

Configuring Network Address Translation

Two key problems facing the Internet are depletion of IP address space and scaling in routing. Network Address Translation (NAT) is a feature that allows the IP network of an organization to appear from the outside to use different IP address space than what it is actually using. Thus, NAT allows an organization with nonglobally routable addresses to connect to the Internet by translating those addresses into globally routable address space. NAT also allows a more graceful renumbering strategy for organizations that are changing service providers or voluntarily renumbering into classless interdomain routing (CIDR) blocks. NAT is also described in RFC 1631.

Beginning with Cisco IOS Release 12.1(5)T, NAT supports all H.225 and H.245 message types, including FastConnect and Alerting as part of the H.323 version 2 specification. Any product that makes use of these message types will be able to pass through a Cisco IOS NAT configuration without any static configuration. Full support for NetMeeting Directory (Internet Locator Service) is also provided through Cisco IOS NAT.

NAT Applications

NAT has several applications. Use it for the following purposes:

- You want to connect to the Internet, but not all your hosts have globally unique IP addresses. NAT enables private IP internetworks that use nonregistered IP addresses to connect to the Internet. NAT is configured on the router at the border of a stub domain (referred to as the *inside network*) and a public network such as the Internet (referred to as the *outside network*). NAT translates the internal local addresses to globally unique IP addresses before sending packets to the outside network.
- You must change your internal addresses. Instead of changing them, which can be a considerable amount of work, you can translate them by using NAT.
- You want to do basic load sharing of TCP traffic. You can map a single global IP address to many local IP addresses by using the TCP load distribution feature.

As a solution to the connectivity problem, NAT is practical only when relatively few hosts in a stub domain communicate outside of the domain at the same time. When this is the case, only a small subset of the IP addresses in the domain must be translated into globally unique IP addresses when outside communication is necessary, and these addresses can be reused when no longer in use.

Benefits

A significant advantage of NAT is that it can be configured without requiring changes to hosts or routers other than those few routers on which NAT will be configured. As discussed previously, NAT may not be practical if large numbers of hosts in the stub domain communicate outside of the domain. Furthermore, some applications use embedded IP addresses in such a way that it is impractical for a NAT device to translate. These applications may not work transparently or at all through a NAT device. NAT also hides the identity of hosts, which may be an advantage or a disadvantage.

A router configured with NAT will have at least one interface to the inside and one to the outside. In a typical environment, NAT is configured at the exit router between a stub domain and backbone. When a packet is leaving the domain, NAT translates the locally significant source address into a globally unique address. When a packet is entering the domain, NAT translates the globally unique destination address into a local address. If more than one exit point exists, each NAT must have the same translation table. If the software cannot allocate an address because it has run out of addresses, it drops the packet and sends an ICMP host unreachable packet.

A router configured with NAT must not advertise the local networks to the outside. However, routing information that NAT receives from the outside can be advertised in the stub domain as usual.

NAT Terminology

As mentioned previously, the term *inside* refers to those networks that are owned by an organization and that must be translated. Inside this domain, hosts will have addresses in the one address space, while on the outside, they will appear to have addresses in another address space when NAT is configured. The first address space is referred to as the *local* address space and the second is referred to as the *global* address space.

Similarly, *outside* refers to those networks to which the stub network connects, and which are generally not under the control of the organization. Hosts in outside networks can be subject to translation also, and can thus have local and global addresses.

To summarize, NAT uses the following definitions:

- Inside local address—The IP address that is assigned to a host on the inside network. The address is probably not a legitimate IP address assigned by the Network Information Center (NIC) or service provider.
- Inside global address—A legitimate IP address (assigned by the NIC or service provider) that represents one or more inside local IP addresses to the outside world.
- Outside local address—The IP address of an outside host as it appears to the inside network. Not necessarily a legitimate address, it was allocated from address space routable on the inside.
- Outside global address—The IP address assigned to a host on the outside network by the owner of the host. The address was allocated from globally routable address or network space.

NAT Configuration Task List

Before configuring any NAT translation, you must know your inside local addresses and inside global addresses. To configure NAT, perform the optional tasks described in the following sections:

- [Translating Inside Source Addresses](#) (Optional)
- [Overloading an Inside Global Address](#) (Optional)
- [Translating Overlapping Addresses](#) (Optional)
- [Providing TCP Load Distribution](#) (Optional)
- [Changing Translation Timeouts](#) (Optional)
- [Monitoring and Maintaining NAT](#) (Optional)
- [Deploying NAT Between an IP Phone and Cisco CallManager](#) (Optional)

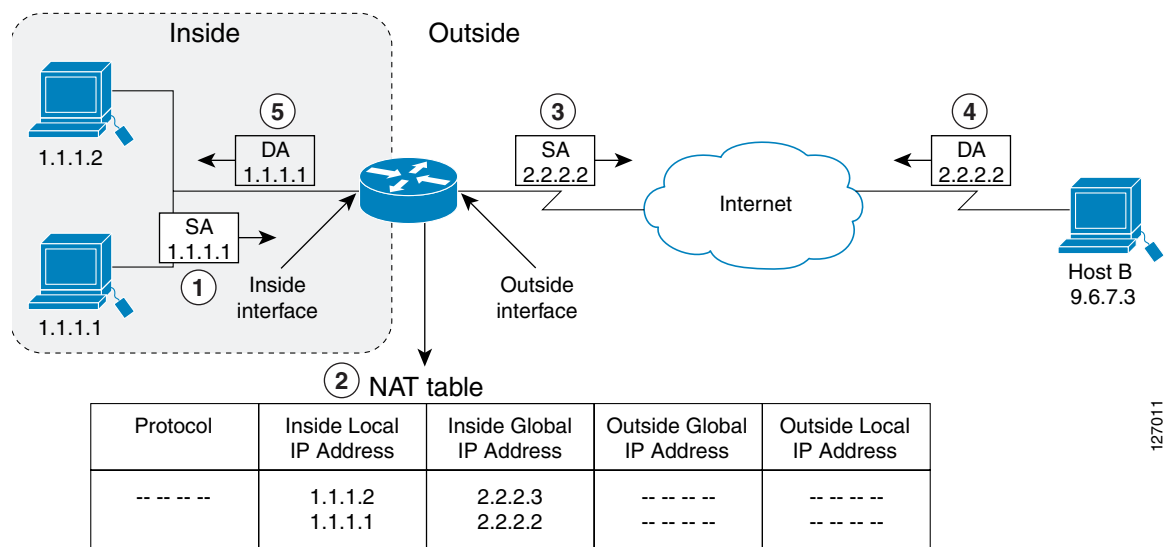
Translating Inside Source Addresses

You can translate your own IP addresses into globally unique IP addresses when communicating outside of your network. You can configure static or dynamic inside source translation as follows:

- *Static translation* establishes a one-to-one mapping between your inside local address and an inside global address. Static translation is useful when a host on the inside must be accessible by a fixed address from the outside.
- *Dynamic translation* establishes a mapping between an inside local address and a pool of global addresses. An access-list or a route-map can be specified for dynamic translations. Route maps allow you to match any combination of access-list, new-hop IP address, and output interface to determine which pool to use.

Figure 4 illustrates a router that is translating a source address inside a network to a source address outside the network.

Figure 4 NAT Inside Source Translation



The following process describes inside source address translation, as shown in [Figure 4](#):

1. The user at host 1.1.1.1 opens a connection to host B.
2. The first packet that the router receives from host 1.1.1.1 causes the router to check its NAT table:
 - If a static translation entry was configured, the router goes to Step 3.
 - If no translation entry exists, the router determines that Source-Address (SA) 1.1.1.1 must be translated dynamically, selects a legal, global address from the dynamic address pool, and creates a translation entry. This type of entry is called a *simple entry*.
3. The router replaces the inside local source address of host 1.1.1.1 with the global address of the translation entry and forwards the packet.
4. Host B receives the packet and responds to host 1.1.1.1 by using the inside global IP Destination-Address (DA) 2.2.2.2.
5. When the router receives the packet with the inside global IP address, it performs a NAT table lookup by using the inside global address as a key. It then translates the address to the inside local address of host 1.1.1.1 and forwards the packet to host 1.1.1.1.

Host 1.1.1.1 receives the packet and continues the conversation. The router performs Steps 2 through 5 for each packet.

Configuring Static Translation

To configure static inside source address translation, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip nat inside source static <i>local-ip global-ip</i>	Establishes static translation between an inside local address and an inside global address.
Step 2	Router(config)# interface <i>type number</i>	Specifies the inside interface and enters interface configuration mode.
Step 3	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 4	Router(config)# interface <i>type number</i>	Specifies the outside interface and enters interface configuration mode.
Step 5	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.

The previous steps are the minimum you must configure. You could also configure multiple inside and outside interfaces.

Configuring Dynamic Translation with an Access List

To configure dynamic inside source address translation with an access list, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip nat pool <i>name start-ip end-ip {netmask netmask prefix-length prefix-length}</i>	Defines a pool of global addresses to be allocated as needed.
Step 2	Router(config)# access-list <i>access-list-number permit source [source-wildcard]</i>	Defines a standard access list permitting those addresses that are to be translated.
Step 3	Router(config)# ip nat inside source list <i>access-list-number pool name</i>	Establishes dynamic source translation, specifying the access list defined in the prior step.
Step 4	Router(config)# interface <i>type number</i>	Specifies the inside interface and enters interface configuration mode.
Step 5	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 6	Router(config)# interface <i>type number</i>	Specifies the outside interface and enters interface configuration mode.
Step 7	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.



Note

The access list must permit only those addresses that are to be translated. (Remember that there is an implicit “deny all” at the end of each access list.) An access list that is too permissive can lead to unpredictable results.

Packets that enter the router through the inside interface and packets sourced from the router are checked against the access list for possible NAT candidates. The access list is used to specify which traffic is to be translated.

Configuring Dynamic Translation with a Route Map

To configure dynamic inside source address translation with a route map, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip nat pool <i>name start-ip end-ip {netmask netmask prefix-length prefix-length}</i>	Defines a pool of global addresses to be allocated as needed.
Step 2	Router(config)# route-map <i>name permit sequence</i>	Defines a route map permitting those addresses that are to be translated.
Step 3	Router(config)# ip nat inside source route-map <i>name pool name</i>	Establishes dynamic source translation, specifying the route map defined in the prior step.
Step 4	Router(config)# interface <i>type number</i>	Specifies the inside interface and enters interface configuration mode.
Step 5	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 6	Router(config)# interface <i>type number</i>	Specifies the outside interface and enters interface configuration mode.
Step 7	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.

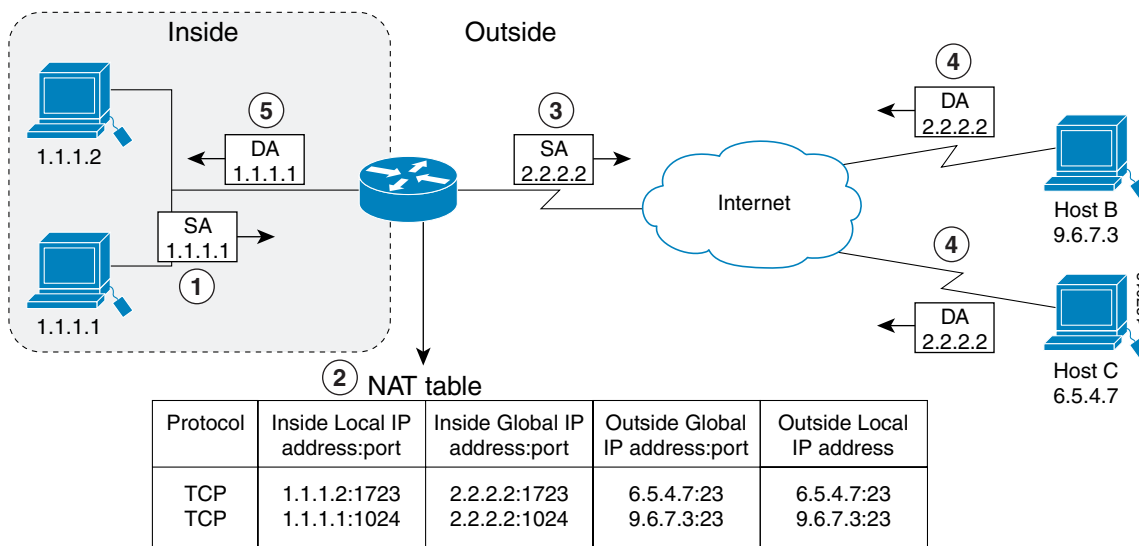
See the “[Dynamic Inside Source Translation Example](#)” section at the end of this chapter for examples of dynamic inside source translation.

Overloading an Inside Global Address

You can conserve addresses in the inside global address pool by allowing the router to use one global address for many local addresses. When this overloading is configured, the router maintains enough information from higher-level protocols (for example, TCP or UDP port numbers) to translate the global address back to the correct local address. When multiple local addresses map to one global address, the TCP or UDP port numbers of each inside host distinguish between the local addresses.

[Figure 5](#) illustrates NAT operation when one inside global address represents multiple inside local addresses. The TCP port numbers act as differentiators.

Figure 5 NAT Overloading Inside Global Addresses



The router performs the following process in overloading inside global addresses, as shown in Figure 5. Both host B and host C believe they are communicating with a single host at address 2.2.2.2. They are actually communicating with different hosts; the port number is the differentiator. In fact, many inside hosts could share the inside global IP address by using many port numbers.

1. The user at host 1.1.1.1 opens a connection to host B.
2. The first packet that the router receives from host 1.1.1.1 causes the router to check its NAT table:
 - If no translation entry exists, the router determines that address 1.1.1.1 must be translated, and sets up a translation of inside local address 1.1.1.1 to a legal global address.
 - If overloading is enabled, and another translation is active, the router reuses the global address from that translation and saves enough information to be able to translate back. This type of entry is called an *extended entry*.
3. The router replaces the inside local source address 1.1.1.1 with the selected global address and forwards the packet.
4. Host B receives the packet and responds to host 1.1.1.1 by using the inside global IP address 2.2.2.2.
5. When the router receives the packet with the inside global IP address, it performs a NAT table lookup, using the protocol, inside global address and port, and outside address and port as a key; translates the address to inside local address 1.1.1.1; and forwards the packet to host 1.1.1.1.

Host 1.1.1.1 receives the packet and continues the conversation. The router performs Steps 2 through 5 for each packet.

To configure overloading of inside global addresses, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip nat pool name start-ip end-ip {netmask netmask prefix-length prefix-length}	Defines a pool of global addresses to be allocated as needed.
Step 2	Router(config)# access-list access-list-number permit source [source-wildcard]	Defines a standard access list.

	Command	Purpose
Step 3	Router(config)# ip nat inside source list <i>access-list-number pool name overload</i>	Establishes dynamic source translation, specifying the access list defined in the prior step.
Step 4	Router(config)# interface <i>type number</i>	Specifies the inside interface.
Step 5	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 6	Router(config)# interface <i>type number</i>	Specifies the outside interface.
Step 7	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.

**Note**

The access list must permit only those addresses that are to be translated. (Remember that there is an implicit “deny all” at the end of each access list.) An access list that is too permissive can lead to unpredictable results.

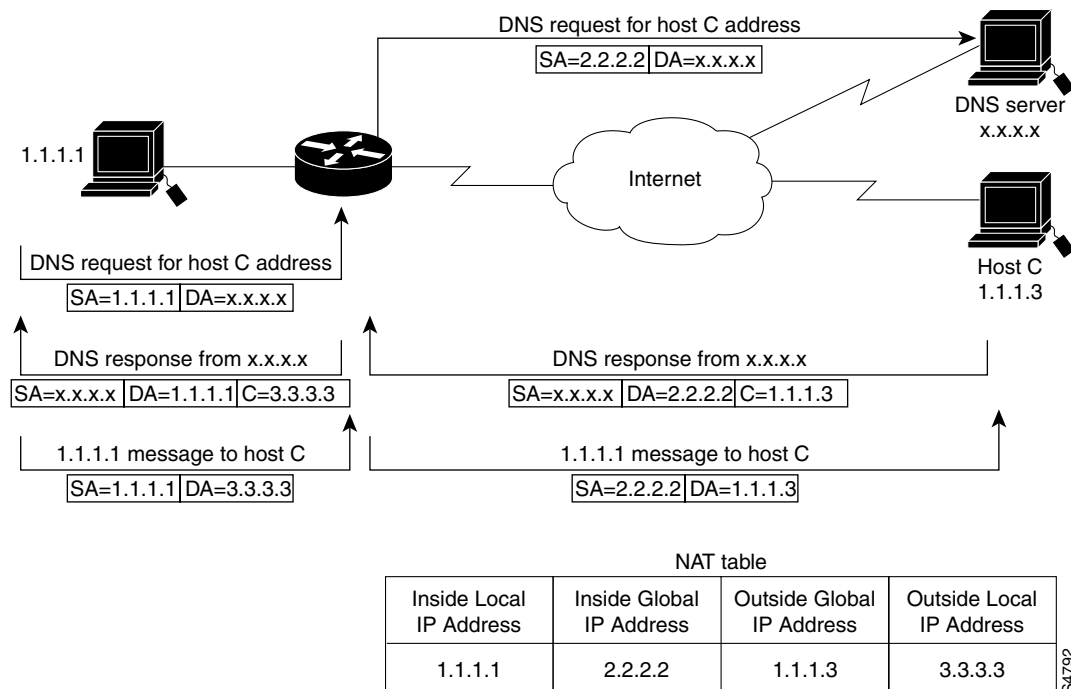
Packets that enter the router through the inside interface and packets sourced from the router are checked against the access list for possible NAT candidates. The access list is used to specify which traffic is to be translated.

See the “[Overloading Inside Global Addresses Example](#)” section at the end of this chapter for an example of overloading inside global addresses.

Translating Overlapping Addresses

The NAT overview discusses translating IP addresses, which could occur because your IP addresses are not legal, officially assigned IP addresses. Perhaps you chose IP addresses that officially belong to another network. The case of an address used both illegally and legally is called *overlapping*. You can use NAT to translate inside addresses that overlap with outside addresses. Use this feature if your IP addresses in the stub network are legitimate IP addresses belonging to another network, and you want to communicate with those hosts or routers.

[Figure 6](#) shows how NAT translates overlapping networks.

Figure 6 NAT Translating Overlapping Addresses

S4792

The router performs the following process when translating overlapping addresses:

1. The user at host 1.1.1.1 opens a connection to host C by name, requesting a name-to-address lookup from a DNS server.
2. The router intercepts the DNS reply and translates the returned address if there is an overlap (that is, the resulting legal address resides illegally in the inside network). To translate the return address, the router creates a simple translation entry mapping the overlapping address 1.1.1.3 to an address from a separately configured, outside local address pool.

The router examines every DNS reply from everywhere, ensuring that the IP address is not in the stub network. If it is, the router translates the address.
3. Host 1.1.1.1 opens a connection to 3.3.3.3.
4. The router sets up translations mapping inside local and global addresses to each other, and outside global and local addresses to each other.
5. The router replaces the SA with the inside global address and replaces the DA with the outside global address.
6. Host C receives the packet and continues the conversation.
7. The router does a lookup, replaces the DA with the inside local address, and replaces the SA with the outside local address.
8. Host 1.1.1.1 receives the packet and the conversation continues, using this translation process.

Configuring Static Translation

To configure static SA address translation, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip nat outside source static <i>global-ip local-ip</i>	Establishes static translation between an outside local address and an outside global address.
Step 2	Router(config)# interface <i>type number</i>	Specifies the inside interface.
Step 3	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 4	Router(config)# interface <i>type number</i>	Specifies the outside interface.
Step 5	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.

Configuring Dynamic Translation

To configure dynamic outside source address translation, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# ip nat pool <i>name start-ip end-ip</i> { netmask <i>netmask</i> prefix-length <i>prefix-length</i> }	Defines a pool of local addresses to be allocated as needed.
Step 2	Router(config)# access-list <i>access-list-number</i> permit <i>source</i> [<i>source-wildcard</i>]	Defines a standard access list.
Step 3	Router(config)# ip nat outside source list <i>access-list-number pool name</i>	Establishes dynamic outside source translation, specifying the access list defined in the prior step.
Step 4	Router(config)# interface <i>type number</i>	Specifies the inside interface.
Step 5	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 6	Router(config)# interface <i>type number</i>	Specifies the outside interface.
Step 7	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.



Note

The access list must permit only those addresses that are to be translated. (Remember that there is an implicit “deny all” at the end of each access list.) An access list that is too permissive can lead to unpredictable results.

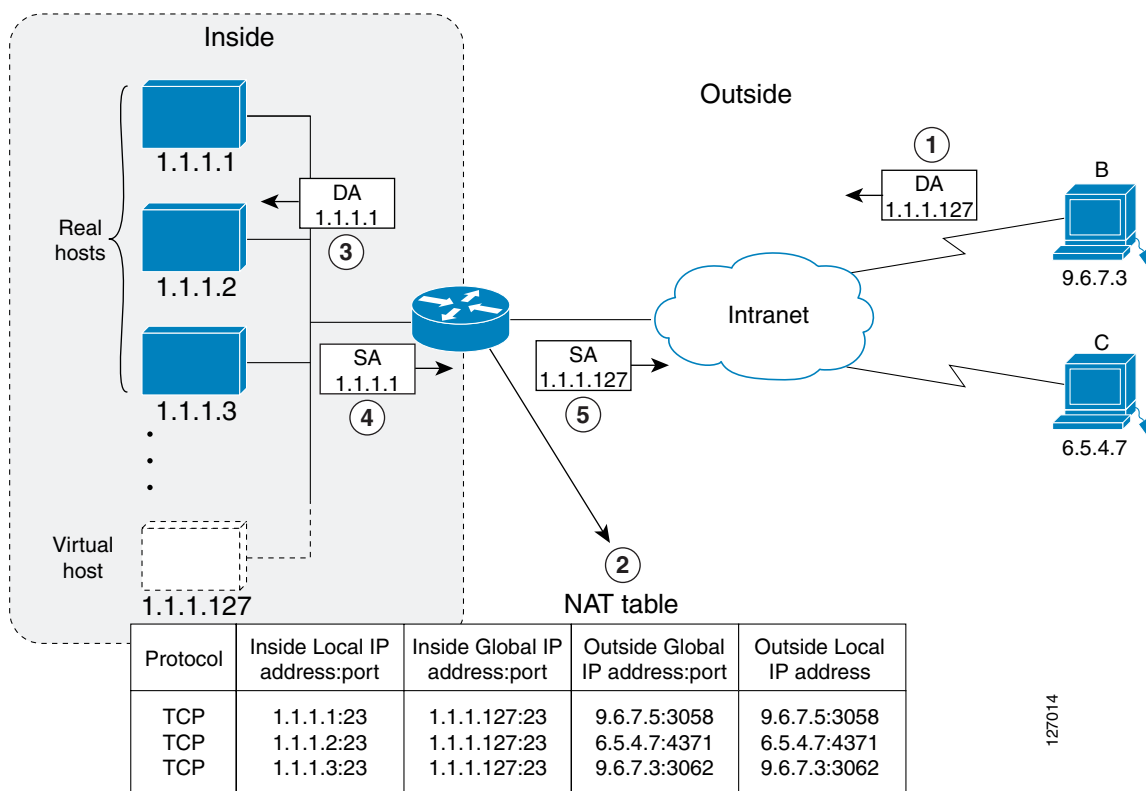
See the “[Translating Overlapping Address Example](#)” section at the end of this chapter for an example of translating an overlapping address.

Providing TCP Load Distribution

Another use of NAT is unrelated to Internet addresses. Your organization may have multiple hosts that must communicate with a heavily used host. Using NAT, you can establish a virtual host on the inside network that coordinates load sharing among real hosts. DAs that match an access list are replaced with

addresses from a rotary pool. Allocation is done on a round-robin basis, and only when a new connection is opened from the outside to the inside. Non-TCP traffic is passed untranslated (unless other translations are in effect). Figure 7 illustrates this feature.

Figure 7 NAT TCP Load Distribution



The router performs the following process when translating rotary addresses:

1. The user on host B (9.6.7.3) opens a connection to the virtual host at 1.1.1.127.
2. The router receives the connection request and creates a new translation, allocating the next real host (1.1.1.1) for the inside local IP address.
3. The router replaces the destination address with the selected real host address and forwards the packet.
4. Host 1.1.1.1 receives the packet and responds.
5. The router receives the packet, performs a NAT table lookup using the inside local address and port number, and the outside address and port number as the key. The router then translates the source address to the address of the virtual host and forwards the packet.

The next connection request will cause the router to allocate 1.1.1.2 for the inside local address.

To configure destination address rotary translation, use the following commands beginning in global configuration mode. These commands allow you to map one virtual host to many real hosts. Each new TCP session opened with the virtual host will be translated into a session with a different real host.

	Command	Purpose
Step 1	Router(config)# ip nat pool <i>name start-ip end-ip {netmask netmask prefix-length prefix-length} type rotary</i>	Defines a pool of addresses containing the addresses of the real hosts.
Step 2	Router(config)# access-list <i>access-list-number permit source [source-wildcard]</i>	Defines an access list permitting the address of the virtual host.
Step 3	Router(config)# ip nat inside destination list <i>access-list-number pool name</i>	Establishes dynamic inside destination translation, specifying the access list defined in the prior step.
Step 4	Router(config)# interface <i>type number</i>	Specifies the inside interface.
Step 5	Router(config-if)# ip nat inside	Marks the interface as connected to the inside.
Step 6	Router(config)# interface <i>type number</i>	Specifies the outside interface.
Step 7	Router(config-if)# ip nat outside	Marks the interface as connected to the outside.

**Note**

The access list must permit only those addresses that are to be translated. (Remember that there is an implicit “deny all” at the end of each access list.) An access list that is too permissive can lead to unpredictable results.

See the “[ping Command Example](#)” section at the end of this chapter for an example of rotary translation.

Changing Translation Timeouts

By default, dynamic address translations time out after some period of nonuse. You can change the default values on timeouts, if necessary. When overloading is not configured, simple translation entries time out after 24 hours. To change this value, use the following command in global configuration mode:

Command	Purpose
Router(config)# ip nat translation timeout <i>seconds</i>	Changes the timeout value for dynamic address translations that do not use overloading.

If you have configured overloading, you have more control over translation entry timeout, because each entry contains more context about the traffic using it. To change timeouts on extended entries, use the following commands in global configuration mode as needed:

Command	Purpose
Router(config)# ip nat translation udp-timeout <i>seconds</i>	Changes the UDP timeout value from 5 minutes.
Router(config)# ip nat translation dns-timeout <i>seconds</i>	Changes the DNS timeout value from 1 minute.
Router(config)# ip nat translation tcp-timeout <i>seconds</i>	Changes the TCP timeout value from 24 hours.
Router(config)# ip nat translation finrst-timeout <i>seconds</i>	Changes the Finish and Reset timeout value from 1 minute.

Command	Purpose
Router(config)# ip nat translation icmp-timeout <i>seconds</i>	Changes the ICMP timeout value from 1 minute.
Router(config)# ip nat translation syn-timeout <i>seconds</i>	Changes the Synchronous (SYN) timeout value from 1 minute.

Monitoring and Maintaining NAT

By default, dynamic address translations will time out from the NAT translation table at some point. To clear the entries before the timeout, use the following commands in EXEC mode as needed:

Command	Purpose
Router# clear ip nat translation *	Clears all dynamic address translation entries from the NAT translation table.
Router# clear ip nat translation inside <i>global-ip local-ip</i> [outside <i>local-ip global-ip</i>]	Clears a simple dynamic translation entry containing an inside translation, or both inside and outside translation.
Router# clear ip nat translation outside <i>local-ip global-ip</i>	Clears a simple dynamic translation entry containing an outside translation.
Router# clear ip nat translation protocol inside <i>global-ip global-port local-ip local-port</i> [outside <i>local-ip local-port global-ip global-port</i>]	Clears an extended dynamic translation entry.

To display translation information, use either of the following commands in EXEC mode:

Command	Purpose
Router# show ip nat translations [<i>verbose</i>]	Displays active translations.
Router# show ip nat statistics	Displays translation statistics.

Deploying NAT Between an IP Phone and Cisco CallManager

Cisco IP phones use the Selsius Skinny Station Protocol to connect with and register to the Cisco CallManager (CCM). Messages flow back and forth that include IP address and port information used to identify other IP phone users with which a call can be placed.

To be able to deploy Cisco IOS NAT between the IP phone and CCM in a scalable environment, NAT needs to be able to detect the Selsius Skinny Station Protocol and understand the information passed within the messages.

When an IP phone attempts to connect to the CCM and it matches the configured NAT translation rules, NAT will translate the original source IP address and replace it with one from the configured pool. This new address will be reflected in the CCM and be visible to other IP phone users.