CHAPTER 26

Information About NAT

This chapter provides an overview of how Network Address Translation (NAT) works on the ASA and includes the following sections:

- Introduction to NAT, page 26-1
- NAT Types, page 26-2
- NAT in Routed Mode, page 26-2
- NAT in Transparent Mode, page 26-3
- Policy NAT, page 26-5
- NAT and Same Security Level Interfaces, page 26-8
- Order of NAT Commands Used to Match Real Addresses, page 26-8
- Mapped Address Guidelines, page 26-8
- DNS and NAT, page 26-9
- Where to Go Next, page 26-11

Introduction to NAT

Address translation substitutes the real address in a packet with a mapped address that is routable on the destination network. NAT is composed of two steps: the process by which a real address is translated into a mapped address and the process to undo translation for returning traffic.

The ASA translates an address when a NAT rule matches the traffic. If no NAT rule matches, processing for the packet continues. The exception is when you enable NAT control. NAT control requires that packets traversing from a higher security interface (inside) to a lower security interface (outside) match a NAT rule, or processing for the packet stops. See the "Security Levels" section on page 6-5 for more information about security levels. See Chapter 27, "Configuring NAT Control," for more information about NAT control.

Note In this document, all types of translation are referred to as NAT. When describing NAT, the terms *inside* and *outside* represent the security relationship between any two interfaces. The higher security level is inside and the lower security level is outside. For example, interface 1 is at 60 and interface 2 is at 50; therefore, interface 1 is "inside" and interface 2 is "outside."

Some of the benefits of NAT are as follows:

- You can use private addresses on your inside networks. Private addresses are not routable on the Internet. See the "Private Networks" section on page C-2 for more information.
- NAT hides the real addresses from other networks, so attackers cannot learn the real address of a host.
- You can resolve IP routing problems, such as overlapping addresses.

See Table 40-1 on page 40-4 for information about protocols that do not support NAT.

NAT Types

You can implement address translation as dynamic NAT, Port Address Translation (PAT), static NAT, static PAT, or as a mix of these types. You can also configure rules to bypass NAT; for example, to enable NAT control when you do not want to perform NAT. The following translation types are available:

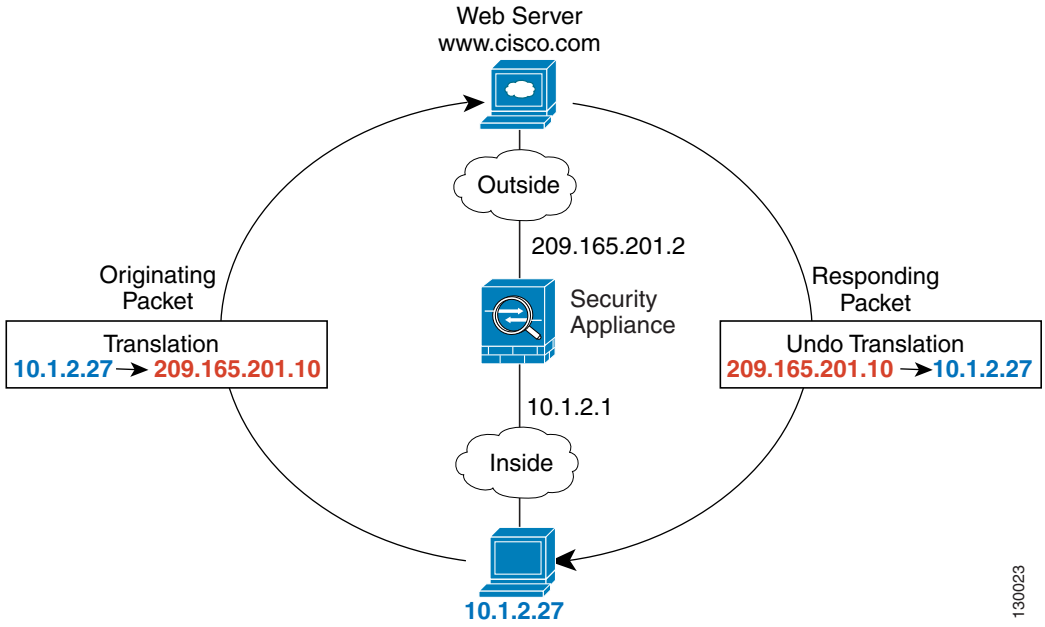
- Dynamic NAT—Dynamic NAT translates a group of real addresses to a pool of mapped addresses that are routable on the destination network. For details about dynamic NAT, see the Chapter 29, "Configuring Dynamic NAT and PAT."
- PAT—PAT translates multiple real address to a single mapped IP address. For details about PAT, see the Chapter 29, "Configuring Dynamic NAT and PAT."
- Static NAT—Static NAT creates a fixed translation of real addresses to mapped addresses. With dynamic NAT and PAT, each host uses a different address or port for each subsequent translation. For details about static NAT, see the Chapter 28, "Configuring Static NAT."
- Static PAT—Static PAT is the same as static NAT, except that it enables you to specify the protocol and port for the real and mapped addresses. For details about static PAT, see the Chapter 30, "Configuring Static PAT."

If you enable NAT control, then inside hosts must match a NAT rule when accessing outside hosts. If you do not want to perform NAT for some hosts, then you can bypass NAT for those hosts, or you can disable NAT control. For details about bypassing NAT, see Chapter 31, "Bypassing NAT."

NAT in Routed Mode

Figure 26-1 shows a typical NAT example in routed mode, with a private network on the inside. When the inside host at 10.1.2.27 sends a packet to a web server, the real source address, 10.1.2.27, of the packet is changed to a mapped address, 209.165.201.10. When the server responds, it sends the response to the mapped address, 209.165.201.10, and the security appliance receives the packet. The security appliance then changes the translation of the mapped address, 209.165.201.10, back to the real address, 10.1.2.27, before sending it to the host.

Figure 26-1 NAT Example: Routed Mode



See the following commands for this example:

```
hostname(config)# nat (inside) 1 10.1.2.0 255.255.255.0
hostname(config)# global (outside) 1 209.165.201.1-209.165.201.15
```

NAT in Transparent Mode

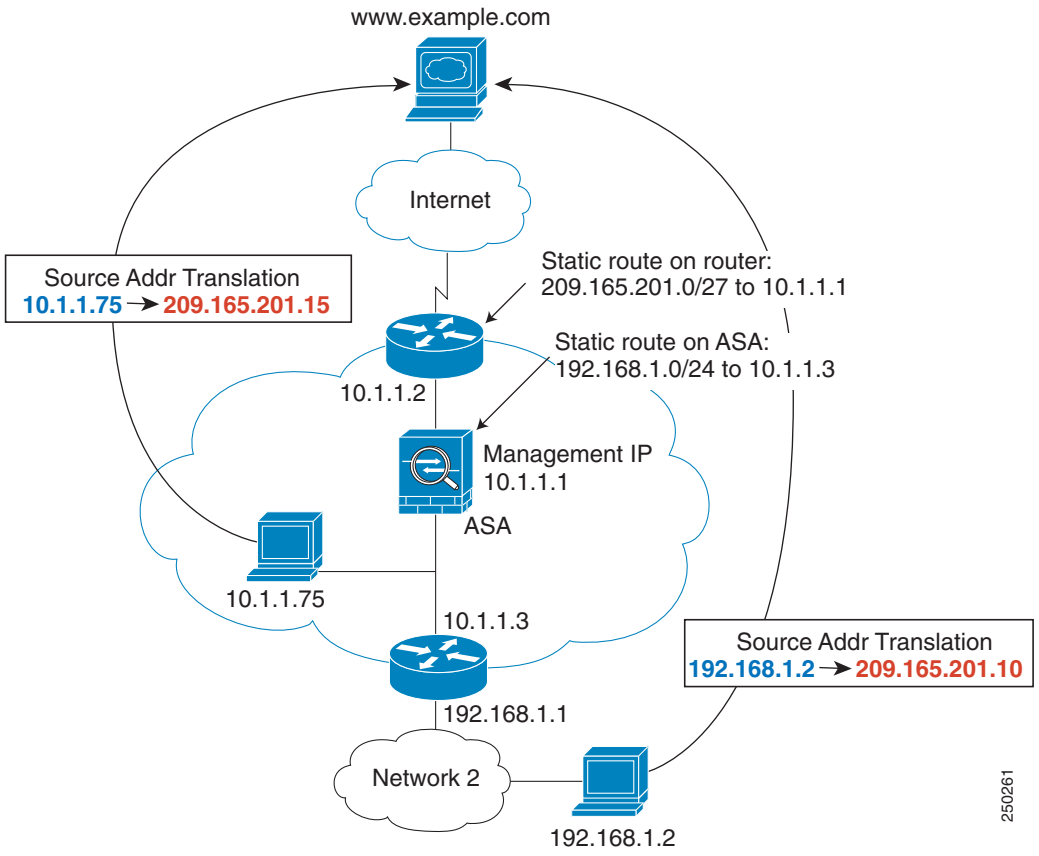
Using NAT in transparent mode eliminates the need for the upstream or downstream routers to perform NAT for their networks. For example, a transparent firewall ASA is useful between two VRFs so tha you can establish BGP neighbor relations between the VRFs and the global table. However, NAT per VRF might not be supported. In this case, using NAT in transparent mode is essential.

NAT in transparent mode has the following requirements and limitations:

- When the mapped addresses are not on the same network as the transparent firewall, then on the upstream router you need to add a static route for the mapped addresses that points to the downstream router (through the ASA).
- When you have VoIP or DNS traffic with NAT and inspection enabled, to successfully translate the IP address inside VoIP and DNS packets, the ASA needs to perform a route lookup. Unless the host is on a directly-connected network, then you need to add a static route on the ASA for the real host address that is embedded in the packet.
- The **alias** command is not supported.
- Because the transparent firewall does not have any interface IP addresses, you cannot use interface PAT.
- ARP inspection is not supported. Moreover, if for some reason a host on one side of the firewall sends an ARP request to a host on the other side of the firewall, and the initiating host real address is mapped to a different address on the same subnet, then the real address remains visible in the ARP request.

Figure 26-2 shows a typical NAT scenario in transparent mode, with the same network on the inside and outside interfaces. The transparent firewall in this scenario is performing the NAT service so that the upstream router does not have to perform NAT.

Figure 26-2 NAT Example: Transparent Mode



1. When the inside host at 10.1.1.75 sends a packet to a web server, the real source address of the packet, 10.1.1.75, is changed to a mapped address, 209.165.201.15.
2. When the server responds, it sends the response to the mapped address, 209.165.201.15, and the ASA receives the packet because the upstream router includes this mapped network in a static route directed through the ASA.
3. The ASA then undoes the translation of the mapped address, 209.165.201.15, back to the real address, 10.1.1.1.75. Because the real address is directly-connected, the ASA sends it directly to the host.
4. For host 192.168.1.2, the same process occurs, except that the ASA looks up the route in its route table and sends the packet to the downstream router at 10.1.1.3 based on the static route.

See the following commands for this example:

```
hostname(config)# route inside 192.168.1.0 255.255.255.0 10.1.1.3 1
hostname(config)# nat (inside) 1 10.1.1.0 255.255.255.0
hostname(config)# nat (inside) 1 192.168.1.0 255.255.255.0
hostname(config)# global (outside) 1 209.165.201.1-209.165.201.15
```

Policy NAT

Policy NAT lets you identify real addresses for address translation by specifying the source and destination addresses in an extended access list. You can also optionally specify the source and destination ports. Regular NAT can only consider the source addresses, not the destination address . For example, with policy NAT you can translate the real address to mapped address A when it accesses server A, but also translate the real address to mapped address B when it accesses server B.

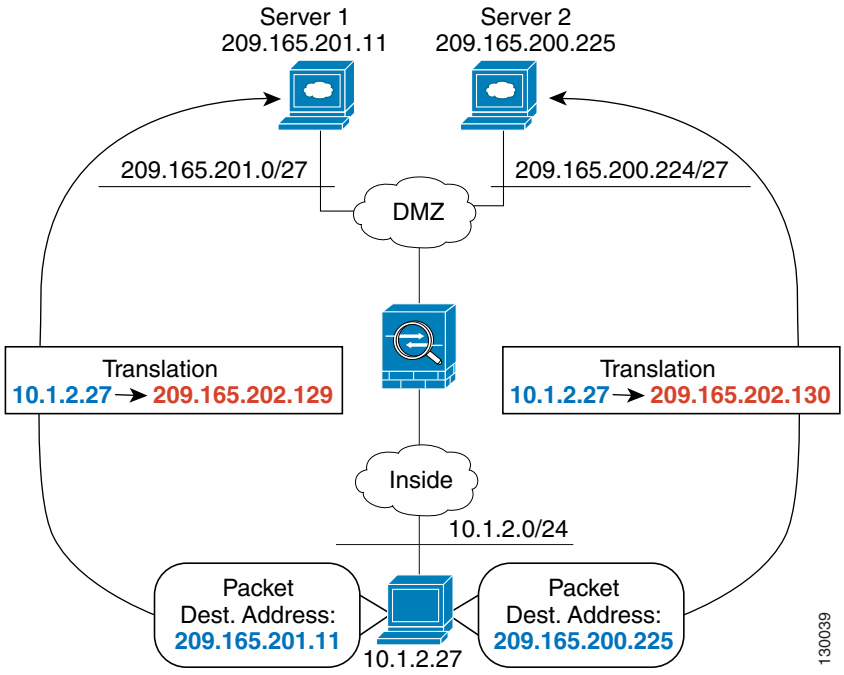
Note Policy NAT does not support time-based access lists.

For applications that require application inspection for secondary channels (for example, FTP and VoIP), the policy specified in the policy NAT statement should include the secondary ports. When the ports cannot be predicted, the policy should specify only the IP addresses for the secondary channel. With this configuration, the security appliance translates the secondary ports.

Note All types of NAT support policy NAT, except for NAT exemption. NAT exemption uses an access list to identify the real addresses, but it differs from policy NAT in that the ports are not considered. See the "Bypassing NAT When NAT Control is Enabled" section on page 27-3 for other differences. You can accomplish the same result as NAT exemption using static identity NAT, which does support policy NAT.

Figure 26-3 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. When the host accesses the server at 209.165.200.225, the real address is translated to 209.165.202.130.

Figure 26-3 Policy NAT with Different Destination Addresses

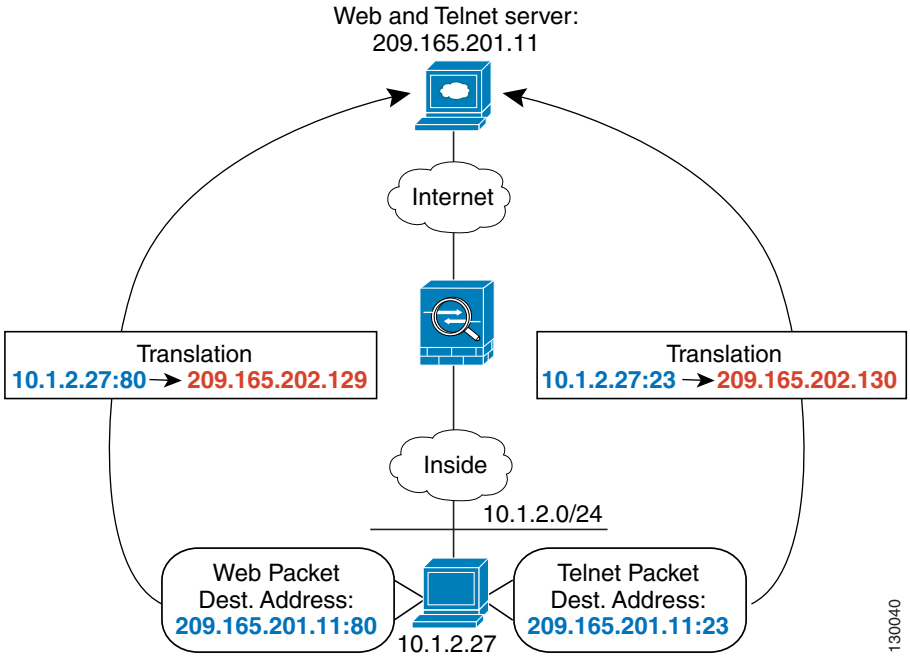


See the following commands for this example:

```
hostname(config)# access-list NET1 permit ip 10.1.2.0 255.255.255.0 209.165.201.0
255.255.255.224
hostname(config)# access-list NET2 permit ip 10.1.2.0 255.255.255.0 209.165.200.224
255.255.255.224
hostname(config)# nat (inside) 1 access-list NET1
hostname(config)# global (outside) 1 209.165.202.129
hostname(config)# nat (inside) 2 access-list NET2
hostname(config)# global (outside) 2 209.165.202.130
```

Figure 26-4 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 web services, the real address is translated to 209.165.202.129. When the host accesses the same server for Telnet services, the real address is translated to 209.165.202.130.

Figure 26-4 Policy NAT with Different Destination Ports



See the following commands for this example:

```
hostname(config)# access-list WEB permit tcp 10.1.2.0 255.255.255.0 209.165.201.11
255.255.255.255 eq 80
hostname(config)# access-list TELNET permit tcp 10.1.2.0 255.255.255.0 209.165.201.11
255.255.255.255 eq 23
hostname(config)# nat (inside) 1 access-list WEB
hostname(config)# global (outside) 1 209.165.202.129
hostname(config)# nat (inside) 2 access-list TELNET
hostname(config)# global (outside) 2 209.165.202.130
```

For policy static NAT (and for NAT exemption, which also uses an access list to identify traffic), you can initiate traffic to and from the real host. However, the destination address in the access list is only used for traffic initiated by the real host. For traffic *to* the real host from the destination network, the source address is not checked, and the first matching NAT rule for the real host address is used. So if you configure static policy NAT such as the following:

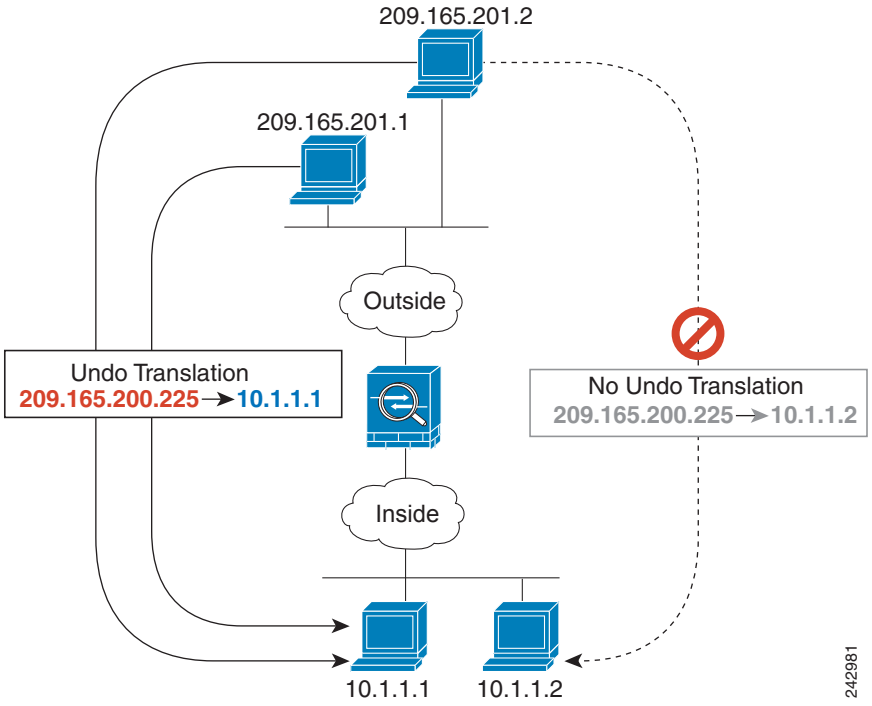
```
hostname(config)# access-list NET1 permit ip 10.1.2.0 255.255.255.224 209.165.201.0
```

```
255.255.255.224
hostname(config)# static (inside,outside) 209.165.202.128 access-list NET1
```

Then when hosts on the 10.1.2.0/27 network access 209.165.201.0/24, they are translated to corresponding addresses on the 209.165.202.128/27 network. But *any* host on the outside can access the mapped addresses 209.165.202.128/27, and not just hosts on the 209.165.201.0/24 network.

For the same reason (the source address is not checked for traffic *to* the real host), you cannot use policy static NAT to translate different real addresses to the same mapped address. For example, Figure 26-5 shows two inside hosts, 10.1.1.1 and 10.1.1.2, that you want to be translated to 209.165.200.225. When outside host 209.165.201.1 connects to 209.165.200.225, then the connection goes to 10.1.1.1. When outside host 209.165.201.2 connects to the same mapped address, 209.165.200.225, you want the connection to go to 10.1.1.2. However, because the destination address in the access list is not checked for traffic to the real host, then the first ACE that matches the real host is used. Since the first ACE is for 10.1.1.1, then all inbound connections sourced from 209.165.201.1 and 209.165.201.2 and destined to 209.165.200.255 will have their destination address translated to 10.1.1.1.

Figure 26-5 Real Addresses Cannot Share the Same Mapped Address



See the following commands for this example. (Although the second ACE in the example does allow 209.165.201.2 to connect to 209.165.200.225, it only allows 209.165.200.225 to be translated to 10.1.1.1.)

```
hostname(config)# static (in,out) 209.165.200.225 access-list policy-nat
hostname(config)# access-list policy-nat permit ip host 10.1.1.1 host 209.165.201.1
hostname(config)# access-list policy-nat permit ip host 10.1.1.2 host 209.165.201.2
```

Note Policy NAT does not support SQL*Net, but it is supported by regular NAT. See the "When to Use Application Protocol Inspection" section on page 40-2 for information about NAT support for other protocols.

NAT and Same Security Level Interfaces

NAT is not required between same security level interfaces even if you enable NAT control. You can optionally configure NAT if desired. However, if you configure dynamic NAT when NAT control is enabled, then NAT is required. See Chapter 27, "Configuring NAT Control," for more information. Also, when you specify a group of IP addresses for dynamic NAT or PAT on a same security interface, then you must perform NAT on that group of addresses when they access any lower or same security level interface (even when NAT control is not enabled). Traffic identified for static NAT is not affected.

See the "Allowing Same Security Level Communication" section on page 6-30 to enable same security communication.

Note The ASA does not support VoIP inspection engines when you configure NAT on same security interfaces. These inspection engines include Skinny, SIP, and H.323. See the "When to Use Application Protocol Inspection" section on page 40-2 for supported inspection engines.

Order of NAT Commands Used to Match Real Addresses

The ASA matches real addresses to NAT commands in the following order:

1. NAT exemption (**nat 0 access-list**)—In order, until the first match. Identity NAT is not included in this category; it is included in the regular static NAT or regular NAT category. We do not recommend overlapping addresses in NAT exemption statements because unexpected results can occur.
2. Static NAT and Static PAT (regular and policy) (**static**)—In order, until the first match. Static identity NAT is included in this category.
3. Policy dynamic NAT (**nat access-list**)—In order, until the first match. Overlapping addresses are allowed.
4. Regular dynamic NAT (**nat**)—Best match. Regular identity NAT is included in this category. The order of the NAT commands does not matter; the NAT statement that best matches the real address is used. For example, you can create a general statement to translate all addresses (0.0.0.0) on an interface. If you want to translate a subset of your network (10.1.1.1) to a different address, then you can create a statement to translate only 10.1.1.1. When 10.1.1.1 makes a connection, the specific statement for 10.1.1.1 is used because it matches the real address best. We do not recommend using overlapping statements; they use more memory and can slow the performance of the ASA.

Mapped Address Guidelines

When you translate the real address to a mapped address, you can use the following mapped addresses:

- Addresses on the same network as the mapped interface.

  If you use addresses on the same network as the mapped interface (through which traffic exits the ASA), the ASA uses proxy ARP to answer any requests for mapped addresses, and thus it intercepts traffic destined for a real address. This solution simplifies routing because the ASA does not have to be the gateway for any additional networks. However, this approach does put a limit on the number of available addresses used for translations.

  For PAT, you can even use the IP address of the mapped interface.

- Addresses on a unique network.

If you need more addresses than are available on the mapped interface network, you can identify addresses on a different subnet. The ASA uses proxy ARP to answer any requests for mapped addresses, and thus it intercepts traffic destined for a real address. If you use OSPF to advertise mapped IP addresses that belong to a different subnet from the mapped interface, you need to create a static route to the mapped addresses that are destined to the mapped interface IP, and then redistribute this static route in OSPF. If the mapped interface is passive (not advertising routes) or you are using static routing, then you need to add a static route on the upstream router that sends traffic destined for the mapped addresses to the ASA.

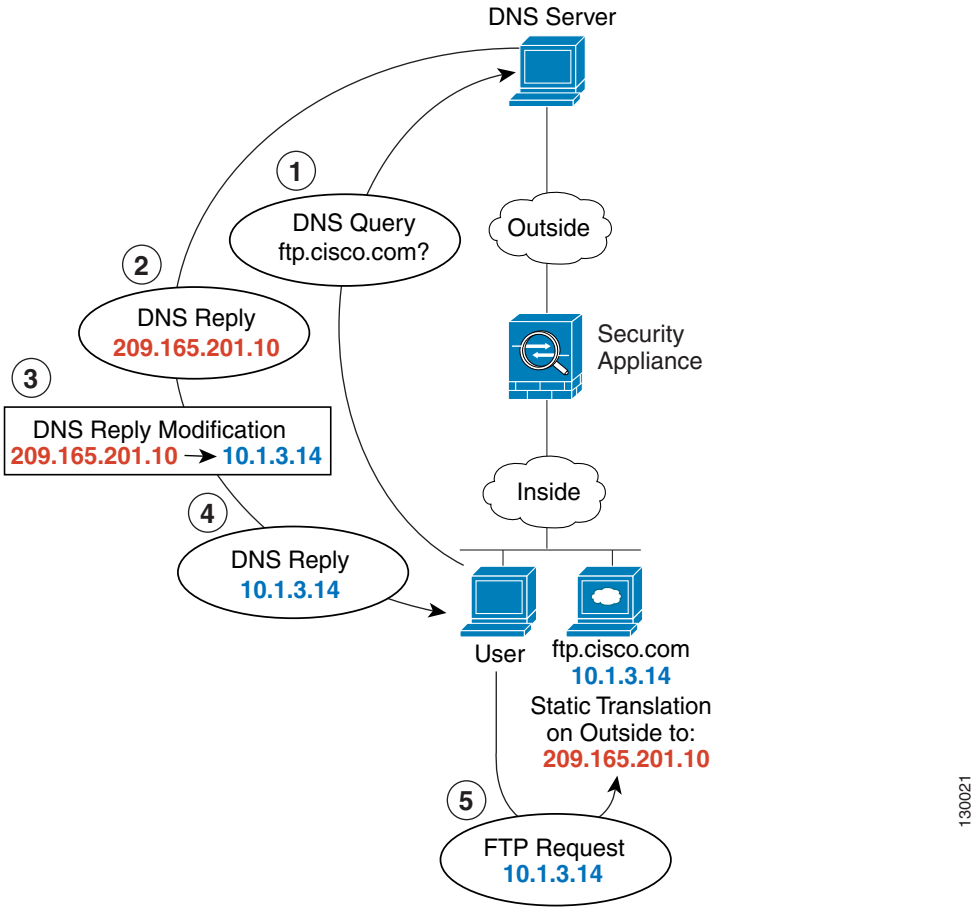
DNS and NAT

You might need to configure the ASA to modify DNS replies by replacing the address in the reply with an address that matches the NAT configuration. You can configure DNS modification when you configure each translation.

For example, a DNS server is accessible from the outside interface. A server, ftp.cisco.com, is on the inside interface. You configure the ASA to statically translate the ftp.cisco.com real address (10.1.3.14) to a mapped address (209.165.201.10) that is visible on the outside network. (See Figure 26-6.) In this case, you want to enable DNS reply modification on this static statement so that inside users who have access to ftp.cisco.com using the real address receive the real address from the DNS server, and not the mapped address.

When an inside host sends a DNS request for the address of ftp.cisco.com, the DNS server replies with the mapped address (209.165.201.10). The ASA refers to the static statement for the inside server and translates the address inside the DNS reply to 10.1.3.14. If you do not enable DNS reply modification, then the inside host attempts to send traffic to 209.165.201.10 instead of accessing ftp.cisco.com directly.

Figure 26-6 DNS Reply Modification



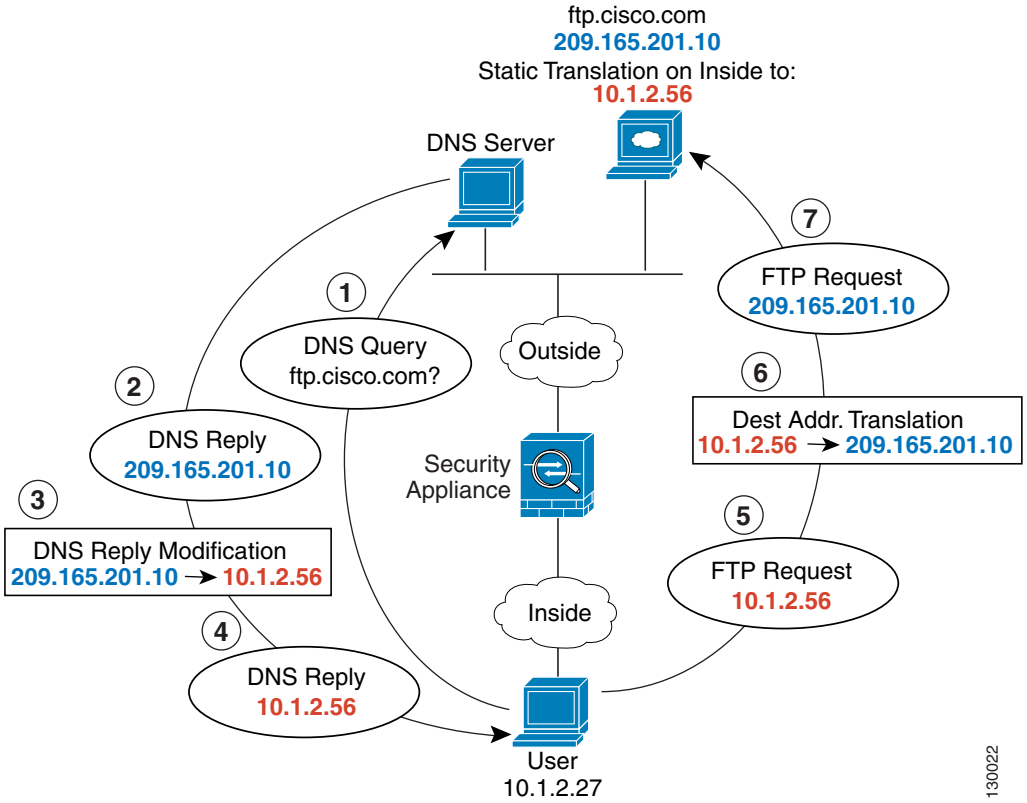
See the following command for this example:

```
hostname(config)# static (inside,outside) 209.165.201.10 10.1.3.14 netmask 255.255.255.255
dns
```

Note If a user on a different network (for example, DMZ) also requests the IP address for ftp.cisco.com from the outside DNS server, then the IP address in the DNS reply is also modified for this user, even though the user is not on the Inside interface referenced by the **static** command.

Figure 26-7 shows a web server and DNS server on the outside. The ASA has a static translation for the outside server. In this case, when an inside user requests the address for ftp.cisco.com from the DNS server, the DNS server responds with the real address, 209.165.20.10. Because you want inside users to use the mapped address for ftp.cisco.com (10.1.2.56) you need to configure DNS reply modification for the static translation.

Figure 26-7 DNS Reply Modification Using Outside NAT



See the following command for this example:

```
hostname(config)# static (outside,inside) 10.1.2.56 209.165.201.10 netmask 255.255.255.255
dns
```

Where to Go Next

- Chapter 27, “Configuring NAT Control”
- Chapter 29, “Configuring Dynamic NAT and PAT”
- Chapter 28, “Configuring Static NAT”
- Chapter 30, “Configuring Static PAT”
- Chapter 31, “Bypassing NAT”