

# Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Quick Start Steps for Configuring and Deploying NAT](#)

[Defining NAT Inside and Outside Interfaces](#)

[Example: Allowing Internal Users to Access the Internet](#)

[Configuring NAT to Allow Internal Users to Access the Internet](#)

[Configuring NAT to Allow Internal Users to Access the Internet Using Overloading](#)

[Example: Allowing the Internet to Access Internal Devices](#)

[Configuring NAT to Allow the Internet to Access Internal Devices](#)

[Example: Redirecting TCP Traffic to Another TCP Port or Address](#)

[Configuring NAT to Redirect TCP Traffic to Another TCP Port or Address](#)

[Example: Using NAT During a Network Transition](#)

[Configuring NAT for Use During a Network Transition](#)

[Example: Using NAT in Overlapping Networks](#)

[Difference between One-to-One Mapping and Many-to-Many](#)

[Verifying NAT Operation](#)

[Conclusion](#)

[Related Information](#)

[Related Cisco Support Community Discussions](#)

## [Introduction](#)

This document explains configuring Network Address Translation (NAT) on a Cisco router for use in common network scenarios. The target audience of this document is first time NAT users.

**Note:** In this document, when the internet, or an internet device is referred to, it means a device on any external network.

## [Prerequisites](#)

## [Requirements](#)

This document requires a basic knowledge of the terms used in connection with NAT. Some of the definitions can be found in [NAT: Local and Global Definitions](#).

## [Components Used](#)

The information in this document is based on these software and hardware versions:

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

## Conventions

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

## Quick Start Steps for Configuring and Deploying NAT

When you configure NAT, it is sometimes difficult to know where to begin, especially if you are new to NAT. These steps guide you to define what you want NAT to do and how to configure it:

### Define NAT inside and outside interfaces.

Do users exist off multiple interfaces?

Are there multiple interfaces going to the internet?

Define what you're trying to accomplish with NAT.

Are you trying to [allow internal users to access the internet](#)?

Are you trying to [allow the internet to access internal devices](#) (such as a mail server or web server)?

Are you trying to [redirect TCP traffic to another TCP port or address](#)?

Are you using [NAT during a network transition](#) (for example, you changed a server's IP address and until you can update all the clients you want the non-updated clients to be able to access the server using the original IP address as well as allow the updated clients to access the server using the new address)?

Are you using NAT to [allow overlapping networks to communicate](#)?

Configure NAT in order to accomplish what you defined above. Based on what you defined in step 2, you need determine which of the following features to use:

Static NAT

Dynamic NAT

Overloading

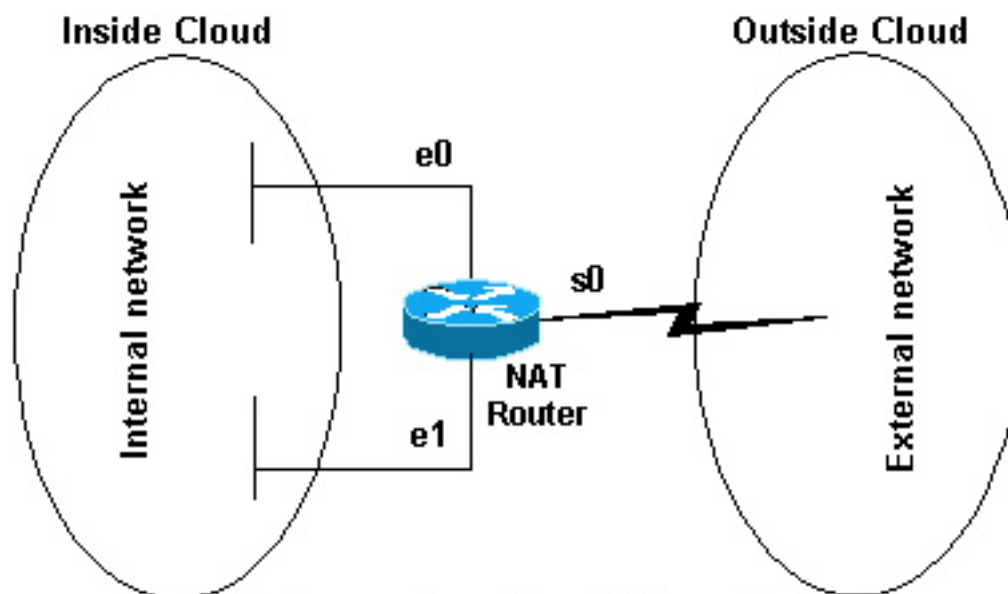
Any combination of the above

Verify the NAT operation.

Each of these NAT examples guides you through steps 1 through 3 of the Quick Start Steps above. These examples describe some common scenarios in which Cisco recommends you deploy NAT.

## Defining NAT Inside and Outside Interfaces

The first step to deploy NAT is to define NAT inside and outside interfaces. You may find it easiest to define your internal network as inside, and the external network as outside. However, the terms internal and external are subject to arbitration as well. This figure shows an example of this.

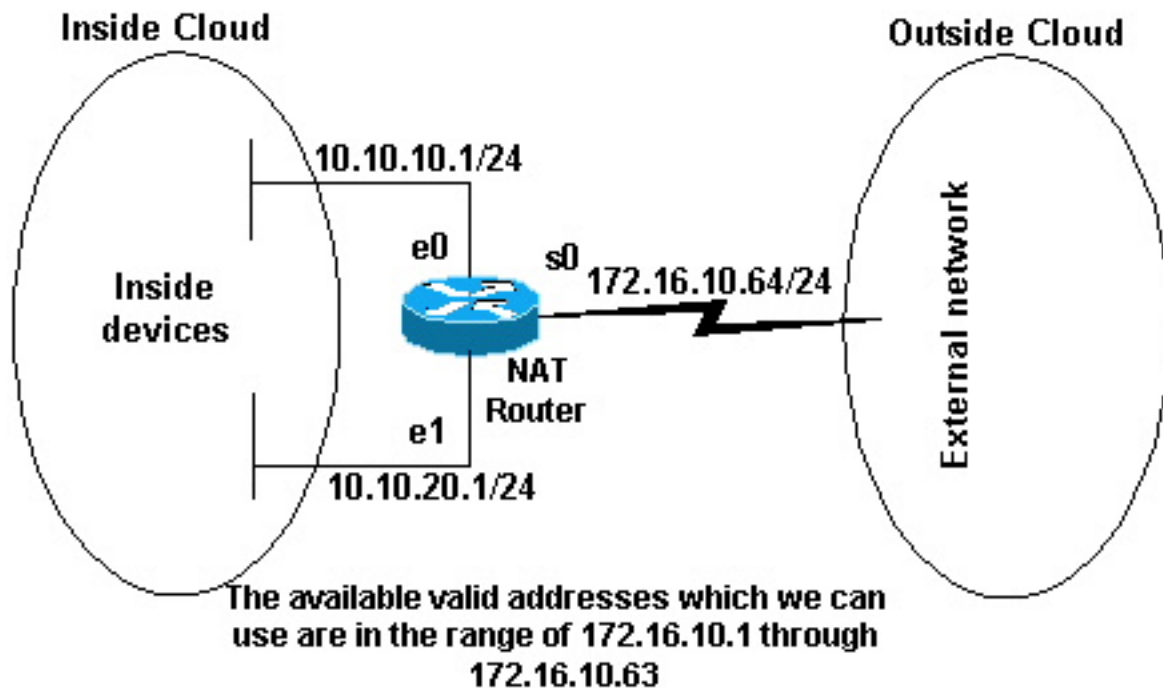


**In this figure, ethernet 0 and ethernet 1 will be defined as NAT inside interfaces and serial 0 will be defined as a NAT outside interface.**

## Example: Allowing Internal Users to Access the Internet

You may want to allow internal users to access the internet, but you may not have enough valid addresses to accommodate everyone. If all communication with devices in the internet originate from the internal devices, you need a single valid address or a pool of valid addresses.

This figure shows a simple network diagram with the router interfaces defined as inside and outside:



In this example, you want NAT to allow certain devices (the first 31 from each subnet) on the inside to originate communication with devices on the outside by translating their invalid address to a valid address or pool of addresses. The pool has been defined as the range of addresses 172.16.10.1 through 172.16.10.63.

Now you are ready to configure NAT. In order to accomplish what is defined above, use dynamic NAT. With dynamic NAT, the translation table in the router is initially empty and gets populated once traffic that needs to be translated passes through the router. As opposed to static NAT, where a translation is statically configured and is placed in the translation table without the need for any traffic.

In this example, you can configure NAT to translate each of the inside devices to a unique valid address, or to translate each of the inside devices to the same valid address. This second method is known as overloading. An example of how to configure each method is given here.

### [Configuring NAT to Allow Internal Users to Access the Internet](#)

NAT Router
------------

```
interface ethernet 0
 ip address 10.10.10.1 255.255.255.0
 ip nat inside
 !--- Defines Ethernet 0 with an IP address and as a NAT
 inside interface. interface ethernet 1 ip address
 10.10.20.1 255.255.255.0 ip nat inside !--- Defines
 Ethernet 1 with an IP address and as a NAT inside
 interface. interface serial 0 ip address 172.16.10.64
 255.255.255.0 ip nat outside !--- Defines serial 0 with
 an IP address and as a NAT outside interface. ip nat
 pool no-overload 172.16.10.1 172.16.10.63 prefix 24 ! !-
 -- Defines a NAT pool named no-overload with a range of
 addresses !--- 172.16.10.1 - 172.16.10.63. ip nat inside
 source list 7 pool no-overload ! !--- Indicates that
 any packets received on the inside interface that !---
 are permitted by access-list 7 has !--- the source
 address translated to an address out of the !--- NAT
 pool "no-overload". access-list 7 permit 10.10.10.0
 0.0.0.31 access-list 7 permit 10.10.20.0 0.0.0.31 !---
 Access-list 7 permits packets with source addresses
 ranging from !--- 10.10.10.0 through 10.10.10.31 and
 10.10.20.0 through 10.10.20.31.
```

**Note:** Cisco highly recommends that you do not configure access lists referenced by NAT commands with **permit any**. Using **permit any** can result in NAT consuming too many router resources which can cause network problems.

Notice in the previous configuration that only the first 32 addresses from subnet 10.10.10.0 and the first 32 addresses from subnet 10.10.20.0 are permitted by **access-list 7**. Therefore, only these source addresses are translated. There may be other devices with other addresses on the inside network, but these are not translated.

The final step is to [verify that NAT is operating as intended](#).

## [Configuring NAT to Allow Internal Users to Access the Internet Using Overloading](#)

**NAT Router**

```
interface ethernet 0
 ip address 10.10.10.1 255.255.255.0
 ip nat inside
 !--- Defines Ethernet 0 with an IP address and as a NAT
 inside interface. interface ethernet 1 ip address
 10.10.20.1 255.255.255.0 ip nat inside !--- Defines
 Ethernet 1 with an IP address and as a NAT inside
 interface. interface serial 0 ip address 172.16.10.64
 255.255.255.0 ip nat outside !--- Defines serial 0 with
 an IP address and as a NAT outside interface. ip nat
 pool ovrlld 172.16.10.1 172.16.10.1 prefix 24 ! !---
 Defines a NAT pool named ovrlld with a range of a single
 IP !--- address, 172.16.10.1. ip nat inside source list
 7 pool ovrlld overload ! ! ! ! !--- Indicates that any
 packets received on the inside interface that !--- are
 permitted by access-list 7 has the source address !---
 translated to an address out of the NAT pool named
 ovrlld. !--- Translations are overloaded, which allows
 multiple inside !--- devices to be translated to the
 same valid IP address. access-list 7 permit 10.10.10.0
 0.0.0.31 access-list 7 permit 10.10.20.0 0.0.0.31 !---
 Access-list 7 permits packets with source addresses
 ranging from !--- 10.10.10.0 through 10.10.10.31 and
 10.10.20.0 through 10.10.20.31.
```

Note in the previous second configuration, the NAT pool "ovrlld" only has a range of one address. The keyword **overload** used in the **ip nat inside source list 7 pool ovrlld overload** command allows NAT to translate multiple inside devices to the single address in the pool.

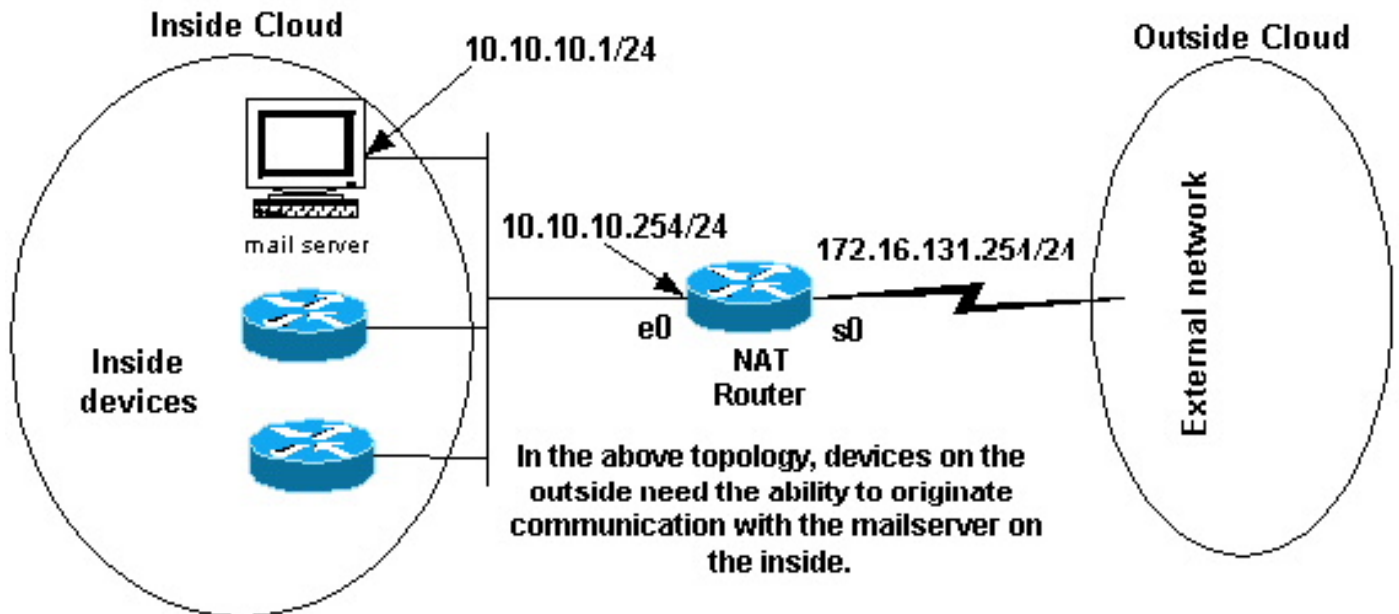
Another variation of this command is **ip nat inside source list 7 interface serial 0 overload**, which configures NAT to overload on the address that is assigned to the serial 0 interface.

When 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. For definitions of global and local address, refer to [NAT: Global and Local Definitions](#).

The final step is to [verify that NAT is operating as intended](#).

## [Example: Allowing the Internet to Access Internal Devices](#)

You may need internal devices to exchange information with devices on the internet, where the communication is initiated from the internet devices, for example, email. It is typical for devices on the internet to send email to a mail server that resides on the internal network.



## [Configuring NAT to Allow the Internet to Access Internal Devices](#)

In this example, you first define the NAT inside and outside interfaces, as shown in the previous network diagram.

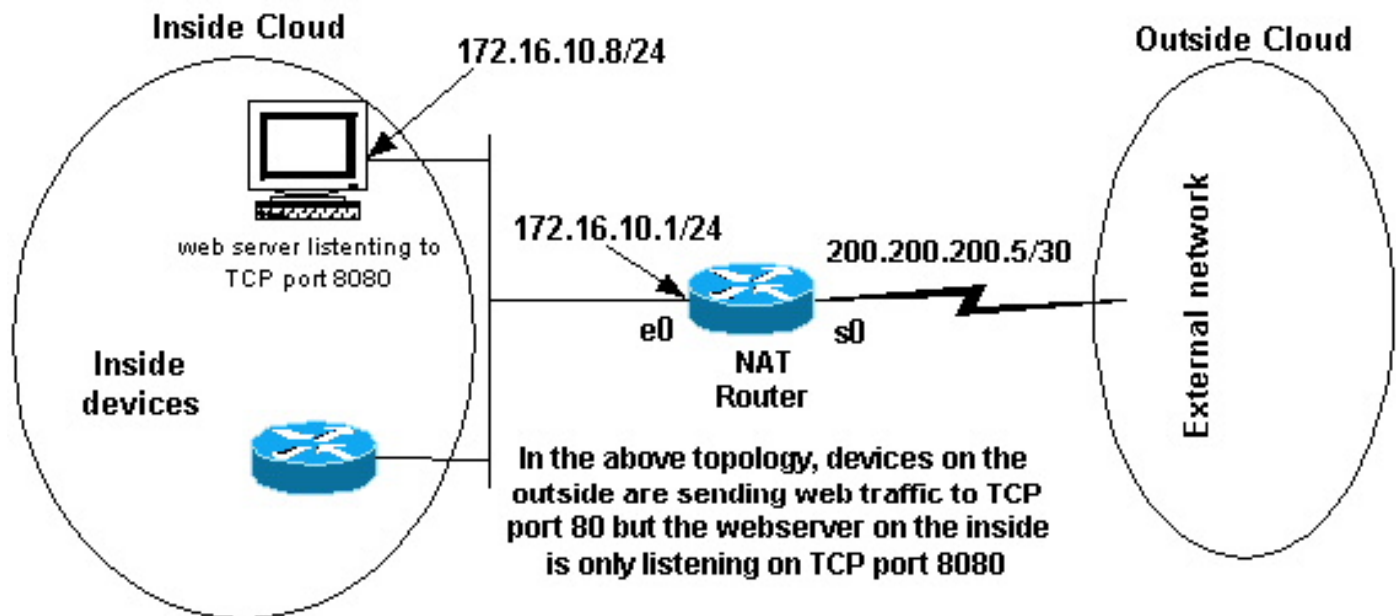
Second, you define that you want users on the inside to be able to originate communication with the outside. Devices on the outside should be able to originate communication with only the mail server on the inside.

The third step is to configure NAT. To accomplish what you have defined, you can configure static and dynamic NAT together. For more information on how to configure this example, refer to [Configuring Static and Dynamic NAT Simultaneously](#).

The final step is to [verify that NAT is operating as intended](#).

## [Example: Redirecting TCP Traffic to Another TCP Port or Address](#)

Having a web server on the internal network is another example of when it may be necessary for devices on the internet to initiate communication with internal devices. In some cases the internal web server may be configured to listen for web traffic on a TCP port other than port 80. For example, the internal web server may be configured to listen to TCP port 8080. In this case, you can use NAT to redirect traffic destined to TCP port 80 to TCP port 8080.



After you define the interfaces as shown in the previous network diagram, you may decide that you want NAT to redirect packets from the outside destined for 172.16.10.8:80 to 172.16.10.8:8080. You can use a **static nat** command in order to translate the TCP port number to achieve this. A sample configuration is shown here.

## [Configuring NAT to Redirect TCP Traffic to Another TCP Port or Address](#)

```

NAT Router
interface ethernet 0
 ip address 172.16.10.1 255.255.255.0
 ip nat inside
 !--- Defines Ethernet 0 with an IP address and as a NAT
 inside interface. interface serial 0 ip address
 200.200.200.5 255.255.255.252 ip nat outside !---
 Defines serial 0 with an IP address and as a NAT outside
 interface. ip nat inside source static tcp 172.16.10.8
 8080 172.16.10.8 80 !--- Static NAT command that states
 any packet received in the inside !--- interface with a
 source IP address of 172.16.10.8:8080 is !--- translated
 to 172.16.10.8:80.

```

Note that the configuration description for the static NAT command indicates any packet received in the inside interface with a source address of 172.16.10.8:8080 is translated to 172.16.10.8:80. This also implies that any packet received on the outside interface with a destination address of 172.16.10.8:80 has the destination translated to 172.16.10.8:8080.

The final step is to [verify that NAT is operating as intended](#).

```

show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
tcp 172.16.10.8:80     172.16.10.8:8080  ---               ---

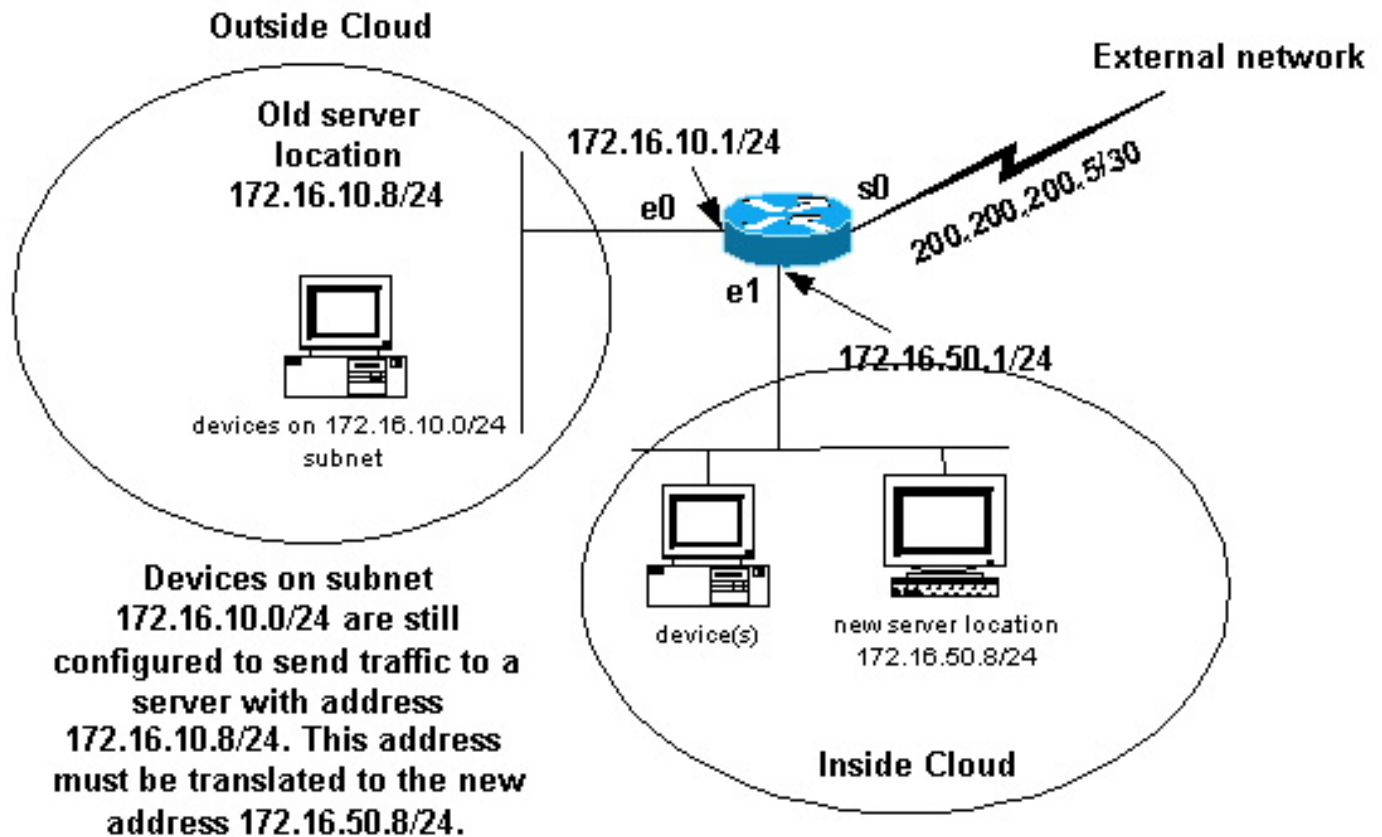
```

## [Example: Using NAT During a Network Transition](#)

Deploying NAT is useful when you need to readdress devices on the network or when you replace one device with another. For instance, if all devices in the network use a particular server and this



server needs to be replaced with a new one that has a new IP address, the reconfiguration of all the network devices to use the new server address takes some time. In the meantime, you can use NAT in order to configure the devices with the old address to translate their packets to communicate with the new server.



Once you have defined the NAT interfaces as shown above, you may decide that you want NAT to allow packets from the outside destined for the old server address (172.16.10.8) to be translated and sent to the new server address. Note that the new server is on another LAN, and devices on this LAN or any devices reachable through this LAN (devices on the inside part of the network), should be configured to use the new server's IP address if possible.

You can use static NAT to accomplish what you need. This is a sample configuration.

## [Configuring NAT for Use During a Network Transition](#)

```

NAT Router
interface ethernet 0
 ip address 172.16.10.1 255.255.255.0
 ip nat outside
 !--- Defines Ethernet 0 with an IP address and as a NAT
 outside interface. interface ethernet 1 ip address
 172.16.50.1 255.255.255.0 ip nat inside !--- Defines
 Ethernet 1 with an IP address and as a NAT inside
 interface. interface serial 0 ip address 200.200.200.5
 255.255.255.252 !--- Defines serial 0 with an IP
 address. This interface is not !--- participating in
 NAT. ip nat inside source static 172.16.50.8 172.16.10.8
 !--- States that any packet received on the inside
 interface with a !--- source IP address of 172.16.50.8
 is translated to 172.16.10.8.
  
```

Note that the inside source NAT command in this example also implies that packets received on the outside interface with a destination address of 172.16.10.8 has the destination address translated to 172.16.50.8.

The final step is to verify that [NAT is operating as intended](#).

## [Example: Using NAT in Overlapping Networks](#)

Overlapping networks result when you assign IP addresses to internal devices that are already being used by other devices within the internet. Overlapping networks also result when two companies, both of whom use [RFC 1918](#) IP addresses in their networks, merge. These two networks need to communicate, preferably without having to readdress all their devices. Refer to [Using NAT in Overlapping Networks](#) for more information on the configuration of NAT for this purpose.

## [Difference between One-to-One Mapping and Many-to-Many](#)

A static NAT configuration creates a one-to-one mapping and translates a specific address to another address. This type of configuration creates a permanent entry in the NAT table as long as the configuration is present and enables both inside and outside hosts to initiate a connection. This is mostly useful for hosts that provide application services like mail, web, FTP and so forth. For example:

```
Router(config)#ip nat inside source static 10.3.2.11 10.41.10.12 Router(config)#ip nat inside source static 10.3.2.12 10.41.10.13
```

Dynamic NAT is useful when fewer addresses are available than the actual number of hosts to be translated. It creates an entry in the NAT table when the host initiates a connection and establishes a one-to-one mapping between the addresses. But, the mapping can vary and it depends upon the registered address available in the pool at the time of the communication. Dynamic NAT allows sessions to be initiated only from inside or outside networks for which it is configured. Dynamic NAT entries are removed from the translation table if the host does not communicate for a specific period of time which is configurable. The address is then returned to the pool for use by another host.

For example, complete these steps of the detailed configuration:

Create a pool of addresses

```
Router(config)#ip nat pool MYPOOLEXAMPLE 10.41.10.1 10.41.10.41 netmask 255.255.255.0
```

Create an access-list for the inside networks that has to be mapped

```
Router(config)#access-list 100 permit ip 10.3.2.0 0.0.0.255 any
```

Associate the access-list 100 that is selecting the internal network 10.3.2.0 0.0.0.255 to be natted to the pool MYPOOLEXAMPLE and then overload the addresses.

```
Router(config)#ip nat inside source list 100 pool MYPOOLEXAMPLE overload
```

## [Verifying NAT Operation](#)

Once you've configured NAT, verify that it is operating as expected. You can do this in a number

of ways: using a network analyzer, **show** commands, or **debug** commands. For a detailed example of NAT verification, refer to [Verifying NAT Operation and Basic NAT Troubleshooting](#).

## Conclusion

The examples in this document demonstrate quick start steps can help you configure and deploy NAT. These quick start steps include:

Defining NAT inside and outside interfaces.

Defining what you are trying to accomplish with NAT.

Configuring NAT in order to accomplish what you defined in Step 2.

Verifying the NAT operation.

In each of the previous examples, various forms of the **ip nat inside** command were used. You can also use the **ip nat outside** command in order to accomplish the same objectives, but keep in mind the NAT order of operations. For configuration examples that use the **ip nat outside** commands, refer to [Sample Configuration Using the ip nat outside source list Command](#) and [Sample Configuration Using the ip nat outside source static Command](#).

The previous examples also demonstrated these actions:

Command	Action
<b>ip nat inside source</b>	<ul style="list-style-type: none"><li>• Translates the source of IP packets that are traveling inside to outside.</li><li>• Translates the destination of the IP packets that are traveling outside to inside.</li></ul>
<b>ip nat outside source</b>	<ul style="list-style-type: none"><li>• Translates the source of the IP packets that are traveling outside to inside.</li><li>• Translates the destination of the IP packets that are traveling inside to outside.</li></ul>

## Related Information

- [NAT Support Page](#)
- [IP Routed Protocols Support Page](#)
- [IP Routing Support Page](#)
- [How NAT Works](#)
- [NAT Order of Operation](#)
- [Frequently Asked Questions about Cisco IOS NAT](#)
- [Technical Support & Documentation - Cisco Systems](#)

