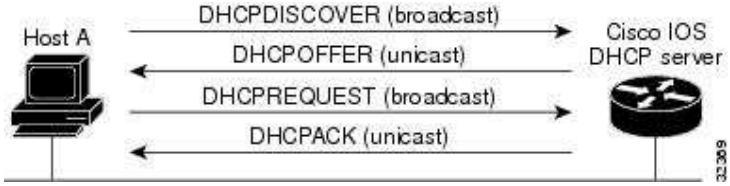
Dynamic Host Configuration Protocol (DHCP) & Network Address Translation (NAT)

1. Objectives

- To learn about **Dynamic Host Configuration Protocol (DHCP)**: why and how used?
- To learn about **Network Address Translation (NAT)**: why and how used?
- To build an **internetwork** and configure **DHCP** and **NAT** using **Packet Tracer**

2. Background: DHCP

Dynamic Host Control Protocol (DHCP) enables you to automatically assign reusable IP addresses to DHCP clients. The **DHCP Server** feature is a full DHCP server implementation that assigns and manages IP addresses from specified **address pools** within the router to DHCP clients. **Figure 1** shows the basic steps that occur when a DHCP client requests an IP address from a DHCP server. The client, Host A, sends a **DHCPDISCOVER** broadcast message to locate a **DHCP Server**. A DHCP server offers configuration parameters (such as an IP address, a MAC address, a domain name, and a lease for the IP address) to the client in a **DHCPOFFER** unicast message.



Benefits of DHCP

- **Reduced Internet access costs:** Using automatic IP address assignment at each remote site substantially reduces Internet access costs. Static IP addresses are considerably more expensive to purchase than are automatically allocated IP addresses.
- **Reduced client configuration tasks and costs:** Because DHCP is easy to configure, it minimizes operational overhead and costs associated with device configuration tasks and eases deployment by nontechnical users.
- **Centralized management:** Because the DHCP server maintains configurations for several subnets, an administrator only needs to update a single, central server when configuration parameters change.

NAT

NAT (Network Address Translation) is a technique for preserving scarce Internet IP addresses.

Why NAT?

The current Internet uses IP addresses in the form xxx.xxx.xxx.xxx. A sample IP address might be 202.187.4.212. Because of the way these IP addresses are allocated, there started to be a shortage of available IP addresses. The current revision of IP (Internet Protocol) in use on the Internet is IPv4. IPv6 is largely a response to this potential IP address shortage. Unfortunately, IPv6 is going to take decades to implement. A much quicker fix was needed, and that fix was NAT.

Private Address Space for NAT

To conserve IP address space, networks which are not directly connected to the Internet are often given private address space. Private address spaces are ranges of IP address which cannot be routed over the Internet. Private address space is often called "RFC 1918" space, because private address space is defined in RFC 1918 - Address Allocation for Private Internets. RFC 1918 defines three sets of private address space:

Start	End	Network Size
10.0.0.0	10.255.255.255	/8
172.16.0.0	172.31.255.255	/12
192.168.0.0	192.168.255.255	/16

The use of private address space conserves IP addresses because any person or company can use the same private address space over and over again. I have a 10.0.0.x network in my house. IBM has a 10.0.0.x network. HP has a 10.0.0.x network. Apple has a 10.0.0.x network. We're all using the same range of IP addresses. The limitation is that private address space is non-routable. This means that any computer on these private IP addresses cannot (directly) connect to the Internet.

Network Address Translation to the Rescue!

The solution to work-around this limitation is NAT (Network Address Translation). A NAT device, usually a firewall or a router, is placed between the private network and the Internet. When computers on the private network want to communicate on the Internet, the NAT device quickly and silently modifies the packets they send to have a normal (public) IP address. When systems on the Internet send reply packets, the NAT device routes those reply packets back to the correct system on the private network. In this way, hundreds or thousands of computers on the private network can share just one IP address on the public Internet. For example, you might have 250 computers on the 192.168.1.x network and one firewall providing NAT services on the IP address 216.17.138.210. Any time one of the hosts communicates across the Internet, the NAT firewall changes the IP address of the packets to 216.17.138.210. When reply packets come from the Internet, the NAT firewall sorts them out and sends them to the correct internal host.

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

NAT Configuration Commands and Examples

1. Static NAT

Configuring static NAT translations is a simple task. You need to define the addresses to translate and then configure NAT on the appropriate interfaces. Packets arriving on an inside interface from the identified IP address are subject to translation. Packets arriving on an outside interface addressed to the identified IP address are subject to translation. The figure explains the commands for the steps. You enter static translations directly into the configuration. Unlike dynamic translations, these translations are always in the NAT table.

Example:

The figure is a simple static NAT configuration applied to both interfaces. The router always translates packets from the host inside the network with the private address of 192.168.10.254 into an outside address of 209.165.200.254. The host on the Internet directs web requests to the public IP address 209.165.200.254, and router R2 always forwards that traffic to the server at 192.168.10.254.


```
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.
```

2. Dynamic NAT

To configure dynamic NAT, you need an ACL to permit only those addresses that are to be translated. When developing your ACL, remember there is an implicit "deny all" at the end of each ACL. An ACL that is too permissive can lead to unpredictable results. Cisco advises against configuring access contol lists referenced by NAT commands with the permit any command. Using permit any can result in NAT consuming too many router resources, which can cause network problems.

Example:

This configuration allows translation for all hosts on the 192.168.10.0 and 192.168.11.0 networks when they generate traffic that enters S0/0/0 and exits S0/1/0. These hosts are translated to an available address in the 209.165.200.226 - 209.165.200.240 range.

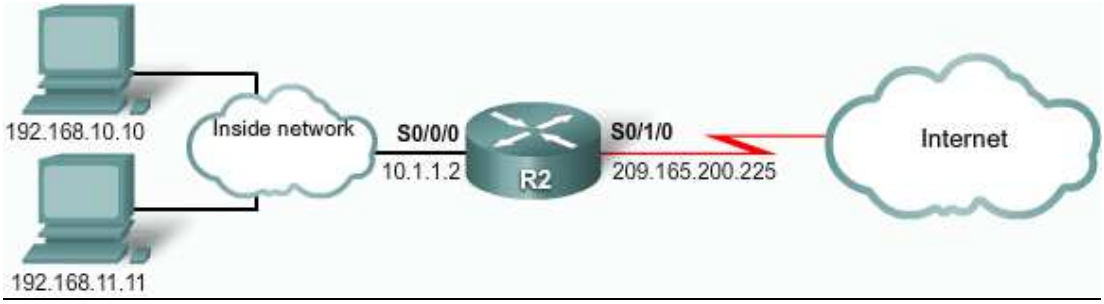

```
ip nat pool NAT-POOL1 209.165.200.226 209.165.200.240 netmask 255.255.255.224
!Defines a pool of public IP addresses under the pool name NAT-POOL1
access-list 1 permit 192.168.0.0 0.0.255.255
!Defines which addresses are eligible to be translated
ip nat inside source list 1 pool NAT-POOL1
!Binds 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
```

3. Overloading

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.

Example:

This example shows how NAT overload is configured. In the example, all hosts from network 192.168.0.0 /16 (matching ACL 1) sending traffic through router R2 to the Internet are translated to IP address 209.165.200.225 (interface S0/1/0 IP address). The traffic flows are identified by port numbers, because the overload keyword was used.



```
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: DHCP

This lab provides an opportunity to revise your understanding of DHCP, and the commands for configuring DHCP functions on a router. One router is the DHCP server. The other router forwards DHCP requests to the server.

Task 1: Create the Topology

Create a topology as shown in the following figure:

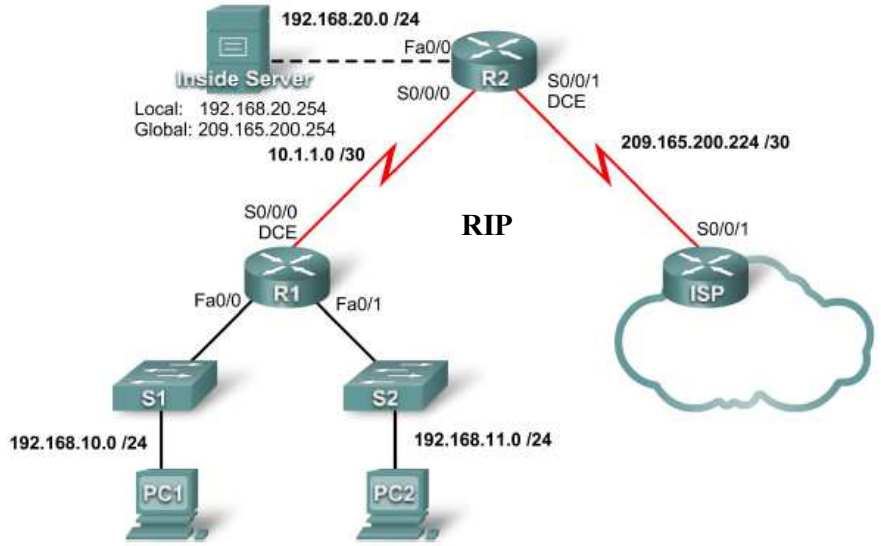


Table – 1

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 PCs and PC3 to receive an IP address through DHCP

Task 2: Configure a Cisco IOS DHCP Server

Step 1: Exclude statically assigned addresses.

The DHCP server assumes that all IP addresses in a DHCP address pool subnet are available for assigning to DHCP clients. You must specify the IP addresses that the DHCP server should not assign to clients. These IP address are usually static addresses reserved for the router interface, switch management IP address, servers, and local network printer. The **ip dhcp excluded-address** command prevents the router from assigning IP addresses within the configured range. The following commands exclude the first 10 IP addresses from each pool for the LANs attached to R1. These addresses will not be assigned to any DHCP clients.

```
R1(config)#ip dhcp excluded-address 192.168.10.1 192.168.10.10
R1(config)#ip dhcp excluded-address 192.168.11.1 192.168.11.10
```

Step 2: Configure the pool.
Create the DHCP pool using the **ip dhcp pool** command and name it **R1Fa0**.
```
R1(config)#ip dhcp pool R1Fa0
```
Specify the subnet to use when assigning IP addresses. DHCP pools automatically associate with an interface based on the network statement. The router now acts as a DHCP server, handing out addresses in the 192.168.10.0/24 subnet starting with 192.168.10.1.
```
R1(dhcp-config)#network 192.168.10.0 255.255.255.0
```

Configure the default router and domain name server for the network. Clients receive these settings via DHCP, along with an IP address.
```
R1(dhcp-config)#dns-server 192.168.11.5
R1(dhcp-config)#default-router 192.168.10.1
```
Note: There is not a DNS server at 192.168.11.5. You are configuring the command for practice only.
```
R1(config)#ip dhcp pool R1Fa1
R1(dhcp-config)#network 192.168.11.0 255.255.255.0
R1(dhcp-config)#dns-server 192.168.11.5
R1(dhcp-config)#default-router 192.168.11.1
```
Step 3: Verify the DHCP configuration.
You can verify the DHCP server configuration in several different ways. The most basic way is to configure a host on the subnet to receive an IP address via DHCP. You can then issue commands on the router to get more information. The **show ip dhcp binding** command provides information on all currently assigned DHCP addresses. For instance, the following output shows that the IP address 192.168.10.11 has been assigned to MAC address 3031.632e.3537.6563. The IP lease expires on September 14, 2007 at 7:33 pm.
```
R1#show ip dhcp binding
IP address Client-ID/ Lease expiration Type
Hardware address
192.168.10.11 0007.EC66.8752 -- Automatic
192.168.11.11 00E0.F724.8EDA – Automatic
```

Task 3: 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 OSPF.
```
R2(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.226
R2(config)#router ospf 1
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 4: 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 5: 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 6: 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.