Implementing NAT-PT for IPv6

First Published: July 25, 2011
Last Updated: July 25, 2011

Network Address Translation—Protocol Translation (NAT-PT) is an IPv6 to IPv4 translation mechanism, as defined in RFC 2765 and RFC 2766, allowing IPv6-only devices to communicate with IPv4-only devices and vice versa.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for Implementing NAT-PT for IPv6" section on page 24.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for Implementing NAT-PT for IPv6

Before implementing NAT-PT, you must configure IPv4 and IPv6 on the router interfaces that need to communicate between IPv4-only and IPv6-only networks.

Restrictions for Implementing NAT-PT for IPv6

- NAT-PT is not supported in Cisco Express Forwarding.
- NAT-PT provides limited Application Layer Gateway (ALG) support—ALG support for Internet Control Message Protocol (ICMP), File Transfer Protocol (FTP), and Domain Naming System (DNS).
- NAT-PT has the same restrictions that apply to IPv4 NAT where NAT-PT does not provide end-to-end security and the NAT-PT router can be a single point of failure in the network.
- Users must decide whether to use Static NAT-PT operation, Dynamic NAT-PT operation, Port Address Translation (PAT), or IPv4-mapped operation. Deciding which operation to use determines how a user will configure and operate NAT-PT.
- Bridge-Group Virtual interfaces (BVIs) in IPv6 are not supported with NAT-PT and wireless interfaces Dot11Radio.

Information About Implementing NAT-PT for IPv6

This section provides an overview of NAT-PT for Cisco IOS software. Users can configure NAT-PT using one of the following operations—static NAT-PT, dynamic NAT-PT, Port Address Translation (PAT), or IPv4-mapped operation—which are described in the following sections:

- NAT-PT Overview, page 2
- Static NAT-PT Operation, page 3
- Dynamic NAT-PT Operation, page 4
- Port Address Translation or Overload, page 5
- IPv4-Mapped Operation, page 5
- IPv6 Virtual Fragmentation Reassembly, page 5

NAT-PT Overview

NAT-PT for Cisco IOS software was designed using RFC 2766 and RFC 2765 as a migration tool to help customers transition their IPv4 networks to IPv6 networks. Using a protocol translator between IPv6 and IPv4 allows direct communication between hosts speaking a different network protocol. Users can use either static definitions or IPv4-mapped definitions for NAT-PT operation.

Figure 1 shows NAT-PT runs on a router between an IPv6 network and an IPv4 network to connect an IPv6-only node with an IPv4-only node.

Figure 1 NAT-PT Basic Operation

Although IPv6 solves addressing issues for customers, a long transition period is likely before customers move to an exclusive IPv6 network environment. During the transition period, any new IPv6-only networks will need to continue to communicate with existing IPv4 networks. NAT-PT is designed to be deployed to allow direct communication between IPv6-only networks and IPv4-only networks. For a service provider customer, an example could be an IPv6-only client trying to access an IPv4-only web server. Enterprise customers will also migrate to IPv6 in stages, and many of their IPv4-only networks will be operational for several years. Dual stack networks may have some IPv6-only hosts configured to take advantage of the IPv6 autoconfiguration, global addressing, and simpler management, and these hosts can use NAT-PT to communicate with existing IPv4-only networks in the same organization.

One of the benefits of NAT-PT is that no changes are required to existing hosts, because all the NAT-PT configurations are performed at the NAT-PT router. Customers with existing stable IPv4 networks can introduce an IPv6 network and use NAT-PT to allow communication without disrupting the existing network. To further illustrate the seamless transition, File Transfer Protocol (FTP) can be used between IPv4 and IPv6 networks, just as within an IPv4 network. Packet fragmentation is enabled by default when IPv6 is configured, allowing IPv6 and IPv4 networks to resolve fragmentation problems between the networks. Without the ability to resolve fragmentation, connectivity could become intermittent when fragmented packets might be dropped or improperly interpreted.

Cisco has developed other transition techniques including dual stack, IPv6 over MPLS, and tunneling. NAT-PT should not be used when other native communication techniques exist. If a host is configured as a dual stack host with both IPv4 and IPv6, we do not recommend using NAT-PT to communicate between the dual stack host and an IPv6-only or IPv4-only host. NAT-PT is not recommended for a scenario in which an IPv6-only network is trying to communicate to another IPv6-only network via an IPv4 backbone or vice versa, because NAT-PT would require a double translation to be performed. In this scenario, tunneling techniques would be recommended.

The following sections describe the operations that may be used to configure NAT-PT. Users have the option to use one of the following operations for NAT-PT operation, but not all four.

Static NAT-PT Operation

Static NAT-PT uses static translation rules to map one IPv6 address to one IPv4 address. IPv6 network nodes communicate with IPv4 network nodes using an IPv6 mapping of the IPv4 address configured on the NAT-PT router.

Figure 2 shows how the IPv6-only node named A can communicate with the IPv4-only node named C using NAT-PT. The NAT-PT device is configured to map the source IPv6 address for node A of 2001:DB8:bbbb:1::1 to the IPv4 address 192.168.99.2. NAT-PT is also configured to map the source address of IPv4 node C, 192.168.30.1 to 2001:DB8::a. When packets with a source IPv6 address of node A are received at the NAT-PT router, they are translated to have a destination address to match node C in the IPv4-only network. NAT-PT can also be configured to match a source IPv4 address and translate the packet to an IPv6 destination address to allow an IPv4-only host communicate with an IPv6-only host.

If you have multiple IPv6-only or IPv4-only hosts that need to communicate, you may need to configure many static NAT-PT mappings. Static NAT-PT is useful when applications or servers require access to a stable IPv4 address, such as accessing an external IPv4 DNS server.

Figure 2 Static NAT-PT Operation

Dynamic NAT-PT Operation

Dynamic NAT-PT allows multiple NAT-PT mappings by allocating addresses from a pool. NAT-PT is configured with a pool of IPv6 and/or IPv4 addresses. At the start of a NAT-PT session a temporary address is dynamically allocated from the pool. The number of addresses available in the address pool determines the maximum number of concurrent sessions. The NAT-PT device records each mapping between addresses in a dynamic state table.

Figure 3 shows how dynamic NAT-PT operates. The IPv6-only node B can communicate with the IPv4-only node D using dynamic NAT-PT. The NAT-PT device is configured with an IPv6 access list, prefix list, or route map to determine which packets are to be translated by NAT-PT. A pool of IPv4 addresses—10.21.8.1 to 10.21.8.10 in Figure 3— is also configured. When an IPv6 packet to be translated is identified, NAT-PT uses the configured mapping rules and assigns a temporary IPv4 address from the configured pool of IPv4 addresses.

Figure 3 Dynamic NAT-PT Operation

Dynamic NAT-PT translation operation requires at least one static mapping for the IPv4 DNS server.

After the IPv6 to IPv4 connection is established, the reply packets going from IPv4 to IPv6 take advantage of the previously established dynamic mapping to translate back from IPv4 to IPv6. If the connection is initiated by an IPv4-only host, then the explanation is reversed.

Port Address Translation or Overload

Port Address Translation (PAT), also known as Overload, allows a single IPv4 address to be used among multiple sessions by multiplexing on the port number to associate several IPv6 users with a single IPv4 address. PAT can be accomplished through a specific interface or through a pool of addresses. Figure 4 shows multiple IPv6 addresses from the IPv6 network linked to a single IPv4 interface into the IPv4 network.

Figure 4 Port Address Translation

IPv4-Mapped Operation

Customers can also send traffic from their IPv6 network to an IPv4 network without configuring IPv6 destination address mapping. A packet arriving at an interface is checked to discover if it has a NAT-PT prefix that was configured with the **ipv6 nat prefix v4-mapped** command. If the prefix matches, then an access-list check is performed to discover if the source address matches the access list or prefix list. If the prefix does not match, the packet is dropped.

If the prefix matches, source address translation is performed. If a rule has been configured for the source address translation, the last 32 bits of the destination IPv6 address is used as the IPv4 destination and a flow entry is created.

With an IPv4-mapping configuration on the router, when the DNS ALG IPv4 address is converted to an IPv6 address, the IPv6 address is processed and the DNS packets from IPv4 network get their ALGs translated into the IPv6 network.

IPv6 Virtual Fragmentation Reassembly

Fragmentation is a process of breaking down an IP datagram into smaller packets to be transmitted over different types of network media. Non-initial fragments of a fragmented IPv6 packet is used to pass through IPsec and NAT64 without any examination due to the lack of the L4 header, which usually is only available on the initial fragment. The IPv6 Virtual Fragmentation Reassembly (VFR) feature provides the ability to collect the fragments and provide L4 info for all fragments for IPsec and NAT64 features.

How to Implement NAT-PT for IPv6

Configuring Basic IPv6 to IPv4 Connectivity for NAT-PT for IPv6

Perform this task to configure basic IPv6 to IPv4 connectivity for NAT-PT, which consists of configuring the NAT-PT prefix globally, and enable NAT-PT on an interface. For NAT-PT to be operational, NAT-PT must be enabled on both the incoming and outgoing interfaces.

An IPv6 prefix with a prefix length of 96 must be specified for NAT-PT to use. The IPv6 prefix can be a unique local unicast prefix, a subnet of your allocated IPv6 prefix, or even an extra prefix obtained from your Internet service provider (ISP). The NAT-PT prefix is used to match a destination address of an IPv6 packet. If the match is successful, NAT-PT will use the configured address mapping rules to translate the IPv6 packet to an IPv4 packet. The NAT-PT prefix can be configured globally or with different IPv6 prefixes on individual interfaces. Using a different NAT-PT prefix on several interfaces allows the NAT-PT router to support an IPv6 network with multiple exit points to IPv4 networks.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 nat prefix** *ipv6-prefix/prefix-length*
4. **interface** *type number*
5. **ipv6 address** *ipv6-prefix* {*/prefix-length* | **link-local**}
6. **ipv6 nat**
7. **exit**
8. **interface** *type number*
9. **ip address** *ip-address mask* [**secondary**]
10. **ipv6 nat**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 nat prefix** *ipv6-prefix*/*prefix-length* **Example:** `Router# ipv6 nat prefix 2001:DB8::/96`	Assigns an IPv6 prefix as a global NAT-PT prefix. • Matching destination prefixes in IPv6 packets are translated by NAT-PT. • The only prefix length supported is 96.
Step 4	**interface** *type number* **Example:** `Router(config)# interface ethernet 3/1`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 5	**ipv6 address** *ipv6-address* {*/prefix-length* | **link-local**} **Example:** `Router(config-if)# ipv6 address 2001:DB8:yyyy:1::9/64`	Specifies an IPv6 address assigned to the interface and enables IPv6 processing on the interface.
Step 6	**ipv6 nat** **Example:** `Router(config-if)# ipv6 nat`	Enables NAT-PT on the interface.
Step 7	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode, and returns the router to global configuration mode.
Step 8	**interface** *type number* **Example:** `Router(config)# interface ethernet 3/3`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 9	**ip address** *ip-address mask* [**secondary**] **Example:** `Router(config-if)# ip address 192.168.30.9 255.255.255.0`	Specifies an IP address and mask assigned to the interface and enables IP processing on the interface.
Step 10	**ipv6 nat** **Example:** `Router(config-if)# ipv6 nat`	Enables NAT-PT on the interface.

Configuring IPv4-Mapped NAT-PT

Perform this task to enable customers to send traffic from their IPv6 network to an IPv4 network without configuring IPv6 destination address mapping. This task shows the **ipv6 nat prefix v4-mapped** command configured on a specified interface, but the command could alternatively be configured globally:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ipv6 nat prefix** *ipv6-prefix* **v4-mapped** {*access-list-name* | *ipv6-prefix*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface ethernet 3/1`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 4	**ipv6 nat prefix** *ipv6-prefix* **v4-mapped** {*access-list-name* | *ipv6-prefix*} **Example:** `Router(config-if)# ipv6 nat prefix 2001::/96 v4-mapped v4mapacl`	Enables customers to send traffic from their IPv6 network to an IPv4 network without configuring IPv6 destination address mapping.

Configuring Mappings for IPv6 Hosts Accessing IPv4 Hosts

Perform this task to configure static or dynamic IPv6 to IPv4 address mappings. The dynamic address mappings include assigning a pool of IPv4 addresses and using an access list, prefix list, or route map to define which packets are to be translated.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **ipv6 nat v6v4 source** *ipv6-address ipv4-address*
   or
   ipv6 nat v6v4 source {**list** *access-list-name* | **route-map** *map-name*} **pool** *name*
4. **ipv6 nat v6v4 pool** *name start-ipv4 end-ipv4* **prefix-length** *prefix-length*
5. **ipv6 nat translation** [**max-entries** *number*] {**timeout** | **udp-timeout** | **dns-timeout** | **tcp-timeout** | **finrst-timeout** | **icmp-timeout**} {*seconds* | **never**}
6. **ipv6 access-list** *access-list-name*
7. **permit** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* | **any** | **host** *destination-ipv6-address*}
8. **exit**
9. **show ipv6 nat translations** [**icmp** | **tcp** | **udp**] [**verbose**]
10. **show ipv6 nat statistics**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ipv6 nat v6v4 source** *ipv6-address ipv4-address*
	or
	ipv6 nat v6v4 source {**list** *access-list-name* | **route-map** *map-name*} **pool** *name*
	
	Example:
	`Router(config)# ipv6 nat v6v4 source 2001:DB8:yyyy:1::1 10.21.8.10`
	or
	
	Example:
	`Router(config)# ipv6 nat v6v4 source list pt-list1 pool v4pool`
		Enables a static IPv6 to IPv4 address mapping using NAT-PT.
		or
		Enables a dynamic IPv6 to IPv4 address mapping using NAT-PT.
		• Use the **list** or **route-map** keyword to specify a prefix list, access list, or a route map to define which packets are translated.
		• Use the **pool** keyword to specify the name of a pool of addresses, created by the **ipv6 nat v6v4 pool** command, to be used in dynamic NAT-PT address mapping.

Step 4	**ipv6 nat v6v4 pool** *name start-ipv4 end-ipv4* **prefix-length** *prefix-length* **Example:** `Router(config)# ipv6 nat v6v4 pool v4pool 10.21.8.1 10.21.8.10 prefix-length 24`	Specifies a pool of IPv4 addresses to be used by NAT-PT for dynamic address mapping.
Step 5	**ipv6 nat translation** [**max-entries** *number*] {**timeout** | **udp-timeout** | **dns-timeout** | **tcp-timeout** | **finrst-timeout** | **icmp-timeout**} {*seconds* | **never**} **Example:** `Router(config)# ipv6 nat translation udp-timeout 600`	(Optional) Specifies the time after which NAT-PT translations time out.
Step 6	**ipv6 access-list** *access-list-name* **Example:** `Router(config)# ipv6 access-list pt-list1`	(Optional) Defines an IPv6 access list and enters IPv6 access list configuration mode. The router prompt changes to Router(config-ipv6-acl)#. • The *access-list name* argument specifies the name of the IPv6 access control list (ACL). IPv6 ACL names cannot contain a space or quotation mark, or begin with a numeral.

	Command or Action	Purpose
Step 7	**permit** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* | **any** | **host** *destination-ipv6-address*} **Example:** `Router(config-ipv6-acl)# permit ipv6 2001:DB8:bbbb:1::/64 any`	(Optional) Specifies permit conditions for an IPv6 ACL. • The *protocol* argument specifies the name or number of an Internet protocol. It can be one of the keywords **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **sctp**, **tcp**, or **udp**, or an integer in the range from 0 to 255 representing an IPv6 protocol number. • The *source-ipv6-prefix*/*prefix-length* and *destination-ipv6-prefix*/*prefix-length* arguments specify the source and destination IPv6 network or class of networks about which to set permit conditions. These arguments must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. • The **any** keyword is an abbreviation for the IPv6 prefix ::/0. • The **host** *source-ipv6-address* keyword and argument combination specifies the source IPv6 host address about which to set permit conditions. The *source-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits access list configuration mode, and returns the router to global configuration mode. Enter the **exit** command twice to return to privileged EXEC mode.
Step 9	**show ipv6 nat translations** [**icmp** | **tcp** | **udp**] [**verbose**] **Example:** `Router# show ipv6 nat translations verbose`	(Optional) Displays active NAT-PT translations. • Use the optional **icmp**, **tcp**, and **udp** keywords to display detailed information about the NAT-PT translation events for the specified protocol. • Use the optional **verbose** keyword to display more detailed information about the active translations.
Step 10	**show ipv6 nat statistics** **Example:** `Router# show ipv6 nat statistics`	(Optional) Displays NAT-PT statistics.

Configuring Mappings for IPv4 Hosts Accessing IPv6 Hosts

Perform this optional task to configure static or dynamic IPv4 to IPv6 address mappings. The dynamic address mappings include assigning a pool of IPv6 addresses and using an access list, prefix list, or route map to define which packets are to be translated.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **ipv6 nat v4v6 source** *ipv4-address ipv6-address*
   or
   ipv6 nat v4v6 source list {*access-list-number* | *name*} **pool** *name*
4. **ipv6 nat v4v6 pool** *name start-ipv6 end-ipv6* **prefix-length** *prefix-length*
5. **access-list** {*access-list-name* | *number*} {**deny** | **permit**} [*source source-wildcard*] [**log**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ipv6 nat v4v6 source** *ipv6-address ipv4-address*
	or
	ipv6 nat v4v6 source list {*access-list-number* | *name*} **pool** *name*
	
	Example:
	`Router(config)# ipv6 nat v4v6 source 10.21.8.11 2001:DB8:yyyy::2`
	or
	`Router(config)# ipv6 nat v4v6 source list 1 pool v6pool`
		Enables a static IPv4 to IPv6 address mapping using NAT-PT.
		or
		Enables a dynamic IPv4 to IPv6 address mapping using NAT-PT.
		• Use the **list** keyword to specify an access list to define which packets are translated.
		• Use the **pool** keyword to specify the name of a pool of addresses, created by the **ipv6 nat v4v6 pool** command, to be used in dynamic NAT-PT address mapping.

Step 4	**ipv6 nat v4v6 pool** *name start-ipv6 end-ipv6* **prefix-length** *prefix-length* **Example:** `Router(config)# ipv6 nat v4v6 pool v6pool 2001:DB8:yyyy::1 2001:DB8:yyyy::2 prefix-length 128`	Specifies a pool of IPv6 addresses to be used by NAT-PT for dynamic address mapping.
Step 5	**access-list** {*access-list-name* | *number*} {**deny** | **permit**} [*source source-wildcard*] [**log**] **Example:** `Router(config)# access-list 1 permit 192.168.30.0 0.0.0.255`	Specifies an entry in a standard IPv4 access list.

Configuring PAT for IPv6 to IPv4 Address Mappings

Perform this task to configure PAT for IPv6 to IPv4 address mappings. Multiple IPv6 addresses are mapped to a single IPv4 address or to a pool of IPv4 addresses and using an access list, prefix list, or route map to define which packets are to be translated.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 nat v6v4 source** {**list** *access-list-name* | **route-map** *map-name*} **pool** *name overload*

   or

   ipv6 nat v6v4 source {**list** *access-list-name* | **route-map** *map-name*} **interface** *interface name* **overload**
4. **ipv6 nat v6v4 pool** *name start-ipv4 end-ipv4* **prefix-length** *prefix-length*
5. **ipv6 nat translation** [**max-entries** *number*] {**timeout** | **udp-timeout** | **dns-timeout** | **tcp-timeout** | **finrst-timeout** | **icmp-timeout**} {*seconds* | **never**}
6. **ipv6 access-list** *access-list-name*
7. **permit** *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	**ipv6 nat v6v4 source** {**list** *access-list-name* | **route-map** *map-name*} **pool** *name overload*
	or
	ipv6 nat v6v4 source {**list** *access-list-name* | **route-map** *map-name*} **interface** *interface name* **overload**
	
	Example:
	`Router(config)# ipv6 nat v6v4 source 2001:DB8:yyyy:1::1 10.21.8.10`
	or
	
	Example:
	`Router(config)# ipv6 nat v6v4 source list pt-list1 pool v4pool overload`
		Enables a dynamic IPv6 to IPv4 address overload mapping using a pool address.
		or
		Enables a dynamic IPv6 to IPv4 address overload mapping using an interface address.
		• Use the **list** or **route-map** keyword to specify a prefix list, access list, or a route map to define which packets are translated.
		• Use the **pool** keyword to specify the name of a pool of addresses, created by the **ipv6 nat v6v4 pool** command, to be used in dynamic NAT-PT address mapping.
		• Use the interface keyword to specify the interface address to be used for overload.

	Command or Action	Purpose
Step 4	**ipv6 nat v6v4 pool** *name start-ipv4 end-ipv4* **prefix-length** *prefix-length* **Example:** `Router(config)# ipv6 nat v6v4 pool v4pool 10.21.8.1 10.21.8.10 prefix-length 24`	Specifies a pool of IPv4 addresses to be used by NAT-PT for dynamic address mapping.
Step 5	**ipv6 nat translation** [**max-entries** *number*] {**timeout** | **udp-timeout** | **dns-timeout** | **tcp-timeout** | **finrst-timeout** | **icmp-timeout**} {*seconds* | **never**} **Example:** `Router(config)# ipv6 nat translation udp-timeout 600`	(Optional) Specifies the time after which NAT-PT translations time out.
Step 6	**ipv6 access-list** *access-list-name* **Example:** `Router(config)# ipv6 access-list pt-list1`	(Optional) Defines an IPv6 access list and enters IPv6 access list configuration mode. The router prompt changes to Router(config-ipv6-acl)#. • The *access-list name* argument specifies the name of the IPv6 access control list (ACL). IPv6 ACL names cannot contain a space or quotation mark, or begin with a numeral.
Step 7	**permit** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix*/*prefix-length* | **any** | **host** *destination-ipv6-address*} **Example:** `Router(config-ipv6-acl)# permit ipv6 2001:DB8:bbbb:1::/64 any`	(Optional) Specifies permit conditions for an IPv6 ACL. • The *protocol* argument specifies the name or number of an Internet protocol. It can be one of the keywords **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **sctp**, **tcp**, or **udp**, or an integer in the range from 0 to 255 representing an IPv6 protocol number. • The *source-ipv6-prefix*/*prefix-length* and *destination-ipv6-prefix*/*prefix-length* arguments specify the source and destination IPv6 network or class of networks about which to set permit conditions. These arguments must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. • The **any** keyword is an abbreviation for the IPv6 prefix ::/0. • The **host** *source-ipv6-address* keyword and argument combination specifies the source IPv6 host address about which to set permit conditions. The *source-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.

Configuring IPv6 Virtual Fragmentation Reassembly

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ipv6 virtual-reassembly** [**in** | **out**] [**max-reassemblies** *maxreassemblies*] [**max-fragments** *max-fragments*] [**timeout** *seconds*] [**drop-fragments**]
5. **exit**
6. **show ipv6 virtual-reassembly interface** *interface-type*
7. **show ipv6 virtual-reassembly features interface** *interface-type*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface gigabitethernet 3/1/1`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 4	**ipv6 virtual-reassembly** [**in** | **out**] [**max-reassemblies** *maxreassemblies*] [**max-fragments** *max-fragments*] [**timeout** *seconds*] [**drop-fragments**] **Example:** `Router(config-if)# ipv6 virtual-reassembly max-reassemblies 32 max-fragments 4 timeout 7`	Enables VFR on an interface.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode and places the router in global configuration mode. • Enter this command twice to reach privileged EXEC mode.
Step 6	**show ipv6 virtual-reassembly interface** *interface-type* **Example:** `Router# show ipv6 virtual-reassembly interface e1/1/1`	Displays VRF configuration and statistical information on a specific interface.
Step 7	**show ipv6 virtual-reassembly features interface** *interface-type* **Example:** `Router# show ipv6 virtual-reassembly features`	Displays VFR information on all interfaces or on a specified interface.

Verifying NAT-PT Configuration and Operation

SUMMARY STEPS

1. **clear ipv6 nat translation ***
2. **enable**
3. **debug ipv6 nat** [**detailed** | **port**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**clear ipv6 nat translation *** **Example:** `Router> clear ipv6 nat translation *`	(Optional) Clears dynamic NAT-PT translations from the dynamic translation state table. • Use the * keyword to clear all dynamic NAT-PT translations. **Note** Static translation configuration is not affected by this command.
Step 2	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 3	**debug ipv6 nat** [**detailed** | **port**] **Example:** `Router# debug ipv6 nat detail`	Displays debugging messages for NAT-PT translation events.

Examples

- Sample Output from the show ipv6 nat translations Command
- Sample Output from the show ipv6 nat statistics Command
- Sample Output from the clear ipv6 nat translation Command
- Sample Output from the debug ipv6 nat Command

Sample Output from the show ipv6 nat translations Command

In the following example, output information about active NAT-PT translations is displayed using the **show ipv6 nat translations** command:

```
Router# show ipv6 nat translations

Prot  IPv4 source              IPv6 source
      IPv4 destination         IPv6 destination
---   ---                      ---
      192.168.123.2            2001:DB8::2

---   ---                      ---
      192.168.122.10           2001:DB8::10

tcp   192.168.124.8,11047      2001:DB8:3::8,11047
      192.168.123.2,23         2001:DB8::2,23

udp   192.168.124.8,52922      2001:DB8:3::8,52922
      192.168.123.2,69         2001::2,69

udp   192.168.124.8,52922      2001:DB8:3::8,52922
      192.168.123.2,52922      2001:DB8::2,52922

---   192.168.124.8            2001:DB8:3::8
      192.168.123.2            2001:DB8::2

---   192.168.124.8            2001:DB8:3::8
      ---                      ---
```

```
---   192.168.121.4           2001:DB8:5::4
      ---                     ---
```

In the following example, detailed output information about active NAT-PT translations is displayed using the **show ipv6 nat translations** command with the **verbose** keyword:

```
Router# show ipv6 nat translations verbose

Prot  IPv4 source             IPv6 source
      IPv4 destination        IPv6 destination
---   ---                     ---
      192.168.123.2           2001:DB8::2
      create 00:04:24, use 00:03:24,

---   ---                     ---
      192.168.122.10          2001:DB8::10
      create 00:04:24, use 00:04:24,

tcp   192.168.124.8,11047     2001:DB8:3::8,11047
      192.168.123.2,23        2001:DB8::2,23
      create 00:03:24, use 00:03:20, left 00:16:39,

udp   192.168.124.8,52922     2001:DB8:3::8,52922
      192.168.123.2,69        2001:DB8::2,69
      create 00:02:51, use 00:02:37, left 00:17:22,

udp   192.168.124.8,52922     2001:DB8:3::8,52922
      192.168.123.2,52922     2001:DB8::2,52922
      create 00:02:48, use 00:02:30, left 00:17:29,

---   192.168.124.8           2001:DB8:3::8
      192.168.123.2           2001:DB8::2
      create 00:03:24, use 00:02:34, left 00:17:25,

---   192.168.124.8           2001:DB8:3::8
      ---                     ---
      create 00:04:24, use 00:03:24,

---   192.168.121.4           2001:DB8:5::4
      ---                     ---
      create 00:04:25, use 00:04:25,
```

Sample Output from the show ipv6 nat statistics Command

In the following example, output information about NAT-PT statistics is displayed using the **show ipv6 nat statistics** command:

```
Router# show ipv6 nat statistics

Total active translations: 4 (4 static, 0 dynamic; 0 extended)
NAT-PT interfaces:
  Ethernet3/1, Ethernet3/3
Hits: 0  Misses: 0
Expired translations: 0
```

Sample Output from the clear ipv6 nat translation Command

In the following example, all dynamic NAT-PT translations are cleared from the dynamic translation state table using the **clear ipv6 nat translation** command with the * keyword. When the output information about active NAT-PT translations is then displayed using the **show ipv6 nat translations** command, only the static translation configurations remain. Compare this **show** command output with the output from the **show ipv6 nat translations** command in Step 1.

```
Router# clear ipv6 nat translation *

Router# show ipv6 nat translations

Prot  IPv4 source              IPv6 source
      IPv4 destination         IPv6 destination
---   ---                      ---
      192.168.123.2            2001:DB8::2

---   ---                      ---
      192.168.122.10           2001:DB8::10

---   192.168.124.8            2001:DB8:3::8
      ---                      ---

---   192.168.121.4            2001:DB8:5::4
      ---                      ---
```

Sample Output from the debug ipv6 nat Command

In the following example, debugging messages for NAT-PT translation events are displayed using the **debug ipv6 nat** command:

```
Router# debug ipv6 nat

00:06:06: IPv6 NAT: icmp src (2001:DB8:3002::8) -> (192.168.124.8), dst (2001:DB8:2001::2)
-> (192.168.123.2)
00:06:06: IPv6 NAT: icmp src (192.168.123.2) -> (2001:DB8:2001::2), dst (192.168.124.8) ->
(2001:DB8:3002::8)
00:06:06: IPv6 NAT: icmp src (2001:DB8:3002::8) -> (192.168.124.8), dst (2001:DB8:2001::2)
-> (192.168.123.2)
00:06:06: IPv6 NAT: icmp src (192.168.123.2) -> (2001:DB8:2001::2), dst (192.168.124.8) ->
(2001:DB8:3002::8)
00:06:06: IPv6 NAT: tcp src (2001:DB8:3002::8) -> (192.168.124.8), dst (2001:DB8:2001::2)
-> (192.168.123.2)
00:06:06: IPv6 NAT: tcp src (192.168.123.2) -> (2001:DB8:2001::2), dst (192.168.124.8) ->
(2001:DB8:3002::8)
00:06:06: IPv6 NAT: tcp src (2001:DB8:3002::8) -> (192.168.124.8), dst (2001:DB8:2001::2)
-> (192.168.123.2)
00:06:06: IPv6 NAT: tcp src (2001:DB8:3002::8) -> (192.168.124.8), dst (2001:DB8:2001::2)
-> (192.168.123.2)
00:06:06: IPv6 NAT: tcp src (2001:DB8:3002::8) -> (192.168.124.8), dst (2001:DB8:2001::2)
-> (192.168.123.2)
00:06:06: IPv6 NAT: tcp src (192.168.123.2) -> (2001:DB8:2001::2), dst (192.168.124.8) ->
(2001:DB8:3002::8)
```

Configuration Examples for NAT-PT for IPv6

Example: Static NAT-PT Configuration

The following example configures the NAT-PT prefix globally, enables NAT-PT on two interfaces, and configures two static NAT-PT mappings. Ethernet interface 3/1 is configured as IPv6 only, and Ethernet interface 3/3 is configured as IPv4 only.

```
interface Ethernet3/1
 ipv6 address 2001:DB8:3002::9/64
 ipv6 enable
 ipv6 nat
!
interface Ethernet3/3
 ip address 192.168.30.9 255.255.255.0
 ipv6 nat
!
ipv6 nat v4v6 source 192.168.30.1 2001:DB8:0::2
ipv6 nat v6v4 source 2001:DB8:bbbb:1::1 10.21.8.10
ipv6 nat prefix 2001:DB8:0::/96
```

Example: Enabling Traffic to be Sent from an IPv6 Network to an IPv4 Network without Using IPv6 Destination Address Mapping

In the following example, the access list permits any IPv6 source address with the prefix 2001::/96 to go to the destination with a 2000::/96 prefix. The destination is then translated to the last 32 bit of its IPv6 address; for example: source address = 2001::1, destination address = 2000::192.168.1.1. The destination then becomes 192.168.1.1 in the IPv4 network:

```
ipv6 nat prefix 2000::/96 v4-mapped v4map_acl

ipv6 access-list v4map_acl
 permit ipv6 2001::/96 2000::/96
```

Example: Dynamic NAT-PT Configuration for IPv6 Hosts Accessing IPv4 Hosts

The following example configures the NAT-PT prefix globally, enables NAT-PT on two interfaces, and configures one static NAT-PT mapping (used, for example, to access a DNS server). A dynamic NAT-PT mapping is also configured to map IPv6 addresses to IPv4 addresses using a pool of IPv4 addresses named v4pool. The packets to be translated by NAT-PT are filtered using an IPv6 access list named pt-list1. The User Datagram Protocol (UDP) translation entries are configured to time out after 10 minutes. Ethernet interface 3/1 is configured as IPv6 only, and Ethernet interface 3/3 is configured as IPv4 only.

```
interface Ethernet3/1
 ipv6 address 2001:DB8:bbbb:1::9/64
 ipv6 enable
 ipv6 nat
!
interface Ethernet3/3
 ip address 192.168.30.9 255.255.255.0
 ipv6 nat
!
ipv6 nat v4v6 source 192.168.30.1 2001:DB8:0::2
ipv6 nat v6v4 source list pt-list1 pool v4pool
ipv6 nat v6v4 pool v4pool 10.21.8.1 10.21.8.10 prefix-length 24
ipv6 nat translation udp-timeout 600
ipv6 nat prefix 2001:DB8:1::/96
```

```
!
ipv6 access-list pt-list1
 permit ipv6 2001:DB8:bbbb:1::/64 any
```

Example: Dynamic NAT-PT Configuration for IPv4 Hosts Accessing IPv6 Hosts

The following example configures the NAT-PT prefix globally, enables NAT-PT on two interfaces, and configures one static NAT-PT mapping (used, for example, to access a DNS server). A dynamic NAT-PT mapping is also configured to map IPv4 addresses to IPv6 addresses using a pool of IPv6 addresses named v6pool. The packets to be translated by NAT-PT are filtered using an access list named pt-list2. Ethernet interface 3/1 is configured as IPv6 only, and Ethernet interface 3/3 is configured as IPv4 only.

```
interface Ethernet3/1
 ipv6 address 2001:DB8:bbbb:1::9/64
 ipv6 enable
 ipv6 nat
!
interface Ethernet3/3
 ip address 192.168.30.9 255.255.255.0
 ipv6 nat
!
ipv6 nat v4v6 source list 72 pool v6pool
ipv6 nat v4v6 pool v6pool 2001:DB8:0::1 2001:DB8:0::2 prefix-length 128
ipv6 nat v6v4 source 2001:DB8:bbbb:1::1 10.21.8.0
ipv6 nat prefix 2001:DB8:0::/96
!
access-list 72 permit 192.168.30.0 0.0.0.255
```

Example: Configuring IPv6 Virtual Fragmentation Reassembly

<<Example OK?>>

```
Router# show ipv6 virtual-reassembly interface gigabitethernet1/1/1

GigabitEthernet1/1/1:
IPv6 Virtual Fragment Reassembly (VFR) is ENABLED(in)
Concurrent reassemblies (max-reassemblies): 64
Fragments per reassembly (max-fragments): 16
Reassembly timeout (timeout): 3 seconds
Drop fragments: OFF

Current reassembly count: 0
Current fragment count: 0
Total reassembly count: 6950
Total reassembly timeout count: 9

GigabitEthernet1/1/1:
IPv6 Virtual Fragment Reassembly (VFR) is ENABLED(out)
Concurrent reassemblies (max-reassemblies): 64
Fragments per reassembly (max-fragments): 16
Reassembly timeout (timeout): 3 seconds
Drop fragments: OFF

Current reassembly count: 0
Current fragment count: 0
Total reassembly count: 0
Total reassembly timeout count: 0
```

Additional References

Related Documents

Related Topic	Document Title
IPv6 supported features	"Start Here: Cisco IOS XE Software Release Specifics for IPv6 Features," *Cisco IOS XE IPv6 Configuration Guide*
Basic IPv6 configuration tasks	"Implementing IPv6 Addressing and Basic Connectivity," *Cisco IOS XE IPv6 Configuration Guide*
IPv6 commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*Cisco IOS IPv6 Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2765	*Stateless IP/ICMP Translation Algorithm (SIIT)*
RFC 2766	*Network Address Translation - Protocol Translation (NAT-PT)*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Implementing NAT-PT for IPv6

Table 1 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 1 Feature Information for Implementing NAT-PT for IPv6

Feature Name	Releases	Feature Information
NAT Protocol Translation	Cisco IOS XE Release 3.4S	NAT-PT is an IPv6-IPv4 translation mechanism that allows IPv6-only devices to communicate with IPv4-only devices and vice versa. NAT-PT is not supported in Cisco Express Forwarding.
NAT-PT—Support for DNS ALG	Cisco IOS XE Release 3.4S	IPv6 provides DNS ALG support. The following section provides information about this feature: • Restrictions for Implementing NAT-PT for IPv6, page 2
NAT-PT—Support for FTP ALG	Cisco IOS XE Release 3.4S	IPv6 provides FTP ALG support. The following section provides information about this feature: • Restrictions for Implementing NAT-PT for IPv6, page 2
NAT-PT—Support for Fragmentation	Cisco IOS XE Release 3.4S	Packet fragmentation is enabled by default when IPv6 is configured, allowing IPv6 and IPv4 networks to resolve fragmentation problems between the networks. The following section provides information about this feature: • NAT-PT Overview, page 2
IPv6 Virtual Fragmentation Reassembly	Cisco IOS XE Release 3.4S	The IPv6 VFR feature provides the ability to collect the fragments and provide L4 info for all fragments for IPsec and NAT64 features. The following section provides information about this feature: • IPv6 Virtual Fragmentation Reassembly, page 5 • Configuring IPv6 Virtual Fragmentation Reassembly, page 15 • Example: Configuring IPv6 Virtual Fragmentation Reassembly, page 21