Twice NAT

Twice NAT lets you identify both the source and destination address in a single rule. This chapter shows you how to configure twice NAT and includes the following sections:

- Information About Twice NAT, page 6-1
- Licensing Requirements for Twice NAT, page 6-2
- Prerequisites for Twice NAT, page 6-2
- Guidelines and Limitations, page 6-2
- Default Settings, page 6-4
- Configuring Twice NAT, page 6-4
- Monitoring Twice NAT, page 6-24
- Configuration Examples for Twice NAT, page 6-25
- Feature History for Twice NAT, page 6-29

Note
For detailed information about how NAT works, see Chapter 4, “Information About NAT.”

Information About Twice NAT

Twice NAT lets you identify both the source and destination address in a single rule. Specifying both the source and destination addresses lets you specify that a source address should be translated to A when going to destination X, but be translated to B when going to destination Y, for example.

Note
For static NAT, the rule is bidirectional, so be aware that “source” and “destination” are used in commands and descriptions throughout this guide even though a given connection might originate at the “destination” address. For example, if you configure static NAT with port address translation, and specify the source address as a Telnet server, and you want all traffic going to that Telnet server to have the port translated from 2323 to 23, then in the command, you must specify the source ports to be translated (real: 23, mapped: 2323). You specify the source ports because you specified the Telnet server address as the source address.

The destination address is optional. If you specify the destination address, you can either map it to itself (identity NAT), or you can map it to a different address. The destination mapping is always a static mapping.
Licensing Requirements for Twice NAT

<table>
<thead>
<tr>
<th>Model</th>
<th>License Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASA v</td>
<td>Standard or Premium License.</td>
</tr>
<tr>
<td>All other models</td>
<td>Base License.</td>
</tr>
</tbody>
</table>

Prerequisites for Twice NAT

- For both the real and mapped addresses, configure network objects or network object groups (the `object network` or `object-group network` command). Network object groups are particularly useful for creating a mapped address pool with discontinuous IP address ranges or multiple hosts or subnets. To create a network object or group, see the general operations configuration guide.
- For static NAT-with-port-translation, configure TCP or UDP service objects (the `object service` command). To create a service object, see the general operations configuration guide.

For specific guidelines for objects and groups, see the configuration section for the NAT type you want to configure. See also the Guidelines and Limitations, page 6-2 section.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

**Context Mode Guidelines**

Supported in single and multiple context mode.

**Firewall Mode Guidelines**

- Supported in routed and transparent firewall mode.
- In transparent mode, you must specify the real and mapped interfaces; you cannot use any.
- In transparent mode, you cannot configure interface PAT, because the transparent mode interfaces do not have IP addresses. You also cannot use the management IP address as a mapped address.
- In transparent mode, translating between IPv4 and IPv6 networks is not supported. Translating between two IPv6 networks, or between two IPv4 networks is supported.
### IPv6 Guidelines

- Supports IPv6.
- For routed mode, you can also translate between IPv4 and IPv6.
- For transparent mode, translating between IPv4 and IPv6 networks is not supported. Translating between two IPv6 networks, or between two IPv4 networks is supported.
- For transparent mode, a PAT pool is not supported for IPv6.
- For static NAT, you can specify an IPv6 subnet up to /64. Larger subnets are not supported.
- When using FTP with NAT46, when an IPv4 FTP client connects to an IPv6 FTP server, the client must use either the extended passive mode (EPSV) or extended port mode (EPRT); PASV and PORT commands are not supported with IPv6.

### Additional Guidelines

- You cannot configure FTP destination port translation when the source IP address is a subnet (or any other application that uses a secondary connection); the FTP data channel establishment does not succeed. For example, the following configuration does not work:

  object network MyInsNet
  subnet 10.1.2.0 255.255.255.0

  object network MapInsNet
  subnet 209.165.202.128 255.255.255.224

  object network Server1
  host 209.165.200.225

  object network Server1_mapped
  host 10.1.2.67

  object service REAL_ftp
  service tcp destination eq ftp

  object service MAPPED_ftp
  service tcp destination eq 2021

  object network MyOutNet
  subnet 209.165.201.0 255.255.255.224

  nat (inside,outside) source static MyInsNet MapInsNet destination static Server1_mapped Server1 service MAPPED_ftp REAL_ftp

- If you change the NAT configuration, and you do not want to wait for existing translations to time out before the new NAT information is used, you can clear the translation table using the `clear xlate` command. However, clearing the translation table disconnects all current connections that use translations.

  **Note**

  If you remove a dynamic NAT or PAT rule, and then add a new rule with mapped addresses that overlap the addresses in the removed rule, then the new rule will not be used until all connections associated with the removed rule time out or are cleared using the `clear xlate` command. This safeguard ensures that the same address is not assigned to multiple hosts.

- You cannot use an object group with both IPv4 and IPv6 addresses; the object group must include only one type of address.
- When using the `any` keyword in a NAT rule, the definition of “any” traffic (IPv4 vs. IPv6) depends on the rule. Before the ASA performs NAT on a packet, the packet must be IPv6-to-IPv6 or IPv4-to-IPv4; with this prerequisite, the ASA can determine the value of `any` in a NAT rule. For example, if you configure a rule from “any” to an IPv6 server, and that server was mapped from an
IPv4 address, then **any** means “any IPv6 traffic.” If you configure a rule from “any” to “any,” and you map the source to the interface IPv4 address, then **any** means “any IPv4 traffic” because the mapped interface address implies that the destination is also IPv4.

- Objects and object groups used in NAT cannot be undefined; they must include IP addresses.
- You can use the same objects in multiple rules.
- The mapped IP address pool cannot include:
  - The mapped interface IP address. If you specify **any** interface for the rule, then all interface IP addresses are disallowed. For interface PAT (routed mode only), use the `interface` keyword instead of the IP address.
  - (Transparent mode) The management IP address.
  - (Dynamic NAT) The standby interface IP address when VPN is enabled.
  - Existing VPN pool addresses.

**Default Settings**

- By default, the rule is added to the end of section 1 of the NAT table.
- (Routed mode) The default real and mapped interface is Any, which applies the rule to all interfaces.
- If you specify an optional interface, then the ASA uses the NAT configuration to determine the egress interface, but you have the option to always use a route lookup instead.

**Configuring Twice NAT**

This section describes how to configure twice NAT. This section includes the following topics:

- Adding Network Objects for Real and Mapped Addresses, page 6-4
- (Optional) Adding Service Objects for Real and Mapped Ports, page 6-6
- Configuring Dynamic NAT, page 6-7
- Configuring Dynamic PAT (Hide), page 6-11
- Configuring Static NAT or Static NAT-with-Port-Translation, page 6-18
- Configuring Identity NAT, page 6-21
- Configuring Per-Session PAT Rules, page 6-24

**Adding Network Objects for Real and Mapped Addresses**

For each NAT rule, configure up to four network objects or groups for:

- **Source real address**
- **Source mapped address**
- **Destination real address**
- **Destination mapped address**
Objects are required unless you specify the any keyword inline to represent all traffic, or for some types of NAT, the interface keyword to represent the interface address. For more information about configuring a network object or group, see the general operations configuration guide.

**Guidelines**

- A network object group can contain objects and/or inline addresses of either IPv4 or IPv6 addresses. The group cannot contain both IPv4 and IPv6 addresses; it must contain one type only.
- See Guidelines and Limitations, page 6-2 for information about disallowed mapped IP addresses.
- Source Dynamic NAT:
  - You typically configure a larger group of real addresses to be mapped to a smaller group.
  - The mapped object or group cannot contain a subnet; the object must define a range; the group can include hosts and ranges.
  - If a mapped network object contains both ranges and host IP addresses, then the ranges are used for dynamic NAT, and the host IP addresses are used as a PAT fallback.
- Source Dynamic PAT (Hide):
  - The mapped object or group cannot contain a subnet; a network object must define a host, or for a PAT pool, a range; a network object group (for a PAT pool) can include hosts and ranges.
- Source Static NAT or Static NAT with port translation:
  - The mapped object or group can contain a host, range, or subnet.
  - The static mapping is typically one-to-one, so the real addresses have the same quantity as the mapped addresses. You can, however, have different quantities if desired. For more information, see Static NAT, page 4-3.
- Source Identity NAT
  - The real and mapped objects must match; you can use the same object for both, or you can create separate objects that contain the same IP addresses.
- Destination Static NAT or Static NAT with port translation (the destination translation is always static):
  - Although the main feature of twice NAT is the inclusion of the destination IP address, the destination address is optional. If you do specify the destination address, you can configure static translation for that address or just use identity NAT for it. You might want to configure twice NAT without a destination address to take advantage of some of the other qualities of twice NAT, including the use of network object groups for real addresses, or manually ordering of rules. For more information, see Main Differences Between Network Object NAT and Twice NAT, page 4-13.
  - For identity NAT, the real and mapped objects must match; you can use the same object for both, or you can create separate objects that contain the same IP addresses.
  - The static mapping is typically one-to-one, so the real addresses have the same quantity as the mapped addresses. You can, however, have different quantities if desired. For more information, see Static NAT, page 4-3.
  - For static interface NAT with port translation (routed mode only), you can specify the interface keyword instead of a network object/group for the mapped address. For more information, see Static Interface NAT with Port Translation, page 4-5.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **object network** **obj_name**  
(host **ip_address** | subnet  
**(subnet_address netmask | range  
**ip_address_1 ip_address_2)**) | Adds a network object, either IPv4 or IPv6. |

**Example:**
```plaintext
hostname(config)# object network MyInsNet  
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0
```

| **object-group network** **grp_name**  
(network-object **{object net_obj_name | subnet_address netmask | host ip_address | group-object grp_obj_name}** | Adds a network object group, either IPv4 or IPv6. |

**Example:**
```plaintext
hostname(config)# object network TEST  
hostname(config-network-object)# range 10.1.1.1 10.1.1.70
```

```plaintext
hostname(config)# object network TEST2  
hostname(config-network-object)# range 10.1.2.1 10.1.2.70
```

```plaintext
hostname(config-network-object)#  
object-group network MAPPED_IPS  
hostname(config-network)# network-object  
object TEST  
hostname(config-network)# network-object  
object TEST2  
hostname(config-network)# network-object  
host 10.1.2.79
```

## (Optional) Adding Service Objects for Real and Mapped Ports

Configure service objects for:

- **Source real port (Static only) or Destination real port**
- **Source mapped port (Static only) or Destination mapped port**

For more information about configuring a service object, see the general operations configuration guide.

## Guidelines

- NAT only supports TCP or UDP. When translating a port, be sure the protocols in the real and mapped service objects are identical (both TCP or both UDP).
- The “not equal” (**neq**) operator is not supported.
- For identity port translation, you can use the same service object for both the real and mapped ports.
- Source Dynamic NAT—Source Dynamic NAT does not support port translation.
- Source Dynamic PAT (Hide)—Source Dynamic PAT does not support port translation.
- Source Static NAT or Static NAT with port translation—A service object can contain both a source and destination port; however, you should specify either the source or the destination port for both service objects. You should only specify both the source and destination ports if your application uses a fixed source port (such as some DNS servers); but fixed source ports are rare. For example, if you want to translate the port for the source host, then configure the source service.
- Source Identity NAT—A service object can contain both a source and destination port; however, you should specify either the source or the destination port for both service objects. You should only specify both the source and destination ports if your application uses a fixed source port (such as some DNS servers); but fixed source ports are rare. For example, if you want to translate the port for the source host, then configure the source service.
- Destination Static NAT or Static NAT with port translation (the destination translation is always static)—For non-static source NAT, you can only perform port translation on the destination. A service object can contain both a source and destination port, but only the destination port is used in this case. If you specify the source port, it will be ignored.

Detailed Steps

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>`object service obj_name service {tcp</td>
<td>udp} [source operator port] [destination operator port]`</td>
</tr>
</tbody>
</table>

Example:
```
hostname(config)# object service REAL_SRC_SVC
hostname(config-service-object)# service tcp source eq 80

hostname(config)# object service MAPPED_SRC_SVC
hostname(config-service-object)# service tcp source eq 8080
```

Configuring Dynamic NAT

This section describes how to configure twice NAT for dynamic NAT. For more information, see Dynamic NAT, page 4-7.
## Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Create network objects or groups for the:</td>
</tr>
<tr>
<td>• Source real addresses</td>
<td>See Adding Network Objects for Real and Mapped Addresses, page 6-4.</td>
</tr>
<tr>
<td>• Source mapped addresses</td>
<td>If you want to translate all source traffic, you can skip adding an object for the source real addresses, and instead specify the <strong>any</strong> keyword in the <strong>nat</strong> command.</td>
</tr>
<tr>
<td>• Destination real addresses</td>
<td>If you want to configure destination static interface NAT with port translation only, you can skip adding an object for the destination mapped addresses, and instead specify the <strong>interface</strong> keyword in the <strong>nat</strong> command.</td>
</tr>
<tr>
<td>• Destination mapped addresses</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>(Optional) Create service objects for the:</td>
</tr>
<tr>
<td>• Destination real ports</td>
<td>See (Optional) Adding Service Objects for Real and Mapped Ports, page 6-6.</td>
</tr>
<tr>
<td>• Destination mapped ports</td>
<td></td>
</tr>
</tbody>
</table>
Command

<table>
<thead>
<tr>
<th>Step 3</th>
<th>目的</th>
</tr>
</thead>
<tbody>
<tr>
<td>nat [(real_ifc,mapped_ifc)] [line</td>
<td>(after-auto [line])]</td>
</tr>
<tr>
<td>source dynamic (real_obj</td>
<td>any) {mapped_obj [interface [ipv6]]} [destination static {mapped_obj</td>
</tr>
<tr>
<td>[service mapped_dest_svc_obj real_dest_svc_obj] [dns] [unidirectional]</td>
<td>• Section and Line—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table (see NAT Rule Order, page 4-18). If you want to add the rule into section 3 instead (after the network object NAT rules), then use the <strong>after-auto</strong> keyword. You can insert a rule anywhere in the applicable section using the <strong>line</strong> argument.</td>
</tr>
<tr>
<td>[inactive] [description desc]</td>
<td>• Source addresses:</td>
</tr>
</tbody>
</table>

**Example:**

```
hostname(config)# nat (inside, outside)
source dynamic MyInsNet NAT_POOL
destination static Server1_mapped Server1
service MAPPED_SVC REAL_SVC
```

- Real—Specify a network object, group, or the **any** keyword.
- Mapped—Specify a different network object or group. You can optionally configure the following fallback method:

  Interface PAT fallback—(Routed mode only) The **interface** keyword enables interface PAT fallback. If you specify **ipv6**, then the IPv6 address of the interface is used. After the mapped IP addresses are used up, then the IP address of the mapped interface is used. For this option, you must configure a specific interface for the **mapped_ifc**.
### Configuring Twice NAT

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Continued)</td>
<td></td>
</tr>
<tr>
<td>• Destination addresses (Optional):</td>
<td></td>
</tr>
<tr>
<td>– Mapped—Specify a network object or group, or for static interface NAT with port translation only, specify the <strong>interface</strong> keyword. If you specify <strong>ipv6</strong>, then the IPv6 address of the interface is used. If you specify <strong>interface</strong>, be sure to also configure the <strong>service</strong> keyword. For this option, you must configure a specific interface for the <strong>real_ifc</strong>. See Static Interface NAT with Port Translation, page 4-5 for more information.</td>
<td></td>
</tr>
<tr>
<td>– Real—Specify a network object or group. For identity NAT, simply use the same object or group for both the real and mapped addresses.</td>
<td></td>
</tr>
<tr>
<td>• Destination port—(Optional) Specify the <strong>service</strong> keyword along with the mapped and real service objects. For identity port translation, simply use the same service object for both the real and mapped ports.</td>
<td></td>
</tr>
<tr>
<td>• DNS—(Optional; for a source-only rule) The <strong>dns</strong> keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). You cannot configure the <strong>dns</strong> keyword if you configure a <strong>destination</strong> address. See DNS and NAT, page 4-28 for more information.</td>
<td></td>
</tr>
<tr>
<td>• Unidirectional—(Optional) Specify <strong>unidirectional</strong> so the destination addresses cannot initiate traffic to the source addresses.</td>
<td></td>
</tr>
<tr>
<td>• Inactive—(Optional) To make this rule inactive without having to remove the command, use the <strong>inactive</strong> keyword. To reactivate it, reenter the whole command without the <strong>inactive</strong> keyword.</td>
<td></td>
</tr>
<tr>
<td>• Description—Optional) Provide a description up to 200 characters using the <strong>description</strong> keyword.</td>
<td></td>
</tr>
</tbody>
</table>
Examples

The following example configures dynamic NAT for inside network 10.1.1.0/24 when accessing servers on the 209.165.201.1/27 network as well as servers on the 203.0.113.0/24 network:

```
hostname(config)# object network INSIDE_NW
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0

hostname(config)# object network MAPPED_1
hostname(config-network-object)# range 209.165.200.225 209.165.200.254

hostname(config)# object network MAPPED_2
hostname(config-network-object)# range 209.165.202.129 209.165.200.158

hostname(config)# object network SERVERS_1
hostname(config-network-object)# subnet 209.165.201.0 255.255.255.224

hostname(config)# object network SERVERS_2
hostname(config-network-object)# subnet 203.0.113.0 255.255.255.0

hostname(config)# nat (inside,outside) source dynamic INSIDE_NW MAPPED_1 destination static SERVERS_1 SERVERS_1
hostname(config)# nat (inside,outside) source dynamic INSIDE_NW MAPPED_2 destination static SERVERS_2 SERVERS_2
```

The following example configures dynamic NAT for an IPv6 inside network 2001:DB8:AAAA::/96 when accessing servers on the IPv4 209.165.201.1/27 network as well as servers on the 203.0.113.0/24 network:

```
hostname(config)# object network INSIDE_NW
hostname(config-network-object)# subnet 2001:DB8:AAAA::/96

hostname(config)# object network MAPPED_1
hostname(config-network-object)# range 209.165.200.225 209.165.200.254

hostname(config)# object network MAPPED_2
hostname(config-network-object)# range 209.165.202.129 209.165.200.158

hostname(config)# object network SERVERS_1
hostname(config-network-object)# subnet 209.165.201.0 255.255.255.224

hostname(config)# object network SERVERS_2
hostname(config-network-object)# subnet 203.0.113.0 255.255.255.0

hostname(config)# nat (inside,outside) source dynamic INSIDE_NW MAPPED_1 destination static SERVERS_1 SERVERS_1
hostname(config)# nat (inside,outside) source dynamic INSIDE_NW MAPPED_2 destination static SERVERS_2 SERVERS_2
```

Configuring Dynamic PAT (Hide)

This section describes how to configure twice NAT for dynamic PAT (hide). For more information, see Dynamic PAT, page 4-8.

Guidelines

For a PAT pool:
- If available, the real source port number is used for the mapped port. However, if the real port is not available, by default the mapped ports are chosen from the same range of ports as the real port number: 0 to 511, 512 to 1023, and 1024 to 65535. Therefore, ports below 1024 have only a small PAT pool that can be used. (8.4(3) and later, not including 8.5(1) or 8.6(1)) If you have a lot of traffic that uses the lower port ranges, you can now specify a flat range of ports to be used instead of the three unequal-sized tiers: either 1024 to 65535, or 1 to 65535.
- If you use the same PAT pool object in two separate rules, then be sure to specify the same options for each rule. For example, if one rule specifies extended PAT and a flat range, then the other rule must also specify extended PAT and a flat range.

For extended PAT for a PAT pool:
- Many application inspections do not support extended PAT. See Default Settings and NAT Limitations, page 7-4 in Chapter 7, “Getting Started with Application Layer Protocol Inspection,” for a complete list of unsupported inspections.
- If you enable extended PAT for a dynamic PAT rule, then you cannot also use an address in the PAT pool as the PAT address in a separate static NAT-with-port-translation rule. For example, if the PAT pool includes 10.1.1.1, then you cannot create a static NAT-with-port-translation rule using 10.1.1.1 as the PAT address.
- If you use a PAT pool and specify an interface for fallback, you cannot specify extended PAT.
- For VoIP deployments that use ICE or TURN, do not use extended PAT. ICE and TURN rely on the PAT binding to be the same for all destinations.

For round robin for a PAT pool:
- If a host has an existing connection, then subsequent connections from that host will use the same PAT IP address if ports are available. Note: This “stickiness” does not survive a failover. If the ASA fails over, then subsequent connections from a host may not use the initial IP address.
- Round robin, especially when combined with extended PAT, can consume a large amount of memory. Because NAT pools are created for every mapped protocol/IP address/port range, round robin results in a large number of concurrent NAT pools, which use memory. Extended PAT results in an even larger number of concurrent NAT pools.
### Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> Create network objects or groups for the:</td>
<td>See Adding Network Objects for Real and Mapped Addresses, page 6-4.</td>
</tr>
<tr>
<td>• Source real addresses</td>
<td>If you want to translate all source traffic, you can skip adding an object for the source real addresses, and instead specify the any keyword in the <code>nat</code> command.</td>
</tr>
<tr>
<td>• Source mapped addresses</td>
<td>If you want to use the interface address as the mapped address, you can skip adding an object for the source mapped addresses, and instead specify the <code>interface</code> keyword in the <code>nat</code> command.</td>
</tr>
<tr>
<td>• Destination real addresses</td>
<td>If you want to configure destination static interface NAT with port translation only, you can skip adding an object for the destination mapped addresses, and instead specify the <code>interface</code> keyword in the <code>nat</code> command.</td>
</tr>
<tr>
<td>• Destination mapped addresses</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> (Optional) Create service objects for the:</td>
<td>See (Optional) Adding Service Objects for Real and Mapped Ports, page 6-6.</td>
</tr>
<tr>
<td>• Destination real ports</td>
<td></td>
</tr>
<tr>
<td>• Destination mapped ports</td>
<td></td>
</tr>
</tbody>
</table>
### Step 3

**Command**

```
nat [(real_ifc,mapped_ifc)]
    [line | (after-auto [line])]
source dynamic (real-obj | any)
    (mapped_obj [interface [ipv6]] | [pat-pool
        mapped_obj [round-robin] [extended]
    [flat [include-reserve]] [interface
        [ipv6]] | [interface [ipv6]] [destination
        static {mapped_obj | interface [ipv6]}
        real_obj] [service mapped_dest_svc_obj
        real_dest_svc_obj] [dns] [unidirectional]
    [inactive] [description desc]
```

**Purpose**

Configures dynamic PAT (hide). See the following guidelines:

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword any for one or both of the interfaces.

- **Section and Line**—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table (see NAT Rule Order, page 4-18). If you want to add the rule into section 3 instead (after the network object NAT rules), then use the after-auto keyword. You can insert a rule anywhere in the applicable section using the line argument.

- **Source addresses:**
  - Real—Specify a network object, group, or the any keyword. Use the any keyword if you want to translate all traffic from the real interface to the mapped interface.
  - Mapped—Configure one of the following:
    - Network object—Specify a network object that contains a host address.
    - **pat-pool**—Specify the pat-pool keyword and a network object or group that contains multiple addresses.
    - **interface**—(Routed mode only) Specify the interface keyword alone to only use interface PAT. If you specify ipv6, then the IPv6 address of the interface is used. When specified with a PAT pool or network object, the interface keyword enables interface PAT fallback. After the PAT IP addresses are used up, then the IP address of the mapped interface is used. For this option, you must configure a specific interface for the mapped_ifc.

(continued)
For a PAT pool, you can specify one or more of the following options:

-- Round robin—The round-robin keyword enables round-robin address allocation for a PAT pool. Without round robin, by default all ports for a PAT address will be allocated before the next PAT address is used. The round-robin method assigns an address/port from each PAT address in the pool before returning to use the first address again, and then the second address, and so on.

-- Extended PAT—The extended keyword enables extended PAT. Extended PAT uses 65535 ports per service, as opposed to per IP address, by including the destination address and port in the translation information. Normally, the destination port and address are not considered when creating PAT translations, so you are limited to 65535 ports per PAT address. For example, with extended PAT, you can create a translation of 10.1.1.1:1027 when going to 192.168.1.7:23 as well as a translation of 10.1.1.1:1027 when going to 192.168.1.7:80.

-- Flat range—The flat keyword enables use of the entire 1024 to 65535 port range when allocating ports. When choosing the mapped port number for a translation, the ASA uses the real source port number if it is available. However, without this option, if the real port is not available, by default the mapped ports are chosen from the same range of ports as the real port number: 1 to 511, 512 to 1023, and 1024 to 65535. To avoid running out of ports at the low ranges, configure this setting. To use the entire range of 1 to 65535, also specify the include-reserve keyword.

(continued)
### Configuring Twice NAT

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>(continued)</td>
<td></td>
</tr>
<tr>
<td>- Destination addresses (Optional):</td>
<td></td>
</tr>
<tr>
<td>- Mapped—Specify a network object or group, or for static interface NAT with port translation only (routed mode), specify the <code>interface</code> keyword. If you specify <code>ipv6</code>, then the IPv6 address of the interface is used. If you specify <code>interface</code>, be sure to also configure the <code>service</code> keyword. For this option, you must configure a specific interface for the <code>real_ifc</code>. See Static Interface NAT with Port Translation, page 4-5 for more information.</td>
<td></td>
</tr>
<tr>
<td>- Real—Specify a network object or group. For identity NAT, simply use the same object or group for both the real and mapped addresses.</td>
<td></td>
</tr>
<tr>
<td>- Destination port—(Optional) Specify the <code>service</code> keyword along with the real and mapped service objects. For identity port translation, simply use the same service object for both the real and mapped ports.</td>
<td></td>
</tr>
<tr>
<td>- DNS—(Optional; for a source-only rule) The <code>dns</code> keyword translates DNS replies. Be sure DNS inspection is enabled (it is enabled by default). You cannot configure the <code>dns</code> keyword if you configure a <code>destination</code> address. See DNS and NAT, page 4-28 for more information.</td>
<td></td>
</tr>
<tr>
<td>- Unidirectional—(Optional) Specify <code>unidirectional</code> so the destination addresses cannot initiate traffic to the source addresses.</td>
<td></td>
</tr>
<tr>
<td>- Inactive—(Optional) To make this rule inactive without having to remove the command, use the <code>inactive</code> keyword. To reactivate it, reenter the whole command without the <code>inactive</code> keyword.</td>
<td></td>
</tr>
<tr>
<td>- Description—(Optional) Provide a description up to 200 characters using the <code>description</code> keyword.</td>
<td></td>
</tr>
</tbody>
</table>
Examples

The following example configures interface PAT for inside network 192.168.1.0/24 when accessing outside Telnet server 209.165.201.23, and Dynamic PAT using a PAT pool when accessing any server on the 203.0.113.0/24 network.

```
hostname(config)# object network INSIDE_NW
hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0

hostname(config)# object network PAT_POOL
hostname(config-network-object)# range 209.165.200.225 209.165.200.254

hostname(config)# object network TELNET_SVR
hostname(config-network-object)# host 209.165.201.23

hostname(config)# object service TELNET
hostname(config-service-object)# service tcp destination eq 23

hostname(config)# object network SERVERS
hostname(config-network-object)# subnet 203.0.113.0 255.255.255.0

hostname(config)# nat (inside,outside) source dynamic INSIDE_NW interface destination static TELNET_SVR TELNET_SVR service TELNET TELNET
hostname(config)# nat (inside,outside) source dynamic INSIDE_NW pat-pool PAT_POOL destination static SERVERS SERVERS
```

The following example configures interface PAT for inside network 192.168.1.0/24 when accessing outside IPv6 Telnet server 2001:DB8::23, and Dynamic PAT using a PAT pool when accessing any server on the 2001:DB8:AAAA::/96 network.

```
hostname(config)# object network INSIDE_NW
hostname(config-network-object)# subnet 192.168.1.0 255.255.255.0

hostname(config)# object network PAT_POOL
hostname(config-network-object)# range 2001:DB8:AAAA::1 2001:DB8:AAAA::200

hostname(config)# object network TELNET_SVR
hostname(config-network-object)# host 2001:DB8::23

hostname(config)# object service TELNET
hostname(config-service-object)# service tcp destination eq 23

hostname(config)# object network SERVERS
hostname(config-network-object)# subnet 2001:DB8:AAAA::/96

hostname(config)# nat (inside,outside) source dynamic INSIDE_NW interface destination static TELNET_SVR TELNET_SVR service TELNET TELNET
hostname(config)# nat (inside,outside) source dynamic INSIDE_NW pat-pool PAT_POOL destination static SERVERS SERVERS
```
Configuring Static NAT or Static NAT-with-Port-Translation

This section describes how to configure a static NAT rule using twice NAT. For more information about static NAT, see Static NAT, page 4-3.

Detailed Steps

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create network objects or groups for the:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Source real addresses</td>
<td>See Adding Network Objects for Real and Mapped Addresses, page 6-4.</td>
<td></td>
</tr>
<tr>
<td>• Source mapped addresses</td>
<td>If you want to configure source static interface NAT with port translation only, you can skip adding an object for the source mapped addresses, and instead specify the interface keyword in the nat command.</td>
<td></td>
</tr>
<tr>
<td>• Destination real addresses</td>
<td>If you want to configure destination static interface NAT with port translation only, you can skip adding an object for the destination mapped addresses, and instead specify the interface keyword in the nat command.</td>
<td></td>
</tr>
<tr>
<td>• Destination mapped addresses</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 2</th>
<th>(Optional) Create service objects for the:</th>
<th>See (Optional) Adding Service Objects for Real and Mapped Ports, page 6-6.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Source or Destination real ports</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Source or Destination mapped ports</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Step 3

**nat** {
  \((\text{real}_\text{ifc},\text{mapped}_\text{ifc})\)

  \[\text{line} | \text{after-object} [\text{line}]]\)

  source static real_obj

  [mapped_obj | interface [ipv6]]

  [destination static {mapped_obj | interface [ipv6]} real_obj]

  [service real_src_mapped_dest_svc_obj

  mapped_src_real_dest_svc_obj][net-to-net]

  [dns] [unidirectional | no-proxy-arp]

  [inactive] [description desc]

---

**Example:**

```
hostname(config)# nat (inside,dmz) source
static MyInsNet MyInsNet_mapped
destination static Server1 Server1 service
REAL_SRC_SVC MAPPED_SRC_SVC
```

---

**Command** | **Purpose**
--- | ---

**nat** {
  \((\text{real}_\text{ifc},\text{mapped}_\text{ifc})\)

  \[\text{line} | \text{after-object} [\text{line}]]\)

  source static real_obj

  [mapped_obj | interface [ipv6]]

  [destination static {mapped_obj | interface [ipv6]} real_obj]

  [service real_src_mapped_dest_svc_obj

  mapped_src_real_dest_svc_obj][net-to-net]

  [dns] [unidirectional | no-proxy-arp]

  [inactive] [description desc]

---

**Command** | **Purpose**
--- | ---

**nat** {
  \((\text{real}_\text{ifc},\text{mapped}_\text{ifc})\)

  \[\text{line} | \text{after-object} [\text{line}]]\)

  source static real_obj

  [mapped_obj | interface [ipv6]]

  [destination static {mapped_obj | interface [ipv6]} real_obj]

  [service real_src_mapped_dest_svc_obj

  mapped_src_real_dest_svc_obj][net-to-net]

  [dns] [unidirectional | no-proxy-arp]

  [inactive] [description desc]

---

**Configures static NAT. See the following guidelines:**

- **Interfaces**—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword any for one or both of the interfaces.

- **Section and Line**—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table. See NAT Rule Order, page 4-18 for more information about sections. If you want to add the rule into section 3 instead (after the network object NAT rules), then use the after-auto keyword. You can insert a rule anywhere in the applicable section using the line argument.

- **Source addresses:**
  - Real—Specify a network object or group.
  - Mapped—Specify a different network object or group. For static interface NAT with port translation only, you can specify the interface keyword (routed mode only). If you specify ipv6, then the IPv6 address of the interface is used. If you specify interface, be sure to also configure the service keyword (in this case, the service objects should include only the source port). For this option, you must configure a specific interface for the mapped_ifc. See Static Interface NAT with Port Translation, page 4-5 for more information.

- **Destination addresses** (Optional):
  - Mapped—Specify a network object or group, or for static interface NAT with port translation only, specify the interface keyword. If you specify ipv6, then the IPv6 address of the interface is used. If you specify interface, be sure to also configure the service keyword (in this case, the service objects should include only the destination port). For this option, you must configure a specific interface for the real_ifc.
  - Real—Specify a network object or group. For identity NAT, simply use the same object or group for both the real and mapped addresses.
Example:
The following example shows the use of static interface NAT with port translation. Hosts on the outside access an FTP server on the inside by connecting to the outside interface IP address with destination port 65000 through 65004. The traffic is untranslated to the internal FTP server at 192.168.10.100:6500 through 65004. Note that you specify the source port range in the service object (and not the destination port) because you want to translate the source address and port as identified in the command; the destination port is “any.” Because static NAT is bidirectional, “source” and “destination” refers primarily
to the command keywords; the actual source and destination address and port in a packet depends on which host sent the packet. In this example, connections are originated from outside to inside, so the “source” address and port of the FTP server is actually the destination address and port in the originating packet.

hostname(config)# object service FTP_PASV_PORT_RANGE
hostname(config-service-object)# service tcp source range 65000 65004

hostname(config)# object network HOST_FTP_SERVER
hostname(config-network-object)# host 192.168.10.100

hostname(config)# nat (inside,outside) source static HOST_FTP_SERVER interface service FTP_PASV_PORT_RANGE FTP_PASV_PORT_RANGE

The following example shows a static translation of one IPv6 network to another IPv6 when accessing an IPv6 network, and the dynamic PAT translation to an IPv4 PAT pool when accessing the IPv4 network:

hostname(config)# object network INSIDE_NW
hostname(config-network-object)# subnet 2001:DB8:AAAA::/96

hostname(config)# object network MAPPED_IPv6_NW
hostname(config-network-object)# subnet 2001:DB8:BBBB::/96

hostname(config)# object network OUTSIDE_IPv6_NW
hostname(config-network-object)# subnet 2001:DB8:CCCC::/96

hostname(config)# object network OUTSIDE_IPv4_NW
hostname(config-network-object)# subnet 10.1.1.0 255.255.255.0

hostname(config)# object network MAPPED_IPv4_POOL
hostname(config-network-object)# range 10.1.2.1 10.1.2.254

hostname(config)# nat (inside,outside) source static INSIDE_NW MAPPED_IPv6_NW destination static OUTSIDE_IPv6_NW OUTSIDE_IPv6_NW
hostname(config)# nat (inside,outside) source dynamic INSIDE_NW pat-pool MAPPED_IPv4_POOL destination static OUTSIDE_IPv4_NW OUTSIDE_IPv4_NW

Configuring Identity NAT

This section describes how to configure an identity NAT rule using twice NAT. For more information about identity NAT, see Identity NAT, page 4-10.
Detailed Steps

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>Create network objects or groups for the:</td>
</tr>
<tr>
<td></td>
<td>• Source real addresses (you will typically use the same object for the source mapped addresses)</td>
</tr>
<tr>
<td></td>
<td>• Destination real addresses</td>
</tr>
<tr>
<td></td>
<td>• Destination mapped addresses</td>
</tr>
<tr>
<td></td>
<td>See Adding Network Objects for Real and Mapped Addresses, page 6-4.</td>
</tr>
<tr>
<td></td>
<td>If you want to perform identity NAT for all addresses, you can skip creating an object for the source real addresses and instead use the keywords <strong>any any</strong> in the <strong>nat</strong> command.</td>
</tr>
<tr>
<td></td>
<td>If you want to configure destination static interface NAT with port translation only, you can skip adding an object for the destination mapped addresses, and instead specify the <strong>interface</strong> keyword in the <strong>nat</strong> command.</td>
</tr>
</tbody>
</table>

| **Step 2** | (Optional) Create service objects for the: |
| | • Source or Destination real ports |
| | • Source or Destination mapped ports |
| | See (Optional) Adding Service Objects for Real and Mapped Ports, page 6-6. |
### Step 3

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>nat [({real_ifc,mapped_ifc})]</code>&lt;br&gt;`[line</td>
<td>{after-object [line]}]<code>&lt;br&gt;</code>source static {nw_obj nw_obj</td>
</tr>
<tr>
<td>Example: hostname(config)# nat {inside, outside} source static MyInsNet MyInsNet destination static Server1 Server1</td>
<td></td>
</tr>
</tbody>
</table>

- Interfaces—(Required for transparent mode) Specify the real and mapped interfaces. Be sure to include the parentheses in your command. In routed mode, if you do not specify the real and mapped interfaces, all interfaces are used; you can also specify the keyword any for one or both of the interfaces.

- Section and Line—(Optional) By default, the NAT rule is added to the end of section 1 of the NAT table. See NAT Rule Order, page 4-18 for more information about sections. If you want to add the rule into section 3 instead (after the network object NAT rules), then use the after-auto keyword. You can insert a rule anywhere in the applicable section using the line argument.

- Source addresses—Specify a network object, group, or the any keyword for both the real and mapped addresses.

- Destination addresses (Optional):
  - Mapped—Specify a network object or group, or for static interface NAT with port translation only, specify the interface keyword (routed mode only). If you specify ipv6, then the IPv6 address of the interface is used. If you specify interface, be sure to also configure the service keyword (in this case, the service objects should include only the destination port). For this option, you must configure a specific interface for the real_ifc. See Static Interface NAT with Port Translation, page 4-5 for more information.
  - Real—Specify a network object or group. For identity NAT, simply use the same object or group for both the real and mapped addresses.

- Port—(Optional) Specify the service keyword along with the real and mapped service objects. For source port translation, the objects must specify the source service. The order of the service objects in the command for source port translation is service real_obj mapped_obj. For destination port translation, the objects must specify the destination service. The order of the service objects for destination port translation is service mapped_obj real_obj. In the rare case where you specify both the source and destination ports in the object, the first service object contains the real source port/mapped destination port; the second service object contains the mapped source port/real destination port. For identity port translation, simply use the same service object for both the real and mapped ports (source and/or destination ports, depending on your configuration).
Configuring Per-Session PAT Rules

By default, all TCP PAT traffic and all UDP DNS traffic uses per-session PAT. To use multi-session PAT for traffic, you can configure per-session PAT rules: a permit rule uses per-session PAT, and a deny rule uses multi-session PAT. For more information about per-session vs. multi-session PAT, see Per-Session PAT vs. Multi-Session PAT, page 4-9.

Detailed Steps

To configure a per-session PAT rule, see Configuring Per-Session PAT Rules, page 5-16.

Monitoring Twice NAT

To monitor twice NAT, enter one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>show nat</td>
<td>Shows NAT statistics, including hits for each NAT rule.</td>
</tr>
<tr>
<td>show nat pool</td>
<td>Shows NAT pool statistics, including the addresses and ports allocated, and how many times they were allocated.</td>
</tr>
<tr>
<td>show xlate</td>
<td>Shows current NAT session information.</td>
</tr>
<tr>
<td>show nat divert-table</td>
<td>All NAT rules build an entry in the NAT divert table. If the NAT divert field is set to ignore=yes NAT on the matching rule, the ASA stops the lookup and does a route lookup based on the destination IP to determine the egress interface. If the NAT divert field is set to ignore=no on the matching rule, walk the NAT table based on the found input_ifc and output_ifc and do the necessary translation. Egress interface will be output_ifc.</td>
</tr>
</tbody>
</table>
Configuration Examples for Twice NAT

This section includes the following configuration examples:
- Different Translation Depending on the Destination (Dynamic PAT), page 6-25
- Different Translation Depending on the Destination Address and Port (Dynamic PAT), page 6-27

Different Translation Depending on the Destination (Dynamic PAT)

Figure 6-1 shows a host on the 10.1.2.0/24 network accessing two different servers. When the host accesses the server at 209.165.201.11, the real address is translated to 209.165.202.129:port. When the host accesses the server at 209.165.200.225, the real address is translated to 209.165.202.130:port.

**Figure 6-1 Twice NAT with Different Destination Addresses**

```
Step 1 Add a network object for the inside network:
hostname(config)# object network myInsideNetwork
hostname(config-network-object)# subnet 10.1.2.0 255.255.255.0

Step 2 Add a network object for the DMZ network 1:
hostname(config)# object network DMZnetwork1
hostname(config-network-object)# subnet 209.165.201.0 255.255.255.224

Step 3 Add a network object for the PAT address:
hostname(config)# object network PATaddress1
hostname(config-network-object)# host 209.165.202.129
```
Step 4  Configure the first twice NAT rule:

```
hostname(config)# nat (inside,dmz) source dynamic myInsideNetwork PATaddress1 destination static DMZnetwork1 DMZnetwork1
```

Because you do not want to translate the destination address, you need to configure identity NAT for it by specifying the same address for the real and mapped destination addresses.

By default, the NAT rule is added to the end of section 1 of the NAT table. See Configuring Dynamic PAT (Hide), page 6-11 for more information about specifying the section and line number for the NAT rule.

Step 5  Add a network object for the DMZ network 2:

```
hostname(config)# object network DMZnetwork2
hostname(config-network-object)# subnet 209.165.200.224 255.255.255.224
```

Step 6  Add a network object for the PAT address:

```
hostname(config)# object network PATaddress2
hostname(config-network-object)# host 209.165.202.130
```

Step 7  Configure the second twice NAT rule:

```
hostname(config)# nat (inside,dmz) source dynamic myInsideNetwork PATaddress2 destination static DMZnetwork2 DMZnetwork2
```
Different Translation Depending on the Destination Address and Port (Dynamic PAT)

Figure 6-2 shows the use of source and destination ports. The host on the 10.1.2.0/24 network accesses a single host for both web services and Telnet services. When the host accesses the server for Telnet services, the real address is translated to 209.165.202.129;port. When the host accesses the same server for web services, the real address is translated to 209.165.202.130;port.

![Twice NAT with Different Destination Ports](image)

**Step 1** Add a network object for the inside network:
```
hostname(config)# object network myInsideNetwork
hostname(config-network-object)# subnet 10.1.2.0 255.255.255.0
```

**Step 2** Add a network object for the Telnet/Web server:
```
hostname(config)# object network TelnetWebServer
hostname(config-network-object)# host 209.165.201.11
```

**Step 3** Add a network object for the PAT address when using Telnet:
```
hostname(config)# object network PATaddress1
hostname(config-network-object)# host 209.165.202.11
```

**Step 4** Add a service object for Telnet:
```
hostname(config)# object service TelnetObj
hostname(config-network-object)# service tcp destination eq telnet
```
Step 5  Configure the first twice NAT rule:

```
hostname(config)# nat (inside, outside) source dynamic myInsideNetwork PATaddress1
destination static TelnetWebServer TelnetWebServer service TelnetObj TelnetObj
```

Because you do not want to translate the destination address or port, you need to configure identity NAT for them by specifying the same address for the real and mapped destination addresses, and the same port for the real and mapped service.

By default, the NAT rule is added to the end of section 1 of the NAT table. See Configuring Dynamic PAT (Hide), page 6-11 for more information about specifying the section and line number for the NAT rule.

Step 6  Add a network object for the PAT address when using HTTP:

```
hostname(config)# object network PATaddress2
hostname(config-network-object)# host 209.165.202.130
```

Step 7  Add a service object for HTTP:

```
hostname(config)# object service HTTPObj
hostname(config-network-object)# service tcp destination eq http
```

Step 8  Configure the second twice NAT rule:

```
hostname(config)# nat (inside, outside) source dynamic myInsideNetwork PATaddress2
destination static TelnetWebServer TelnetWebServer service HTTPObj HTTPObj
```
Feature History for Twice NAT

Table 6-1 lists each feature change and the platform release in which it was implemented.

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twice NAT</td>
<td>8.3(1)</td>
<td>Twice NAT lets you identify both the source and destination address in a single rule. We modified or introduced the following commands: <code>nat</code>, <code>show nat</code>, <code>show xlate</code>, <code>show nat pool</code>.</td>
</tr>
<tr>
<td>Identity NAT configurable proxy ARP and route lookup</td>
<td>8.4(2)/8.5(1)</td>
<td>In earlier releases for identity NAT, proxy ARP was disabled, and a route lookup was always used to determine the egress interface. You could not configure these settings. In 8.4(2) and later, the default behavior for identity NAT was changed to match the behavior of other static NAT configurations: proxy ARP is enabled, and the NAT configuration determines the egress interface (if specified) by default. You can leave these settings as is, or you can enable or disable them discretely. Note that you can now also disable proxy ARP for regular static NAT. For pre-8.3 configurations, the migration of NAT exempt rules (the <code>nat 0 access-list</code> command) to 8.4(2) and later now includes the following keywords to disable proxy ARP and to use a route lookup: <code>no-proxy-arp</code> and <code>route-lookup</code>. The <code>unidirectional</code> keyword that was used for migrating to 8.3(2) and 8.4(1) is no longer used for migration. When upgrading to 8.4(2) from 8.3(1), 8.3(2), and 8.4(1), all identity NAT configurations will now include the <code>no-proxy-arp</code> and <code>route-lookup</code> keywords, to maintain existing functionality. The <code>unidirectional</code> keyword is removed. We modified the following command: <code>nat source static [no-proxy-arp] [route-lookup]</code>.</td>
</tr>
<tr>
<td>PAT pool and round robin address assignment</td>
<td>8.4(2)/8.5(1)</td>
<td>You can now specify a pool of PAT addresses instead of a single address. You can also optionally enable round-robin assignment of PAT addresses instead of first using all ports on a PAT address before using the next address in the pool. These features help prevent a large number of connections from a single PAT address from appearing to be part of a DoS attack and makes configuration of large numbers of PAT addresses easy. We modified the following command: <code>nat source dynamic [pat-pool mapped_object [round-robin]]</code>.</td>
</tr>
</tbody>
</table>
Table 6-1  Feature History for Twice NAT (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Round robin PAT pool allocation uses the same IP address for existing hosts</td>
<td>8.4(3)</td>
<td>When using a PAT pool with round robin allocation, if a host has an existing connection, then subsequent connections from that host will use the same PAT IP address if ports are available. We did not modify any commands. This feature is not available in 8.5(1) or 8.6(1).</td>
</tr>
<tr>
<td>Flat range of PAT ports for a PAT pool</td>
<td>8.4(3)</td>
<td>If available, the real source port number is used for the mapped port. However, if the real port is not available, by default the mapped ports are chosen from the same range of ports as the real port number: 0 to 511, 512 to 1023, and 1024 to 65535. Therefore, ports below 1024 have only a small PAT pool. If you have a lot of traffic that uses the lower port ranges, when using a PAT pool, you can now specify a flat range of ports to be used instead of the three unequal-sized tiers: either 1024 to 65535, or 1 to 65535. We modified the following command: <strong>nat source dynamic [pat-pool mapped_object [flat [include-reserve]]]</strong>. This feature is not available in 8.5(1) or 8.6(1).</td>
</tr>
<tr>
<td>Extended PAT for a PAT pool</td>
<td>8.4(3)</td>
<td>Each PAT IP address allows up to 65535 ports. If 65535 ports do not provide enough translations, you can now enable extended PAT for a PAT pool. Extended PAT uses 65535 ports per service, as opposed to per IP address, by including the destination address and port in the translation information. We modified the following command: <strong>nat source dynamic [pat-pool mapped_object [extended]]</strong>. This feature is not available in 8.5(1) or 8.6(1).</td>
</tr>
</tbody>
</table>
Automatic NAT rules to translate a VPN peer’s local IP address back to the peer’s real IP address

8.4(3) In rare situations, you might want to use a VPN peer’s real IP address on the inside network instead of an assigned local IP address. Normally with VPN, the peer is given an assigned local IP address to access the inside network. However, you might want to translate the local IP address back to the peer’s real public IP address if, for example, your inside servers and network security is based on the peer’s real IP address.

You can enable this feature on one interface per tunnel group. Object NAT rules are dynamically added and deleted when the VPN session is established or disconnected. You can view the rules using the `show nat` command.

Note Because of routing issues, we do not recommend using this feature unless you know you need this feature; contact Cisco TAC to confirm feature compatibility with your network. See the following limitations:

- Only supports Cisco IPsec and AnyConnect Client.
- Return traffic to the public IP addresses must be routed back to the ASA so the NAT policy and VPN policy can be applied.
- Does not support load-balancing (because of routing issues).
- Does not support roaming (public IP changing).

We introduced the following command: `nat-assigned-to-public-ip interface` (tunnel-group general-attributes configuration mode).

### Table 6-1  Feature History for Twice NAT (continued)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
</table>
| Automatic NAT rules to translate a VPN peer’s local IP address back to the peer’s real IP address | 8.4(3) | In rare situations, you might want to use a VPN peer’s real IP address on the inside network instead of an assigned local IP address. Normally with VPN, the peer is given an assigned local IP address to access the inside network. However, you might want to translate the local IP address back to the peer’s real public IP address if, for example, your inside servers and network security is based on the peer’s real IP address.

You can enable this feature on one interface per tunnel group. Object NAT rules are dynamically added and deleted when the VPN session is established or disconnected. You can view the rules using the `show nat` command.

Note Because of routing issues, we do not recommend using this feature unless you know you need this feature; contact Cisco TAC to confirm feature compatibility with your network. See the following limitations:

- Only supports Cisco IPsec and AnyConnect Client.
- Return traffic to the public IP addresses must be routed back to the ASA so the NAT policy and VPN policy can be applied.
- Does not support load-balancing (because of routing issues).
- Does not support roaming (public IP changing).

We introduced the following command: `nat-assigned-to-public-ip interface` (tunnel-group general-attributes configuration mode). |
| NAT support for IPv6 | 9.0(1) | NAT now supports IPv6 traffic, as well as translating between IPv4 and IPv6. Translating between IPv4 and IPv6 is not supported in transparent mode.

We modified the following commands: `nat` (global configuration mode), `show nat`, `show nat pool`, `show xlate`. |
NAT support for reverse DNS lookups 9.0(1) NAT now supports translation of the DNS PTR record for reverse DNS lookups when using IPv4 NAT, IPv6 NAT, and NAT64 with DNS inspection enabled for the NAT rule.

Per-session PAT 9.0(1) The per-session PAT feature improves the scalability of PAT and, for clustering, allows each member unit to own PAT connections; multi-session PAT connections have to be forwarded to and owned by the master unit. At the end of a per-session PAT session, the ASA sends a reset and immediately removes the xlate. This reset causes the end node to immediately release the connection, avoiding the TIME_WAIT state. Multi-session PAT, on the other hand, uses the PAT timeout, by default 30 seconds. For “hit-and-run” traffic, such as HTTP or HTTPS, the per-session feature can dramatically increase the connection rate supported by one address. Without the per-session feature, the maximum connection rate for one address for an IP protocol is approximately 2000 per second. With the per-session feature, the connection rate for one address for an IP protocol is \( \frac{65535}{\text{average-lifetime}} \).

By default, all TCP traffic and UDP DNS traffic use a per-session PAT xlate. For traffic that requires multi-session PAT, such as H.323, SIP, or Skinny, you can disable per-session PAT by creating a per-session deny rule.

We introduced the following commands: `xlate per-session`, `show nat pool`.

---

**Feature History for Twice NAT (continued)**

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Platform Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAT support for reverse DNS lookups</td>
<td>9.0(1)</td>
<td>NAT now supports translation of the DNS PTR record for reverse DNS lookups when using IPv4 NAT, IPv6 NAT, and NAT64 with DNS inspection enabled for the NAT rule.</td>
</tr>
<tr>
<td>Per-session PAT</td>
<td>9.0(1)</td>
<td>The per-session PAT feature improves the scalability of PAT and, for clustering, allows each member unit to own PAT connections; multi-session PAT connections have to be forwarded to and owned by the master unit. At the end of a per-session PAT session, the ASA sends a reset and immediately removes the xlate. This reset causes the end node to immediately release the connection, avoiding the TIME_WAIT state. Multi-session PAT, on the other hand, uses the PAT timeout, by default 30 seconds. For “hit-and-run” traffic, such as HTTP or HTTPS, the per-session feature can dramatically increase the connection rate supported by one address. Without the per-session feature, the maximum connection rate for one address for an IP protocol is approximately 2000 per second. With the per-session feature, the connection rate for one address for an IP protocol is ( \frac{65535}{\text{average-lifetime}} ). By default, all TCP traffic and UDP DNS traffic use a per-session PAT xlate. For traffic that requires multi-session PAT, such as H.323, SIP, or Skinny, you can disable per-session PAT by creating a per-session deny rule. We introduced the following commands: <code>xlate per-session</code>, <code>show nat pool</code>.</td>
</tr>
</tbody>
</table>