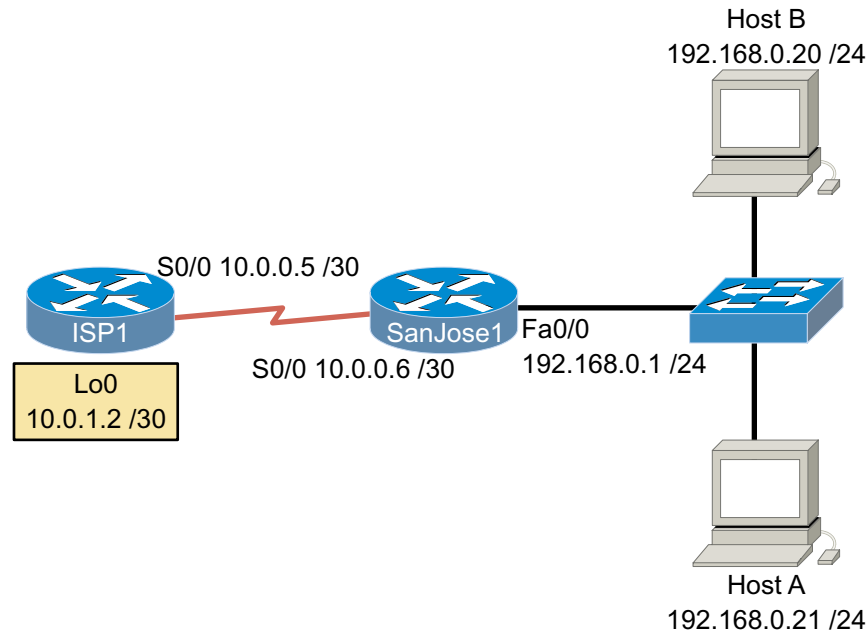


## 11.5.2 Configuring Dynamic NAT



### Objective

Configure dynamic NAT to provide privately addressed users with access to outside resources.

### Scenario

When the International Travel Agency (ITA) expanded and updated their network, they chose to use the 192.168.0.0 /24 private addresses and NAT to handle connectivity with the outside world. In securing the outside IP addresses from their ISP, ITA has to pay a monthly fee per IP address. ITA has asked you to set up a series of prototypes that would demonstrate NAT's capabilities to meet ITA's requirements. The company hopes to be able to get by with 14 real IP addresses (42.0.0.48 /28). For a variety of reasons including security concerns, the company wishes to hide the internal network from the outside.

ITA is hoping to limit user access to the Internet and other outside resources by limiting the number of connections. You have been asked to prototype the basic dynamic translation to see if it will meet ITA's objectives.

### Step 1

Build and configure the network according to the above diagram. This configuration requires the use of subnet zero, so you may need to enter the **ip subnet-zero** command, depending on the IOS version you are using. Both Host A and Host B represent users on the ITA network.

Configure SanJose1 to use a default route to ISP1:

```
SanJose1(config)#ip route 0.0.0.0 0.0.0.0 10.0.0.5
```

On ISP1, configure a static route to the global addresses used by SanJose1 for NAT:

```
ISP1(config)#ip route 42.0.0.48 255.255.255.240 10.0.0.6
```

## Step 2

Define a pool of global addresses to be allocated by the dynamic NAT process. Issue the following command on SanJose1:

```
SanJose1(config)#ip nat pool MYNATPOOL 42.0.0.55 42.0.0.55
netmask255.255.255.240
```

The name MYNATPOOL is the name of the address pool. If you prefer, you can use a different word of your choosing. The first 42.0.0.55 in the command is the first IP address in the pool. The second 42.0.0.55 is the last IP address in the pool. The above command creates a pool that contains only a single address. Typically, you will configure a larger range of addresses in a pool. For now, only one address will be used.

Next, you must configure a standard access list to define which internal source addresses can be translated. Since you are translating any users on the ITA network, use the following command:

```
SanJose1(config)#access-list 2 permit 192.168.0.0 0.0.0.255
```

To establish the dynamic source translation, you must link the access list to the name of the NAT pool, as shown here:

```
SanJose1(config)#ip nat inside source list 2 pool MYNATPOOL
```

Finally, specify an interface on SanJose1 to be used by inside network hosts requiring address translation:

```
SanJose1(config)#interface fastethernet0/0
SanJose1(config-if)#ip nat inside
```

You must also specify an interface to be used as the outside NAT interface:

```
SanJose1(config)#interface serial0/0
SanJose1(config-if)#ip nat outside
```

## Step 3

On SanJose1, enter the **show ip nat translations** command, which should result in no output. Unlike static translations, which are permanent and always remain in the translations table, dynamic translations are only assigned as needed, and only appear when active.

From Host A, **ping** ISP1's serial and loopback IP addresses. Both pings should work; troubleshoot as necessary.

Issue the **show ip nat translations** command on SanJose1 again. You should now get a single translation for that workstation. The result might look like this:

```
SanJose1#show ip nat trans
Pro Inside global      Inside local      Outside local      Outside global
--- 42.0.0.55          192.168.0.21      ---               ---
```

From Host B, **ping** ISP1's serial and loopback IP addresses. They should both fail. The one available IP address in the pool is being used by the other workstation. If you had assigned a larger pool of addresses, Host B could be assigned an address from the pool.

#### Step 4

Issue the **show ip nat translations verbose** command and examine the output:

```
SanJose1#show ip nat translations verbose
Pro Inside global      Inside local      Outside local      Outside global
--- 42.0.0.55          192.168.0.21      ---              ---
    create 00:13:18, use 00:13:06, left 23:46:53,
    flags: none, use_count: 0
```

1. According to the output of this command, how much time is left before the dynamic translation times out?

---

The default timeout value for dynamic NAT translations is 24 hours. This means the second workstation will have to wait until the next day before it can be assigned the address.

Next, issue the **show ip nat statistics** command. Notice that it summarizes the translation information, shows the pool of global addresses, and indicates that only one address has been allocated (translated), as shown here:

```
SanJose1#show ip nat statistics
Total active translations: 1 (0 static, 1 dynamic; 0 extended)
Outside interfaces:
  Serial0/0
Inside interfaces:
  FastEthernet0/0
Hits: 45 Misses: 0
Expired translations: 0
Dynamic mappings:
-- Inside Source
access-list 2 pool MYNATPOOL refcount 1
 pool MYNATPOOL: netmask 255.255.255.240
   start 42.0.0.55 end 42.0.0.55
   type generic, total addresses 1, allocated 1 (100%), misses 4
```

To change the default NAT timeout value from 24 hours (86,400 seconds) to 120 seconds, issue the following command:

```
SanJose1(config)#ip nat translation timeout 120
```

You must clear the existing address allocation before the new timer can take effect. Type **clear ip nat translation \*** to immediately clear the translation table.

Now, from Host B, try pinging either interface of ISP1 again. The ping should be successful.

Use the **show ip nat translations** and **show ip nat translations verbose** commands to confirm the translation and to see that the new translations expire in two minutes.

Next, perform a ping from Host B and issue the **show ip nat translations verbose** command again. You should see that the “time left” timer has been reset. This means that additional hosts will not be allocated an address until a translation has been inactive for the timeout period.

## Step 5

In this step, configure the NAT pool to include the complete range of global addresses available to ITA. Issue the following command on SanJose1:

```
SanJose1(config)#ip nat pool MYNATPOOL 42.0.0.55 42.0.0.62 netmask  
255.255.255.240
```

The preceding command redefines MYNATPOOL to include a range of eight addresses. You should now be able to **ping** ISP1 from both workstations.

The **show ip nat translations** command confirms that two translations have occurred, as shown here:

```
SanJose1#show ip nat translations  
Pro Inside global      Inside local      Outside local      Outside  
global  
--- 42.0.0.55          192.168.0.20      ---              ---  
--- 42.0.0.56          192.168.0.21      ---              ---
```

Increasing the address range in the pool allows more hosts to be translated. However, if every address in the pool is allocated, the timeout period must expire before any other hosts can be allocated an address. As you saw in the last step, an allocated address cannot be released until its host is inactive for the duration of the timeout period.

In the next lab, you will learn to use many-to-one NAT, or NAT overload. An overload configuration can allow hundreds of hosts to use a handful of global addresses, without hosts waiting for timeouts.