this document, the following platforms are supported:

- M Series
- T Series
- MX Series
- J Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   system {
     scripts {
       commit {
         file ex-script.xsl;
       }
     }
   }
   interfaces {
     fxp0 {
       disable;
       unit 0 {
         family inet {
           address 10.0.0.1/24;
         }
       }
     }
   }
   ```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

   ```
   [edit]
   user@host# load merge /var/tmp/ex-script.conf
   load complete
   ```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   commit {
     file ex-script-snippet.xsl; }
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

   ```
   [edit]
   user@host# edit system scripts
   [edit system scripts]
   ```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

   ```
   [edit system scripts]
   user@host# load merge relative /var/tmp/ex-script-snippet.conf
   load complete
   ```

For more information about the **load** command, see the CLI User Guide.

Documentation Conventions

Table 1 on page xiii defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.

Table 2 on page xiii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the**configure** command: user@host> **configure**
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`

Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
Italic text like this	• Introduces or emphasizes important new terms. • Identifies book names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS System Basics Configuration Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Enclose optional keywords or variables.	**stub <default-metric** ***metric*****>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Enclose a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

Indention and braces ({ })	Identify a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at https://www.juniper.net/cgi-bin/docbugreport/ . If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit http://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://www.juniper.net/alerts/

- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://tools.juniper.net/SerialNumberEntitlementSearch/

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at http://www.juniper.net/cm/.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see http://www.juniper.net/support/requesting-support.html.

PART 1

Overview

CHAPTER 1

Network Address Translation

Network Address Translation Overview

Types of NAT

The types of Network Address Translation (NAT) supported by the Junos OS are described in the following sections:

NAT Concept and Facilities Overview

NAT is a mechanism for translating IP addresses. NAT provides the technology used to support a wide range of networking goals, including:

- Concealing a set of host addresses on a private network behind a pool of public addresses.
- Providing a security measure to protect the host addresses from direct targeting in network attacks.
- Providing a tool set for coping with IPv4 address depletion and IPV6 transition issues.

The Junos OS provides carrier-grade NAT (CGN) for IPv4 and IPv6 networks, and facilitates the transit of traffic between different types of networks.

The multiservices Dense Port Concentrator (DPC) and multiservices PIC interfaces support the following types of traditional CGN:

- Static-source translation—Allows you to hide a private network. It features a one-to-one mapping between the original address and the translated address; the mapping is configured statically. For more information, see "Basic NAT" on page 4.
- Dynamic-source translation—Includes two options: dynamic address-only source translation and Network Address Port Translation (NAPT):
  - Dynamic address-only source translation—A NAT address is picked up dynamically from a source NAT pool and the mapping from the original source address to the translated address is maintained as long as there is at least one active flow that uses this mapping. For more information, see "Dynamic NAT" on page 6.
  - NAPT—Both the original source address and the source port are translated. The translated address and port are picked up from the corresponding NAT pool. For more information, see "NAPT" on page 5.
- Static destination translation—Allows you to make selected private servers accessible. It features a one-to-one mapping between the translated address and the destination address; the mapping is configured statically. For more information, see "Static Destination NAT" on page 5.
- Protocol translation—Allows you to assign addresses from a pool on a static or dynamic basis as sessions are initiated across IPv4 or IPv6 boundaries. For more information, see "NAT-PT" on page 5, "NAT-PT with DNS ALG" on page 6, and "Stateful NAT64" on page 7.
- Encapsulation of IPv4 packets into IPv6 packets using softwires—Enables packets to travel over softwires to a carrier-grade NAT endpoint where they undergo source-NAT processing to hide the original source address. For more information, see Tunneling Services for IPv4-to-IPv6 Transition Overview.

The Junos OS supports NAT functionality described in IETF RFCs and Internet drafts, as shown in "Supported NAT and SIP Standards" in *Standards Supported in Junos OS 11.4*.

IPv4-to-IPv4 Basic NAT

Basic Network Address Translation or Basic NAT is a method by which IP addresses are mapped from one group to another, transparent to end users. Network Address Port Translation or NAPT is a method by which many network addresses and their TCP/UDP ports are translated into a single network address and its TCP/UDP ports. Together, these two operations, referred to as traditional NAT, provide a mechanism to connect a realm with private addresses to an external realm with globally unique registered addresses.

Traditional NAT, specified in RFC 3022, *Traditional IP Network Address Translator*, is fully supported by the Junos OS. In addition, NAPT is supported for source addresses.

Basic NAT

With Basic NAT, a block of external addresses is set aside for translating addresses of hosts in a private domain as they originate sessions to the external domain. For packets

outbound from the private network, Basic NAT translates source IP addresses and related fields such as IP, TCP, UDP, and ICMP header checksums. For inbound packets, Basic NAT translates the destination IP address and the checksums listed above.

NAPT

Use NAPT to enable the components of the private network to share a single external address. NAPT translates the transport identifier (for example, TCP port number, UDP port number, or ICMP query ID) of the private network into a single external address. NAPT can be combined with Basic NAT to use a pool of external addresses in conjunction with port translation.

For packets outbound from the private network, NAPT translates the source IP address, source transport identifier (TCP/UDP port or ICMP query ID), and related fields, such as IP, TCP, UDP, and ICMP header checksums. For inbound packets, NAPT translates the destination IP address, the destination transport identifier, and the IP and transport header checksums.

NAT-PT

NAT-Protocol Translation (NAT-PT) is an obsolete IPv4-to-IPv6 transition mechanism and is no longer recommended. NAT64 is the newer, recommended solution. Using a pool of IPv4 addresses, NAT-PT assigns addresses from that pool to IPv6 nodes on a dynamic basis as sessions are initiated across IPv4 or IPv6 boundaries. Inbound and outbound sessions must traverse the same NAT-PT router so that it can track those sessions. RFC 2766, *Network Address Translation - Protocol Translation (NAT-PT)*, recommends the use of NAT-PT for translation between IPv6-only nodes and IPv4-only nodes, and *not* for IPv6-to-IPv6 translation between IPv6 nodes or IPv4-to-IPv4 translation between IPv4 nodes.

NAT-PT, specified in RFC 2766, *Network Address Translation - Protocol Translation (NAT-PT)* and obsoleted by RFC 2766, *Reasons to Move Network Address Translator - Protocol Translator (NAT-PT) to Historic Status*, is still supported by the Junos OS.

Static Destination NAT

Use static destination NAT to translate the destination address for external traffic to an address specified in a destination pool. The destination pool contains one address and no port configuration.

For more information about static destination NAT, see RFC 2663, *IP Network Address Translator (NAT) Terminology and Considerations.*

Twice NAT

In Twice NAT, both the source and destination addresses are subject to translation as packets traverse the NAT router. The source information to be translated can be either address only or address and port. For example, you would use Twice NAT when you are connecting two networks in which all or some addresses in one network overlap with addresses in another network (whether the network is private or public). In traditional NAT, only one of the addresses is translated.

To configure Twice NAT, you must specify both a destination address and a source address for the match direction, pool or prefix, and translation type.

You can configure application-level gateways (ALGs) for ICMP and traceroute under stateful firewall, NAT, or class-of-service (CoS) rules when Twice NAT is configured in the same service set. These ALGs cannot be applied to flows created by the Packet Gateway Control Protocol (PGCP). Twice NAT does not support other ALGs. By default, the Twice NAT feature can affect IP, TCP, and UDP headers embedded in the payload of ICMP error messages.

Twice NAT, specified in RFC 2663, *IP Network Address Translator (NAT) Terminology and Considerations*, is fully supported by the Junos OS.

IPv6 NAT

IPv6-to-IPv6 NAT (NAT66), defined in Internet draft draft-mrw-behave-nat66-01, *IPv6-to-IPv6 Network Address Translation (NAT66)*, is fully supported by the Junos OS.

NAT-PT with DNS ALG

NAT-PT and Domain Name System (DNS) ALG are used to facilitate communication between IPv6 hosts and IPv4 hosts. Using a pool of IPv4 addresses, NAT-PT assigns addresses from that pool to IPv6 nodes on a dynamic basis as sessions are initiated across IPv4 or IPv6 boundaries. Inbound and outbound sessions must traverse the same NAT-PT router so that it can track those sessions. RFC 2766, *Network Address Translation - Protocol Translation (NAT-PT)*, recommends the use of NAT-PT for translation between IPv6-only nodes and IPv4-only nodes, and *not* for IPv6-to-IPv6 translation between IPv6 nodes or IPv4-to-IPv4 translation between IPv4 nodes.

DNS is a distributed hierarchical naming system for computers, services, or any resource connected to the Internet or a private network. The DNS ALG is an application-specific agent that allows an IPv6 node to communicate with an IPv4 node and vice versa.

When DNS ALG is employed with NAT-PT, the DNS ALG translates IPv6 addresses in DNS queries and responses to the corresponding IPv4 addresses and vice versa. IPv4 name-to-address mappings are held in the DNS with “A” queries. IPv6 name-to-address mappings are held in the DNS with “AAAA” queries.

NOTE: For IPv6 DNS queries, use the **do-not-translate-AAAA-query-to-A-query** statement at the **[edit applications application *application-name*]** hierarchy level.

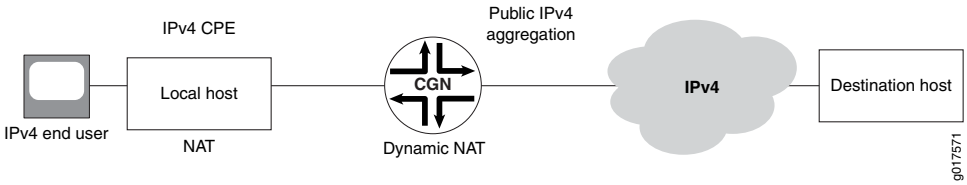
Related Documentation

- Configuring NAT Rules on page 21
- Configuring NAT-PT on page 51
- Example: Configuring NAT-PT on page 71

Dynamic NAT

Dynamic NAT flow is shown in Figure 1 on page 7.

Figure 1: Dynamic NAT Flow



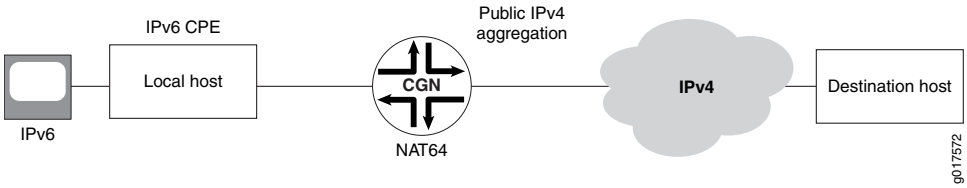
With dynamic NAT, you can map a private IP address (source) to a public IP address drawing from a pool of registered (public) IP addresses. NAT addresses from the pool are assigned dynamically. Assigning addresses dynamically also allows a few public IP addresses to be used by several private hosts, in contrast with an equal-sized pool required by source static NAT.

For more information about dynamic address translation, see RFC 2663, *IP Network Address Translator (NAT) Terminology and Considerations*.

Stateful NAT64

Stateful NAT64 flow is shown in Figure 2 on page 7.

Figure 2: Stateful NAT64 Flow



Stateful NAT64 is a mechanism to move to an IPv6 network and at the same time deal with IPv4 address depletion. By allowing IPv6-only clients to contact IPv4 servers using unicast UDP, TCP, or ICMP, several IPv6-only clients can share the same public IPv4 server address. To allow sharing of the IPv4 server address, NAT64 translates incoming IPv6 packets into IPv4 (and vice versa).

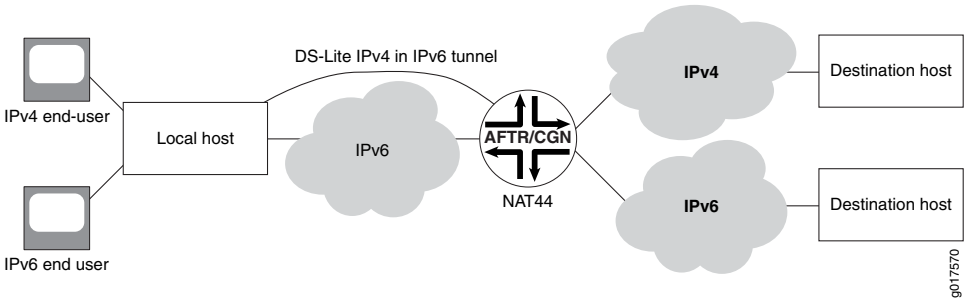
When stateful NAT64 is used in conjunction with DNS64, no changes are usually required in the IPv6 client or the IPv4 server. DNS64 is out of scope of this document because it is normally implemented as an enhancement to currently deployed DNS servers.

Stateful NAT64, specified in RFC 6146, *Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers*, is fully supported by the Junos OS.

Dual-Stack Lite

Dual-stack lite (DS-Lite) flow is shown in Figure 3 on page 8.

Figure 3: DS-Lite Flow



DS-Lite employs IPv4-over-IPv6 tunnels to cross an IPv6 access network to reach a carrier-grade IPv4-IPv4 NAT. This facilitates the phased introduction of IPv6 on the Internet by providing backward compatibility with IPv4.

Related Documentation

- Configuring a DS-Lite Softwire Concentrator

PART 2

Configuration

CHAPTER 2

Configuration Tasks

Configuring Addresses and Ports for Use in NAT Rules

For information about configuring translated addresses, see the following sections:

Configuring Pools of Addresses and Ports

You can use the **pool** statement to define the addresses (or prefixes), address ranges, and ports used for Network Address Translation (NAT). To configure the information, include the **pool** statement at the **[edit services nat]** hierarchy level:

```
[edit services nat]
pool nat-pool-name {
```

```
    address ip-prefix</prefix-length>;
    address-range low minimum-value high maximum-value;
    port (automatic | range low minimum-value high maximum-value);
      preserve-parity;
      preserve-range {
    }
}
```

To configure pools for traditional NAT, specify either a destination pool or a source pool.

With static source NAT and dynamic source NAT, you can specify multiple IPv4 addresses (or prefixes) and IPv4 address ranges. Up to 32 prefixes or address ranges (or a combination) can be supported within a single pool.

With static destination NAT, you can also specify multiple address prefixes and address ranges in a single term. Multiple destination NAT terms can share a destination NAT pool. However, the netmask or range for the **from** address must be smaller than or equal to the netmask or range for the destination pool address. If you define the pool to be larger than required, some addresses will not be used. For example, if you define the pool size as 100 addresses and the rule specifies only 80 addresses, the last 20 addresses in the pool are not used.

For constraints on specific translation types, see "Configuring Actions in NAT Rules" on page 24.

With source static NAT, the prefixes and address ranges cannot overlap between separate pools.

In an address range, the **low** value must be a lower number than the **high** value. When multiple address ranges and prefixes are configured, the prefixes are depleted first, followed by the address ranges.

When you specify a port for dynamic source NAT, address ranges are limited to a maximum of 65,000 addresses, for a total of (65,000 x 65,535) or 4,259,775,000 flows. A dynamic NAT pool with no address port translation supports up to 65,535 addresses. There is no limit on the pool size for static source NAT.

Preserve Range and Preserve Parity

You can configure your carrier-grade NAT (CGN) to preserve the range or parity of the packet source port when it allocates a source port for an outbound connection. You can configure the preserve parity and preserve range options under the NAT pool definition by including the **preserve-range** and **preserve-parity** configuration statements at the **[edit services nat pool *poolname* port]** hierarchy level.

- Preserve range—RFC 4787, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*, defines two ranges: 0 through 1023, and 1024 through 65,535. When the **preserve-range** knob is configured and the incoming port falls into one of these ranges, CGN allocates a port from that range only. However, if there is no available port in the range, the port allocation request fails and that session is not created. The failure is reflected on counters and system logging, but no Internet Control Message Protocol (ICMP) message is generated. If this knob is not configured, allocation is based on the configured port range without regard to the port range that contains the incoming

port. The exception is some application-level gateways (ALGs), such as hello, that have special zones.

- Preserve parity—When the **preserve-parity** knob is configured, CGN allocates a port with the same even or odd parity as the incoming port. If the incoming port number is odd or even, the outgoing port number should correspondingly be odd or even. If a port number of the desired parity is not available, the port allocation request fails, the session is not created, and the packet is dropped.

Configuring Address Pools for Network Address Port Translation

With Network Address Port Translation (NAPT), you can configure up to 32 address ranges with up to 65,536 addresses each.

The **port** statement specifies port assignment for the translated addresses. To configure automatic assignment of ports, include the **port automatic** statement at the **[edit services nat pool *nat-pool-name*]** hierarchy level. To configure a specific range of port numbers, include the **port range low *minimum-value* high *maximum-value*** statement at the **[edit services nat pool *nat-pool-name*]** hierarchy level.

The Junos OS provides several alternatives for allocating ports:

Round-Robin Allocation

To configure round-robin allocation for NAT pools, include the **address-allocation round-robin** configuration statement at the **[edit services nat pool *pool-name*]** hierarchy level. When you use round-robin allocation, one port is allocated from each address in a range before repeating the process for each address in the next range. After ports have been allocated for all addresses in the last range, the allocation process wraps around and allocates the next unused port for addresses in the first range.

- The first connection is allocated to the address:port 100.0.0.1:3333.
- The second connection is allocated to the address:port 100.0.0.2:3333.
- The third connection is allocated to the address:port 100.0.0.3:3333.
- The fourth connection is allocated to the address:port 100.0.0.4:3333.
- The fifth connection is allocated to the address:port 100.0.0.5:3333.
- The sixth connection is allocated to the address:port 100.0.0.6:3333.
- The seventh connection is allocated to the address:port 100.0.0.7:3333.
- The eighth connection is allocated to the address:port 100.0.0.8:3333.
- The ninth connection is allocated to the address:port 100.0.0.9:3333.

- The tenth connection is allocated to the address:port 100.0.0.10:3333.
- The eleventh connection is allocated to the address:port 100.0.0.11:3333.
- The twelfth connection is allocated to the address:port 100.0.0.12:3333.
- Wraparound occurs and the thirteenth connection is allocated to the address:port 100.0.0.1:3334.

Sequential

With sequential allocation, the next available address in the NAT pool is selected only when all the ports available from an address are exhausted.

NOTE: This legacy implementation provides backward compatibility.

The NAT pool called **napt** in the following configuration example uses the sequential implementation:

```
pool napt {
   address-range low 100.0.0.1 high 100.0.0.3;
   address-range low 100.0.0.4 high 100.0.0.6;
   address-range low 100.0.0.8 high 100.0.0.10;
   address-range low 100.0.0.12 high 100.0.0.13;
   port {
      range low 3333 high 3334;
   }
}
```

In this example, the ports are allocated starting from the first address in the first address-range, and allocation continues from this address until all available ports have been used. When all available ports have been used, the next address (in the same address-range or in the following address-range) is allocated and all its ports are selected as needed. In the case of the example **napt** pool, the tuple address, port 100.0.0.4:3333, is allocated only when all ports for all the addresses in the first range have been used.

- The first connection is allocated to the address:port 100.0.0.1:3333.
- The second connection is allocated to the address:port 100.0.0.1:3334.
- The third connection is allocated to the address:port 100.0.0.2:3333.
- The fourth connection is allocated to the address:port 100.0.0.2:3334, and so on.

Port Block Allocation

Carriers track subscribers using the IP address (RADIUS or DHCP) log. If they use CGN, an IP address is shared by multiple subscribers, and the carrier must track the IP address and port, which are part of the NAT log. Because ports are used and reused at a very high rate, tracking subscribers using the log becomes difficult due to the large number of messages, which are difficult to archive and correlate. By enabling the allocation of ports in blocks, port block allocation can significantly reduce the number of logs, making it easier to track subscribers.

Port block allocation is supported on MX series routers wiith MultiServices Dense Port Concentrators (MS-DPCs).

- Secured Port Block Allocation on page 15
- Deterministic Port Block Allocation on page 15

Secured Port Block Allocation

When allocating blocks of ports, the most recently allocated block is the current active block. New requests for NAT ports are served from the active block. Ports are allocated randomly from the current active block.

When you configure secured port block allocation, you can specify the following:

- **block-size**
- **max-blocks-per-address**
- **active-block-timeout**

Related Documentation

- Configuring Secured Port Block Allocation

Deterministic Port Block Allocation

You can configure NAT algorithm-based allocation of blocks of destination ports. By specifying **deterministic-port-block-allocation blocksize *blocksize*** at the **[edit services nat pool *poolname* port]** hierarchy level, you ensure that an incoming (source) IP address and port always map to the same destination IP address and port, thus eliminating the need for the address translation logging. When you use deterministic port block allocation, you must specify **deterministic-nat44** as the **translation-type** in your NAT rule.

For detailed information on how to configure deterministic port block allocation, see Configuring Deterministic Port Block Allocation.

Understanding Deterministic Port Block Allocation Algorithms

The effectiveness of your implementation of deterministic port block allocation depends on your analysis of your subscriber requirements. The block size you provide indicates how many ports will be made available for each incoming subscriber address in the range the **from** clause specified in the applicable NAT rule. The allocation algorithm computes an offset value to determine the outgoing port. A reverse algorithm is used to derive the originating subscriber address.

NOTE: In order to track subscribers without using logs, an ISP must use a reverse algorithm to derive a subscriber (source) addresses from translated addresses.

Deterministic Port Block Allocation Algorithm Usage

When you have configured deterministic port block allocation, you can use the **show services nat deterministic-nat internal-host** and **show services nat deterministic-nat nat-port-block** commands to show forward and reverse mapping. However, mappings will change if you reconfigure your deterministic port block allocation block size or the **from** clause for your NAT rule. In order to provide historical information on mappings, we recommend that you write scripts that can show specific mappings for prior configurations.

The following variables are used in forward calculation (private subscriber IP address to public IP address) and reverse calculation (public IP address to private subscriber IP address):

- Pr_Prefix—Any pre-NAT IPv4 subscriber address
- Pr_Port—Any pre-NAT protocol port
- Block_Size—Number of ports configured to be available for each Pr_Prefix
- Base_PR_Prefix—First usable pre-NAT IPv4 subscriber address in a "from" clause match condition
- Base_PU_Prefix—First usable post-NAT IPv4 subscriber address configured in the NAT pool.
- Pu_Port_Range_Start—1024 (ports 0 through 1023 are not used when **port automatic** is configured)
- Pr_Offset—Pr_Prefix – Base_Pr_Prefix
- PR_Port_Offset—Pr_Offset * Block_Size
- Pu_Prefix—Post-NAT address for a given Pr_Prefix
- Pu_Start_Port—Post-NAT start port for a flow from a given Pr_Prefix
- Pu_Actual_Port—Post-NAT port seen on a reverse flow
- Port_Range_Per_Pu_IP—64,512 (65,536 – 1024), constant as privileged ports are unused due to "port automatic" configuration)
- Pu_Offset—Pu_Prefix – Base_Pu_Prefix
- Pu_Port_Offset—(Pu_Offset * Port_Range_Per_Pu_IP) + (Pu_Actual_Port – Pu_Port_Start_Port)

NOTE: If **block-size** is configured as zero, the block size per user is computed by the Junos OS by the following formula: (64512 * Total no. of IP addresses in the NAT Pool) / Total no. of subscribers, where 64512 is the maximum number of ports available to port block allocation. 64512 is derived from (highest assignable port number - 1023). 1023 is used because regular port assignments start from 1024.

Algorithm Usage–Assume the following configuration:

```
services {
   nat {
      pool src-pool {
         address-range low 32.32.32.1 high 32.32.32.254;
         port {
            automatic {
               random-allocation;
            }
            deterministic-block-allocation {
               block-size 256;
            }
         }
      }
rule det-nat {
match-direction input;
   term t1 {
      from {
         source-address {
            10.1.0.0/16;
         }
      }
      then {
         translated {
            source-pool src-pool;
            translation-type {
               deterministic-napt44;
            }
         }
      }
   }
```

Forward Translation

1. Pr_Offset = Pr_Prefix – Base_Pr_Prefix
2. Pr_Port_Offset = Pr_Offset * Block_Size
3. Pu_Prefix = Base_Public_Prefix + (Pr_Port_Offset / Port_Range_Per_IP)
4. Pu_Start_Port = Pu_Port_Range_Start + (Pr_Port_Offset % Port_Range_Per_IP)

Using the sample configuration and assuming a subscriber flow sourced from 10.1.1.250:5000:

1. Pr_Offset = 10.1.1.250 – 10.1.0.1 = 505
2. Pu_Port_Offset = 505 * 256 = 129,280
3. Pu_Prefix = 32.32.32.1 + (129,280 / 64,512) = 32.32.32.3
4. Pu_Start_Port = 1,024 + (129,280 % 64,512) = 1,280
   - 10.1.1.250 is translated to 32.32.32.3.
   - The starting port is 1280. There are 256 ports available to the subscriber based on the configured block size. The available port range spans ports 1280 through 1535 (inclusive).

- The specific flow 10.1.1.250:5000 randomly assigns any of the ports in its range because random allocation was specified.

Reverse Translation

1. Pu_Offset = Pu_Prefix – Base_Pu_Prefix
2. Pu_Port_Offset = (Pu_Offset * Port_Range_Per_Pu_IP) + (Pu_Actual_Port – Pu_Port_Range_Start)
3. Subscriber_IP = Base_Pr_Prefix + (Pu_Port_Offset / Block_Size)

The reverse translation is determined as follows. Assume a flow returning to 32.32.32.3:1400.

1. Pu_Offset = 32.32.32.3 – 32.32.32.1 = 2
2. Pu_Port_Offset = (2 * 64,512) + (1400 - 1024) = 129,400
3. Subscriber_IP = 10.1.0.1 + (129,400 / 256) = 10.1.1.250

NOTE: In reverse translation, only the original private IP address can be derived, and not the original port in use. This is sufficiently granular for law enforcement requirements.

Deterministic Port Block Allocation Restrictions

When you configure deterministic port block allocation, you must be aware of the following restrictions. Violation of any restriction results in a commit error. The restrictions and their error messages are shown in Table 3 on page 18

Table 3: Deterministic Port Block Allocation Commit Constraints

Restiction	Error Message
The total number of deterministic NAT blocks must be greater than or equal to the 'from' clause addresses configured.	Number of addresses and port blocks combination in the NAT pool is less than number of addresses in 'from' clause
IPv6 addresses should not be used in deterministic NAT pool/from clause.	Invalid IP address in pool p1 with translation type deterministic-napt44 OR There is already a range configured with v4 address range
The **from** clause addresses should be same if the same deterministic NAT pool is used across multiple terms/rules. Only one **from** clause address/range should be specified if the same deterministic NAT pool is used across multiple terms/rules.	With translation-type deterministic-napt44, same 'from' address/range should be configured if pool is shared by multiple rules or terms
There shouldn't be address overlap between **except** entries in the **from** clause addresses.	overlapping address, in the 'from' clause between 'except' entries

Table 3: Deterministic Port Block Allocation Commit Constraints *(continued)*

Restiction	Error Message
A deterministic NAT pool cannot be used with other translation-types	Deterministic NAT pool cannot be used with other translation-types
Deterministic NAPT44 must use a source pool with deterministic-port-block-allocation configuration	Deterministic NAPT44 must use a source pool with deterministic-port-block-allocation configuration
If **address-allocation round-robin** is configured, a commit results in display of a warning indicating that this technique is not needed with translation-type deterministic-napt44 and is ignored.	Address allocation round-robin is not needed with translation-type deterministic-napt44
The total number of IP addresses assigned to a deterministic NAT pool should be less than or equal to 2^{24} (16777216).	Number of addresses in pool with deterministic-napt44 translation are limited to at most 16777216(2^24)

Additional Options for NAPT

The following options are available for NAPT:

- Preserving parity—Use the **preserve-parity** command to allocate even ports for packets with even source ports and odd ports for packets with odd source ports.
- Preserving range—Use the **preserve-range** command to allocate ports within a range from 0 to 1023, assuming the original packet contains a source port in the reserved range. This applies to control sessions, not data sessions.

Comparision of NAPT Implementation Methods

Table 4 on page 19 provides a feature comparison of available NAPT implementation methods.

Table 4: Comparison of NAPT Implementation Methods

Feature/Function	Dynamic Port Allocation	Secured Port Block Allocation	Deterministic Port Block Allocation
Users per IP	High	Medium	Low
Security Risk	Low	Medium	Medium
Log Utilization	High	Low	None (no logs necessary)
Security Risk Reduction	Random allocation	**active-block-timeout** feature	n/a
Increasing Users per IP	n/a	Configure multiples of smaller port blocks to maximize users/ public IP	Algorithm-based port allocation

Specifying Destination and Source Prefixes

You can directly specify the destination or source prefix used in NAT without configuring a pool.

To configure the information, include the **rule** statement at the **[edit services nat]** hierarchy level:

```
[edit services nat]
rule rule-name {
  term term-name {
    then {
      translated {
        destination-prefix prefix;
      }
    }
  }
}
```

Requirements for NAT Addresses

You must configure a specific address, a prefix, or the address-range boundaries:

- The following addresses, while valid in **inet.0**, cannot be used for NAT translation:
  - **0.0.0.0/32**
  - **127.0.0.0/8** (loopback)
  - **128.0.0.0/16** (martian)
  - **191.255.0.0/16** (martian)
  - **192.0.0.0/24** (martian)
  - **223.255.255.0/24** (martian)
  - **224.0.0.0/4** (multicast)
  - **240.0.0.0/4** (reserved)
  - **255.255.255.255** (broadcast)
- You can specify one or more IPv4 address prefixes in the **pool** statement and in the **from** clause of the NAT rule term. This enables you to configure source translation from a private subnet to a public subnet without defining a rule term for each address in the subnet. Destination translation cannot be configured by this method. For more information, see *Examples: Configuring NAT Rules.*
- When you configure static source NAT, the **address** prefix size you configure at the **[edit services nat pool *pool-name*]** hierarchy level must be larger than the **source-address** prefix range configured at the **[edit services nat rule *rule-name* term *term-name* from]** hierarchy level. The **source-address** prefix range must also map to a single subnet or range of IPv4 or IPv6 addresses in the **pool** statement. Any pool addresses that are not used by the **source-address** prefix range are left unused. Pools cannot be shared.

NOTE: When you include a NAT configuration that changes IP addresses, it might affect forwarding path features elsewhere in your router configuration, such as source class usage (SCU), destination class usage (DCU), filter-based forwarding, or other features that target specific IP addresses or prefixes.

NAT configuration might also affect routing protocol operation, because the protocol peering, neighbor, and interface addresses can be altered when routing protocols packets transit the Adaptive Services (AS) or Multiservices PIC.

Related Documentation

- Network Address Translation Overview on page 3

Configuring NAT Rules

To configure a NAT rule, include the **rule *rule-name*** statement at the **[edit services nat]** hierarchy level:

```
[edit services nat]
rule rule-name {
  match-direction (input | output);
  term term-name {
    from (Services NAT) {
      application-sets (Services NAT) set-name;
      applications (Services NAT) [ application-names ];
      destination-address (address | any-unicast) <except>;
      destination-address-range low minimum-value high maximum-value <except>;
      destination-prefix-list list-name <except>;
      source-address (address | any-unicast) <except>;
      source-address-range low minimum-value high maximum-value <except>;
      source-prefix-list list-name <except>;
    }
    then {
      no-translation;
      translated {
        address-pooling paired;
        destination-pool nat-pool-name;
        destination-prefix destination-prefix;
        dns-alg-pool dns-alg-pool;
        dns-alg-prefix dns-alg-prefix;
        filtering-type endpoint-independent;
        mapping-type endpoint-independent;
        overload-pool overload-pool-name;
        overload-prefix overload-prefix;
        source-pool nat-pool-name;
        source-prefix source-prefix;
        translation-type {
          (basic-nat-pt | basic-nat44 | basic-nat66 | dnat-44 | dynamic-nat44 | napt-44 |
            napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 | twice-dynamic-nat-44
            | twice-napt-44);
        }
        use-dns-map-for-destination-translation;
```

```
        }
        syslog;
      }
    }
  }
```

Each rule must include a **match-direction** statement that specifies the direction in which the match is applied.

In addition, each NAT rule consists of a set of terms, similar to a firewall filter. A term consists of the following:

- **from** statement—Specifies the match conditions and applications that are included and excluded.
- **then** statement—Specifies the actions and action modifiers to be performed by the router software.

The following sections explain how to configure the components of NAT rules:

- Configuring Match Direction for NAT Rules on page 22
- Configuring Match Conditions in NAT Rules on page 23
- Configuring Actions in NAT Rules on page 24

Configuring Match Direction for NAT Rules

Each rule must include a **match-direction** statement that specifies the direction in which the match is applied. To configure where the match is applied, include the **match-direction** statement at the **[edit services nat rule *rule-name*]** hierarchy level:

```
[edit services nat rule rule-name]
match-direction (input | output);
```

The match direction is used with respect to the traffic flow through the Multiservices DPC and Multiservices PICs. When a packet is sent to the PIC, direction information is carried along with it. The packet direction is determined based on the following criteria:

- With an interface service set, packet direction is determined by whether a packet is entering or leaving the interface on which the service set is applied.
- With a next-hop service set, packet direction is determined by the interface used to route the packet to the Multiservices DPC or Multiservices PIC. If the inside interface is used to route the packet, the packet direction is input. If the outside interface is used to direct the packet to the PIC or DPC, the packet direction is output. For more information about inside and outside interfaces, see "Configuring Service Sets to be Applied to Services Interfaces."
- On the Multiservices DPC and Multiservices PIC, a flow lookup is performed. If no flow is found, rule processing is performed. All rules in the service set are considered. During rule processing, the packet direction is compared against rule directions. Only rules with direction information that matches the packet direction are considered.

Configuring Match Conditions in NAT Rules

To configure NAT match conditions, include the **from** statement at the **[edit services nat rule *rule-name* term *term-name*]** hierarchy level:

```
[edit services nat rule rule-name term term-name]
from {
  application-sets set-name;
  applications [ application-names ];
  destination-address (address | any-unicast) <except>;
  destination-address-range low minimum-value high maximum-value <except>;
  destination-prefix-list list-name <except>;
  source-address (address | any-unicast) <except>;
  source-address-range low minimum-value high maximum-value <except>;
  source-prefix-list list-name <except>;
}
```

To configure traditional NAT, you can use the destination address, a range of destination addresses, the source address, or a range of source addresses as a match condition, in the same way that you would configure a firewall filter; for more information, see the Routing Policy Configuration Guide.

Alternatively, you can specify a list of source or destination prefixes by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level and then including either the **destination-prefix-list** or **source-prefix-list** statement in the NAT rule. For an example, see "Examples: Configuring Stateful Firewall Rules."

You can include application protocol definitions that you have configured at the **[edit applications]** hierarchy level; for more information, see "Configuring Application Protocol Properties":

- To apply one or more specific application protocol definitions, include the **applications** statement at the **[edit services nat rule *rule-name* term *term-name* from]** hierarchy level.
- To apply one or more sets of application protocol definitions that you have defined, include the **application-sets** statement at the **[edit services nat rule *rule-name* term *term-name* from]** hierarchy level.

NOTE: If you include one of the statements that specifies application protocols, the router derives port and protocol information from the corresponding configuration at the **[edit applications]** hierarchy level; you cannot specify these properties as match conditions. When matched rules include more than one ALG, the more specific ALG takes effect; for example, if the stateful firewall rule includes TCP and the NAT rule includes FTP, the NAT rule takes precedence.

You can configure ALGs for ICMP and traceroute under stateful firewall and NAT.

By default, NAT can restore IP, TCP, and UDP headers embedded in the payload of ICMP error messages. You can include the **protocol tcp** and **protocol udp** statements with the **application** statement for NAT configurations.

Configuring Actions in NAT Rules

To configure NAT actions, include the **then** statement at the **[edit services nat rule** *rule-name* **term** *term-name***]** hierarchy level:

```
[edit services nat rule rule-name term term-name]
then {
  no-translation;
  syslog;
  translated {
    destination-pool nat-pool-name;
    destination-prefix destination-prefix;
    source-pool nat-pool-name;
    source-prefix source-prefix;
    translation-type (basic-nat-pt | basic-nat44 | basic-nat66 | dnat-44 | dynamic-nat44
      | napt-44 | napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 |
      twice-dynamic-nat-44 | twice-napt-44);
  }
 }
}
```

The **no-translation** statement allows you to specify addresses that you want excluded from NAT.

The **syslog** statement enables you to record an alert in the system logging facility.

The **destination-pool**, **destination-prefix**, and **source-prefix** statements specify addressing information that you define by including the **pool** statement at the **[edit services nat]** hierarchy level; for more information, see "Configuring Addresses and Ports for Use in NAT Rules" on page 11.

The **translation-type** statement specifies the type of NAT used for source or destination traffic. The options are **basic-nat-pt**, **basic-nat66**, **dynamic-nat44**, **napt-44**, **napt-66**, **napt-pt**, **stateful-nat64**, **twice-basic-nat-44**, **twice-dynamic-nat-44**, and **twice-napt-44**.

The implementation details of the nine options of the **translation-type** statement are as follows:

- **basic-nat44**—This option implements the static translation of source IP addresses without port mapping. You must configure the **from source-address** statement in the match condition for the rule. The size of the address range specified in the statement must be the same as or smaller than the source pool. You must specify either a source pool or a destination prefix. The referenced pool can contain multiple addresses but you cannot specify ports for translation.

  NOTE: In an interface service set, all packets destined for the source address specified in the match condition are automatically routed to the services PIC, even if no service set is associated with the interface.

  NOTE: Prior to Junos OS Release 11.4R3, you could only use a source NAT pool in a single service set. As of Junos OS Release 11.4R3 and subsequent releases, you can reuse a source NAT pool in multiple service sets.

- **basic-nat66**—This option implements the static translation of source IP addresses without port mapping in IPv6 networks. The configuration is similar to the **basic-nat44** implementation, but with IPv6 addresses.
- **basic-nat-pt**—This option implements translation of addresses of IPv6 hosts, as they originate sessions to the IPv4 hosts in an external domain and vice versa. This option is always implemented with DNS ALG. You must define the source and destination pools of IPv4 addresses. You must configure one rule and define two terms. Configure the IPv6 addresses in the **from** statement in both the **term** statements. In the **then** statement of the first term within the rule, reference both the source and destination pools and configure **dns-alg-prefix**. Configure the source prefix in the **then** statement of the second term within the same rule.
- **dnat-44**—This option implements static translation of destination IP addresses without port mapping. The size of the pool address space must be greater than or equal to the destination address space. You must specify a name for the **destination pool** statement. The referenced pool can contain multiple addresses, ranges, or prefixes, as long as the number of NAT addresses in the pool is larger than the number of destination addresses in the **from** statement. You must include exactly one **destination-address** value at the **[edit services nat rule *rule-name* term *term-name* from]** hierarchy level; if it is a prefix, the size must be less than or equal to the pool prefix size. Any addresses in the pool that are not matched in the **destination-address** value remain unused, because a pool cannot be shared among multiple terms or rules.
- **dynamic-nat44**—This option implements dynamic translation of source IP addresses without port mapping. You must specify a **source-pool** name. The referenced pool must include an **address** configuration (for address-only translation).

  The **dynamic-nat44** address-only option supports translating up to 16,777,216 addresses to a smaller size pool. The requests from the source address range are assigned to the

addresses in the pool until the pool is used up, and any additional requests are rejected. A NAT address assigned to a host is used for all concurrent sessions from that host. The address is released to the pool only after all the sessions for that host expire. This feature enables the router to share a few public IP addresses between several private hosts. Because all the private hosts might not simultaneously create sessions, they can share a few public IP addresses.

- **napt-44**—This option implements dynamic translation of source IP addresses with port mapping. You must specify a name for the **source-pool** statement. The referenced pool must include a **port** configuration. If the port is configured as automatic or a port range is specified, then it implies that Network Address Port Translation (NAPT) is used.
- **napt-66**—This option implements dynamic address translation of source IP addresses with port mapping for IPv6 addresses. The configuration is similar to the **napt-44** implementation, but with IPv6 addresses.
- **napt-pt**—This option implements dynamic address and port translation for source and static translation of destination IP address. You must specify a name for the **source-pool** statement. The referenced pool must include a port configuration (for NAPT). Additionally, you must configure two rules, one for the DNS traffic and the other for the rest of the traffic. The rule meant for the DNS traffic should be DNS ALG enabled and the **dns-alg-prefix** statement should be configured. Moreover, the prefix configured in the **dns-alg-prefix** statement must be used in the second rule to translate the destination IPv6 addresses to IPv4 addresses.
- **stateful-nat64**—This option implements dynamic address and port translation for source IP addresses and prefix removal translation for destination IP addresses. You must specify the IPv4 addresses used for translation at the **[edit services nat pool]** hierarchy level. This pool must be referenced in the rule that translates the IPv6 addresses to IPv4.
- **twice-basic-nat-44**—This option implements static source and static destination translation for IPv4 addesses, thus combining **basic-nat44** for source and **dnat-44** for destination addresses.
- **twice-dynamic-nat-44**—This option implements source dynamic and destination static translation for IPv4 addresses, combining **dynamic-nat44** for source and **dnat-44** for destination addresses.
- **twice-napt-44**—This option implements source NAPT and destination static translation for IPv4 address, combining **napt-44** for source and **dnat-44** for destination addresses.

NOTE: When configuring NAT, if any traffic is destined for the following addresses and does not match a NAT flow or NAT rule, the traffic is dropped:

- Addresses specified in the from destination-address statement when you are using destination translation
- Addresses specified in the source NAT pool when you are using source translation

For more information on NAT methods, see RFC 2663, *IP Network Address Translator (NAT) Terminology and Considerations*.

Related Documentation

- Network Address Translation Overview on page 3

Configuring NAT Rule Sets

The **rule-set** statement defines a collection of NAT rules that determine what actions the router software performs on packets in the data stream. You define each rule by specifying a rule name and configuring terms. You then specify the order of the rules by including the **rule-set** statement at the **[edit services nat]** hierarchy level with a **rule** statement for each rule:

```
rule-set rule-set-name {
  rule rule-name;
}
```

The router software processes the rules in the order in which you specify them in the configuration. If a term in a rule matches the packet, the router performs the corresponding action and the rule processing stops. If no term in a rule matches the packet, processing continues to the next rule in the rule set. If none of the rules match the packet, no NAT action is performed on the packet. If a packet is destined to a NAT pool address, it is dropped.

Configuring Static Source Translation in IPv4 Networks

To configure the translation type as **basic-nat44**, you must configure the NAT pool and rule, service set with service interface, and trace options. This topic includes the following tasks:

- Configuring the NAT Pool and Rule on page 27
- Configuring the Service Set for NAT on page 29
- Configuring Trace Options on page 30

Configuring the NAT Pool and Rule

To configure the NAT pool, rule, and term:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# edit services nat
   ```

2. Configure the NAT pool with an address.

   ```
   [edit services nat]
   user@host# set pool pool name address address
   ```

   In the following example, the pool name is **src_pool** and the address is **10.10.10.2/32**.

   ```
   [edit services nat]
   user@host# set pool src_pool address 10.10.10.2/32
   ```

3. Configure the NAT rule and the match direction.

```
[edit services nat]
user@host# set rule rule-name match-direction match-direction
```

In the following example, the NAT rule name is **rule-basic-nat44** and the match direction is **input**.

```
[edit services nat]
user@host# set rule rule-basic-nat44 match-direction input
```

4. Configure the source address in the **from** statement.

```
[edit services nat]
user@host# set rule rule-basic-nat44 term term-name from from
```

In the following example, the term name is **t1** and the input condition is **source-address 3.1.1.2/32**.

```
[edit services nat]
user@host# set rule rule-basic-nat44 term t1 from source-address 3.1.1.2/32
```

5. Configure the NAT term action and properties of the translated traffic.

```
[edit services nat]
user@host# set rule rule-basic-nat44 term t1 then term-action translated-property
```

In the following example, the term action is **translated** and the property of the translated traffic is **source-pool src_pool**.

```
[edit services nat]
user@host# set rule rule-basic-nat44 term t1 then translated source-pool src_pool
```

6. Configure the translation type.

```
[edit services nat]
user@host# set rule rule-basic-nat44 term t1 then translated translation-type
  translation-type
```

In the following example, the translation type is **basic-nat44**.

```
[edit services nat]
user@host# set rule rule-basic-nat44 term t1 then translated translation-type
  basic-nat44
```

7. Verify the configuration by using the **show** command at the **[edit services nat]** hierarchy level.

```
[edit services]
user@host# show
nat {
    pool src_pool {
        address 10.10.10.2/32;
    }
    rule rule-basic-nat44 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    3.1.1.2/32;
                }
            }
            then {
                translated {
```

```
                    source-pool src_pool;
                    translation-type {
                        basic-nat44;
                    }
                }
            }
        }
    }
}
```

Configuring the Service Set for NAT

To configure the service set for NAT:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   ```
   [edit]
   user@host# edit services
   ```

2. Configure the service set.

   ```
   [edit services]
   user@host# edit service-set service-set-name
   ```

   In the following example, the service set name is **s1**.

   ```
   [edit services]
   user@host# edit service-set s1
   ```

3. For the **s1** service set, set the reference to the NAT rules configured at the **[edit services nat]** hierarchy level.

   ```
   [edit services service-set s1]
   user@host# set nat-rules rule-name
   ```

   In the following example, the rule name is **rule-basic-nat44**.

   ```
   [edit services service-set s1]
   user@host# set nat-rules rule-basic-nat44
   ```

4. Configure the service interface.

   ```
   [edit services service-set s1]
   user@host# set interface-service service-interface service-interface-name
   ```

   In the following example, the service interface name is **ms-1/2/0**.

   ```
   [edit services service-set s1]
   user@host# set interface-service service-interface ms-1/2/0
   ```

NOTE: If you have a Trio-based line card, you can configure an inline-services interface on that card:

```
[edit]
user@host# set interfaces si-0/0/0
[edit services service-set s1]
user@host# set interface-service service-interface si-0/0/0
```

5. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-basic-nat44;
    interface-service {
        service-interface ms-1/2/0;
    }
}
```

Configuring Trace Options

To configure the trace options:

1. In configuration mode, go to the **[edit services adaptive-services-pics]** hierarchy level.

```
[edit]
user@host# edit services adaptive-services-pics
```

2. Configure the trace options.

```
[edit services adaptive-services-pics]
user@host# set traceoptions flag tracing parameter
```

In the following example, the tracing parameter is **all**.

```
[edit services adaptive-services-pics]
user@host# set traceoptions flag all
```

3. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
adaptive-services-pics {
    traceoptions {
        flag all;
    }                                   }
```

Configuring Static Source Translation in IPv6 Networks

To configure the translation type as **basic-nat66**, you must configure the NAT pool and rule, service set with service interface, and trace options. This topic includes the following tasks:

Configuring the NAT Pool and Rule

To configure the NAT pool, rule, and term:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# edit services nat
   ```

2. Configure the NAT pool with an address.

   ```
   [edit services nat]
   user@host# set pool pool name address address
   ```

   In the following example, the pool name is **src_pool** and the address is **10.10.10.2/32**.

   ```
   [edit services nat]
   user@host# set pool src_pool address 10.10.10.2/32
   ```

3. Configure the NAT rule and the match direction.

   ```
   [edit services nat]
   user@host# set rule rule-name match-direction match-direction
   ```

   In the following example, the rule name is **rule-basic-nat66** and the match direction is **input**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 match-direction input
   ```

4. Configure the source address in the **from** statement.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term term-name from from
   ```

   In the following, the term name is **t1** and the input condition is **source-address 10:10:10::0/96**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term t1 from source-address 10:10:10::0/96
   ```

5. Configure the NAT term action and properties of the translated traffic.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term t1 then term-action translated-property
   ```

In the following example, the term action is **translated** and the property of the translated traffic is **source-pool src_pool**.

```
[edit services nat]
user@host# set rule rule-basic-nat66 term t1 then translated source-pool src_pool
```

6. Configure the translation type.

```
[edit services nat]
user@host# set rule rule-basic-nat66 term t1 then translated translation-type
  translation-type
```

In the following example, the translation type is **basic-nat66**.

```
[edit services nat]
user@host# set rule rule-basic-nat66 term t1 then translated translation-type
  basic-nat66
```

7. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
nat {
    pool src_pool {
        address 10.10.10.2/32;
    }
    rule rule-basic-nat66 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    10:10:10::0/96;
                }
            }
            then {
                translated {
                    source-pool src_pool;
                    translation-type {
                        basic-nat66;
                    }
                }
            }
        }
    }
}
```

Configuring the Service Set for NAT

To configure the service set for NAT:

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

In the following example, the service set name is **s1**.

```
[edit services]
user@host# edit service-set s1
```

3. For the **s1** service set, set the reference to the NAT rules configured at the **[edit services nat]** hierarchy level.

   ```
   [edit services service-set s1]
   user@host# set nat-rules rule-name
   ```

   In the following example, the rule name is **rule-basic-nat66**.

   ```
   [edit services service-set s1]
   user@host# set nat-rules rule-basic-nat66
   ```

4. Configure the service interface.

   ```
   [edit services service-set s1]
   user@host# set interface-service service-interface service-interface-name
   ```

   In the following example, the service interface name is **sp-1/2/0**.

   ```
   [edit services service-set s1]
   user@host# set interface-service service-interface sp-1/2/0
   ```

5. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

   ```
   [edit services]
   user@host# show
   service-set s1 {
       nat-rules rule-basic-nat66;
       interface-service {
           service-interface sp-1/2/0;
       }
   }
   ```

Configuring Trace Options

To configure the trace options at the **[edit services adaptive-services-pics]** hierarchy level:

1. In configuration mode, go to the **[edit services adaptive-services-pics]** hierarchy level.

   ```
   [edit]
   user@host# edit services adaptive-services-pics
   ```

2. Configure the trace options.

   ```
   [edit services adaptive-services-pics]
   user@host# set traceoptions flag tracing parameter
   ```

   In the following example, the tracing parameter is **all**.

   ```
   [edit services adaptive-services-pics]
   user@host# set traceoptions flag all
   ```

3. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

   ```
   [edit services]
   user@host# show
   ```

```
adaptive-services-pics {
    traceoptions {
        flag all;
    }
}
```

Configuring Dynamic Source Address and Port Translation in IPv4 Networks

Network Address Port Translation (NAPT) is a method by which many network addresses and their TCP/UDP ports are translated into a single network address and its TCP/UDP ports. This translation can be configured in both IPv4 and IPv6 networks. This section describes the steps for configuring NAPT in IPv4 networks.

To configure NAPT, you must configure a rule at the **[edit services nat]** hierarchy level for dynamically translating the source IPv4 addresses.

To configure the NAPT in IPv4 networks:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   ```
   [edit]
   user@host# edit services
   ```

2. Configure the service set and NAT rule.

   ```
   [edit services]
   user@host# set service-set service-set-name nat-rules rule-name
   ```

   In the following example, the name of the service set is **s1** and the name of the NAT rule is **rule-napt-44**.

   ```
   [edit services]
   user@host# set service-set s1 nat-rules rule-napt-44
   ```

3. Go to the **[interface-service]** hierarchy level of the service set.

   ```
   [edit services]
   user@host# edit service-set s1 interface-service
   ```

4. Configure the service interface.

   ```
   [edit services service-set s1 interface service]
   user@host# set service-interface service-interface-name
   ```

   In the following example, the name of the service interface is **ms-0/1/0**.

   NOTE: If the service interface is not present in the router, or the specified interface is not functional, the following command can result in an error.

   ```
   [edit services service-set s1 interface service]
   user@host# set service-interface ms-0/1/0
   ```

5. Go to the **[edit services nat]** hierarchy level. Issue the command from the top of the services hierarchy, or use the **top** keyword.

   ```
   [edit services service-set s1 interface service]
   user@host# top editservices nat
   ```

6. Configure the NAT pool with an address.

   ```
   [edit services nat]
   user@host# set pool pool-name address address
   ```

   In the following example, the name of the pool is **napt-pool** and the address is **10.10.10.0**.

   ```
   [edit services nat]
   user@host# set pool napt-pool address 10.10.10.0
   ```

7. Configure the port.

   ```
   [edit services nat]
   user@host# set pool pool-name port port-type
   ```

   In the following example, the port type is selected as **automatic**.

   ```
   [edit services nat]
   user@host# set pool napt-pool port automatic
   ```

8. Configure the rule and the match direction.

   ```
   [edit services nat]
   user@host# set rule rule-name match-direction match-direction
   ```

   In the following example, the name of the rule is **rule-napt-44** and the match direction is **input**.

   ```
   [edit services nat]
   user@host# set rule rule-napt-44 match-direction input
   ```

9. Configure the term, the action for the translated traffic, and the translation type.

   ```
   [edit services nat]
   user@host# set rule rule-name term term-name then translated translated-action
     translation-type translation- type
   ```

   In the following example, the name of the term is **t1**, the action for the translated traffic is **translated**, the name of the source pool is **napt-pool**, and the translation type is **napt-44**.

   ```
   [edit services nat]
   user@host# set rule rule-napt-44 match-direction input term t1 then translated
     source-pool napt-pool translation-type napt-44
   ```

10. Go to the **[edit services adaptive-services-pics]** hierarchy level. In the command, the **top** keyword ensures that the command is run from the top of the hierarchy.

    ```
    [edit services nat]
    user@host# top edit services adaptive-services-pics
    ```

11. Configure the trace options.

    ```
    [edit services adaptive-services-pics]
    user@host# set traceoptions flag tracing parameter
    ```

    In the following example, the tracing parameter is configured as **all**.

    ```
    [edit services adaptive-services-pics]
    ```

```
user@host# set traceoptions flag all
```

12. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-napt-44;
    interface-service {
        service-interface ms-0/1/0;
    }
}
nat {
    pool napt-pool {
        address 10.10.10.0/32;
        port {
            automatic;
        }
    }
    rule rule-napt-44 {
        match-direction input;
        term t1 {
            then {
                translated {
                    source-pool napt-pool;
                    translation-type {
                        napt-44;
                    }
                }
            }
        }
    }
}
adaptive-services-pics {
    traceoptions {
        flag all;
    }
}
```

Related Documentation

- Example: Configuring Dynamic Source Translation for an IPv4 Network on page 63

Configuring Dynamic Source Address and Port Translation for IPv6 Networks

Network Address Port Translation (NAPT) is a method by which many network addresses and their TCP/UDP ports are translated into a single network address and its TCP/UDP ports. This translation can be configured in both IPv4 and IPv6 networks. This section describes the steps for configuring NAPT in IPv6 networks. For information about configuring NAPT in IPv4 networks, see "Configuring Dynamic Source Address and Port Translation in IPv4 Networks" on page 34.

To configure NAPT, you must configure a rule at the **[edit services nat]** hierarchy level for dynamically translating the source IPv6 addresses.

To configure NAPT in IPv6 networks:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# edit services nat
   ```

2. Define the pool of IPv6 source addresses that must be used for dynamic translation. For NAPT, also specify port numbers when configuring the source pool.

   ```
   [edit services nat]
   user@host# set pool pool name address IPv6 source addresses
   user@host# set pool pool name port source ports
   ```

   For example:

   ```
   [edit services nat]
   user@host#  set pool IPV6-NAPT-Pool address 2002::1/96
   user@host#  set pool IPV6-NAPT-Pool port automatic
   ```

3. Define a NAT rule for translating the source addresses. To do this, set the **match-direction** statement of the rule as **input**. In addition, define a term that uses **napt-66** as the translation type for translating the addresses of the pool defined in the previous step.

   ```
   [edit services nat]
   user@host# set rule rule name match-direction input
   user@host# set rule rule name term term name then translated source-pool pool name
   user@host# set rule rule name term term name then translated translation-type napt-66
   ```

   For example:

   ```
   [edit services nat]
   user@host# set rule IPV6-NAPT-Rule match-direction input
   user@host# set rule IPV6-NAPT-Rule term t1 then translated source-pool
     IPV6-NAPT-Pool
   user@host# set rule IPV6-NAPT-Rule term t1 then translated translation-type napt-66
   ```

4. Enter the **up** command to navigate to the **[edit services]** hierarchy level.

   ```
   [edit services nat]
   user@host# up
   ```

5. Define a service set to specify the services interface that must be used, and reference the NAT rule implemented for NAPT translation.

```
[edit services]
user@host# set service-set service-set name interface- service service-interface
  services interface
user@host# set service-set service-set name nat-rules rule name
```

For example:

```
[edit services]
user@host# set service-set IPV6-NAPT-ServiceSet interface-service service-interface
  ms-0/1/0
user@host# set service-set IPV6-NAPT-ServiceSet nat-rules IPV6-NAPT-Rule
```

6. Define the trace options for the adaptive services PIC.

```
[edit services]
user@host# set adaptive-services-pics traceoptions flag tracing parameter
```

For example:

```
[edit services]
user@host# set adaptive-services-pics traceoptions flag all
```

Related Documentation

- Example: Configuring Dynamic Source Address and Port Translation for an IPv6 Network on page 63

Configuring Dynamic Address-Only Source Translation in IPv4 Networks

In IPv4 networks, dynamic address translation (dynamic NAT) is a mechanism to dynamically translate the destination traffic without port mapping. To use dynamic NAT, you must specify a source pool name, which includes an address configuration.

To configure dynamic NAT in IPv4 networks:

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure the service set and NAT rule.

```
[edit services]
user@host# set service-set service-set-name nat-rules rule-name
```

In the following example, the name of the service set is **s1**, and the name of the NAT rule is **rule-dynamic-nat44**.

```
[edit services]
user@host# set service-set s1 nat-rules rule-dynamic-nat44
```

3. Go to the **[interface-service]** hierarchy level for the service set.

```
[edit services]
user@host# edit service-set s1 interface-service
```

4. Configure the service interface.

```
[edit services service-set s1 interface-service]
user@host# set service-interface service-interface-name
```

In the following example, the name of the service interface is **ms-0/1/0**.

NOTE: If the service interface is not present in the router, or the specified interface is not functional, the following command can result in an error.

```
[edit services service-set s1 interface-service]
user@host# set service-interface ms-0/1/0
```

5. Go to the **[edit services nat]** hierarchy level. Issue the following command from the top of the services hierarchy, or use the **top** keyword.

   ```
   [edit services service-set s1 interface-service]
   user@host# top editservices nat
   ```

6. Configure the NAT pool with an address.

   ```
   [edit services nat]
   user@host# set pool pool-name address address
   ```

   In the following example, the name of the pool is **source-dynamic-pool**, and the address is **10.10.10.0**.

   ```
   [edit services nat]
   user@host# set pool source-dynamic-pool address 10.10.10.0
   ```

7. Configure the rule, match direction, term, and source address.

   ```
   [edit services nat]
   user@host# set rule rule-name match-direction match-direction term term-name from
     source-address address
   ```

   In the following example, the name of the rule is **rule-dynamic-nat44**, the match direction is **input**, the name of the term is **t1**, and the source address is **3.1.1.0**.

   ```
   [edit services nat]
   user@host# set rule rule-dynamic-nat44 match-direction input term t1 from
     source-address 3.1.1.0
   ```

8. Go to the **[edit rule rule-dynamic-nat-44 term t1]** hierarchy level.

   ```
   [edit services nat]
   user@host# edit rule rule-dynamic-nat44 term t1
   ```

9. Configure the source pool and the translation type.

   ```
   [edit services nat rule rule-dynamic-nat44 term t1]
   user@host# set then translated source-pool src-pool-name translation-type
     translation-type
   ```

   In the following example, the name of the source pool is **source-dynamic-pool** and the translation type is **dynamic-nat44**.

   ```
   [edit services nat rule rule-dynamic-nat44 term t1]
   user@host# set then translated source-pool  source-dynamic-pool translation-type
     dynamic-nat44
   ```

10. Go to the **[edit services adaptive-services-pics]** hierarchy level. In the following command, the **top** keyword ensures that the command is run from the top of the hierarchy.

```
[edit services nat rule rule-dynamic-nat44 term t1]
user@host# top editservices adaptive-services-pics
```

11. Configure the trace options.

```
[edit services adaptive-services-pics]
user@host# set traceoptions flag tracing parameter
```

In the following example, the tracing parameter is configured as **all**.

```
[edit services adaptive-services-pics]
user@host# set traceoptions flag all
```

12. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-dynamic-nat44;
    interface-service {
        service-interface ms-0/1/0;
    }
}
 nat {
    pool source-dynamic-pool {
        address 10.1.1.0/24;
    }
    rule rule-dynamic-nat44 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    3.1.1.0/24;
                }
            }
            then {
                translated {
                    destination-pool source-dynamic-pool;
                    translation-type {
                        dynamic-nat44;
                    }
                }
            }
        }
    }
}
 adaptive-services-pics {
    traceoptions {
        flag all;
    }
  }
```

Related Documentation

- Example: Configuring Dynamic Address-Only Source Translation in an IPv4 Network on page 64

Configuring Static Destination Address Translation in IPv4 Networks

In IPv4 networks, destination address translation is a mechanism used to implement address translation for destination traffic without port mapping. To use destination address translation, the size of the pool address space must be greater than or equal to the destination address space. You must specify a name for the **destination-pool** statement, which can contain multiple addresses, ranges, or prefixes, as long as the number of NAT addresses in the pool is larger than the number of destination addresses in the **from** statement.

To configure destination address translation in IPv4 networks:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   ```
   [edit]
   user@host# edit services
   ```

2. Configure the service set and the NAT rule.

   ```
   [edit services]
   user@host# set service-set service-set-name nat-rules rule-name
   ```

   In the following example, the name of the service set is **s1** and the name of the NAT rule is **rule-dnat44**.

   ```
   [edit services]
   user@host# set service-set s1 nat-rules rule-dnat44
   ```

3. Go to the **[interface-service]** hierarchy level of the service set.

   ```
   [edit services]
   user@host# edit service-set s1 interface-service
   ```

4. Configure the service interface.

   ```
   [edit services service-set s1 interface-service]
   user@host# set service-interface service-interface-name
   ```

   In the following example, the name of the service interface is **ms-0/1/0**.

   NOTE: If the service interface is not present in the router, or the specified interface is not functional, the following command can result in an error.

   ```
   [edit services service-set s1 interface-service]
   user@host# set service-interface ms-0/1/0
   ```

5. Go to the **[edit services nat]** hierarchy level. Issue the following command from the top of the services hierarchy, or use the **top** keyword.

   ```
   [edit services service-set s1]
   user@host# top editservices nat
   ```

6. Configure the NAT pool with an address.

   ```
   [edit services nat]
   user@host# set pool pool-name address address
   ```

In the following example, **dest-pool** is used as the pool name and **4.1.1.2** as the address.

```
user@host# set pool dest-pool address 4.1.1.2
```

7. Configure the rule, match direction, term, and destination address.

```
[edit services nat]
user@host# set rule rule-name match-direction match-direction term term-name from
  destination-address address
```

In the following example, the name of the rule is **rule-dnat44**, the match direction is **input**, the name of the term is **t1**, and the address is **20.20.20.20**.

```
[edit services nat]
user@host# set rule rule-dnat44 match-direction input term t1 from
  destination-address 20.20.20.20
```

8. Go to the **[edit services nat rule rule-dnat44 term t1]** hierarchy level.

```
[edit services nat]
user@host# edit rule rule-dnat44 term t1
```

9. Configure the destination pool and the translation type.

```
[edit services nat rule rule-dnat44 term t1]
user@host# set then translated destination-pool dest-pool-name translation-type
  translation-type
```

In the following example, the destination pool name is **dest-pool**, and the translation type is **dnat-44**.

```
[edit services nat rule rule-dnat44 term t1]
user@host# set then translated destination-pool dest-pool translation-type dnat-44
```

10. Go to the **[edit services adaptive-services-pics]** hierarchy level. In the following command, the **top** keyword ensures that the command is run from the top of the hierarchy.

```
[edit services nat rule rule-dnat44 term t1]
user@host# top edit services adaptive-services-pics
```

11. Configure the trace options.

```
[edit services adaptive-services-pics]
user@host# set traceoptions flag tracing parameter
```

In the following example, the tracing parameter is configured as **all**.

```
[edit services adaptive-services-pics]
user@host# set traceoptions flag all
```

12. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-dnat44;
    interface-service {
        service-interface ms-0/1/0;
    }
}
 nat {
```

```
    pool dest-pool {
        address 4.1.1.2/32;
    }
    rule rule-dnat44 {
        match-direction input;
        term t1 {
            from {
                destination-address {
                    20.20.20.20/32;
                }
            }
            then {
                translated {
                    destination-pool dest-pool;
                    translation-type {
                        dnat-44;
                    }
                }
            }
        }
    }
}
  adaptive-services-pics {
    traceoptions {
        flag all;
    }
  }
```

Related Documentation

- Example: Configuring Static Destination Address Translation on page 65

Configuring Port Forwarding for Static Destination Address Translation

Starting with Junos OS Release 11.4, you can map an external IP address and port with an IP address and port in a private network. This allows the destination address and port of a packet to be changed to reach the correct host in a Network Address Translation (NAT) gateway. The translation facilitates reaching a host within a masqueraded, typically private, network, based on the port number on which the packet was received from the originating host. Port forwarding allows remote computers, such as public machines on the Internet, to connect to a non-standard port (port other than 80) of a specific computer within a private network. An example of this type of destination is the host of a public HTTP server within a private network. Port forwarding is supported only with **dnat-44** and **twice-napt-44** on IPv4 networks. Port forwarding works only with the FTP application-level gateway (ALG). Port forwarding also supports endpoint-independent mapping (EIM), endpoint-independent filltering (EIF), and address pooling paired (APP). Port forwarding has no support for technologies such as IPv6 rapid deployment (6rd) and dual-stack lite (DS-Lite) that offer IPv6 services over IPv4 infrastructure.

To configure destination address translation with port forwarding in IPv4 networks:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# editservices nat
   ```

2. Configure the NAT pool with an address.

```
[edit services nat]
user@host# set pool pool-name address address
```

In the following example, **dest-pool** is used as the pool name and **4.1.1.2** as the address.

```
user@host# set pool dest-pool address 4.1.1.2
```

3. Configure the rule, match direction, term, and destination address.

```
[edit services nat]
user@host# set rule rule-name match-direction match-direction term term-name from
  destination-address  address
```

In the following example, the name of the rule is **rule-dnat44**, the match direction is **input**, the name of the term is **t1**, and the address is **20.20.20.20**.

```
[edit services nat]
user@host# set rule rule-dnat44 match-direction input term t1 from
  destination-address 20.20.20.20
```

4. Configure the destination port range.

```
[edit services nat]
user@host# set rule rule-name match-direction match-direction term term-name from
  destination-port range range high | low
```

In the following example, the upper port range is **50** and the lower port range is **20**.

```
[edit services nat]
user@host# set rule rule-dnat44 match-direction input term t1 from destination-port
  range range high 50 low 20
```

5. Go to the **[edit services nat rule rule-dnat44 term t1]** hierarchy level.

```
[edit services nat]
user@host# edit rule rule-dnat44 term t1
```

6. Configure the destination pool.

```
[edit services nat rule rule-dnat44 term t1]
user@host# set then translated destination-pool dest-pool-name
```

In the following example, the destination pool name is **dest-pool**, and the translation type is **dnat-44**.

```
[edit services nat rule rule-dnat44 term t1]
user@host# set then translated destination-pool dest-pool
```

7. Configure the mapping for port forwarding and the translation type.

```
[edit services nat rule rule-dnat44 term t1]
user@host# set then port-forwarding-mappings  map-name translation-type
  translation-type
```

In the following example, the port forwarding map name is **map1**, and the translation type is **dnat-44**.

```
[edit services nat rule rule-dnat44 term t1]
user@host# set then port-forwarding-mappings map1 translation-type dnat-44
```

8. Go to the **[edit services nat port-forwarding map1]** hierarchy level.

```
[edit services nat]
user@host# edit port-forwarding map1
```

9. Configure the mapping for port forwarding.

   ```
   [edit port-forwarding map1]
   user@host# set destined-port port-id
   user@host# set translated-port port-id
   ```

   In the following example, the destination port is **45** and the translated port is **23**.

   ```
   [edit port-forwarding map1]
   user@host# set destined-port 23
   user@host# set translated-port 45
   ```

   NOTE:
   - Multiple port mappings are supported with port forwarding. Up to 32 port maps can be configured for port forwarding.
   - The destination port should not overlap the port range configured for NAT.

10. Verify the configuration by using the **show** command at the **[edit services nat]** hierarchy level.

```
[edit services]
user@host# show
nat {
 pool dest-pool {
     address 4.1.1.2/32;
 }
 rule rule-dnat44 {
     match-direction input;
     term t1 {
  from {
      destination-address {
          20.20.20.20/32;
      }
      destination-port {
          range low 20 high 50;
      }
  }
  then {
      port-forwarding-mappings map1;
      translated {
           destination-pool dest-pool;
           translation-type {
               dnat-44;
             }
         }
     }
     }
 }
 port-forwarding map1 {
     destined-port 45;
   translated-port 23;
     }
}
```

NOTE:

- A similar configuration is possible with twice NAT for IPv4. See "Example: Configuring Port Forwarding with Twice NAT" on page 85.
- Port forwarding and stateful firewall can be configured together. Stateful firewall has precedence over port forwarding.

Related Documentation

- Example: Configuring Static Destination Address Translation on page 65

Configuring Translation Type for Translation Between IPv6 and IPv4 Networks

To configure the translation type as **basic-nat-pt**, you must configure the DNS ALG application, the NAT pools and rules, a service set with a service interface, and trace options. This topic includes the following tasks:

- Configuring the DNS ALG Application on page 46
- Configuring the NAT Pool and NAT Rule on page 47
- Configuring the Service Set for NAT on page 50
- Configuring Trace Options on page 51

Configuring the DNS ALG Application

To configure the DNS ALG application:

1. In configuration mode, go to the **[edit applications]** hierarchy level.

   ```
   [edit]
   user@host# edit applications
   ```

2. Configure the ALG to which the DNS traffic is destined at the **[edit applications]** hierarchy level. Define the application name and specify the application protocol to use in match conditions in the first NAT rule or term.

   ```
   [edit applications]
   user@host# set application application-name application-protocol application-protocol
   ```

   In the following example, the application name is **dns-alg** and application protocol is **dns**.

   ```
   [edit applications]
   user@host# set application dns-alg application-protocol dns
   ```

3. Verify the configuration by using the **show** command at the **[edit applications]** hierarchy level.

   ```
   [edit applications]
   user@host# show
   application dns-alg {
       application-protocol dns;
   }
   ```

Configuring the NAT Pool and NAT Rule

To configure the NAT pool and NAT rule:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# edit services nat
   ```

2. Configure the NAT pool and its address.

   ```
   [edit services nat]
   user@host# set pool pool-name address address
   ```

   In the following example, the name of the NAT pool is **p1** and the address is **10.10.10.2/32**.

   ```
   [edit services nat]
   user@host# set pool p1 address 10.10.10.2/32
   ```

3. Configure the source pool and its address.

   ```
   [edit services nat]
   user@host# set pool source-pool-name address address
   ```

   In the following example, the name of the source pool is **src_pool0** and the source pool address is **20.1.1.1/32**.

   ```
   [edit services nat]
   user@host# set pool src_pool0 address 20.1.1.1/32
   ```

4. Configure the destination pool and its address.

   ```
   [edit services nat]
   user@host# set pool destination-pool-name address address
   ```

   In the following example, the name of the destination pool is **dst_pool0** and the destination pool address is **50.1.1.2/32**.

   ```
   [edit services nat]
   user@host# set pool dst_pool0 address 50.1.1.2/32
   ```

5. Configure the rule and the match direction.

   ```
   [edit services nat]
   user@host# set rule rule-name match-direction match-direction
   ```

   In the following example, the rule name is **rule-basic-nat-pt** and the match direction is **input**.

   ```
   [edit services nat]
   user@host# set rule basic-nat-pt match-direction input
   ```

6. Configure the term and the input conditions for the NAT term.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term term from from
   ```

   In the following example, the term is **t1** and the input conditions are **source-address 2000::2/128**, **destination-address 4000::2/128**, and **applications dns_alg**.

   ```
   [edit services nat]
   ```

```
user@host# set rule rule-basic-nat-pt term t1 from source-address 2000::2/128
[edit services nat]
user@host# set rule rule-basic-nat-pt term t1 from destination-address 4000::2/128
[edit services nat]
user@host# set rule rule-basic-nat-pt term t1 from applications dns_alg
```

7. Configure the NAT term action and the properties of the translated traffic.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t1 then term-action translated-property
   ```

   In the following example, the term action is **translated** and the properties of the translated traffic are **source-pool src_pool0**, **destination-pool dst_pool0**, and **dns-alg-prefix 10:10:10::0/96**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t1 then translated source-pool src_pool0
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t1 then translated destination-pool
     dst_pool0
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t1 then translated dns-alg-prefix
     10:10:10::0/96
   ```

8. Configure the translation type.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t1 then translated translation-type
     translation-type
   ```

   In the following example, the translation type is **basic-nat-pt**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t1 then translated translation-type
     basic-nat-pt
   ```

9. Configure another term and the input conditions for the NAT term.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term term-name from from
   ```

   In the following example, the term name is **t2** and the input conditions are **source-address 2000::2/128** and **destination-address 10:10:10::0/96**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t2 from source-address 2000::2/128
   [edit services nat]
   user@host# set rule rule-basic-nat-pt term t2 from destination-address 10:10:10::0/96
   ```

10. Configure the NAT term action and the property of the translated traffic.

    ```
    [edit services nat]
    user@host# set rule rule-basic-nat-pt term t2 then term-action translated-property
    ```

    In the following example, the term action is **translated** and the property of the translated traffic is **source-prefix 19.19.19.1/32**.

    ```
    [edit services nat]
    user@host# set rule rule-basic-nat-pt term t2 then translated source-prefix
      19.19.19.1/32
    ```

11. Configure the translation type.

    [edit services nat]
    user@host# **set rule rule-basic-nat-pt term t2 then translated translation-type *translation-type***

    In the following example, the translation type is **basic-nat-pt**.

    [edit services nat]
    user@host# **set rule rule-basic-nat-pt term t2 then translated translation-type basic-nat-pt**

12. Verify the configuration by using the **show** command at the **[edit services nat]** hierarchy level.

```
[edit services nat]
user@host# show
pool p1 {
    address 10.10.10.2/32;
}
pool src_pool0 {
    address 20.1.1.1/32;
}
pool dst_pool0 {
    address 50.1.1.2/32;
}
rule rule-basic-nat-pt {
    match-direction input;
    term t1 {
        from {
            source-address {
                2000::2/128;
            }
            destination-address {
                4000::2/128;
            }
            applications dns_alg;
        }
        then {
            translated {
                source-pool src_pool0;
                destination-pool dst_pool0;
                dns-alg-prefix 10:10:10::0/96;
                translation-type {
                    basic-nat-pt;
                }
            }
        }
    }
    term t2 {
        from {
            source-address {
                2000::2/128;
            }
            destination-address {
                10:10:10::0/96;
            }
        }
        then {
            translated {
                source-prefix 19.19.19.1/32;
```

```
                translation-type {
                    basic-nat-pt;
                }
            }
        }
    }
}
```

Configuring the Service Set for NAT

To configure the service set for NAT:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   ```
   [edit]
   user@host# edit services
   ```

2. Configure the service set.

   ```
   [edit services]
   user@host# edit service-set service-set-name
   ```

   In the following example, the name of the service set is **ss_dns**.

   ```
   [edit services]
   user@host# edit service-set ss_dns
   ```

3. Configure the service set with NAT rules.

   ```
   [edit services service-set ss_dns]
   user@host# set nat-rules rule-name
   ```

   In the following example, the rule name is **rule-basic-nat-pt**.

   ```
   [edit services service-set ss_dns]
   user@host# set nat-rules rule-basic-nat-pt
   ```

4. Configure the service interface.

   ```
   [edit services service-set ss_dns]
   user@host# set interface-service service-interface service-interface-name
   ```

   In the following example, the name of service interface is **sp-1/2/0**.

   ```
   [edit services service-set ss_dns]
   user@host# set interface-service service-interface sp-1/2/0
   ```

5. Verify the configuration by using the **show services** command from the **[edit]** hierarchy level.

   ```
   [edit]
   user@host# show services
       service-set ss_dns {
           nat-rules rule-basic-nat-pt;
           interface-service {
               service-interface sp-1/2/0;
           }
       }
   ```

Configuring Trace Options

To configure the trace options:

1. In configuration mode, go to the **[edit services adaptive-services-pics]** hierarchy level.

   ```
   [edit]
   user@host# edit services adaptive-services-pics
   ```

2. Configure the trace options.

   ```
   [edit services adaptive-services-pics]
   user@host# set traceoptions flag tracing parameter
   ```

   In the following example, the tracing parameter is **all**.

   ```
   [edit services adaptive-services-pics]
   user@host# set traceoptions flag all
   ```

3. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

   ```
   [edit services]
   user@host# show
   adaptive-services-pics {
       traceoptions {
           flag all;
       }
   }
   ```

Configuring NAT-PT

To configure Network Address Translation–Protocol Translation (NAT-PT), you must configure a Domain Name System application-level gateway (DNS ALG) application to map addresses returned in the DNS response to an IPv6 address. DNS ALG is used with NAT-PT to facilitate name-to-address mapping. When configuring NAT-PT, network address translation can either be an address-only translation or an address and port translation. The Junos OS implementation is described in RFC 2766 and RFC 2694.

Before you begin configuring NAT-PT with DNS ALG, you must have the following configured:

- NAT with two rules or one rule and two terms. The first NAT rule or term ensures that the DNS query and response packets are translated correctly. For this rule to work, you must configure a DNS ALG application and reference it in the first rule. The second rule or term is required to ensure that NAT sessions are destined to the address mapped by the DNS ALG application.
- A service set that references the first NAT rule or term and a multiservices interface.

To configure NAT-PT with DNS ALG:

1. Configure the DNS session that processes packets to the DNS server:

   a. Configure the ALG to which the DNS traffic is destined at the **[edit applications]** hierarchy level. Define the application name and specify the application protocol to use in match conditions in the first NAT rule or term.

      [edit applications]
      user@host# **set application** *application-name* **application-protocol** *application-protocol*

      For example:

      [edit applications]
      user@host# **set application dns_alg application-protocol dns**

   b. Reference the ALG in the first NAT rule or term.

      [edit services nat rule *rule-name* term *term-name*]
      user@host# **set from applications** *application-name*

      In the following example, the application name is **dns_alg**.

      [edit services nat rule rule1 term term1]
      user@host# **set from applications dns_alg**

   c. Define the DNS ALG pool or prefix for mapping IPv4 addresses to IPv6 addresses.

      [edit services nat rule *rule-name* term *term-name*]
      user@host# **set then translated dns-alg-prefix** *dns-alg-prefix*
      user@host# **set then translated dns-alg-pool** *dns-alg-pool*

      The following example shows the configuration of the 96-bit prefix for mapping IPv4 address to IPv6 addresses.

      [edit services nat rule rule1 term term1]
      user@host# **set then translated dns-alg-prefix 10:10:10::0/96**

The following sample output shows the minimum configuration of the application.

```
[edit applications]
user@host# show
application dns_alg {
    application-protocol dns;
}
```

The following sample output shows the minimum configuration of the first NAT rule.

```
[edit services nat]
user@host# show
rule rule1 {
            applications dns_alg;
        }
        then {
            translated {
                dns-alg-prefix 10:10:10::0/96;
                }
            }
        }
```



```
    }
}
```

The following sample output shows the minimum configuration of the second NAT rule.

```
[edit services nat]
user@host# show
rule rule2 {
    term term1 {
        from {
            destination-address {
                10:10:10::c0a8:108/128;
            }
        }
        then {
            translated {
                source-prefix 19.19.19.1/32;
                }
            }
        }
    }
}
```

Related Documentation

- Network Address Translation Overview on page 3
- Example: Configuring NAT-PT on page 71
- dns-alg-prefix on page 104
- dns-alg-pool on page 103

Configuring Dynamic Source Address and Static Destination Address Translation (IPv6 to IPV4)

Stateful NAT64 is a mechanism used to move to an IPv6 network and at the same time deal with IPv4 address depletion. By allowing IPv6-only clients to contact IPv4 servers using unicast UDP, TCP, or ICMP, several IPv6-only clients can share the same public IPv4 server address. To allow sharing of the IPv4 server address, stateful NAT64 translates incoming IPv6 packets into IPv4, and vice versa.

To configure stateful NAT64, you must configure a rule at the **[edit services nat]** hierarchy level for translating the source address dynamically and the destination address statically.

BEST PRACTICE: When you configure the service set that includes your NAT rule, include the **set stateful-nat64 clear-dont-fragment-bit** at the **[edit services service-set *service-set-name*]** hierarchy level.This clears the DF (don't fragment) bit in order to prevent unnecessary creation of an IPv6 fragmentation header when translating IPv4 packets that are less than 1280 bytes. RFC 6145, *IP/ICMP Translation Algorithm*, provides a full discussion of the use of the DF flag to control generation of fragmentation headers. For more information on service sets for NAT, see Configuring NAT Service Sets.

To configure stateful NAT64:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# edit services nat
   ```

2. Define the pool of source addresses to be used for dynamic translation.

   ```
   [edit services nat]
   user@host# set pool pool name address source addresses
   user@host# set pool pool name port source ports
   ```

   For example:

   ```
   [edit services nat]
   user@host#  set pool src-pool-nat64 address 203.0.113.0/24
   user@host#  set pool src-pool-nat64 port automatic
   ```

3. Define a NAT rule for translating the source addresses. Set the **match-direction** statement of the rule as **input**. Then define a term that uses **stateful-nat64** as the translation type for translating the addresses of the pool defined in the previous step.

   ```
   [edit services nat]
   user@host# set rule rule name match-direction input
   user@host# set rule rule name term term name from source-address source address
   user@host# set rule rule name term term name from destination-address destination
     address
   user@host# set rule rule name term term name then translated source-pool pool name
   user@host# set rule rule name term term name then translated destination-prefix
     destination prefix
   ```

user@host# set rule *rule name* term *term name* then translated translation-type stateful-nat64

For example:

```
[edit services nat]
user@host# set rule stateful-nat64 match-direction input
user@host# set rule stateful-nat64 term t1 from source-address 2001:DB8::0/96
user@host# set rule stateful-nat64 term t1 from destination-address 64:FF9B::/96
user@host# set rule stateful-nat64 term t1 then translated source-pool src-pool-nat64
user@host# set rule stateful-nat64 term t1 then translated destination-prefix
  64:FF9B::/96
user@host# set rule stateful-nat64 term t1 then translated translation-type
  stateful-nat64
```

Related Documentation

- Example: Configuring Dynamic Source Address and Static Destination Address Translation (IPv6 to IPV4) on page 70

CHAPTER 3

NAT Rules Examples

Example: Configuring Static Source Translation in an IPv4 Network

The following configuration sets up one-to-one mapping between a private subnet and a public subnet.

```
[edit]
user@host# show services
service-set s1 {
    nat-rules rule-basic-nat44;
    interface-service {
        service-interface ms-1/2/0;
    }
}
nat {
    pool src_pool {
        address 10.10.10.2/32;
    }
    rule rule-basic-nat44 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    3.1.1.2/32;
                }
            }
            then {
                translated {
                    source-pool src_pool;
                    translation-type {
                        basic-nat44;
                    }
                }
            }
        }
    }
}
adaptive-services-pics {
    traceoptions {
        flag all;
    }
}
```

Configuring Static Source Translation in IPv6 Networks

To configure the translation type as **basic-nat66**, you must configure the NAT pool and rule, service set with service interface, and trace options. This topic includes the following tasks:

- Configuring the NAT Pool and Rule on page 59
- Configuring the Service Set for NAT on page 60
- Configuring Trace Options on page 61

Configuring the NAT Pool and Rule

To configure the NAT pool, rule, and term:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   [edit]
   user@host# edit services nat
   ```

2. Configure the NAT pool with an address.

   ```
   [edit services nat]
   user@host# set pool pool name address address
   ```

   In the following example, the pool name is **src_pool** and the address is **10.10.10.2/32**.

   ```
   [edit services nat]
   user@host# set pool src_pool address 10.10.10.2/32
   ```

3. Configure the NAT rule and the match direction.

   ```
   [edit services nat]
   user@host# set rule rule-name match-direction match-direction
   ```

   In the following example, the rule name is **rule-basic-nat66** and the match direction is **input**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 match-direction input
   ```

4. Configure the source address in the **from** statement.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term term-name from from
   ```

   In the following, the term name is **t1** and the input condition is **source-address 10:10:10::0/96**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term t1 from source-address 10:10:10::0/96
   ```

5. Configure the NAT term action and properties of the translated traffic.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term t1 then term-action translated-property
   ```

   In the following example, the term action is **translated** and the property of the translated traffic is **source-pool src_pool**.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term t1 then translated source-pool src_pool
   ```

6. Configure the translation type.

   ```
   [edit services nat]
   user@host# set rule rule-basic-nat66 term t1 then translated translation-type
     translation-type
   ```

   In the following example, the translation type is **basic-nat66**.

   ```
   [edit services nat]
   ```

```
user@host# set rule rule-basic-nat66 term t1 then translated translation-type
  basic-nat66
```

7. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
nat {
    pool src_pool {
        address 10.10.10.2/32;
    }
    rule rule-basic-nat66 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    10:10:10::0/96;
                }
            }
            then {
                translated {
                    source-pool src_pool;
                    translation-type {
                        basic-nat66;
                    }
                }
            }
        }
    }
}
```

Configuring the Service Set for NAT

To configure the service set for NAT:

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure the service set.

```
[edit services]
user@host# edit service-set service-set-name
```

In the following example, the service set name is **s1**.

```
[edit services]
user@host# edit service-set s1
```

3. For the **s1** service set, set the reference to the NAT rules configured at the **[edit services nat]** hierarchy level.

```
[edit services service-set s1]
user@host# set nat-rules rule-name
```

In the following example, the rule name is **rule-basic-nat66**.

```
[edit services service-set s1]
user@host# set nat-rules rule-basic-nat66
```

4. Configure the service interface.

   [edit services service-set s1]
   user@host# **set interface-service service-interface** *service-interface-name*

   In the following example, the service interface name is **sp-1/2/0**.

   [edit services service-set s1]
   user@host# **set interface-service service-interface sp-1/2/0**

5. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-basic-nat66;
    interface-service {
        service-interface sp-1/2/0;
    }
}
```

Configuring Trace Options

To configure the trace options at the **[edit services adaptive-services-pics]** hierarchy level:

1. In configuration mode, go to the **[edit services adaptive-services-pics]** hierarchy level.

   [edit]
   user@host# **edit services adaptive-services-pics**

2. Configure the trace options.

   [edit services adaptive-services-pics]
   user@host# **set traceoptions flag** *tracing parameter*

   In the following example, the tracing parameter is **all**.

   [edit services adaptive-services-pics]
   user@host# **set traceoptions flag all**

3. Verify the configuration by using the **show** command at the **[edit services]** hierarchy level.

```
[edit services]
user@host# show
adaptive-services-pics {
    traceoptions {
        flag all;
    }
}
```

Example: Configuring Static Source Translation with Multiple Prefixes and Address Ranges

The following configuration creates a static pool with an address prefix and an address range and uses static source NAT translation.

[edit services nat]

```
pool p1 {
   address 30.30.30.252/30;
   address-range low 20.20.20.1 high 20.20.20.2;
}
rule r1 {
   match-direction input;
   term {
      from {
         source-address {
            10.10.10.252/30;
         }
      }
      then {
         translated {
            source-pool p1;
            translation-type basic-nat44;
         }
      }
   }
}
```

Example: Configuring Dynamic Source Address and Port Translation (NAPT) for an IPv4 Network

The following example configures dynamic source (address and port) translation, or NAPT.

```
[edit services nat]
pool public {
   address-range low 192.16.2.1 high 192.16.2.32;
   port automatic;
}
rule Private-Public {
   match-direction input;
   term Translate {
      then {
         translated {
            source-pool public;
            translation-type napt-44;
         }
      }
   }
}
```

NOTE: The only difference between the configurations for dynamic address-only source translation and NAPT is the inclusion of the **port** statement for NAPT.

Example: Configuring Dynamic Source Translation for an IPv4 Network

The following example configures the translation type as **napt-44**.

```
[edit services]
    user@host# show
    service-set s1 {
    nat-rules rule-napt-44;
    interface-service {
        service-interface ms-0/1/0;
    }
}
nat {
    pool napt-pool {
        address 10.10.10.0/32;
        port {
            automatic;
        }
    }
    rule rule-napt-44 {
        match-direction input;
        term t1 {
            then {
                translated {
                    source-pool napt-pool;
                    translation-type {
                        napt-44;
                    }
                }
            }
        }
    }
}
adaptive-services-pics {
    traceoptions {
        flag all;
    }
}
```

Example: Configuring Dynamic Source Address and Port Translation for an IPv6 Network

The following example configures dynamic source (address and port) translation or NAPT for an IPv6 network.

```
[edit services]
user@host# show
    service-set IPV6-NAPT-ServiceSet {
        nat-rules IPV6-NAPT-Rule;
        interface-service {
            service-interface ms-0/1/0;
        }
    }
    nat {
        pool IPV6-NAPT-Pool {
            address 2002::1/96;
            port automatic;
        }
```

```
        rule IPV6-NAPT-Rule {
            match-direction input;
            term term1 {
                then {
                    translated {
                        source-pool IPV6-NAPT-Pool;
                        translation-type {
                            napt-66;
                        }
                    }
                }
            }
        }
    }
    adaptive-services-pics {
        traceoptions {
            flag all;
        }
    }
}
```

Example: Configuring Dynamic Address-Only Source Translation

The following example configures dynamic address-only source translation.

```
[edit services nat]
pool public {
  address-range low 192.16.2.1 high 192.16.2.32;
}
rule Private-Public {
  match-direction input;
  term Translate {
    then {
      translated {
        source-pool public;
        translation-type dynamic-nat44 ;
      }
    }
  }
}
```

Example: Configuring Dynamic Address-Only Source Translation in an IPv4 Network

The following example configures the translation type as **dynamic-nat44**.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-dynamic-nat44;
    interface-service {
        service-interface ms-0/1/0;
    }
}
 nat {
    pool source-dynamic-pool {
        address 10.1.1.0/24;
    }
    rule rule-dynamic-nat44 {
```

```
        match-direction input;
        term t1 {
            from {
                source-address {
                    3.1.1.0/24;
                }
            }
            then {
                translated {
                    destination-pool source-dynamic-pool;
                    translation-type {
                        dynamic-nat44;
                    }
                }
            }
        }
    }
}
 adaptive-services-pics {
    traceoptions {
        flag all;
    }
  }
```

Example: Configuring Static Destination Address Translation

The following example configures the translation type as **dnat-44**.

```
[edit services]
user@host# show
service-set s1 {
    nat-rules rule-dnat44;
    interface-service {
        service-interface ms-0/1/0;
    }
}
 nat {
    pool dest-pool {
        address 4.1.1.2/32;
    }
    rule rule-dnat44 {
        match-direction input;
        term t1 {
            from {
                destination-address {
                    20.20.20.20/32;
                }
            }
            then {
                translated {
                    destination-pool dest-pool;
                    translation-type {
                        dnat-44;
                    }
                }
            }
        }
    }
}
  adaptive-services-pics {
```

```
    traceoptions {
        flag all;
    }
}
```

Example: Configuring NAT in Mixed IPv4 and IPv6 Networks

The following example shows NAT configuration in a network that combines IPv4 and IPv6 addressing.

```
interfaces {
    sp-2/0/0 {
        traceoptions {
            flag all;
        }
        services-options {
            syslog {
                host local {
                    services any;
                    log-prefix IPV6-SS;
                }
            }
            inactivity-timeout 200;
        }
        unit 0 {
            family inet;
            family inet6;
        }
        unit 1 {
            family inet;
            family inet6;
        }
        unit 1001 {
            family inet;
            family inet6;
        }
    }
    so-2/1/0 {
        description "services-art1 201/1";
        encapsulation ppp;
        sonet-options {
            fcs 32;
        }
        unit 0 {
            family inet {
                address 192.168.1.1/30;
            }
            family inet6 {
                service {
                    input {
                        service-set ss-ipv6;
                    }
                    output {
                        service-set ss-ipv6;
                    }
```

```
            }
            address 3ffe::1:1/64;
          }
        }
      }
      so-2/1/3 {
        description "services-art1 201/2";
        encapsulation ppp;
        sonet-options {
          fcs 32;
        }
        unit 0 {
          family inet {
            address 192.168.1.1/30;
          }
          family inet6 {
            address 4ffe::1:1/64;
          }
        }
      }
    }
    routing-options {
      rib inet6.0 {
        static {
          route 5ffe::0:0/64 next-hop 3ffe::1:1;
          route 6ffe::0:0/64 next-hop 4ffe::1:1;
          route 192::168:1:0/112 next-hop 4ffe::1:2;
        }
      }
    }
    services {
      service-set ss-ipv6 {
        stateful-firewall-rules test-ipv6;
        nat-rules src-nat;
        nat-rules src-nat-v6;
        interface-service {
          service-interface sp-2/0/0;
        }
      }
      stateful-firewall {
        rule test-ipv6 {
          match-direction input-output;
          term 1 {
            from {
              source-address {
                any-unicast;
              }
            }
            then {
              accept;
            }
          }
        }
      }
      nat {
        pool dst_pool {
```

```
    address 192.168.1.2/32;
}
pool src_pool {
    address 192.168.1.0/27;
    port automatic;
}
pool dst_pool_v6 {
    address 192::168:1:2/128;
}
pool src_pool_v6 {
    address 192::168:1:2/100;
}
rule src-nat {
    match-direction input;
    term t1 {
        from {
            source-address {
                3ffe::0:0/96;
            }
            destination-address {
                4ffe::1:2/128;
            }
        }
        then {
            translated {
                source-pool src_pool;
                destination-pool dst_pool;
                translation-type {
                    source dynamic;
                    destination static;
                }
            }
        }
    }
}
rule src-nat-v6 {
    match-direction input;
    term t1 {
        from {
            source-address {
                5ffe::0:0/96;
            }
            destination-address {
                6ffe::1:2/128;
            }
        }
        then {
            translated {
                source-pool src_pool_v6;
                destination-pool dst_pool_v6;
                translation-type {
                    source static;
                    destination static;
                }
            }
        }
```

```
        }
      }
    }
  }
```

Example: Configuring the Translation Type Between IPv6 and IPv4 Networks

The following example configures the translation type as **basic-nat-pt**.

```
[edit]
user@host# show services
service-set ss_dns {
    nat-rules rule-basic-nat-pt;
    interface-service {
        service-interface sp-1/2/0;
    }
}
nat {
    pool p1 {
        address 10.10.10.2/32;
    }
    pool src_pool0 {
        address 20.1.1.1/32;
    }
    pool dst_pool0 {
        address 50.1.1.2/32;
    }
    rule rule-basic-nat-pt {
        match-direction input;
        term t1 {
            from {
                source-address {
                    2000::2/128;
                }
                destination-address {
                    4000::2/128;
                }
                applications dns_alg;
            }
            then {
                translated {
                    source-pool src_pool0;
                    destination-pool dst_pool0;
                    dns-alg-prefix 10:10:10::0/96;
                    translation-type {
                        basic-nat-pt;
                    }
                }
            }
        }
        term t2 {
            from {
                source-address {
                    2000::2/128;
                }
                destination-address {
                    10:10:10::0/96;
                }
            }
```

```
            then {
                translated {
                    source-prefix 19.19.19.1/32;
                    translation-type {
                        basic-nat-pt;
                    }
                }
            }
        }
    }
}
adaptive-services-pics {
    traceoptions {
        flag all;
    }
}
```

Example: Configuring Dynamic Source Address and Static Destination Address Translation (IPv6 to IPV4)

The following example configures dynamic source address (IPv6-to-IPv4) and static destination address (IPv6-to-IPv4) translation.

```
[edit services]
user@host# show
nat {
    pool src-pool-nat64 {
        address 203.0.113.0/24;
        port {
            automatic;
        }
    }
    rule stateful-nat64 {
        match-direction input;
        term t1 {
            from {
                source-address {
                    2001:db8::0/96;
                }
                destination-address {
                    64:ff9b::/96;
                }
            }
            then {
                translated {
                    source-pool src-pool-nat64;
                    destination-prefix 64:ff9b::/96;
                    translation-type {
                        stateful-nat64;
                    }
                }
            }
        }
    }
}
service-set sset-nat64 {
    nat-options {
        stateful-nat64 {
            clear-dont-fragment-bit;
        }
```

```
    }
    service-set-options;
    nat-rules stateful-nat64;
    interface-service {
        service-interface ms-0/1/0;
    }
}
```

Example: Configuring Source Dynamic and Destination Static Translation

In the following configuration, **term1** configures source address translation for traffic from any private address to any public address. The translation is applied for all services. **term2** performs destination address translation for Hypertext Transfer Protocol (HTTP) traffic from any public address to the server's virtual IP address. The virtual server IP address is translated to an internal IP address.

```
[edit services nat]
rule my-nat-rule {
  match-direction input;
  term my-term1 {
    from {
      source-address private;
      destination-address public;
    }
    then {
      translated {
        source-pool my-pool; # pick address from a pool
        translation-type napt-44; # dynamic NAT with port translation
      }
    }
  }
  term my-term2 {
    from {
      destination-address 192.168.137.3; # my server's virtual address
      application http;
    }
    then {
      translated {
        destination-pool nat-pool-name;
        translation-type dnat-44; # static destination NAT
      }
    }
  }
}
```

Example: Configuring NAT-PT

A Domain Name System application-level gateway (DNS ALG) is used with Network Address Translation-Protocol Translation (NAT-PT) to facilitate name-to-address mapping. You can configure the DNS ALG to map addresses returned in the DNS response to an IPv6 address.

When you configure NAT-PT with DNS ALG support, you must configure two NAT rules or one rule with two terms. In this example, you configure two rules. The first NAT rule

ensures that the DNS query and response packets are translated correctly. For this rule to work, you must configure a DNS ALG application and reference it in the rule. The second rule is required to ensure that NAT sessions are destined to the address mapped by the DNS ALG.

Then, you must configure a service set, and then apply the service set to the interfaces.

This example describes how to configure NAT-PAT with DNS ALG:

Requirements

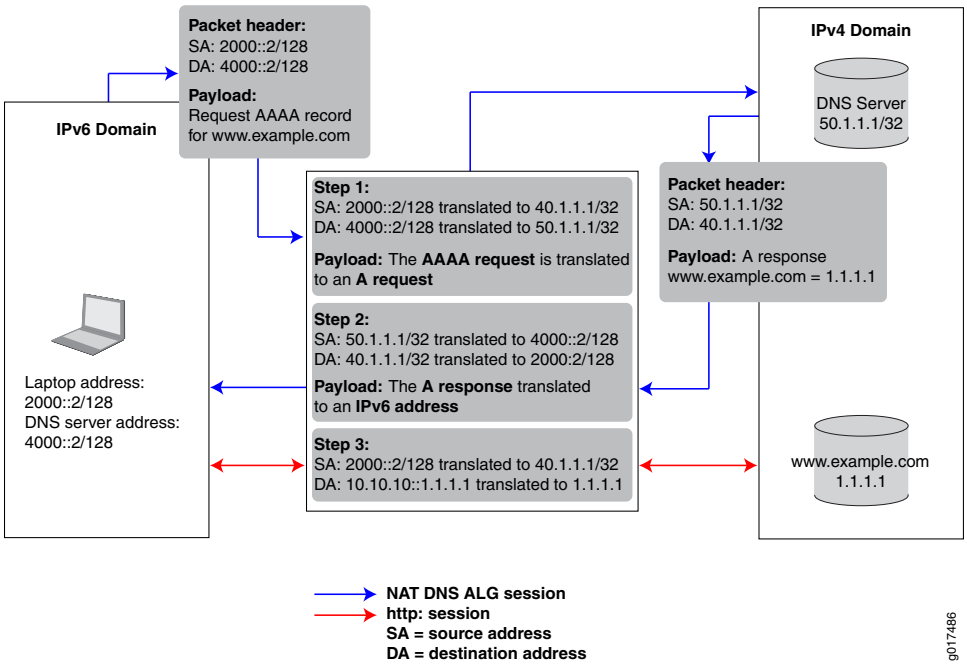
This example uses the following hardware and software components:

- Junos OS Release 11.2
- A multiservices interface (**ms-**)

Overview and Topology

The following scenario shows the process of NAT-PT with DNS ALG when a laptop in an IPv6-only domain requests access to a server in an IPv4-only domain.

Figure 4: Configuring DNS ALGs with NAT-PT Network Topology



The Juniper Networks router in the center of the illustration performs address translation in two steps. When the laptop requests a session with the **www.example.com** server that is in an IPv4-only domain, the Juniper Networks router performs the following:

- Translates the IPv6 laptop and DNS server addresses into IPv4 addresses.
- Translates the AAAA request from the laptop into an A request so that the DNS server can provide the IPv4 address.

When the DNS server responds with the A request, the Juniper Networks router performs the following:

- Translates the IPv4 DNS server address back into an IPv6 address.
- Translates the A request back into a AAAA request so that the laptop now has the 96-bit IPv6 address of the **www.example.com** server.

After the laptop receives the IPv6 version of the **www.example.com** server address, the laptop initiates a second session using the 96-bit IPv6 address to access that server. The Juniper Networks router performs the following:

- Translates the laptop IPv4 address directly into its IPv4 address.
- Translates the 96-bit IPv6 **www.example.com** server address into its IPv4 address.

Configuration of NAT-PT with DNS ALGs

To configure NAT-PT with DNS ALG , perform the following tasks:

Configuring the Application-Level Gateway

Step-by-Step Procedure

Configure the DNS application as the ALG to which the DNS traffic is destined. The DNS application protocol closes the DNS flow as soon as the DNS response is received. When you configure the DNS application protocol, you must specify the UDP protocol as the network protocol to match in the application definition.

To configure the DNS application:

1. In configuration mode, go to the **[edit applications]** hierarchy level.

   ```
   user@host# edit applications
   ```

2. Define the application name and specify the application protocol to use in match conditions in the first NAT rule.

   ```
   [edit applications]
   user@host#  set application application-name application-protocol protocol-name
   ```

   For example:

```
[edit applications]
user@host#  set application dns_alg application-protocol dns
```

3. Specify the protocol to match, in this case UDP.

```
[edit applications]
user@host#  set application application-name protocol type
```

For example:

```
[edit applications]
user@host# set application dns_alg protocol udp
```

4. Define the UDP destination port for additional packet matching, in this case the domain port.

```
[edit applications]
user@host#  set application application-name destination-port value
```

For example:

```
[edit applications]
user@host# set application dns_alg destination-port 53
```

Results

```
[edit applications]
user@host# show
application dns_alg {
    application-protocol dns;
    protocol udp;
    destination-port 53;
}
```

Configuring the NAT Pools

Step-by-Step Procedure

In this configuration, you configure two pools that define the addresses (or prefixes) used for NAT. These pools define the IPv4 addresses that are translated into IPv6 addresses. The first pool includes the IPv4 address of the source. The second pool defines the IPv4 address of the DNS server. To configure NAT pools:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

```
user@host# edit services nat
```

2. Specify the name of the first pool and the IPv4 source address (laptop).

```
[edit services nat]
user@host# set pool nat-pool-name address ip-prefix
```

For example:

```
[edit services nat]
user@host# set pool pool1 address 40.1.1.1/32
```

3. Specify the name of the second pool and the IPv4 address of the DNS server.

```
[edit services nat]
user@host# set pool nat-pool-name address ip-prefix
```

For example:

```
[edit services nat]
```

```
user@host# set pool pool2 address 50.1.1.1/32
```

Results The following sample output shows the configuration of NAT pools.

```
[edit services nat]
user@host# show
pool pool1 {
    address 40.1.1.1/32;
}
pool pool2 {
    address 50.1.1.1/32;
}
```

Configuring the DNS Server Session: First NAT Rule

Step-by-Step Procedure The first NAT rule is applied to DNS traffic going to the DNS server. This rule ensures that the DNS query and response packets are translated correctly. For this rule to work, you must configure a DNS ALG application and reference it in the rule. The DNS application was configured in "Configuring the DNS ALG Application" on page 46. In addition, you must specify the direction in which traffic is matched, the source address of the laptop, the destination address of the DNS server, and the actions to take when the match conditions are met.

To configure the first NAT rule:

1. In configuration mode, go to the **[edit services nat]** hierarchy level.

   ```
   user@host# edit services nat
   ```

2. Specify the name of the NAT rule.

   ```
   [edit services nat]
   user@host# edit rule rule-name
   ```

   For example:

   ```
   [edit services nat]
   user@host# edit rule rule1
   ```

3. Specify the name of the NAT term.

   ```
   [edit services nat rule rule-name]
   user@host# edit term term-name
   ```

   For example:

   ```
   [edit services nat rule rule1]
   user@host# edit term term1
   ```

4. Define the match conditions for this rule.

   a. Specify the IPv6 source address of the device (laptop) attempting to access an IPv4 address.

      ```
      [edit services nat rule rule-name term term-name]
      user@host# set from source-address source-address
      ```

      For example:

```
[edit services nat rule rule1 term term1]
user@host# set from source-address 2000::2/128
```

b. Specify the IPv6 destination address of the DNS server.

```
[edit services nat rule rule-name term term-name]
user@host# set from destination-address prefix
```

For example:

```
[edit services nat rule rule1 term term1]
user@host# set from destination-address 4000::2/128
```

c. Reference the DNS application to which the DNS traffic destined for port 53 is applied.

```
[edit services nat rule rule1 term term1]
user@host# set from applications application-name
```

In this example, the application name configured in the *Configuring the DNS Application* step is **dns_alg**:

```
[edit services nat rule rule1 term term1]
user@host# set from applications dns_alg
```

5. Define the actions to take when the match conditions are met. The source and destination pools you configured in "Configuring the NAT Pools" on page 74 are applied here.

a. Apply the NAT pool configured for source translation.

```
[edit services nat rule rule-name term term-name]
user@host# set then translated source-pool nat-pool-name
```

For example:

```
[edit services nat rule rule1 term term1]
user@host# set then translated source-pool pool1
```

b. Apply the NAT pool configured for destination translation.

```
[edit services nat rule rule-name term term-name]
user@host# set then translated destination-pool nat-pool-name
```

For example:

```
[edit services nat rule rule1 term term1]
user@host# set then translated source-pool pool2
```

6. Define the DNS ALG 96-bit prefix for IPv4-to-IPv6 address mapping.

```
[edit services nat rule rule-name term term-name]
user@host# set then translated dns-alg-prefix dns-alg-prefix
```

For example:

```
[edit services nat rule rule1 term term1]
user@host# set then translated dns-alg-prefix 10:10:10::0/96
```

7. Specify the type of NAT used for source and destination traffic.

```
[edit services nat rule rule-name term term-name]
user@host# set then translated translation-type basic-nat-pt
```


For example:

```
[edit services nat rule rule1 term term1]
user@host# set then translated translation-type basic-nat-pt
```

NOTE: In this example, since NAT is achieved using address-only translation, the **basic-nat-pt** translation type is used. To achieve NAT using address and port translation (NAPT), use the **napt-pt** translation type.

8. Specify the direction in which to match traffic that meets the rule conditions.

   ```
   [edit services nat rule rule-name]
   user@host# set match-direction (input | output)
   ```

   For example:

   ```
   [edit services nat rule rule1]
   user@host# set match-direction input
   ```

9. Configure system logging to record information from the services interface to the /var/log directory.

   ```
   [edit services nat rule rule-name term term-name]
   user@host# set then syslog
   ```

   For example:

   ```
   [edit services nat rule rule1 term term1]
   user@host# set then syslog
   ```

Results The following sample output shows the configuration of the first NAT rule that goes to the DNS server.

```
[edit services nat]
user@host# show
rule rule1 {
    match-direction input;
    term term1 {
        from {
            source-address {
                2000::2/128;
            }
            destination-address {
                4000::2/128;
            }
            applications dns_alg;
        }
        then {
            translated {
                source-pool pool1;
                destination-pool pool2;
                dns-alg-prefix 10:10:10::0/96;
                translation-type {
                    basic-nat-pt;

                }
            }
            syslog;
        }
    }
}
```

Configuring the HTTP Session: Second NAT Rule

Step-by-Step Procedure The second NAT rule is applied to destination traffic going to the IPv4 server (**www.example.com**). This rule ensures that NAT sessions are destined to the address mapped by the DNS ALG. For this rule to work, you must configure the DNS ALG address map that correlates the DNS query or response processing done by the first rule with the actual data sessions processed by the second rule. In addition, you must specify the direction in which traffic is matched: the IPv4 address for the IPv6 source address (laptop), the 96-bit prefix to prepend to the IPv4 destination address (www.example.com), and the translation type.

To configure the second NAT rule:

1. In configuration mode, go to the following hierarchy level.

   user@host# edit services nat

2. Specify the name of the NAT rule and term.

   [edit services nat]
   user@host# edit rule *rule-name* term *term-name*

   For example:

   [edit services nat]
   user@host# edit rule rule2 term term1

3. Define the match conditions for this rule:

   a. Specify the IPv6 address of the device attempting to access the IPv4 server.

      [edit services nat rule *rule-name* term *term-name*]
      user@host# **set from source-address *source-address***

      For example:

      [edit services nat rule rule2 term term1]
      user@host# **set from source-address 2000::2/128**

   b. Specify the 96-bit IPv6 prefix to prepend to the IPv4 server address.

      [edit services nat rule *rule-name* term *term-name*]
      user@host# **set from destination-address *prefix***

      For example:

      [edit services nat rule rule2 term term1]
      user@host# **set from destination-address 10:10:10::c0a8:108/128**

4. Define the actions to take when the match conditions are met.

   - Specify the prefix for the translation of the IPv6 source address.

     [edit services nat rule *rule-name* term *term-name*]
     user@host# **set then translated source-prefix *source-prefix***

     For example:

     [edit services nat rule rule2 term term1]
     user@host# **set then translated source-prefix 19.19.19.1/32**

5. Specify the type of NAT used for source and destination traffic.

   [edit services nat rule *rule-name* term *term-name*]
   user@host# **set then translated translation-type basic-nat-pt**

   For example:

   [edit services nat rule rule2 term term1]
   user@host# **set then translated translation-type basic-nat-pt**

   NOTE: In this example, since NAT is achieved using address-only translation, the **basic-nat-pt** translation type is used. To achieve NAT using address and port translation (NAPT), you must use the **napt-pt** translation type.

6. Specify the direction in which to match traffic that meets the conditions in the rule.

   [edit services nat rule *rule-name*]
   user@host# **set match-direction *(input | output)***

   For example:

   [edit services nat rule rule2]
   user@host# **set match-direction *input***

Results The following sample output shows the configuration of the second NAT rule.

```
[edit services nat]
user@host# show
rule rule2 {
    match-direction input;
    term term1 {
        from {
            source-address {
                2000::2/128;
            }
            destination-address {
                10:10:10::c0a8:108/128;
            }
        }
        then {
            translated {
                source-prefix 19.19.19.1/32;
                translation-type {
                    basic-nat-pt;
                }
            }
        }
    }
}
```

Configuring the Service Set

Step-by-Step Procedure This service set is an interface service set used as an action modifier across the entire services (**ms-**) interface. Stateful firewall and NAT rule sets are applied to traffic processed by the services interface.

To configure the service set:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   user@host# edit services

2. Define a service set.

   [edit services]
   user@host# edit service-set *service-set-name*

   For example:

   [edit services]
   user@host# edit service-set ss

3. Specify properties that control how system log messages are generated for the service set.

   [edit services service-set ss]
   user@host# set syslog host local services *severity-level*

   The example below includes all severity levels.

   [edit services service-set ss]
   user@host# set syslog host local services any

4. Specify the stateful firewall rule included in this service set.


```
[edit services service-set ss]
user@host# set stateful-firewall-rules rule1 severity-level
```

The example below references the stateful firewall rule defined in "Configuring the Stateful Firewall Rule" on page 82.

```
[edit services service-set ss]
user@host# set stateful-firewall-rules rule1
```

5. Define the NAT rules included in this service set.

```
[edit services service-set ss]
user@host# set nat-rules rule-name
```

The example below references the two rules defined in this configuration example.

```
[edit services service-set ss
user@host# set nat-rules rule1
user@host# set nat-rules rule2
```

6. Configure an adaptive services interface on which the service is to be performed.

```
[edit services service-set ss]
user@host# set interface-service service-interface interface-name
```

For example:

```
[edit services service-set ss
user@host# interface-service service-interface ms-2/0/0
```

Only the device name is needed, because the router software manages logical unit numbers automatically. The services interface must be an adaptive services interface for which you have configured **unit 0 family inet** at the **[edit interfaces *interface-name*]** hierarchy level in "Configuring Interfaces" on page 83.

Results The following sample output shows the configuration of the service set.

```
[edit services]
user@host# show
service-set ss {
    syslog {
        host local {
            services any;
        }
    }
    stateful-firewall-rules rule1;
    nat-rules rule1;
    nat-rules rule2;
    interface-service {
        service-interface ms-2/0/0;
    }
}
```

Configuring the Stateful Firewall Rule

Step-by-Step Procedure This example uses a stateful firewall to inspect packets for state information derived from past communications and other applications. The NAT-PT router checks the traffic flow matching the direction specified by the rule, in this case both input and output. When a packet is sent to the services (**ms-**) interface, direction information is carried along with it.

To configure the stateful firewall rule:

1. In configuration mode, go to the **[edit services stateful firewall]** hierarchy level.

   user@host# edit services stateful firewall

2. Specify the name of the stateful firewall rule.

   [edit services stateful-firewall]
   user@host# edit rule *rule-name*

   For example:

   [edit services stateful-firewall]
   user@host# edit rule rule1

3. Specify the direction in which traffic is to be matched.

   [edit services stateful-firewall rule *rule-name*]
   user@host# set match-direction (input | input-output | output)

   For example:

   [edit services stateful-firewall rule rule1]
   user@host# set match-direction *input-output*

4. Specify the name of the stateful firewall term.

   [edit services stateful-firewall rule *rule-name*]
   user@host# edit term *term-name*

   For example:

   [edit services stateful-firewall rule rule1]
   user@host# edit term term1

5. Define the terms that make up this rule.

   [edit services stateful-firewall rule *rule-name* term *term-name*]
   user@host# **set then accept**

   For example:

   [edit services stateful-firewall rule rule1 term term1]
   user@host# **set then accept**

Results The following sample output shows the configuration of the services stateful firewall.

```
[edit services]
user@host# show
stateful-firewall {
    rule rule1 {
        match-direction input-output;
        term term1 {
            then {
                accept;
            }
        }
    }
}
```

Configuring Interfaces

Step-by-Step Procedure After you have defined the service set, you must apply services to one or more interfaces installed on the router. In this example, you configure one interface on which you apply the service set for input and output traffic. When you apply the service set to an interface, it automatically ensures that packets are directed to the services (**ms-**) interface.

To configure the interfaces:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level.

   user@host# **edit interfaces**

2. Configure the interface on which the service set is applied to automatically ensure that packets are directed to the services (**ms-**) interface.

   a. For IPv4 traffic, specify the IPv4 address.

      [edit interfaces]
      user@host# **set ge-1/0/9 unit 0 family inet address 30.1.1.1/24**

   b. Apply the service set defined in "Configuring Interfaces" on page 83.

      [edit interfaces]
      user@host# **set ge-1/0/9 unit 0 family inet6 service input service-set ss**
      user@host# **set ge-1/0/9 unit 0 family inet6 service output service-set ss**

   c. For IPv6 traffic, specify the IPv6 address.

      [edit interfaces]
      user@host# **set ge-1/0/9 unit 0 family inet6 address 2000::1/64**

3. Specify the interface properties for the services interface that performs the service.

   [edit interfaces]


```
user@host# set ms-2/0/0 services-options syslog host local services any
user@host# set ms-2/0/0 unit 0 family inet
user@host# set ms-2/0/0 unit 0 family inet6
```

Results The following sample output shows the configuration of the interfaces for this example.

```
[edit interfaces]
user@host# show

ge-1/0/9 {
    unit 0 {
        family inet {
            address 30.1.1.1/24;
        }
        family inet6 {
            service {
                input {
                    service-set ss;
                }
                output {
                    service-set ss;
                }
            }
            address 2000::1/64;
        }
    }
}

ms-2/0/0 {
    services-options {
        syslog {
            host local {
                services any;
            }
        }
    }
    unit 0 {
        family inet;
        family inet6;
    }
}
```

Related Documentation

- Network Address Translation Overview on page 3
- Configuring NAT-PT on page 51
- Configuring Service Sets to be Applied to Services Interfaces
- Example: Configuring Layer 3 Services and the Services SDK on Two PICs
- dns-alg-prefix on page 104
- dns-alg-pool on page 103

Example: Configuring Port Forwarding with Twice NAT

The following example configures port forwarding with **twice-napt-44** as the translation type. The example also has stateful firewall and multiple port maps configured.

```
[edit services]
user@host# show
service-set in {
    syslog {
        host local {
            services any;
        }
    }
    stateful-firewall-rules r;
    nat-rules r;
    interface-service {
        service-interface sp-10/0/0.0;
    }
}
stateful-firewall {
    rule r {
        match-direction input;
        term t {
            from {
                destination-port {
                    range low 1 high 57000;
                }
            }
            then {
                reject;
            }
        }
    }
}
nat {
    pool x {
        address 12.0.0.2/32;
    }
    rule r {
        match-direction input;
        term t {
            from {
                destination-address {
                    14.0.0.2/32;
                }
                destination-port {
                    range low 10 high 20000;
                }
            }
            then {
                port-forwarding-mappings y;
             translated {
                        destination-pool x;
                        translation-type {
                            twice-napt-44;
                        }
                    }
                }
            }
```

```
    }
    port-forwarding y {
        destined-port 45;
        translated-port 23;
        destined-port 55;
        translated-port 33;
        destined-port 65;
        translated-port 43;
   }
}
adaptive-services-pics {
    traceoptions {
        file sp-trace;
        flag all;
    }
}
```

NOTE:

- Stateful firewall has precedence over port forwarding. In this example, for instance, no traffic destined to any port between 1 and 57000 will be translated.
- Up to 32 port maps can be configured.

Related Documentation

- Configuring Port Forwarding for Static Destination Address Translation on page 43

Example: Configuring an Oversubscribed Pool with Fallback to NAPT

The following configuration shows dynamic address translation from a large prefix to a small pool, translating a **/24** subnet to a pool of 10 addresses. When the addresses in the source pool (**src-pool**) are exhausted, NAT is provided by the NAPT overload pool (**pat-pool**).

```
[edit services nat]
pool src-pool {
  address-range low 192.16.2.1 high 192.16.2.10;
}
pool pat-pool {
  address-range low 192.2.11 high 192.16.2.12;
  port automatic;
}
rule myrule {
  match-direction input;
  term myterm {
    from {
      source-address 10.150.1.0/24;
    }
    then {
      translated {
        source-pool src-pool;
        overload-pool pat-pool;
        translation-type napt-44;
```

```
        }
      }
    }
  }
```

Example: Configuring an Oversubscribed Pool with No Fallback

The following configuration shows dynamic address translation from a large prefix to a small pool, translating a **/24** subnet to a pool of 10 addresses. Sessions from the first 10 host sessions are assigned an address from the pool on a first-come, first-served basis, and any additional requests are rejected. Each host with an assigned NAT can participate in multiple sessions.

```
[edit services nat]
pool my-pool {
  address-range low 10.10.10.1 high 10.10.10.10;
}
rule src-nat {
  match-direction input;
  term t1 {
    from {
      source-address 192.168.1.0/24;
    }
    then {
      translated {
        translation-type dynamic-nat44;
        source-pool my-pool;
      }
    }
  }
}
```

Example: Assigning Addresses from a Dynamic Pool for Static Use

The following configuration statically assigns a subset of addresses that are configured as part of a dynamic pool (**dynamic-pool**) to two separate static pools (**static-pool** and **static-pool2**).

```
[edit services nat]
pool dynamic-pool {
  address 20.20.10.0/24;
}
pool static-pool {
  address-range low 20.20.10.10 high 10.20.10.12;
}
pool static-pool2 {
  address 20.20.10.15/32;
}
rule src-nat {
  match-direction input;
  term t1 {
    from {
      source-address 30.30.30.0/24;
    }
```

```
        then {
            translation-type dynamic-nat44;
            source-pool dynamic-pool;
        }
    }
    term t2 {
        from {
            source-address 10.10.10.2;
        }
        then {
            translation-type basic-nat44;
            source-pool static-pool;
        }
    }
    term t3 {
        from {
            source-address 10.10.10.10;
        }
        then {
            translation-type basic-nat44;
            source-pool static-pool2;
        }
    }
}
```

Example: Configuring NAT Rules Without Defining a Pool

The following configuration performs NAT using the source prefix **20.20.10.0/24** without defining a pool.

```
[edit services nat]
rule src-nat {
    match-direction input;
    term t1 {
        then {
            translation-type dynamic-nat44;
            source-prefix 20.20.10.0/24;
        }
    }
}
```

The following configuration performs NAT using the destination prefix **20.20.10.0/32** without defining a pool.

```
[edit services nat]
rule src-nat {
    match-direction input;
    term t1 {
        from {
            destination-address 10.10.10.10/32;
            then {
                translation-type dnat44;
                destination-prefix 20.20.10.0/24;
            }
        }
    }
```

```
}
```

Example: Preventing Translation of Specific Addresses

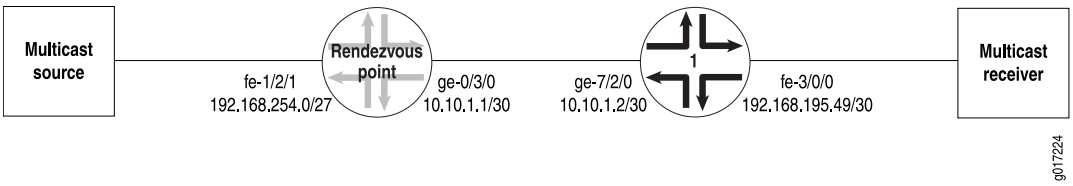
The following configuration specifies that NAT is not performed on incoming traffic from the source address **192.168.20.24/32**. Dynamic NAT is performed on all other incoming traffic.

```
[edit services nat]
pool my-pool {
   address-range low 10.10.10.1 high 10.10.10.16;
   port-automatic;
}
rule src-nat {
   match-direction input;
   term t0 {
      from {
         source-address 192.168.20.24/32;
      }
      then {
         no-translation;
      }
   }
   term t1 {
      then {
         translated {
            translation-type dynamic-nat44;
            source-pool my-pool;
         }
      }
   }
}
```

Example: Configuring NAT for Multicast Traffic

Figure 5 on page 89 illustrates the network setup for the following configuration, which allows IP multicast traffic to be sent to the Multiservices PIC.

Figure 5: Configuring NAT for Multicast Traffic



- Rendezvous Point Configuration on page 89
- Router 1 Configuration on page 92

Rendezvous Point Configuration

On the rendezvous point (RP), all incoming traffic from the multicast source at **192.168.254.0/27** is sent to the static NAT pool **mcast_pool**, where its source is translated

to **20.20.20.0/27**. The service set **nat_ss** is a next-hop service set that allows IP multicast traffic to be sent to the Multiservices DPC or Multiservices PIC. The inside interface on the PIC is **ms-1/1/0.1** and the outside interface is **ms-1/1/0.2**.

```
[edit services]
nat {
   pool mcast_pool {
      address 20.20.20.0/27;
   }
   rule nat_rule_1 {
      match-direction input;
      term 1 {
         from {
            source-address 192.168.254.0/27;
         }
      }
      then {
         translated {
            source-pool mcast_pool;
            translation-type basic-nat44;
         }
         syslog;
      }
   }
}
service-set nat_ss {
   allow-multicast;
   nat-rules nat_rule_1;
   next-hop-service {
      inside-service-interface ms-1/1/0.1;
      outside-service-interface ms-1/1/0.2;
   }
}
```

The Gigabit Ethernet interface **ge-0/3/0** carries traffic out of the RP to Router 1. The multiservices interface **ms-1/1/0** has two logical interfaces: **unit 1** is the inside interface for next-hop services and **unit 2** is the outside interface for next-hop services. Multicast source traffic comes in on the Fast Ethernet interface **fe-1/2/1**, which has the firewall filter **fbf** applied to incoming traffic.

```
[edit interfaces]
ge-0/3/0 {
   unit 0 {
      family inet {
         address 10.10.1.1/30;
      }
   }
}
ms-1/1/0 {
   unit 0 {
      family inet;
   }
   unit 1 {
      family inet;
      service-domain inside;
```



```
    }
    unit 2 {
      family inet;
      service-domain outside;
    }
  }
  fe-1/2/1 {
    unit 0 {
      family inet {
        filter {
          input fbf;
        }
        address 192.168.254.27/27;
      }
    }
  }
```

Multicast packets can only be directed to the Multiservices DPC or the Multiservices PIC using a next-hop service set. In the case of NAT, you must also configure a VPN routing and forwarding instance (VRF). Therefore, the routing instance **stage** is created as a "dummy" forwarding instance. To direct incoming packets to **stage**, you configure filter-based forwarding through a firewall filter called **fbf**, which is applied to the incoming interface **fe-1/2/1**. A lookup is performed in **stage.inet.0**, which has a multicast static route that is installed with the next hop pointing to the PIC's inside interface. All multicast traffic matching this route is sent to the PIC.

```
[edit firewall]
filter fbf {
  term 1 {
    then {
      routing-instance stage;
    }
  }
}
```

The routing instance **stage** forwards IP multicast traffic to the inside interface **ms-1/1/0.1** on the Multiservices DPC or Multiservices PIC:

```
[edit]
routing-instances stage {
  instance-type forwarding;
  routing-options {
    static {
      route 224.0.0.0/4 next-hop ms-1/1/0.1;
    }
  }
}
```

You enable OSPF and Protocol Independent Multicast (PIM) on the Fast Ethernet and Gigabit Ethernet logical interfaces over which IP multicast traffic enters and leaves the RP. You also enable PIM on the outside interface (**ms-1/1/0.2**) of the next-hop service set.

```
[edit protocols]
ospf {
  area 0.0.0.0 {
```

```
        interface fe-1/2/1.0 {
            passive;
        }
        interface lo0.0;
        interface ge-0/3/0.0;
    }
}
pim {
    rp {
        local {
            address 10.255.14.160;
        }
    }
    interface fe-1/2/1.0;
    interface lo0.0;
    interface ge-0/3/0.0;
    interface ms-1/1/0.2;
}
```

As with any filter-based forwarding configuration, in order for the static route in the forwarding instance **stage** to have a reachable next hop, you must configure routing table groups so that all interface routes are copied from **inet.0** to the routing table in the forwarding instance. You configure routing tables **inet.0** and **stage.inet.0** as members of **fbf_rib_group**, so that all interface routes are imported into both tables.

```
[edit routing-options]
interface-routes {
    rib-group inet fbf_rib_group;
}
rib-groups fbf_rib_group {
    import-rib [ inet.0 stage.inet.0 ];
}
multicast {
    rpf-check-policy no_rpf;
}
```

Reverse path forwarding (RPF) checking must be disabled for the multicast group on which source NAT is applied. You can disable RPF checking for specific multicast groups by configuring a policy similar to the one in the example that follows. In this case, the **no_rpf** policy disables RPF check for multicast groups belonging to **224.0.0.0/4**.

```
[edit policy-options]
policy-statement no_rpf {
    term 1 {
        from {
            route-filter 224.0.0.0/4 orlonger;
        }
        then reject;
    }
}
```

Router 1 Configuration

The Internet Group Management Protocol (IGMP), OSPF, and PIM configuration on Router 1 is as follows. Because of IGMP static group configuration, traffic is forwarded out

fe-3/0/0.0 to the multicast receiver without receiving membership reports from host members.

```
[edit protocols]
igmp {
  interface fe-3/0/0.0 {
  }
}
ospf {
  area 0.0.0.0 {
    interface fe-3/0/0.0 {
      passive;
    }
    interface lo0.0;
    interface ge-7/2/0.0;
  }
  pim {
    rp {
      static {
        address 10.255.14.160;
      }
    }
    interface fe-3/0/0.0;
    interface lo0.0;
    interface ge-7/2/0.0;
  }
}
```

The routing option creates a static route to the NAT pool, **mcast_pool**, on the RP.

```
[edit routing-options]
static {
  route 20.20.20.0/27 next-hop 10.10.1.1;
}
```

Example: Configuring Port Forwarding with Twice NAT

The following example configures port forwarding with **twice-napt-44** as the translation type. The example also has stateful firewall and multiple port maps configured.

```
[edit services]
user@host# show
service-set in {
    syslog {
        host local {
            services any;
        }
    }
    stateful-firewall-rules r;
    nat-rules r;
    interface-service {
        service-interface sp-10/0/0.0;
    }
}
stateful-firewall {
    rule r {
        match-direction input;
```

```
        term t {
            from {
                destination-port {
                    range low 1 high 57000;
                }
            }
            then {
                reject;
            }
        }
    }
}
nat {
    pool x {
        address 12.0.0.2/32;
    }
    rule r {
        match-direction input;
        term t {
            from {
                destination-address {
                    14.0.0.2/32;
                }
                destination-port {
                    range low 10 high 20000;
                }
            }
            then {
                port-forwarding-mappings y;
             translated {
                    destination-pool x;
                    translation-type {
                        twice-napt-44;
                    }
                }
            }
        }
    }
    port-forwarding y {
        destined-port 45;
        translated-port 23;
        destined-port 55;
        translated-port 33;
        destined-port 65;
        translated-port 43;
   }
}
adaptive-services-pics {
    traceoptions {
        file sp-trace;
        flag all;
    }
}
```

NOTE:

- Stateful firewall has precedence over port forwarding. In this example, for instance, no traffic destined to any port between 1 and 57000 will be translated.
- Up to 32 port maps can be configured.

Related Documentation

- Configuring Port Forwarding for Static Destination Address Translation on page 43

CHAPTER 4

Configuration Statements

address (Services NAT Pool)

Syntax address *ip-prefix</prefix-length>*;

Hierarchy Level [edit services nat pool *nat-pool-name*]

Release Information Statement introduced before Junos OS Release 7.4.
prefix option enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the NAT pool prefix value.

Options ***prefix***—Specify an IPv4 or IPv6 prefix value.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Addresses and Ports for Use in NAT Rules on page 11

address-allocation

Syntax address-allocation round-robin;

Hierarchy Level [edit services nat pool *pool-name*]

Release Information Statement introduced in Junos OS Release 11.2.

Description When you use round-robin allocation, one port is allocated from each address in a range before repeating the process for each address in the next range. After ports have been allocated for all addresses in the last range, the allocation process wraps around and allocates the next unused port for addresses in the first range.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Addresses and Ports for Use in NAT Rules on page 11

address-range

Syntax address-range low *minimum-value* high *maximum-value*;

Hierarchy Level [editservices nat pool *nat-pool-name*]

Release Information Statement introduced before Junos OS Release 7.4.
minimum-value and ***maximum-value*** options enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the NAT pool address range.

Options ***minimum-value***—Lower boundary for the IPv4 or IPv6 address range.

maximum-value—Upper boundary for the IPv4 or IPv6 address range.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Addresses and Ports for Use in NAT Rules on page 11

application-sets (Services NAT)

Syntax applications-sets *set-name*;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define one or more target application sets.

Options ***set-name***—Name of the target application set.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23

applications (Services NAT)

Syntax applications [*application-names*];

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define one or more application protocols to which the NAT services apply.

Options ***application-name***—Name of the target application.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23

cgn-pic

Syntax cgn-pic;

Hierarchy Level [edit interfaces *interface-name* services-options]

Release Information Statement introduced in Junos OS Release 11.2.

Description Restrict usage of the service PIC to carrier-grade NAT (CGN) or associated services (intrusion detection, stateful firewall, and softwire). All memory is available for CGN or related services and can be used for CGN scaling.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Service Sets

destination-address

Syntax destination-address (*address* | any-unicast) <except>;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced before Junos OS Release 7.4.
any-unicast and **except** options introduced in Junos OS Release 7.6.
address option enhanced to support IPv6 and addresses in Junos OS Release 8.5.

Description Specify the destination address for rule matching.

Options ***address***—Destination IPv4 or IPv6 address or prefix value.

any-unicast—Any unicast packet.

except—(Optional) Prevent the specified address, prefix, or unicast packets from being translated.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23

destination-address-range

Syntax destination-address-range low *minimum-value* high *maximum-value* <except>;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced in Junos OS Release 7.6.
minimum-value and ***maximum-value*** options enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the destination address range for rule matching.

Options ***minimum-value***—Lower boundary for the IPv4 or IPv6 address range.

maximum-value—Upper boundary for the IPv4 or IPv6 address range.

except—(Optional) Prevent the specified address range from being translated.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23

destination-pool

Syntax destination-pool *nat-pool-name*;

Hierarchy Level [edit services nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the destination address pool for translated traffic.

Options ***nat-pool-name***—Destination pool name.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

destination-port range

Syntax destination-port range *high* | *low*;

Hierarchy Level [edit services nat rule *rule-name* term *term-name* from (Services NAT)]

Release Information Statement introduced in Junos OS Release 11.4.

Description Specify the destination port range for rule matching.

Options ***high***—Upper limit of port range for matching.

low—Lower limit of port range for matching.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Port Forwarding for Static Destination Address Translation on page 43

destination-prefix

Syntax destination-prefix *destination-prefix*;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 7.6.
destination-prefix option enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the destination prefix for translated traffic.

Options ***destination-prefix***—IPv4 or IPv6 destination prefix value.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

destination-prefix-list

Syntax destination-prefix-list *list-name* <except>;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced in Junos OS Release 8.2.

Description Specify the destination prefix list for rule matching. You configure the prefix list by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level.

Options ***list-name***—Destination prefix list.

except—(Optional) Exclude the specified prefix list from rule matching.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23
- Routing Policy Configuration Guide

destined-port

Syntax destined-port *port id*;

Hierarchy Level [editservices nat port-forwarding *map-name*]

Release Information Statement introduced in Junos OS Release 11.4.

Description Specify the port from where traffic has to be forwarded.

Options ***port id***—The destination port number from where traffic will be forwarded.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- port-forwarding on page 111
- translated-port on page 119

dns-alg-pool

Syntax dns-alg-pool *dns-alg-pool*;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 10.4.

Description Specify the Network Address Translation (NAT) pool for destination translation.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Rules on page 21

dns-alg-prefix

Syntax dns-alg-prefix *dns-alg-prefix*;

Hierarchy Level [edit services nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 10.4.

Description Set the Domain Name System (DNS) application-level gateway (ALG) 96-bit prefix for mapping IPv4 addresses to IPv6 addresses.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Rules on page 21

from (Services NAT)

Syntax

```
from {
  application-sets set-name;
  applications [ application-names ];
  destination-address (address | any-unicast) <except>;
  destination-address-range low minimum-value high maximum-value <except>;
  source-address address (address | any-unicast) <except>;
  source-address-range low minimum-value high maximum-value <except>;
}
```

Hierarchy Level [edit services nat rule *rule-name* term *term-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify input conditions for the NAT term.

Options For information on match conditions, see the description of firewall filter match conditions in the Routing Policy Configuration Guide.

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Rules on page 21

hint

Syntax hint [*hint-strings*];

Hierarchy Level [editservices nat pool *nat-pool-name* pgcp]

Release Information Statement introduced in Junos OS Release 9.0.

Description Configure a hint that enables the border gateway function (BGF) to choose a NAT pool by direction rather than by virtual interface. The BGF matches the configured hint with a termination hint located in the Direction field of a nonstandard termination ID.

Default When no hint is configured, the BGF can choose any NAT pool associated with the virtual interface.

Options ***hint-string***—Alphanumeric string of up to three characters that the BGF uses to match with a termination hint located in the Direction field of a nonstandard termination ID. You can also include underscores (_) and hyphens (-) within the string. To specify a list of hints, use the format: **[hint *xx* hint *yy*]**.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

ipv6-multicast-interfaces

Syntax
```
ipv6-multicast-interfaces (all | interface-name) {
    disable;
}
```

Hierarchy Level [editservices nat],
[edit services softwire]

Release Information Statement introduced in Junos OS Release 9.1.

Description Enable multicast filters on Ethernet interfaces when IPv6 NAT is used for neighbor discovery.

Options **all**—Enable filters on all interfaces.

disable—Disable filters on the specified interfaces.

interface-name—Enable filters on a specific interface only.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring IPv6 Multicast Filters on page 11
- Configuring IPv6 Multicast Interfaces

match-direction

Syntax match-direction (input | output);

Hierarchy Level [editservices nat rule *rule-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the direction in which the rule match is applied.

Options **input**—Apply the rule match on input.

output—Apply the rule match on output.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Rules on page 21

nat-type

Syntax nat-type (full-cone | symmetric);

Hierarchy Level [edit services nat rule *rule-name* term *term-name*]

Release Information Statement introduced in JUNOS Release 8.5.

Description Specify whether the term supports full-cone or traditional (symmetric) NAT.

Default **symmetric**

Options **full-cone**—Support full-cone NAT processing, in which all requests from the same internal IP address and port are mapped to the same external IP address and port.

symmetric—Support traditional NAT address matching only.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Type for Terms in NAT Rules on page 21

no-translation

Syntax no-translation;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then]

Release Information Statement introduced in Junos OS Release 7.6.

Description Specify that traffic is not to be translated.

Options none

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

overload-pool

Syntax overload-pool *overload-pool-name*;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 7.6.

Description Specify an address pool that can be used if the source pool becomes exhausted.

Options ***overload-pool-name***—Name of the overload pool.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

overload-prefix

Syntax overload-prefix *overload-prefix*;

Hierarchy Level [edit services nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 7.6.

Description Specify the prefix that can be used if the source pool becomes exhausted.

Options ***overload-prefix***—Prefix value.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

pgcp

Syntax
```
pgcp {
  hint [ hint-strings ];
  ports-per-session ports;
  remotely-controlled;
  transport [ transport-protocols ];
}
```

Hierarchy Level [edit services nat pool *nat-pool-name*]

Release Information Statement introduced in Junos OS Release 8.4.
remotely-controlled and **ports-per-session** statements added in Junos OS Release 8.5.
hint statement added in Junos OS Release 9.0.

Description Specify that the NAT pool is used exclusively by the BGF.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

pool

Syntax

```
pool nat-pool-name {
  address ip-prefix</prefix-length>;
  address-allocation round-robin;
  address-range low minimum-value high maximum-value;
  mapping-timeout seconds;
  pgcp {
    hint [ hint-strings ];
    ports-per-session ports;
    remotely-controlled:
    transport [ transport-protocols ];
  }
  port (automatic | range low minimum-value high maximum-value) {
    preserve-parity;
    preserve-range;
    secured-port-block-allocation {
      active-block-timeout timeout-seconds;
      block-size block-size;
      max-blocks-per-user max-blocks;
    }
  }
}
```

Hierarchy Level [edit services nat]

Release Information

Statement introduced before Junos OS Release 7.4.
pgcp statement added in Junos OS Release 8.4.
remotely-controlled and **ports-per-session** statements added in Junos OS Release 8.5.
hint statement added in Junos OS Release 9.0.
address-allocation statement added in Junos OS Release 11.2.

Description Specify the NAT name and properties.

Options

nat-pool-name—Identifier for the NAT address pool.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Addresses and Ports for Use in NAT Rules on page 11

port

Syntax

```
port (automatic | range low minimum-value high maximum-value random-allocation) {
  preserve-parity;
  preserve-range;
  deterministic-port-block-allocation <block-size block-size>;
  secured-port-block-allocation {
    active-block-timeout timeout-seconds;
    block-size block-size;
    max-blocks-per-user max-blocks;
  }
}
```

Hierarchy Level [edit services nat pool *nat-pool-name*]

Release Information

port statement introduced before Junos OS Release 7.4.
random-allocation statement introduced in Junos OS Release 9.3.
secured-port-block-allocation statement introduced in Junos OS Release 11.2.
deterministic-port-block-allocation statement introduced in Junos OS Release 12.1.

Description Specify the NAT pool port or range. You can configure an automatically assigned port or specify a range with minimum and maximum values.

Options

automatic—Router-assigned port.

minimum-value—Lower boundary for the port range.

maximum-value—Upper boundary for the port range.

preserve-parity—Allocate ports with same parity as the original port.

preserve-range—Preserve privileged port range after translation.

random-allocation—Allocate ports within a specified range randomly.

Other options are described separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Addresses and Ports for Use in NAT Rules on page 11

port-forwarding

Syntax

```
port-forwarding map-name {
  destined-port;
  translated-port;
}
```

Hierarchy Level [edit services nat]

Release Information Statement introduced in Junos OS Release 11.4.

Description Specify the mapping for port forwarding.

Options *map-name*—Identifier for the port forwarding map.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Port Forwarding for Static Destination Address Translation on page 43
- Configuring Port Forwarding Without Destination Address Translation

port-forwarding-mappings

Syntax port-forwarding-mappings *map-name*;

Hierarchy Level [edit services nat rule *rule-name* term *term-name* then]

Release Information Statement introduced in Junos OS Release 11.4.

Description Specify the name for mapping port forwarding in a Network Address Translation configuration.

Options *map-name*—Identifier for the port forwarding mapping.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Port Forwarding for Static Destination Address Translation on page 43
- Configuring Port Forwarding Without Destination Address Translation

ports-per-session

Syntax ports-per-session *ports*;

Hierarchy Level [editservices nat pool *nat-pool-name* pgcp]

Release Information Statement introduced in Junos OS Release 8.4.

Description Configure the number of ports required to support Real-Time Transport Protocol (RTP), Real-Time Control Protocol (RTCP), Real-Time Streaming Protocol (RTSP), and forward error correction (FEC) for voice and video flows on the Multiservices PIC.

Options ***number-of-ports***—Number of ports to enable: 2 or 4 for combined voice and video services.
Default: 2

Required Privilege Level interface—To view this statement in the configuration.
interface–control—To add this statement to the configuration.

remotely-controlled

Syntax remotely-controlled;

Hierarchy Level [editservices nat pool *nat-pool-name* pgcp]

Release Information Statement introduced in Junos OS Release 8.5.

Description Configure the addresses and ports in a NAT pool to be remotely controlled by the gateway controller.

Required Privilege Level interface—To view this statement in the configuration.
interface–control—To add this statement to the configuration.

rule

Syntax

```
rule rule-name {
  match-direction (input | output);
  term term-name {
    from {
      application-sets set-name;
      applications [ application-names ];
      destination-address (address | any-unicast) <except>;
      destination-address-range low minimum-value high maximum-value <except>;
      source-address (address | any-unicast) <except>;
      source-address-range low minimum-value high maximum-value <except>;
    }
    then {
      no-translation;
      translated {
        address-pooling paired;
        destination-pool nat-pool-name;
        destination-prefix destination-prefix; destination-prefix;
        dns-alg-pool dns-alg-pool;
        dns-alg-prefix dns-alg-prefix;
        filtering-type endpoint-independent;
        mapping-type endpoint-independent;
        overload-pool overload-pool;
        overload-prefix overload-prefix;
        source-pool nat-pool-name;
        source-prefix source-prefix;
        translation-type (basic-nat-pt | basic-nat44 | basic-nat66 | dnat-44 | dynamic-nat44
          | napt-44 | napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 |
          twice-dynamic-nat-44 | twice-napt-44);
        }
      }
      syslog;
    }
  }
}
```

Hierarchy Level

[edit services nat],
[edit services nat rule-set *rule-set-name*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the rule the router uses when applying this service.

Options

rule-name—Identifier for the collection of terms that make up this rule.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring NAT Rules on page 21

rule-set

Syntax
```
rule-set rule-set-name {
    [ rule rule-names ];
}
```

Hierarchy Level [edit services nat]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the rule set the router uses when applying this service.

Options ***rule-set-name***—Identifier for the collection of rules that constitute this rule set.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring NAT Rule Sets on page 27

services (NAT)

Syntax services nat { ... }

Hierarchy Level [edit]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the service rules to be applied to traffic.

Options **nat**—Identifies the NAT set of rules statements.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

secured-port-block-allocation

Syntax

```
secured-port-block-allocation {
  active-block-timeout timeout-seconds;
  block-size block-size;
  max-blocks-per-address max-blocks;
}
```

Hierarchy Level [edit services nat pool *pool-name* port]

Release Information Statement introduced in Junos OS Release 11.2.

Description When you use block allocation, one or more blocks of ports in a NAT pool address range are available for assignment to a subscriber.

Options

block-size—Number of ports included in a block.

Default: 128

Range: 1 to 60,000

max-blocks—Maximum number of blocks that can be allocated to a user address.

Default: 8

Range: 1 to 512

timeout-seconds—Interval, in seconds, during which a block is active. After timeout, a new block is allocated, even if ports are available in the active block.

Default: 0—The default timeout of the active block is 0 (infinite). In this case, the active block transitions to inactive only when it runs out of ports and a new block is allocated. Any inactive block without any ports in use will be freed to the NAT pool.

Range: Any value greater than or equal to 120.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Addresses and Ports for Use in NAT Rules on page 11

source-address (NAT)

Syntax source-address (*address* | any-unicast) <except>;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced before Junos OS Release 7.4.
any-unicast and **except** options introduced in Junos OS Release 7.6.
address option enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the source address for rule matching.

Options ***address***—Source IPv4 or IPv6 address or prefix value.

any-unicast—Any unicast packet.

except—(Optional) Prevent the specified address or unicast packets from being translated.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23

source-address-range

Syntax source-address-range low *minimum-value* high *maximum-value* <except>;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced in Junos OS Release 7.6.
minimum-value and ***maximum-value*** options enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the source address range for rule matching.

Options ***minimum-value***—Lower boundary for the IPv4 or IPv6 address range.

maximum-value—Upper boundary for the IPv4 or IPv6 address range.

except—(Optional) Prevent the specified address range from being translated.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23

source-pool

Syntax source-pool *nat-pool-name*;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the source address pool for translated traffic.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

source-prefix

Syntax source-prefix *source-prefix*;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 7.6.
source-prefix option enhanced to support IPv6 addresses in Junos OS Release 8.5.

Description Specify the source prefix for translated traffic.

Options ***source-prefix***—IPv4 or IPv6 source prefix value.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

source-prefix-list

Syntax source-prefix-list *list-name* <except>;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* from]

Release Information Statement introduced in Junos OS Release 8.2.

Description Specify the source prefix list for rule matching. You configure the prefix list by including the **prefix-list** statement at the **[edit policy-options]** hierarchy level.

Options ***list-name***—Destination prefix list.

except—(Optional) Exclude the specified prefix list from rule matching.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Match Conditions in NAT Rules on page 23
- Routing Policy Configuration Guide

syslog

Syntax syslog;

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then]

Release Information Statement introduced before Junos OS Release 7.4.

Description Enable system logging. The system log information from the Multiservices PIC is passed to the kernel for logging in the **/var/log** directory.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

translated-port

Syntax translated-port *port id*;

Hierarchy Level [edit services nat port-forwarding *map-name*]

Release Information Statement introduced in Junos OS Release 11.4.

Description Specify the port to which all traffic will be translated.

Options ***port id***—The port number to which traffic will be translated.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- port-forwarding on page 111
- destined-port on page 103

term

Syntax

```
term term-name {
  from {
    application-sets set-name;
    applications [ application-names ];
    destination-address (address | any-unicast) <except>;
    destination-address-range low minimum-value high maximum-value <except>;
    source-address (address | any-unicast) <except>;
    source-address-range low minimum-value high maximum-value <except>;
  }
  then {
    no-translation;
    translated {
      address-pooling paired;
      destination-pool nat-pool-name;
      destination-prefix destination-prefix;
      dns-alg-pool dns-alg-pool;
      dns-alg-prefix dns-alg-prefix;
      filtering-type endpoint-independent;
      mapping-type endpoint-independent;
      source-pool nat-pool-name;
      source-prefix source-prefix;
      translation-type (basic-nat-pt | basic-nat44 | basic-nat66 | dnat-44 | dynamic-nat44
        | napt-44 | napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 |
        twice-dynamic-nat-44 | twice-napt-44);
      }
    }
    syslog;
  }
}
```

Hierarchy Level [edit services nat rule *rule-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the NAT term properties.

Options *term-name*—Identifier for the term.

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring NAT Rules on page 21

then

Syntax

```
then {
  no-translation;
  translated {
    address-pooling paired;
    destination-pool nat-pool-name;
    destination-prefix destination-prefix;
    dns-alg-pool dns-alg-pool;
    dns-alg-prefix dns-alg-prefix;
    filtering-type endpoint-independent;
    mapping-type endpoint-independent;
    source-pool nat-pool-name;
    source-prefix source-prefix;
    translation-type (basic-nat-pt | basic-nat44 | basic-nat66 | dnat-44 | dynamic-nat44
      | napt-44 | napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 |
      twice-dynamic-nat-44 | twice-napt-44);
    }
  }
  syslog;
}
```

Hierarchy Level [edit services nat rule *rule-name* term *term-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define the NAT term actions.

Options The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring NAT Rules on page 21

translated

Syntax

```
translated {
  address-pooling paired;
  destination-pool nat-pool-name;
  dns-alg-pool dns-alg-pool;
  dns-alg-prefix dns-alg-prefix;
  filtering-type endpoint-independent;
  mapping-type endpoint-independent;
  source-pool nat-pool-name;
  translation-type (basic-nat-pt | basic-nat44 | basic-nat66 | dnat-44 | dynamic-nat44 |
    napt-44 | napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 | twice-dynamic-nat-44
    | twice-napt-44)
  }
}
```

Hierarchy Level [edit services nat rule *rule-name* term *term-name* then]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define properties for translated traffic.

Options The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Actions in NAT Rules on page 24

translation-type

Syntax translation-type (basic-nat-pt | basic-nat44 | basic-nat66 | nat-44 | deterministic-napt44 | dnat-44 | dynamic-nat44 | napt-44 | napt-66 | napt-pt | stateful-nat64 | twice-basic-nat-44 | twice-dynamic-nat-44 | twice-napt-44)

Hierarchy Level [editservices nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced before Junos OS Release 7.4.
The following options introduced in Junos OS Release 11.2, replacing all previous options:

- **basic-nat44**
- **basic-nat66**
- **basic-nat-pt**
- **dnat-44**
- **dynamic-nat44**
- **napt-44**
- **napt-66**
- **napt-pt**
- **stateful-nat64**

twice-basic-nat-44 option introduced in Junos OS Release 11.4.
twice-dynamic-nat-44 option introduced in Junos OS Release 11.4.
twice-napt-44 option introduced in Junos OS Release 11.4.
deterministic-napt44 option introduced in Junos OS Release 12.1.

Description Specify the NAT translation types.

Options

- **basic-nat44**—Translate the source address statically (IPv4 to IPv4).
- **basic-nat66**—Translate the source address statically (IPv6 to IPv6).
- **basic-nat-pt**—Translate the addresses of IPv6 hosts as they originate sessions to the IPv4 hosts in the external domain. The **basic-nat-pt** option is always implemented with DNS ALG.
- **deterministic-napt44**—Translate as **napt-44**, and use deterministic port block allocation for port translation.
- **dnat-44**—Translate the destination address statically (IPv4 to IPv4).
- **dynamic-nat44**—Translate only the source address by dynamically choosing the NAT address from the source address pool.
- **napt-44**—Translate the transport identifier of the IPv4 private network to a single IPv4 external address.

- **napt-66**—Translate the transport identifier of the IPv6 private network to a single IPv6 external address.
- **napt-pt**—Bind addresses in an IPv6 network with addresses in an IPv4 network and vice versa to provide transparent routing for the datagrams traversing between the address realms.
- **stateful-nat64**—Implement dynamic address and port translation for source IP addresses (IPv6-to-IPv4) and prefix removal translation for the destination IP addresses (IPv6-to-IPv4).
- **twice-basic-nat-44**—Translate the source and destination addresses statically (IPv4 to IPv4).
- **twice-dynamic-nat-44**—Translate the source address by dynamically choosing the NAT address from the source address pool. Translate the destination address statically.
- **twice-dynamic-napt-44**—Translate the transport identifier of the IPv4 private network to a single IPv4 external address. Translate the destination address statically.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring Actions in NAT Rules on page 24

transport

Syntax
transport [*transport-protocols*];

Hierarchy Level
[edit services nat pool *nat-pool-name* pgcp]

Release Information
Statement introduced in Junos OS Release 9.2.

Description
Configure the BGF to select a NAT pool based on transport protocol type.

Options
[*transport-protocol*]—One or more transport protocols.

Values: **rtp-avp**, **tcp**, **udp**

Syntax: One or more protocols. If you specify more than one protocol, you must enclose all protocols in brackets.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

use-dns-map-for-destination-translation

Syntax use-dns-map-for-destination-translation;

Hierarchy Level [edit services nat rule *rule-name* term *term-name* then translated]

Release Information Statement introduced in Junos OS Release 10.4.

Description Enable the Domain Name System (DNS) application-level gateway (ALG) address map for destination translation.

NOTE: This statement is deprecated and might be removed completely in a future release.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

PART 3

Administration

CHAPTER 5

Network Address Translation Operational Mode Commands

show services nat pool

Syntax
show services nat pool
<brief | detail>
<*pool-name*>
pgcp <ports-per-session | remotely-controlled>

Release Information
Command introduced before Junos OS Release 7.4.
pgcp option added in Junos OS Release 8.5.

Description
Display information about Network Address Translation (NAT) pools.

Options
none—Display standard information about all NAT pools.

brief | detail—(Optional) Display the specified level of output.

pool-name—(Optional) Display information about the specified NAT pool.

pgcp—(Optional) Display information about a NAT pool that is exclusive to the BGF.

ports-per-session—(Optional) Display the number of ports allocated per session from the NAT pool.

remotely-controlled—(Optional) Display if the NAT pool is explicitly specified by the gateway controller.

Required Privilege Level
view

List of Sample Output
show services nat pool brief on page 132
show services nat pool detail on page 132
show services nat pool for Secured Port Block Allocation on page 132
show services nat pool for Deterministic Port Block Allocation on page 132
show services nat pool detail for Port Block Allocation on page 132

Output Fields
Table 5 on page 130 lists the output fields for the **show services nat pool** command. Output fields are listed in the approximate order in which they appear.

Table 5: show services nat pool Output Fields

Field Name	Field Description	Level of Output
Interface	Name of an adaptive services interface.	All levels
Service set	Name of a service set. Individual empty service sets are not displayed, but if none of the service sets has any flows, a flow table header is printed for each service set.	All levels
NAT pool	Name of the Network Address Translation pool.	All levels

Table 5: show services nat pool Output Fields *(continued)*

Field Name	Field Description	Level of Output
Type or **Translation type**	Address translation type: **basic-nat-pt**, **basic-nat44**, **basic-nat66**, **detministic-napt44**, **dnat-44**, **dynamic-nat44**, **napt44**, **napt-66**, **napt-pt**, **stateful-nat64**, **twice-basic-nat-44**, **twice-dynamic-nat-44**, **twice-dynamic-napt-44**.	All levels
Address or **Address range**	IPv4 address range of the pool.	All levels
Port or **Port range**	Port range of the pool. Applicable only for dynamic NAT pools. Not displayed for static NAT pools.	All levels
Ports used' or **Ports in use**	Number of ports allocated in this pool with this name. Applicable only for dynamic NAT pools. Not displayed for static NAT pools.	All levels
Port block type	Type of port block allocation: secured or deterministic	All levels
Out of port errors	Number of port allocation errors. Applicable only for dynamic NAT pools. Not displayed for static NAT pools.	**detail**
Max ports used	Maximum number of ports used. Applicable only for dynamic NAT pools. Not displayed for static NAT pools.	**detail**
Addresses in use	Number of addresses in use for dynamic source address NAT pools.	**detail**
Current EIF Inbound flows count	Current count of EIF inbound flows, including all EIF flows per pool.	
EIF flow limit exceeded drops	Current number of flow drops due to exceeded flow limit. This number is per pool, not per EIF mapping.	

Sample Output

show services nat pool brief

```
user@host> show services nat pool brief

Interface: ms-1/0/0, Service set: s1
NAT pool          Type    Address                          Port        Ports used
dest-pool         DNAT-44 10.10.10.2-10.10.10.2
napt-pool         NAPT-44 50.50.50.1-50.50.50.254          1024-63487  0
source-dynamic-pool DYNAMIC NAT44 40.40.40.1-40.40.40.254
source-static-pool BASIC NAT44 30.30.30.1-30.30.30.254
```

show services nat pool detail

```
user@host> show services nat pool detail

Interface: ms-1/0/0, Service set: s1
  NAT pool: dest-pool, Translation type: DNAT-44
    Address range: 10.10.10.2-10.10.10.2
  NAT pool: napt-pool, Translation type: NAPT-44
    Address range: 50.50.50.1-50.50.50.254
    Port range: 1024-63487, Ports in use: 0, Out of port errors: 0, Max ports
used: 0
  NAT pool: source-dynamic-pool, Translation type: DYNAMIC NAT44
    Address range: 40.40.40.1-40.40.40.254
    Out of address errors: 0, Addresses in use: 0
  NAT pool: source-static-pool, Translation type: BASIC NAT44
    Address range: 30.30.30.1-30.30.30.254
```

show services nat pool for Secured Port Block Allocation

```
user@host> show services nat pool
Interface: sp-2/0/0, Service set: in
NAT pool          Type    Address                          Port        Ports used
mypool            dynamic 3.3.3.3-3.3.3.10                 512-65535   0
                          3.3.3.15-3.3.3.20
                          3.3.3.25-3.3.3.30
                          3.3.3.95-3.3.3.200
Port block size: 64, Max port blocks per address: 1, Active block timeout: 86400,
 Effective port range: 1024-65471,
Effective number of port blocks: 126882, Effective number of ports: 8120448, Port
 block efficiency: nan

Interface: sp-2/1/0, Service set: in1
NAT pool          Type    Address                          Port        Ports used
mypool1           dynamic 9.9.9.1-9.9.9.254                512-65535   0
Port block size: 64, Max port blocks per address: 1, Active block timeout: 86400,
 Effective port range: 1024-65471,
Effective number of port blocks: 255778, Effective number of ports: 16369792,
Port block efficiency: nan
```

show services nat pool for Deterministic Port Block Allocation

```
user@host> show services nat pool

Interface: sp-2/0/0, Service set: ss2
NAT pool          Type    Address                          Port        Ports Used
pba               dynamic 33.33.33.1-33.33.33.128          512-65535   6604
Port block type: Deterministic port block, Port block size: 200
```

show services nat pool detail for Port Block

```
user@host> show services nat pool detail

Interface: sp-2/0/0, Service set: s
```

Allocation

```
  NAT pool: napt_pool, Translation type: dynamic
    Address range: 44.1.1.1-44.1.1.1
    Port range: 1024-65535, Ports in use: 0, Out of port errors: 0, Max ports
used: 0
    AP-P out of port errors: 0
    Current EIF Inbound flows count: 0
    EIF flow limit exceeded drops: 0
```

Sample Output

show services nat mappings

Syntax show services nat mappings
<brief | detail | summary>
<*pool-name*>

Release Information Command introduced in Junos OS Release 10.1.
summary option introduced in Junos OS Release 11.1.

Description Display information about Network Address Translation (NAT) address and port mappings.

Options **none**—Display standard information about all NAT pools.

brief | detail | summary—(Optional) Display the specified level of output.

pool-name—(Optional) Display information about the specified NAT pool.

Required Privilege Level view

List of Sample Output
show services nat mappings brief on page 136
show services nat mappings detail on page 136
show services nat mappings pool-name on page 136
show services nat mappings summary on page 136

Output Fields Table 6 on page 134 lists the output fields for the **show services nat mappings** command. Output fields are listed in the approximate order in which they appear.

Table 6: show services nat mappings Output Fields

Field Name	Field Description	Level of Output
Interface	Name of a service interface.	All levels
Service set	Name of a service set. Individual empty service sets are not displayed, but if none of the service sets has any flows, a flow table header is printed for each service set.	All levels
NAT pool	Name of the NAT pool.	All levels
Address Mapping	Mapping performed by NAT to conceal the network address.	All levels
No. of Port Mappings	Number of port mappings.	All levels
Port mapping	Port mapping performed by NAT.	**detail**
Flow Count	Number of flows.	**detail**
Flow Count		

Table 6: show services nat mappings Output Fields *(continued)*

Field Name	Field Description	Level of Output
Total number of address mappings:	Total number of address mappings, by service interface.	**summary**
Total number of endpoint independent port mappings:	Total number of port mappings by service interface.	**summary**
Total number of endpoint independent filters:	Total number of independent filters that filter out only packets that are not destined to the internal address and port, regardless of the external IP address and port source, by service interface.	**summary**

Sample Output

show services nat mappings brief

```
user@host> show services nat mappings brief
Interface: sp-2/3/0, Service set: s1

    NAT pool: p1
      Address Mapping: 2.1.20.10 ---> 34.34.34.34
        No. of port mappings: 1
```

show services nat mappings detail

```
user@host> show services nat mappings detail
Interface: sp-2/0/0, Service set: s1
   NAT pool: napt_p1, Translation type: dynamic
     Address range: 5.5.5.1-5.5.5.254
     Port range: 512-65535, Ports in use: 0, Out of port errors: 0, Max ports
used: 0
     AP-P out of port errors: 0
```

show services nat mappings pool-name

```
user@host> show services nat mappings p1
Interface: sp-2/3/0, Service set: s1

    NAT pool: p1
      Address Mapping: 2.1.20.10 ---> 34.34.34.34
        No. of port mappings: 1
```

show services nat mappings summary

```
user@host> show services nat mapping summary
Service Interface:                                      sp-1/0/0
   Total number of address mappings:                       790
   Total number of endpoint independent port mappings:     1580
   Total number of endpoint independent filters:           1580

   Service Interface:                                      sp-1/1/0
   Total number of address mappings:                       914
   Total number of endpoint independent port mappings:     1828
   Total number of endpoint independent filters:           1828

   Service Interface:                                      sp-4/0/0
   Total number of address mappings:                       688
   Total number of endpoint independent port mappings:     1376
   Total number of endpoint independent filters:           1376

   Service Interface:                                      sp-4/1/0
   Total number of address mappings:                       648
   Total number of endpoint independent port mappings:     1296
   Total number of endpoint independent filters:           1296
```

PART 4

Index

Index

Symbols

A

B

C

D

P

R

S

T