NAT Examples and Reference

The following topics provide examples for configuring NAT, plus information on advanced configuration and troubleshooting.

- Examples for Network Object NAT, page 5-1
- Examples for Twice NAT, page 5-6
- NAT in Routed and Transparent Mode, page 5-9
- Routing NAT Packets, page 5-11
- NAT for VPN, page 5-15
- DNS and NAT, page 5-21

Examples for Network Object NAT

Following are some configuration examples for network object NAT.

- Providing Access to an Inside Web Server (Static NAT), page 5-1
- NAT for Inside Hosts (Dynamic NAT) and NAT for an Outside Web Server (Static NAT), page 5-2
- Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many), page 5-4
- Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation), page 5-5

Providing Access to an Inside Web Server (Static NAT)

The following example performs static NAT for an inside web server. The real address is on a private network, so a public address is required. Static NAT is necessary so hosts can initiate traffic to the web server at a fixed address.
Figure 5-1  Static NAT for an Inside Web Server

Procedure

**Step 1**  Create a network object for the internal web server.
```
hostname(config)# object network myWebServ
hostname(config-network-object)# host 10.1.2.27
```

**Step 2**  Configure static NAT for the object:
```
hostname(config-network-object)# nat (inside,outside) static 209.165.201.10
```

**NAT for Inside Hosts (Dynamic NAT) and NAT for an Outside Web Server (Static NAT)**

The following example configures dynamic NAT for inside users on a private network when they access the outside. Also, when inside users connect to an outside web server, that web server address is translated to an address that appears to be on the inside network.
**Figure 5-2 Dynamic NAT for Inside, Static NAT for Outside Web Server**

**Procedure**

**Step 1** Create a network object for the dynamic NAT pool to which you want to translate the inside addresses.

```bash
hostname(config)# object network myNatPool
hostname(config-network-object)# range 209.165.201.20 209.165.201.30
```

**Step 2** Create a network object for the inside network.

```bash
hostname(config)# object network myInsNet
hostname(config-network-object)# subnet 10.1.2.0 255.255.255.0
```

**Step 3** Enable dynamic NAT for the inside network using the dynamic NAT pool object.

```bash
hostname(config-network-object)# nat (inside,outside) dynamic myNatPool
```

**Step 4** Create a network object for the outside web server.

```bash
hostname(config)# object network myWebServ
hostname(config-network-object)# host 209.165.201.12
```

**Step 5** Configure static NAT for the web server.

```bash
hostname(config-network-object)# nat (outside,inside) static 10.1.2.20
```
Inside Load Balancer with Multiple Mapped Addresses (Static NAT, One-to-Many)

The following example shows an inside load balancer that is translated to multiple IP addresses. When an outside host accesses one of the mapped IP addresses, it is untranslated to the single load balancer address. Depending on the URL requested, it redirects traffic to the correct web server.

**Figure 5-3  Static NAT with One-to-Many for an Inside Load Balancer**

**Procedure**

**Step 1**  Create a network object for the addresses to which you want to map the load balancer.

```
hostname(config)# object network myPublicIPs
hostname(config-network-object)# range 209.165.201.3 209.265.201.8
```

**Step 2**  Create a network object for the load balancer.

```
hostname(config)# object network myLBHost
hostname(config-network-object)# host 10.1.2.27
```

**Step 3**  Configure static NAT for the load balancer applying the range object.

```
hostname(config-network-object)# nat (inside,outside) static myPublicIPs
```
Single Address for FTP, HTTP, and SMTP (Static NAT-with-Port-Translation)

The following static NAT-with-port-translation example provides a single address for remote users to access FTP, HTTP, and SMTP. These servers are actually different devices on the real network, but for each server, you can specify static NAT-with-port-translation rules that use the same mapped IP address, but different ports.

**Figure 5-4 Static NAT-with-Port-Translation**

**Step 1** Create a network object for the FTP server and configure static NAT with port translation, mapping the FTP port to itself.

```plaintext
hostname(config)# object network FTP_SERVER
hostname(config-network-object)# host 10.1.2.27
hostname(config-network-object)# nat (inside,outside) static 209.165.201.3 service tcp ftp ftp
```

**Step 2** Create a network object for the HTTP server and configure static NAT with port translation, mapping the HTTP port to itself.

```plaintext
hostname(config)# object network HTTP_SERVER
hostname(config-network-object)# host 10.1.2.28
hostname(config-network-object)# nat (inside,outside) static 209.165.201.3 service tcp http http
```

**Step 3** Create a network object for the SMTP server and configure static NAT with port translation, mapping the SMTP port to itself.

```plaintext
hostname(config)# object network SMTP_SERVER
hostname(config-network-object)# host 10.1.2.29
hostname(config-network-object)# nat (inside,outside) static 209.165.201.3 service tcp smtp smtp
```
Examples for Twice NAT

This section includes the following configuration examples:

- Different Translation Depending on the Destination (Dynamic Twice PAT), page 5-6
- Different Translation Depending on the Destination Address and Port (Dynamic PAT), page 5-7
- Example: Twice NAT with Destination Address Translation, page 5-9

Different Translation Depending on the Destination (Dynamic Twice PAT)

The following figure 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 5-5 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.

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)

The following figure 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.
**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 209.165.202.129
```

**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.

**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:

\[\text{hostname(config)}# \text{object service HTTPObj}\]
\[\text{hostname(config-network-object)}# \text{service tcp destination eq http}\]

Step 8  Configure the second twice NAT rule:

\[\text{hostname(config)}# \text{nat (inside,outside) source dynamic myInsideNetwork PATaddress2 destination static TelnetWebServer TelnetWebServer service HTTPObj HTTPObj}\]

---

**Example: Twice NAT with Destination Address Translation**

The following figure shows a remote host connecting to a mapped host. The mapped host has a twice static NAT translation that translates the real address only for traffic to and from the 209.165.201.0/27 network. A translation does not exist for the 209.165.200.224/27 network, so the translated host cannot connect to that network, nor can a host on that network connect to the translated host.

![Twice Static NAT with Destination Address Translation](image.png)

---

**NAT in Routed and Transparent Mode**

You can configure NAT in both routed and transparent firewall mode. This section describes typical usage for each firewall mode.

- NAT in Routed Mode, page 5-10
- NAT in Transparent Mode, page 5-10
NAT in Routed Mode

The following figure shows a typical NAT example in routed mode, with a private network on the inside.

**Figure 5-8  NAT Example: Routed Mode**

1. When the inside host at 10.1.2.27 sends a packet to a web server, the real source address of the packet, 10.1.2.27, is changed to a mapped address, 209.165.201.10.
2. When the server responds, it sends the response to the mapped address, 209.165.201.10, and the ASA receives the packet because the ASA performs proxy ARP to claim the packet.
3. The ASA 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.

NAT in Transparent Mode

Using NAT in transparent mode eliminates the need for the upstream or downstream routers to perform NAT for their networks.

NAT in transparent mode has the following requirements and limitations:

- 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 ASA sends an ARP request to a host on the other side of the ASA, 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.
- Translating between IPv4 and IPv6 networks is not supported. Translating between two IPv6 networks, or between two IPv4 networks is supported.

The following figure 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.
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 to the ASA management IP address. See Mapped Addresses and Routing, page 5-12 for more information about required routes.

3. The ASA then undoes the translation of the mapped address, 209.165.201.15, back to the real address, 10.1.1.75. 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 for returning traffic, the ASA looks up the route in its routing table and sends the packet to the downstream router at 10.1.1.3 based on the ASA static route for 192.168.1.0/24. See Transparent Mode Routing Requirements for Remote Networks, page 5-14 for more information about required routes.

Routing NAT Packets

The ASA needs to be the destination for any packets sent to the mapped address. The ASA also needs to determine the egress interface for any packets it receives destined for mapped addresses. This section describes how the ASA handles accepting and delivering packets with NAT.

- Mapped Addresses and Routing, page 5-12
• Transparent Mode Routing Requirements for Remote Networks, page 5-14
• Determining the Egress Interface, page 5-14

Mapped Addresses and Routing

When you translate the real address to a mapped address, the mapped address you choose determines how to configure routing, if necessary, for the mapped address.

See additional guidelines about mapped IP addresses in Additional Guidelines for NAT, page 4-8.

The following topics explain the mapped address types:
• Addresses on the Same Network as the Mapped Interface, page 5-12
• Addresses on a Unique Network, page 5-12
• The Same Address as the Real Address (Identity NAT), page 5-13

Addresses on the Same Network as the Mapped Interface

If you use addresses on the same network as the mapped interface, the ASA uses proxy ARP to answer any ARP requests for the mapped addresses, thus intercepting traffic destined for a mapped address. This solution simplifies routing because the ASA does not have to be the gateway for any additional networks. This solution is ideal if the outside network contains an adequate number of free addresses, a consideration if you are using a 1:1 translation like dynamic NAT or static NAT. Dynamic PAT greatly extends the number of translations you can use with a small number of addresses, so even if the available addresses on the outside network is small, this method can be used. For PAT, you can even use the IP address of the mapped interface.

Note

If you configure the mapped interface to be any interface, and you specify a mapped address on the same network as one of the mapped interfaces, then if an ARP request for that mapped address comes in on a different interface, then you need to manually configure an ARP entry for that network on the ingress interface, specifying its MAC address (see the `arp` command). Typically, if you specify any interface for the mapped interface, then you use a unique network for the mapped addresses, so this situation would not occur.

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 upstream router needs a static route for the mapped addresses that points to the ASA. Alternatively for routed mode, you can configure a static route on the ASA for the mapped addresses using any IP address on the destination network as the gateway, and then redistribute the route using your routing protocol. For example, if you use NAT for the inside network (10.1.1.0/24) and use the mapped IP address 209.165.201.5, then you can configure the following static route that can be redistributed:

```
route inside 209.165.201.5 255.255.255.255 10.1.1.99
```

For transparent mode, if the real host is directly-connected, configure the static route on the upstream router to point to the ASA: specify the bridge group IP address. For remote hosts in transparent mode, in the static route on the upstream router, you can alternatively specify the downstream router IP address.
The Same Address as the Real Address (Identity NAT)

The default behavior for identity NAT has proxy ARP enabled, matching other static NAT rules. You can disable proxy ARP if desired. You can also disable proxy ARP for regular static NAT if desired, in which case you need to be sure to have proper routes on the upstream router.

Normally for identity NAT, proxy ARP is not required, and in some cases can cause connectivity issues. For example, if you configure a broad identity NAT rule for “any” IP address, then leaving proxy ARP enabled can cause problems for hosts on the network directly connected to the mapped interface. In this case, when a host on the mapped network wants to communicate with another host on the same network, then the address in the ARP request matches the NAT rule (which matches “any” address). The ASA will then proxy ARP for the address, even though the packet is not actually destined for the ASA. (Note that this problem occurs even if you have a twice NAT rule; although the NAT rule must match both the source and destination addresses, the proxy ARP decision is made only on the “source” address). If the ASA ARP response is received before the actual host ARP response, then traffic will be mistakenly sent to the ASA (see the following figure).

*Figure 5-10 Proxy ARP Problems with Identity NAT*

In rare cases, you need proxy ARP for identity NAT; for example for virtual Telnet. When using AAA for network access, a host needs to authenticate with the ASA using a service like Telnet before any other traffic can pass. You can configure a virtual Telnet server on the ASA to provide the necessary login. When accessing the virtual Telnet address from the outside, you must configure an identity NAT rule for the address specifically for the proxy ARP functionality. Due to internal processes for virtual Telnet, proxy ARP lets the ASA keep traffic destined for the virtual Telnet address rather than send the traffic out the source interface according to the NAT rule. (See the following figure).
Routing NAT Packets

Chapter 5      NAT Examples and Reference

Figure 5-11 Proxy ARP and Virtual Telnet

Transparent Mode Routing Requirements for Remote Networks

When you use NAT in transparent mode, some types of traffic require static routes. See the general operations configuration guide for more information.

Determining the Egress Interface

When the ASA receives traffic for a mapped address, the ASA untranslated the destination address according to the NAT rule, and then it sends the packet on to the real address. The ASA determines the egress interface for the packet in the following ways:

- Transparent mode—The ASA determines the egress interface for the real address by using the NAT rule; you must specify the source and destination interfaces as part of the NAT rule.
- Routed mode—The ASA determines the egress interface in one of the following ways:
  - You configure the interface in the NAT rule—The ASA uses the NAT rule to determine the egress interface. However, you have the option to always use a route lookup instead. In certain scenarios, a route lookup override is required; for example, see NAT and VPN Management Access, page 5-19.
  - You do not configure the interface in the NAT rule—The ASA uses a route lookup to determine the egress interface.

The following figure shows the egress interface selection method in routed mode. In almost all cases, a route lookup is equivalent to the NAT rule interface, but in some configurations, the two methods might differ.
NAT for VPN

The following topics explain NAT usage with the various types of VPN.

- NAT and Remote Access VPN, page 5-15
- NAT and Site-to-Site VPN, page 5-17
- NAT and VPN Management Access, page 5-19
- Troubleshooting NAT and VPN, page 5-21

NAT and Remote Access VPN

The following figure shows both an inside server (10.1.1.6) and a VPN client (209.165.201.10) accessing the Internet. Unless you configure split tunneling for the VPN client (where only specified traffic goes through the VPN tunnel), then Internet-bound VPN traffic must also go through the ASA. When the VPN traffic enters the ASA, the ASA decrypts the packet; the resulting packet includes the VPN client local address (10.3.3.10) as the source. For both inside and VPN client local networks, you need a public IP address provided by NAT to access the Internet. The below example uses interface PAT rules. To allow the VPN traffic to exit the same interface it entered, you also need to enable intra-interface communication (also known as “hairpin” networking).
The following figure shows a VPN client that wants to access an inside mail server. Because the ASA expects traffic between the inside network and any outside network to match the interface PAT rule you set up for Internet access, traffic from the VPN client (10.3.3.10) to the SMTP server (10.1.1.6) will be dropped due to a reverse path failure: traffic from 10.3.3.10 to 10.1.1.6 does not match a NAT rule, but returning traffic from 10.1.1.6 to 10.3.3.10 should match the interface PAT rule for outgoing traffic. Because forward and reverse flows do not match, the ASA drops the packet when it is received. To avoid this failure, you need to exempt the inside-to-VPN client traffic from the interface PAT rule by using an identity NAT rule between those networks. Identity NAT simply translates an address to the same address.
See the following sample NAT configuration for the above network:

```plaintext
! Enable hairpin for non-split-tunneled VPN client traffic:
same-security-traffic permit intra-interface

! Identify local VPN network, & perform object interface PAT when going to Internet:
object network vpn_local
  subnet 10.3.3.0 255.255.255.0
  nat (outside,outside) dynamic interface

! Identify inside network, & perform object interface PAT when going to Internet:
object network inside_nw
  subnet 10.1.1.0 255.255.255.0
  nat (inside,outside) dynamic interface

! Use twice NAT to pass traffic between the inside network and the VPN client without
! address translation (identity NAT):
nat (inside,outside) source static inside_nw inside_nw destination static vpn_local
  vpn_local
```

**NAT and Site-to-Site VPN**

The following figure shows a site-to-site tunnel connecting the Boulder and San Jose offices. For traffic that you want to go to the Internet (for example from 10.1.1.6 in Boulder to www.example.com), you need a public IP address provided by NAT to access the Internet. The below example uses interface PAT rules. However, for traffic that you want to go over the VPN tunnel (for example from 10.1.1.6 in Boulder to 10.2.2.78 in San Jose), you do not want to perform NAT; you need to exempt that traffic by creating an identity NAT rule. Identity NAT simply translates an address to the same address.
The following figure shows a VPN client connected to ASA1 (Boulder), with a Telnet request for a server (10.2.2.78) accessible over a site-to-site tunnel between ASA1 and ASA2 (San Jose). Because this is a hairpin connection, you need to enable intra-interface communication, which is also required for non-split-tunneled Internet-bound traffic from the VPN client. You also need to configure identity NAT between the VPN client and the Boulder & San Jose networks, just as you would between any networks connected by VPN to exempt this traffic from outbound NAT rules.

See the following sample NAT configuration for ASA1 (Boulder):

```plaintext
! Enable hairpin for VPN client traffic:
same-security-traffic permit intra-interface

! Identify local VPN network, & perform object interface PAT when going to Internet:
```
object network vpn_local
  subnet 10.3.3.0 255.255.255.0
  nat (outside,outside) dynamic interface

! Identify inside Boulder network, & perform object interface PAT when going to Internet:
object network boulder_inside
  subnet 10.1.1.0 255.255.255.0
  nat (inside,outside) dynamic interface

! Identify inside San Jose network for use in twice NAT rule:
object network sanjose_inside
  subnet 10.2.2.0 255.255.255.0

! Use twice NAT to pass traffic between the Boulder network and the VPN client without
! address translation (identity NAT):
  nat (inside,outside) source static boulder_inside boulder_inside destination static
  vpn_local vpn_local

! Use twice NAT to pass traffic between the Boulder network and San Jose without
! address translation (identity NAT):
  nat (inside,outside) source static boulder_inside boulder_inside destination static
  sanjose_inside sanjose_inside

! Use twice NAT to pass traffic between the VPN client and San Jose without
! address translation (identity NAT):
  nat (outside,outside) source static vpn_local vpn_local destination static sanjose_inside
  sanjose_inside

See the following sample NAT configuration for ASA2 (San Jose):

! Identify inside San Jose network, & perform object interface PAT when going to Internet:
object network sanjose_inside
  subnet 10.2.2.0 255.255.255.0
  nat (inside,outside) dynamic interface

! Identify inside Boulder network for use in twice NAT rule:
object network boulder_inside
  subnet 10.1.1.0 255.255.255.0

! Identify local VPN network for use in twice NAT rule:
object network vpn_local
  subnet 10.3.3.0 255.255.255.0

! Use twice NAT to pass traffic between the San Jose network and Boulder without
! address translation (identity NAT):
  nat (inside,outside) source static sanjose_inside sanjose_inside destination static
  boulder_inside boulder_inside

! Use twice NAT to pass traffic between the San Jose network and the VPN client without
! address translation (identity NAT):
  nat (inside,outside) source static sanjose_inside sanjose_inside destination static
  vpn_local vpn_local

---

**NAT and VPN Management Access**

When using VPN, you can allow management access to an interface other than the one from which you entered the ASA (see the `management-access` command). For example, if you enter the ASA from the outside interface, the management-access feature lets you connect to the inside interface using ASDM, SSH, Telnet, or SNMP; or you can ping the inside interface.
The following figure shows a VPN client Telnetting to the ASA inside interface. When you use a management-access interface, and you configure identity NAT according to NAT and Remote Access VPN, page 5-15 or NAT and Site-to-Site VPN, page 5-17, you must configure NAT with the route lookup option. Without route lookup, the ASA sends traffic out the interface specified in the NAT command, regardless of what the routing table says; in the below example, the egress interface is the inside interface. You do not want the ASA to send the management traffic out to the inside network; it will never return to the inside interface IP address. The route lookup option lets the ASA send the traffic directly to the inside interface IP address instead of to the inside network. For traffic from the VPN client to a host on the inside network, the route lookup option will still result in the correct egress interface (inside), so normal traffic flow is not affected. See the Determining the Egress Interface, page 5-14 for more information about the route lookup option.

Figure 5-17  VPN Management Access

1. Telnet request to ASA inside ifc; management-access config req'd
   
2. ASA decrypts packet; src address is now local address
   
3. Identity NAT between inside & VPN client NWs; route-lookup req'd
   
4. Telnet request to 10.1.1.1
   
5. Telnet response to VPN Client
   
6. Identity NAT

7. ASA encrypts packet; dst address is now real address

See the following sample NAT configuration for the above network:

! Enable hairpin for non-split-tunneled VPN client traffic:
  same-security-traffic permit intra-interface

! Enable management access on inside ifc:
  management-access inside

! Identify local VPN network, & perform object interface PAT when going to Internet:
  object network vpn_local
      subnet 10.3.3.0 255.255.255.0
      nat (outside,inside) dynamic interface

! Identify inside network, & perform object interface PAT when going to Internet:
  object network inside_nw
      subnet 10.1.1.0 255.255.255.0
      nat (inside,outside) dynamic interface
! Use twice NAT to pass traffic between the inside network and the VPN client without ! address translation (identity NAT), w/route-lookup:

```
nat (outside,inside) source static vpn_local vpn_local destination static inside_nw inside_nw route-lookup
```

**Troubleshooting NAT and VPN**

See the following monitoring tools for troubleshooting NAT issues with VPN:

- Packet tracer—When used correctly, a packet tracer shows which NAT rules a packet is hitting.
- `show nat detail`—Shows hit counts and untranslated traffic for a given NAT rule.
- `show conn all`—Lets you see active connections including to and from the box traffic.

To familiarize yourself with a non-working configuration vs. a working configuration, you can perform the following steps:

1. Configure VPN without identity NAT.
2. Enter `show nat detail` and `show conn all`.
3. Add the identity NAT configuration.
4. Repeat `show nat detail` and `show conn all`.

**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 rule.

This feature rewrites the address in DNS queries and replies that match a NAT rule (for example, the A record for IPv4, the AAAA record for IPv6, or the PTR record for reverse DNS queries). For DNS replies traversing from a mapped interface to any other interface, the record is rewritten from the mapped value to the real value. Inversely, for DNS replies traversing from any interface to a mapped interface, the record is rewritten from the real value to the mapped value.

Following are some limitations with DNS rewrite:

- DNS rewrite is not applicable for PAT because multiple PAT rules are applicable for each A-record, and the PAT rule to use is ambiguous.
- If you configure a twice NAT rule, you cannot configure DNS modification if you specify the source address as well as the destination address. These kinds of rules can potentially have a different translation for a single address when going to A vs. B. Therefore, the ASA cannot accurately match the IP address inside the DNS reply to the correct twice NAT rule; the DNS reply does not contain information about which source/destination address combination was in the packet that prompted the DNS request.
- DNS rewrite requires DNS application inspection to be enabled, which it is on by default. See DNS Inspection, page 7-1 for more information.
- DNS rewrite is actually done on the xlate entry, not the NAT rule. Thus, if there is no xlate for a dynamic rule, rewrite cannot be done correctly. The same problem does not occur for static NAT.
The following topics provide examples of DNS rewrite:

- DNS Reply Modification, DNS Server on Outside, page 5-22
- DNS Reply Modification, DNS Server, Host, and Server on Separate Networks, page 5-23
- DNS Reply Modification, DNS Server on Host Network, page 5-24
- DNS64 Reply Modification Using Outside NAT, page 5-25
- PTR Modification, DNS Server on Host Network, page 5-27

**DNS Reply Modification, DNS Server on Outside**

The following figure shows a DNS server that 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.

In this case, you want to enable DNS reply modification on this static rule 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 rule 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.
Chapter 5      NAT Examples and Reference

DNS and NAT

Figure 5-18  DNS Reply Modification, DNS Server on Outside

Procedure

Step 1  Create a network object for the FTP server.

```
hostname(config)# object network FTP_SERVER
hostname(config-network-object)# host 10.1.3.14
```

Step 2  Configure static NAT with DNS modification.

```
hostname(config-network-object)# nat (inside,outside) static 209.165.201.10 dns
```

DNS Reply Modification, DNS Server, Host, and Server on Separate Networks

The following figure shows a user on the inside network requesting the IP address for ftp.cisco.com, which is on the DMZ network, from an outside DNS server. The DNS server replies with the mapped address (209.165.201.10) according to the static rule between outside and DMZ even though the user is not on the DMZ network. The ASA translates the address inside the DNS reply to 10.1.3.14.
If the user needs to access ftp.cisco.com using the real address, then no further configuration is required. If there is also a static rule between the inside and DMZ, then you also need to enable DNS reply modification on this rule. The DNS reply will then be modified two times. In this case, the ASA again translates the address inside the DNS reply to 192.168.1.10 according to the static rule between inside and DMZ.

**Figure 5-19  DNS Reply Modification, DNS Server, Host, and Server on Separate Networks**

**DNS Reply Modification, DNS Server on Host Network**

The following figure shows an FTP 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 5-20  DNS Reply Modification, DNS Server on Host Network**

**Procedure**

**Step 1**  Create a network object for the FTP server.

```
hostname(config)# object network FTP_SERVER
hostname(config-network-object)# host 209.165.201.10
```

**Step 2**  Configure static NAT with DNS modification.

```
hostname(config-network-object)# nat (outside,inside) static 10.1.2.56 dns
```

**DNS64 Reply Modification Using Outside NAT**

The following figure shows an FTP server and DNS server on the outside IPv4 network. The ASA has a static translation for the outside server. In this case, when an inside IPv6 user requests the address for ftp.cisco.com from the DNS server, the DNS server responds with the real address, 209.165.200.225.

Because you want inside users to use the mapped address for ftp.cisco.com (2001:DB8::D1A5:C8E1) you need to configure DNS reply modification for the static translation. This example also includes a static NAT translation for the DNS server, and a PAT rule for the inside IPv6 hosts.
Figure 5-21  DNS64 Reply Modification Using Outside NAT

Procedure

Step 1  Create a network object for the FTP server and configure static NAT with DNS modification. Because this is a one-to-one translation, include the net-to-net option for NAT46.

```
hostname(config)# object network FTP_SERVER
hostname(config-network-object)# host 209.165.200.225
hostname(config-network-object)# nat (outside,inside) static 2001:DB8::D1A5:C8E1/128 net-to-net dns
```

Step 2  Create a network object for the DNS server and configure static NAT. Include the net-to-net option for NAT46.

```
hostname(config)# object network DNS_SERVER
hostname(config-network-object)# host 209.165.201.15
hostname(config-network-object)# nat (outside,inside) static 2001:DB8::D1A5:C8E1/128 net-to-net
```

Step 3  Configure an IPv4 PAT pool for translating the inside IPv6 network.

```
hostname(config)# object network IPv4_POOL
hostname(config-network-object)# range 203.0.113.1 203.0.113.254
```
Step 4  Create a network object for the inside IPv6 network, and configure dynamic NAT with a PAT pool.

```bash
hostname(config)# object network IPv6_INSIDE
hostname(config-network-object)# subnet 2001:DB8::/96
hostname(config-network-object)# nat (inside,outside) dynamic pat-pool IPv4_POOL
```

**PTR Modification, DNS Server on Host Network**

The following figure shows an FTP server and DNS server on the outside. The ASA has a static translation for the outside server. In this case, when an inside user performs a reverse DNS lookup for 10.1.2.56, the ASA modifies the reverse DNS query with the real address, and the DNS server responds with the server name, ftp.cisco.com.

*Figure 5-22  PTR Modification, DNS Server on Host Network*