Computer Networks Laboratory

CSE 324 Lab Experiment # 7 Fall 2017

Network Address Translation (NAT)

1. Objectives

- To learn about **Network Address Translation (NAT)**: why and how used?
- To build an **internetwork** using **NAT** using **Packet Tracer**

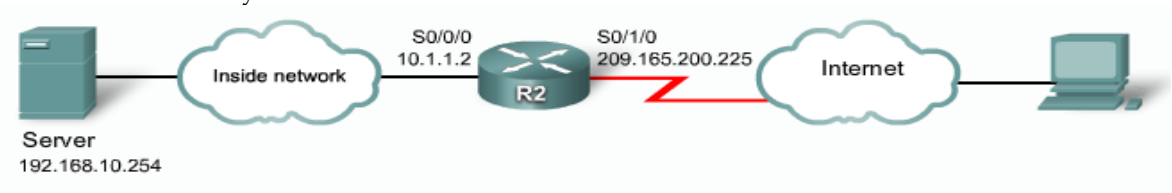
2. Background: NAT

NAT (Network Address Translation) is a technique for preserving scarce Internet IP addresses. It converts private IP addresses (not routable to Internet) to public IP addresses so that Internet can be accessed from private network.

Types of NAT

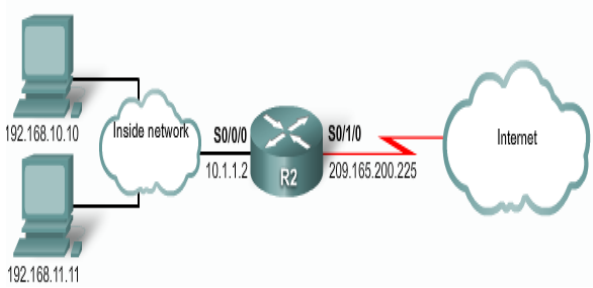
Developed by Cisco, Network Address Translation is used by a device (firewall, router or computer) that sits between an internal network and the rest of the world. NAT has many forms and can work in several ways:

- **Static NAT -** Mapping an unregistered (private) IP address to a registered (public) IP address on a one-to-one basis. Particularly useful when a device needs to be accessible from outside the network.



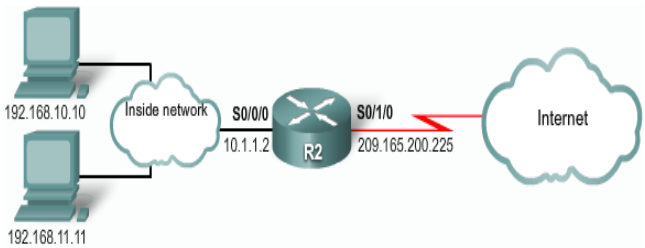

```
ip nat inside source static 192.168.10.254 209.165.200.254
!Establishes static translation between an inside local address and an inside global address.
interface serial 0/0/0
ip nat inside
!Identifies Serial 0/0/0 as an inside NAT interface.
interface serial 0/1/0
ip nat outside
!Identifies Serial 0/1/0 as an inside NAT interface.
```

- **Dynamic NAT** - Maps an unregistered IP address to a registered IP address from a group of registered IP addresses.




```
ip nat pool NAT-POOL1 209.165.200.226 209.165.200.240 netmask 255.255.255.224
!Defines a pool of public IP addresses under the pool name NAT-POOL1
access-list 1 permit 192.168.0.0 0.0.255.255
!Defines which addresses are eligible to be translated
ip nat inside source list 1 pool NAT-POOL1
!Binds the NAT pool with ACL 1
interface serial 0/0/0
  ip nat inside
!Identifies interface Serial 0/0/0 as an inside NAT interface
interface serial 0/1/0
  ip nat outside
!Identifies interface Serial 0/1/0 as the outside NAT interface
```

- **Overloading -** A form of dynamic NAT that maps multiple unregistered IP addresses to a single registered IP address by using different ports. This is known also as PAT (Port Address Translation), single address NAT or port-level multiplexed NAT.




```
access-list 1 permit 192.168.0.0 0.0.255.255
!Defines which addresses are eligible to be translated
ip nat inside source list 1 interface serial 0/1/0 overload
!Identifies the outside interface Serial 0/1/0 as the inside global address to be overloaded
interface serial 0/0/0
  ip nat inside
!Identifies Serial 0/0/0 as an inside NAT interface.
interface serial 0/1/0
  ip nat outside
!Identifies Serial 0/1/0 as an inside NAT interface.
```

3a. Instructions:

This lab provides an opportunity to revise your understanding of NAT/PAT, and the commands for configuring NAT/PAT on a router. The router R2 translates private IP to public IP using NAT/PAT.

Task 1: Create the Topology

Create a topology as shown in the following figure:

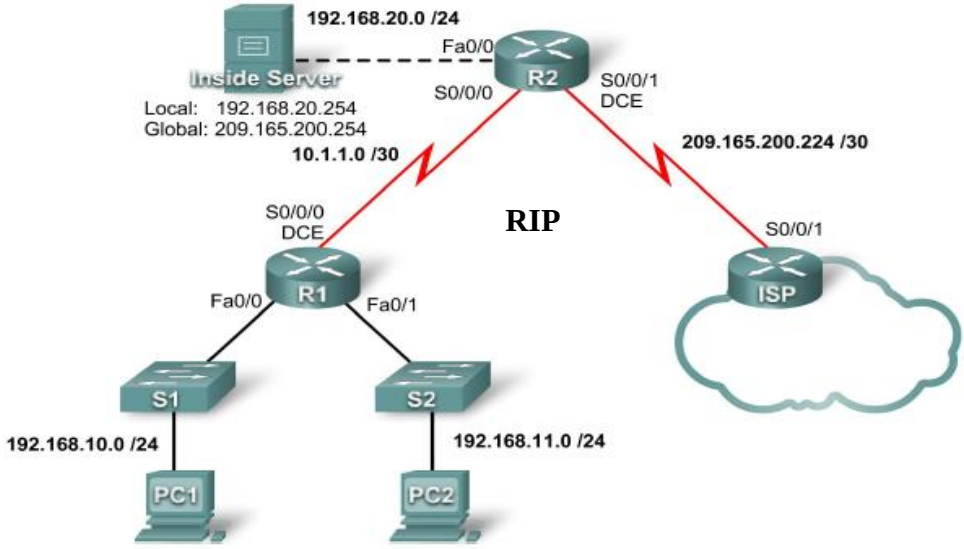


Table – 1

Device	Interface	IP Address	Subnet Mask
R1	S0/0/0	10.1.1.1	255.255.255.252
	Fa0/0	192.168.10.1	255.255.255.0
	Fa0/1	192.168.11.1	255.255.255.0
R2	S0/0/0	10.1.1.2	255.255.255.252
	S0/0/1	209.165.200.225	255.255.255.252
	Fa0/0	192.168.20.1	255.255.255.0
ISP	S0/0/1	209.165.200.226	255.255.255.252

Task 2: Configure Static and Default Routing

ISP uses static routing to reach all networks beyond R2. However, R2 translates private addresses into public addresses before sending traffic to ISP. Therefore, ISP must be configured with the public addresses that are part of the NAT configuration on R2. Enter the following **static route** on ISP:

```
ISP(config)#ip route 209.165.200.240 255.255.255.240 serial 0/0/1
```

This static route includes all addresses assigned to R2 for public use. Configure a **default route** on R2 and **propagate** the route in RIP.

```
R2(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.226
R2(config)#router rip
R2(config-router)#default-information originate
```

Allow a few seconds for R1 to learn the default route from R2 and then check the R1 routing table. Alternatively, you can clear the routing table with the **clear ip route *** command. A default route pointing to R2 should appear in the R1 routing table. From R1, ping the serial 0/0/1 interface on R2 (209.165.200.225). The pings should be successful. Troubleshoot if the pings fail.

Task 3: Configure Static NAT

Step 1: Statically map a public IP address to a private IP address.

The inside server attached to R2 is accessible by outside hosts beyond ISP. Statically assign the public IP address 209.165.200.254 as the address for NAT to use to map packets to the private IP address of the inside server at 192.168.20.254.

```
R2(config)#ip nat inside source static 192.168.20.254 209.165.200.254
```

Step 2: Specify inside and outside NAT interfaces.

Before NAT can work, you must specify which interfaces are inside and which interfaces are outside.

```
R2(config)#interface serial 0/0/1
R2(config-if)#ip nat outside
R2(config-if)#interface fa0/0
R2(config-if)#ip nat inside
```

Step 3: Verify the static NAT configuration.

From ISP, ping the public IP address 209.165.200.254.

Task 4: Configure Dynamic NAT with a Pool of Addresses

While static NAT provides a permanent mapping between an internal address and a specific public address, dynamic NAT maps private IP addresses to public addresses. These public IP addresses come from a NAT pool.

Step 1: Define a pool of global addresses.

Create a pool of addresses to which matched source addresses are translated. The following command creates a pool named **MY-NAT-POOL** that translates matched addresses to an available IP address in the 209.165.200.241 - 209.165.200.246 range.

```
R2(config)#ip nat pool MY-NAT-POOL 209.165.200.241 209.165.200.246 netmask
255.255.255.248
```

Step 2: Create a standard access control list to identify which inside addresses are translated.

```
R2(config)#ip access-list extended NAT
R2(config-std-nacl)#permit ip 192.168.10.0 0.0.0.255 any
R2(config-std-nacl)#permit ip 192.168.11.0 0.0.0.255 any
```

Step 3: Establish dynamic source translation by binding the pool with the access control list.

A router can have more than one NAT pool and more than one ACL. The following command tells the router which address pool to use to translate hosts that are allowed by the ACL.

```
R2(config)#ip nat inside source list NAT pool MY-NAT-POOL
```

Step 4: Specify inside and outside NAT interfaces.

You have already specified the inside and outside interfaces for your static NAT configuration. Now add the serial interface linked to R1 as an inside interface.

```
R2(config)#interface serial 0/0/0
R2(config-if)#ip nat inside
```

Step 5: Verify the configuration.

Ping ISP from PC1 and PC2. Then use the **show ip nat translations** command on R2 to verify NAT.

```
R2#show ip nat translations
Pro Inside global Inside local Outside local Outside global
--- 209.165.200.241 192.168.10.11 --- ---
--- 209.165.200.242 192.168.11.11 --- ---
--- 209.165.200.254 192.168.20.254 --- ---
```

Task 5: Configure NAT Overload

In the previous example, what would happen if you needed more than the **six public IP addresses** that the pool allows? By tracking port numbers, NAT overloading allows multiple inside users to reuse a public IP address. In this task, you will remove the pool and mapping statement configured in the previous task. Then you will configure NAT overload on R2 so that all internal IP addresses are translated to the R2 S0/0/1 address when connecting to any outside device.

Step 1: Remove the NAT pool and mapping statement.

Use the following commands to remove the NAT pool and the map to the NAT ACL.

```
R2(config)#no ip nat pool MY-NAT-POOL 209.165.200.241 209.165.200.246 netmask
255.255.255.248
R2(config)#no ip nat inside source list NAT pool MY-NAT-POOL
If you receive the following message, clear your NAT translations.
%Pool MY-NAT-POOL in use, cannot destroy
R2#clear ip nat translation *
```

Step 2: Configure PAT on R2 using the serial 0/0/1 interface public IP address.

The configuration is similar to dynamic NAT, except that instead of a pool of addresses, the **interface** keyword is used to identify the outside IP address. Therefore, no NAT pool is defined. The **overload** keyword enables the addition of the port number to the translation. Because you already configured an ACL to identify which inside IP addresses to translate as well as which interfaces are inside and outside, you only need to configure the following:

```
R2(config)#ip nat inside source list NAT interface S0/0/1 overload
```

Step 3: Verify the configuration.

Ping ISP from PC1 and PC2. Then use the **show ip nat translations** command on R2 to verify NAT.

```
R2#show ip nat translations
Pro Inside global Inside local Outside local Outside global
icmp 209.165.200.225:3 192.168.10.11:3 209.165.200.226:3
209.165.200.226:3
icmp 209.165.200.225:1024192.168.11.11:3 209.165.200.226:3
209.165.200.226:1024
--- 209.165.200.254 192.168.20.254 --- ---
```

Note: In the previous task, you could have added the keyword **overload** to the **ip nat inside source list NAT pool MY-NAT-POOL** command to allow for more than six concurrent users.