CHAPTER 28

Configuring Static NAT

This chapter describes how to configure a static network translation and includes the following topics:

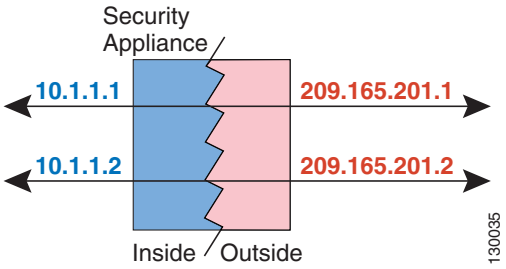
Information About Static NAT

Static NAT creates a fixed translation of real address(es) to mapped address(es).With dynamic NAT and PAT, each host uses a different address or port for each subsequent translation. Because the mapped address is the same for each consecutive connection with static NAT, and a persistent translation rule exists, static NAT allows hosts on the destination network to initiate traffic to a translated host (if an access list exists that allows it).

The main difference between dynamic NAT and a range of addresses for static NAT is that static NAT allows a remote host to initiate a connection to a translated host (if an access list exists that allows it), while dynamic NAT does not. You also need an equal number of mapped addresses as real addresses with static NAT.

Figure 28-1 shows a typical static NAT scenario. The translation is always active so both translated and remote hosts can originate connections, and the mapped address is statically assigned by the **static** command.

Figure 28-1 Static NAT



Licensing Requirements for Static NAT

The following table shows the licensing requirements for this feature:

Model	License Requirement
All models	Base License.

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.

Additional Guidelines and Limitations

The following features are not supported for static NAT:

- You cannot use the same real or mapped address in multiple **static** commands between the same two interfaces unless you use static PAT. (For more information, see Chapter 30, "Configuring Static PAT.")
- Do not use a mapped address in the **static** command that is also defined in a **global** command for the same mapped interface.

If your **nat** command includes the address of a host that has an entry in a DNS server, and the DNS server is on a different interface from a client, then the client and the DNS server need different addresses for the host; one needs the mapped address and one needs the real address. This option rewrites the address in the DNS reply to the client. The translated host needs to be on the same interface as either the client or the DNS server. Typically, hosts that need to allow access from other interfaces use a static translation, so this option is more likely to be used with the **static** command. (See the "DNS and NAT" section on page 26-9 for more information.)

-
- If you remove a **static** command, existing connections that use the translation are not affected. To remove these connections, enter the **clear local-host** command.
- You cannot clear static translations from the translation table with the **clear xlate** command; you must remove the **static** command instead. Only dynamic translations created by the **nat** and **global** commands can be removed with the **clear xlate** command.

Default Settings

Table 28-1 lists the command options and defaults for static NAT.

Table 28-1 Command Options and Defaults for Policy NAT

Command	Purpose
norandomseq, **tcp** *tcp_max_conns*, **udp** *udp_max_conns*, and *emb_limit*	These keywords set connection limits. However, we recommend using a more versatile method for setting connection limits; for more information, see Chapter 53, "Configuring Connection Limits and Timeouts." For *tcp_max_conns*, *emb_limit*, and *udp_max_conns*, the default value is 0 (unlimited), which is the maximum available.

Table 28-2 Command Options and Defaults for Regular NAT

nat_id	An integer between 1 and 2147483647. The NAT ID must match a **global** command NAT ID. See the "Information About Implementing Dynamic NAT and PAT" section on page 29-5 for more information about how NAT IDs are used. **0** is reserved for identity NAT. See the "Configuring Identity NAT" section on page 31-1 for more information about identity NAT. See Table 28-1, "Command Options and Defaults for Policy NAT," for information about other command options.

Configuring Static NAT

This section describes how to configure a static translation and includes the following topics:

Configuring Policy Static NAT

When you configure “policy NAT,” you identify the real addresses and destination/source addresses using an extended access list. To configure policy static NAT, enter the following command:

Command	Purpose
static (*real_interface,mapped_interface*) {*mapped_ip* | **interface**} **access-list** *acl_name* [**dns**] [**norandomseq**] [[**tcp**] *tcp_max_conns* [*emb_limit*]] [**udp** *udp_max_conns*] **Example:** hostname(config)# static (inside,outside) 209.165.202.129 access-list NET1	Configures a persistent one-to-one address translation rule by mapping a real IP address to a mapped IP address. Identify the real addresses and destination/source addresses using an extended access list. Create the extended access list using the **access-list extended** command. The first address in the access list is the real address; the second address is either the source or destination address, depending on where the traffic originates. (For more information, see Chapter 11, "Adding an Extended Access List."). This access list should include only **permit** ACEs. You can optionally specify the real and destination ports in the access list using the **eq** operator. Policy NAT considers the **inactive** and **time-range** keywords, but it does not support ACL with all **inactive** and **time-range** ACEs. The *real_ifc* argument specifies the name of the interface connected to the real IP address network. The *mapped_ifc* argument specifies the name of the interface connected to the mapped IP address network. The *mapped_ip* argument specifies the address to which the real address is translated. The **interface** keyword uses the interface IP address as the mapped address. Use this keyword if you want to use the interface address, but the address is dynamically assigned using DHCP. The **dns** option rewrites the A record, or address record, in DNS replies that match this static. For DNS replies traversing from a mapped interface to any other interface, the A record is rewritten from the mapped value to the real value. Inversely, for DNS replies traversing from any interface to a mapped interface, the A record is rewritten from the real value to the mapped value. The **norandomseq** disables TCP ISN randomization protection. The **tcp** *tcp_max_cons* option specifies the maximum number of simultaneous TCP connections allowed to the local-host. (See the **local-host** command). (Idle connections are closed after the idle timeout specified by the **timeout conn** command.) The *emb_limit* is the maximum number of embryonic connections per host. **Note** An embryonic limit applied using static NAT is applied to all connections to or from the real IP address, and not just connections between the specified interfaces. To apply limits to specific flows, see the "Configuring Connection Limits and Timeouts" section on page 53-3. The **udp** *tcp_max_cons* option specifies the maximum number of simultaneous UDP connections allowed to the local-host. (See the **local-host** command.) (Idle connections are closed after the idle timeout specified by the **timeout conn** command.) If this interface is on a lower security level than the interface you identify by the matching **global** statement, then you must enter **outside** to identify the NAT instance as outside NAT.

Example

To translate the real address 10.1.1.1 to the mapped address 192.168.1.1 when 10.1.1.1 sends traffic to the 209.165.200.224 network, the **access-list** and **static** commands are as follows:

```
hostname(config)# access-list TEST extended ip host 10.1.1.1 209.165.200.224
255.255.255.224
hostname(config)# static (inside,outside) 192.168.1.1 access-list TEST
```

In this case, the second address is the destination address. However, the same configuration is used for hosts to originate a connection to the mapped address. For example, when a host on the 209.165.200.224/27 network initiates a connection to 192.168.1.1, then the second address in the access list is the source address.

This access list should include only **permit** ACEs. You can optionally specify the real and destination ports in the access list using the **eq** operator. Policy NAT does not consider the **inactive** or **time-range** keywords; all ACEs are considered to be active for policy NAT configuration. See the "Policy NAT" section on page 26-5 for more information.

If you specify a network for translation (for example, 10.1.1.0 255.255.255.0), then the ASA translates the .0 and .255 addresses. If you want to prevent access to these addresses, be sure to configure an access list to deny access.

See Chapter 29, "Configuring Dynamic NAT and PAT," for information about the other options.

Configuring Regular Static NAT

To configure regular static NAT, enter the following command:

Command	Purpose
static (*real_interface,mapped_interface*) {*mapped_ip* | **interface**} *real_ip* [**netmask** *mask*][**dns**] [**norandomseq**] [[**tcp**] *tcp_max_conns* [*emb_limit*]] [**udp** *udp_max_conns*] **Example:** hostname(config)# static (inside,outside) 209.165.201.12 10.1.1.3 netmask 255.255.255.255	Configures a persistent one-to-one address translation rule by mapping a real IP address to a mapped IP address. The *real_ifc* argument specifies the name of the interface connected to the real IP address network. The *mapped_ifc* argument specifies the name of the interface connected to the mapped IP address network. The *mapped_ip* argument specifies the address to which the real address is translated. The **interface** keyword uses the interface IP address as the mapped address. Use this keyword if you want to use the interface address, but the address is dynamically assigned using DHCP. The *real_ip* specifies the real address that you want to translate. The **netmask** *mask* specifies the subnet mask for the real and mapped addresses. For single hosts, use 255.255.255.255. If you do not enter a mask, then the default mask for the IP address class is used, with one exception. If a host-bit is non-zero after masking, a host mask of 255.255.255.255 is used. If you use the **access-list** keyword instead of the real_ip, then the subnet mask used in the access list is also used for the mapped_ip. The **dns** option rewrites the A record, or address record, in DNS replies that match this static. For DNS replies traversing from a mapped interface to any other interface, the A record is rewritten from the mapped value to the real value. Inversely, for DNS replies traversing from any interface to a mapped interface, the A record is rewritten from the real value to the mapped value. The **norandomseq** disables TCP ISN randomization protection. The **tcp** *tcp_max_cons* option specifies the maximum number of simultaneous TCP connections allowed to the local-host. (See the **local-host** command). (Idle connections are closed after the idle timeout specified by the **timeout conn** command.) The *emb_limit* is the maximum number of embryonic connections per host. **Note** An embryonic limit applied using static NAT is applied to all connections to or from the real IP address, and not just connections between the specified interfaces. To apply limits to specific flows, see the "Configuring Connection Limits and Timeouts" section on page 53-3. The **udp** *tcp_max_cons* option specifies the maximum number of simultaneous UDP connections allowed to the local-host. (See the **local-host** command.) (Idle connections are closed after the idle timeout specified by the **timeout conn** command.)

Monitoring Static NAT

To monitor static NAT, perform one of the following tasks:

Command	Purpose
`show running-config static`	Displays all static commands in the configuration

Configuration Examples for Static NAT

This section contains configuration examples for static NAT and contains these sections:

Typical Static NAT Examples

For example, the following policy static NAT example shows a single real address that is translated to two mapped addresses depending on the destination address (see Figure 26-3 on page 26-5, "Policy NAT with Different Destination Addresses," for a related figure):

```
hostname(config)# access-list NET1 permit ip host 10.1.2.27 209.165.201.0 255.255.255.224
hostname(config)# access-list NET2 permit ip host 10.1.2.27 209.165.200.224
255.255.255.224
hostname(config)# static (inside,outside) 209.165.202.129 access-list NET1
hostname(config)# static (inside,outside) 209.165.202.130 access-list NET2
```

The following command maps an inside IP address (10.1.1.3) to an outside IP address (209.165.201.12):

```
hostname(config)# static (inside,outside) 209.165.201.12 10.1.1.3 netmask 255.255.255.255
```

The following command maps the outside address (209.165.201.15) to an inside address (10.1.1.6):

```
hostname(config)# static (outside,inside) 10.1.1.6 209.165.201.15 netmask 255.255.255.255
```

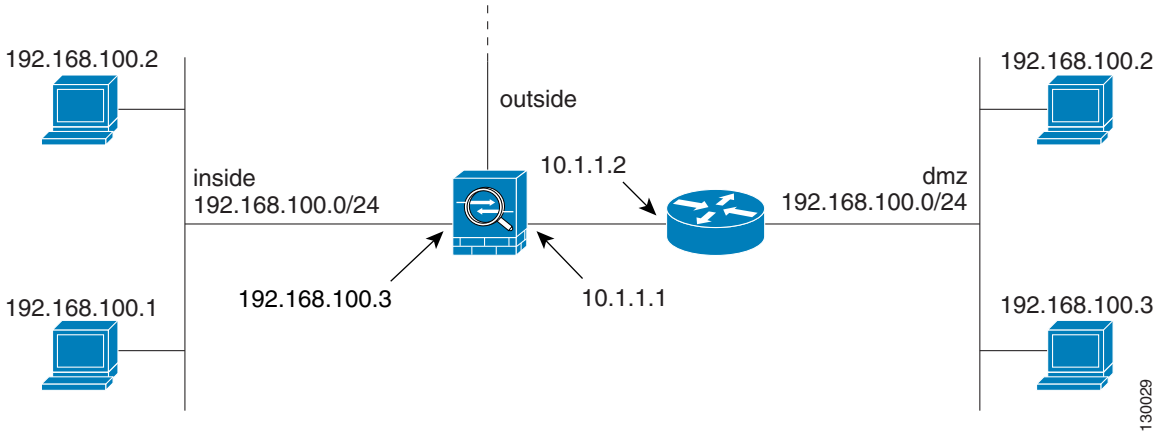
The following command statically maps an entire subnet:

```
hostname(config)# static (inside,dmz) 10.1.1.0 10.1.2.0 netmask 255.255.255.0
```

Example of Overlapping Networks

In Figure 28-2, the ASA connects two private networks with overlapping address ranges.

Figure 28-2 Using Outside NAT with Overlapping Networks



Two networks use an overlapping address space (192.168.100.0/24), but hosts on each network must communicate (as allowed by access lists). Without NAT, when a host on the inside network tries to access a host on the overlapping DMZ network, the packet never makes it past the ASA, which sees the packet as having a destination address on the inside network. Moreover, if the destination address is being used by another host on the inside network, that host receives the packet.

To solve this problem, use NAT to provide non-overlapping addresses. If you want to allow access in both directions, use static NAT for both networks. If you only want to allow the inside interface to access hosts on the DMZ, then you can use dynamic NAT for the inside addresses, and static NAT for the DMZ addresses you want to access. This example shows static NAT.

To configure static NAT for these two interfaces, perform the following steps. The 10.1.1.0/24 network on the DMZ is not translated.

Step 1 Translate 192.168.100.0/24 on the inside to 10.1.2.0/24 when it accesses the DMZ by entering the following command:

```
hostname(config)# static (inside,dmz) 10.1.2.0 192.168.100.0 netmask 255.255.255.0
```

Step 2 Translate the 192.168.100.0/24 network on the DMZ to 10.1.3.0/24 when it accesses the inside by entering the following command:

```
hostname(config)# static (dmz,inside) 10.1.3.0 192.168.100.0 netmask 255.255.255.0
```

Step 3 Configure the following static routes so that traffic to the dmz network can be routed correctly by the ASA:

```
hostname(config)# route dmz 192.168.100.128 255.255.255.128 10.1.1.2 1
hostname(config)# route dmz 192.168.100.0 255.255.255.128 10.1.1.2 1
```

The ASA already has a connected route for the inside network. These static routes allow the ASA to send traffic for the 192.168.100.0/24 network out the DMZ interface to the gateway router at 10.1.1.2. (You need to split the network into two because you cannot create a static route with the exact same network as a connected route.) Alternatively, you could use a more broad route for the DMZ traffic, such as a default route.

If host 192.168.100.2 on the DMZ network wants to initiate a connection to host 192.168.100.2 on the inside network, the following events occur:

1. The DMZ host 192.168.100.2 sends the packet to IP address 10.1.2.2.
2. When the ASA receives this packet, the ASA translates the source address from 192.168.100.2 to 10.1.3.2.
3. Then the ASA translates the destination address from 10.1.2.2 to 192.168.100.2, and the packet is forwarded.

Additional References

For additional information related to implementing Static NAT, see the following sections:

- Related Documents, page 28-11

Related Documents

Related Topic	Document Title
static command	*Cisco Security Appliance Command Reference*

Feature History for Static NAT

Table 28-3 lists the release history for this feature.

Table 28-3 Feature History for Static NAT

Feature Name	Releases	Feature Information
Regular static NAT and policy static NAT	7.0	Static NAT creates a fixed translation of real addresses to mapped addresses. The **static** command was introduced.
Regular static NAT and policy static NAT	7.3.1	NAT began support in transparent firewall mode.