# Data Communication and Computer Networks Lecture

## DHCP and NAT, Standard ACL, Extended ACL and Named ACL

***Introduction***

DHCP service is a key component of your network infrastructure by allowing centralized ip address management on a single pool of servers. DHCP configuration is also part of CCNA and CCNP Enterprise certification curricula.

Packet Tracer 8.0 implements two methods for setting up a DHCP server in your network:

- Configuration of DHCP pools on Cisco routers or multilayer switches.
- Configuration of a standalone DHCP server appliance on the network and usage of the "ip helper-address" command on network devices for DHCP traffic forwarding outside each local broadcast domain. This lecture will describe this method for implementing DHCP service in your network.

DHCP server and DHCP client should be in the same vlan to be able to communicate as the initial DHCP discover is a layer 2 broadcast packet to ff:ff:ff:ff:ff:ff MAC address. Cisco routers and layer 3 switches are able to act as DHCP relay and forward DHCP requests to a DHCP server located in another VLAN : a single DHCP server can now be deployed to deliver IP addresses to many subnet.

**Description**

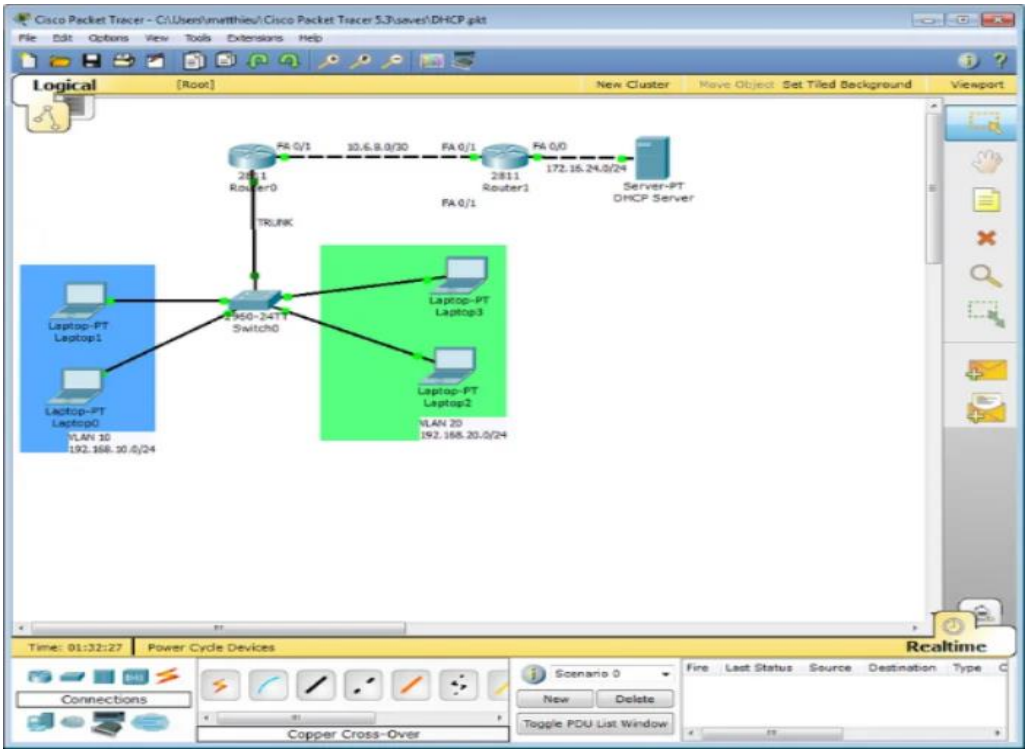
Configure dynamic IP address assignment on multiple VLAN with a unique DHCP server appliance on the network.

Two VLANs are configured on Switch0 with Router0 as default gateway:

VLAN 10 - Network: 192.168.10.0/24 - Gateway: 192.168.10.1 (FA 0/0.10)

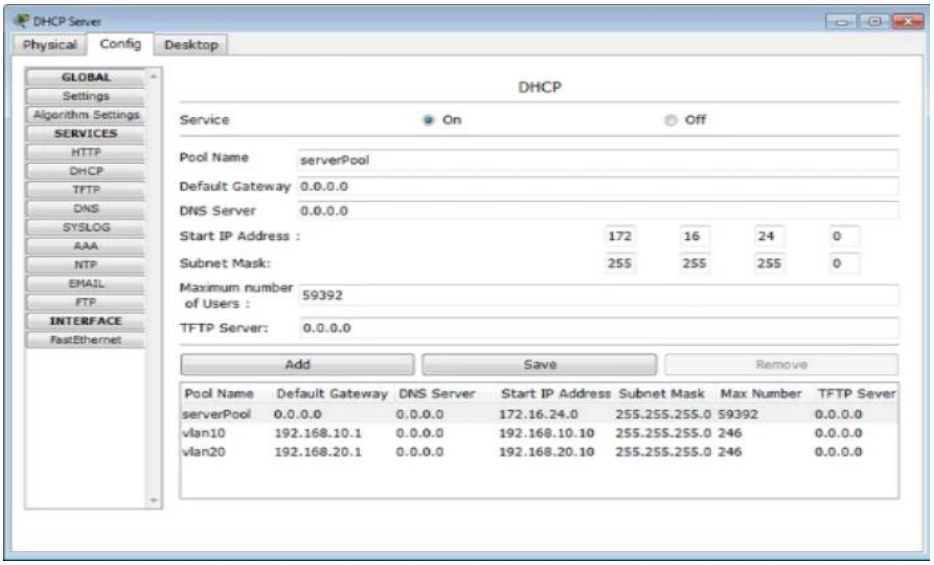
VLAN 20 - Network: 192.168.20.0/24 - Gateway: 192.168.20.1 (FA 0/0.20)

The unique DHCP server is located on a remote subnet with IP 172.16.24.2.




**DHCP configuration**

Declare IP address pools on the DHCP management tab of the server like on the picture below. One pool has to be declared for each VLAN. Don't forget to configure the right network settings and default gateway (Router0 FA 0/0.10 and FA 0.0.20 IP address) for each VLAN.

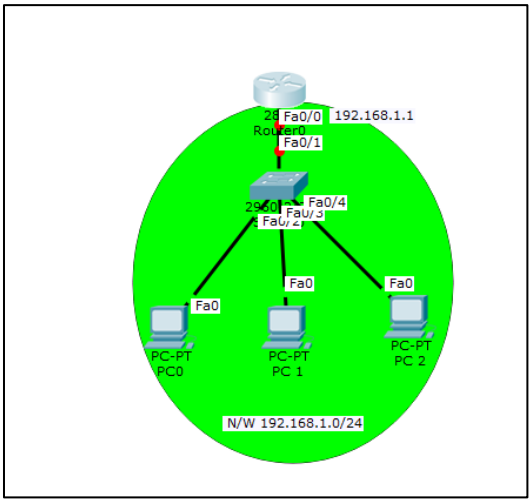


# Configure router0 for DHCP forwarding to the DHCP server (DHCP relay)

The **ip helper-address <IP address>** configures DHCP request forwarding to the configured <IP address> DHCP server.

```
Router(config)# interface FastEthernet0/0.10

Router(config-subif)# encapsulation dot1Q 10

Router(config-subif)# ip address 192.168.10.1 255.255.255.0

Router(config-subif)# ip helper-address 172.16.24.2


Router(config)# interface FastEthernet0/0.20

Router(config-subif)# encapsulation dot1Q 20

Router(config-subif)# ip address 192.168.20.1 255.255.255.0

Router(config-subif)# ip helper-address 172.16.24.2
```

Example 2:




## Configuring DHCP server on a Router.

1. Build the network topology:
2. On the router, configure *interface fa0/0* to act as the default gateway for our LAN.

```
3. Router>enable
```

```
4. Router#config terminal
5. Router(config)#int fa0/0
6. Router(config-if)#ip add 192.168.1.1 255.255.255.0
7. Router(config-if)#no shutdown
8. Router(config-if)#exit
```

3. Configure DHCP server on the Router. In the server we will define a **DHCP pool** of IP addresses to be assigned to hosts, a **Default gateway** for the LAN and a **DNS Server**.

```
Router(config)#

Router(config)#ip dhcp pool MY_LAN

Router(dhcp-config)#network 192.168.1.0 255.255.255.0

Router(dhcp-config)#default-router 192.168.1.1

Router(dhcp-config)#dns-server 192.168.1.10
```

We can add ip dhcp excluded-address command to our configuration so as to configure the router to exclude addresses 192.168.1.1 through 192.168.1.10 when assigning addresses to clients. The **ip dhcp excluded-address** command may be used to reserve addresses that are statically assigned to key hosts.

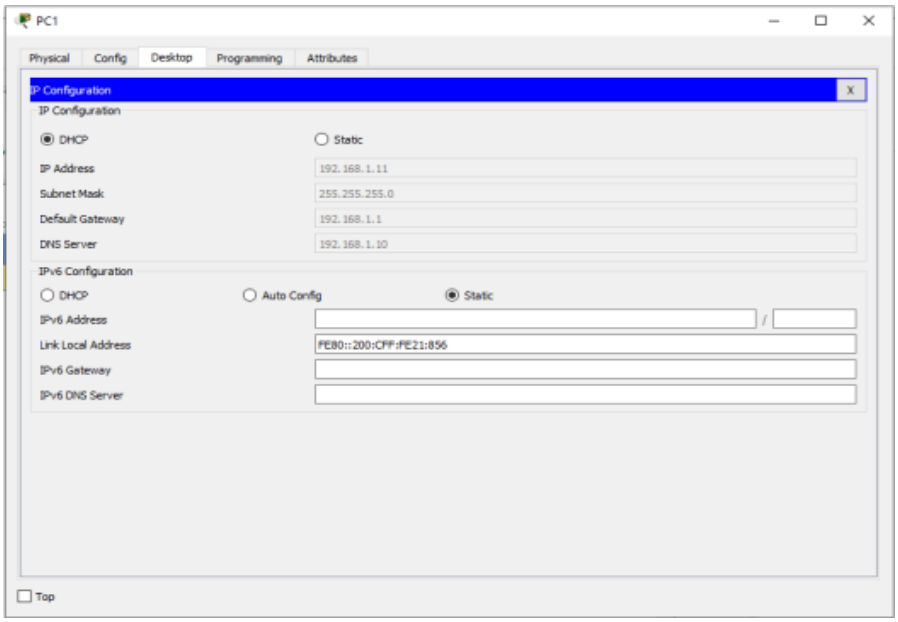
So add the above command under the **global configuration mode.**

```
Router(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.10
```

4. Now go to every PC and on their **IP configuration** tabs, enable **DHCP**. Every PC should be able to obtain an IP address, default gateway and DNS server, as defined in step 2.

For example, to enable DHCP on PC1:

Click **PC1->Desktop->IP configuration.** Then enable DHCP:




Do this for the other PCs.

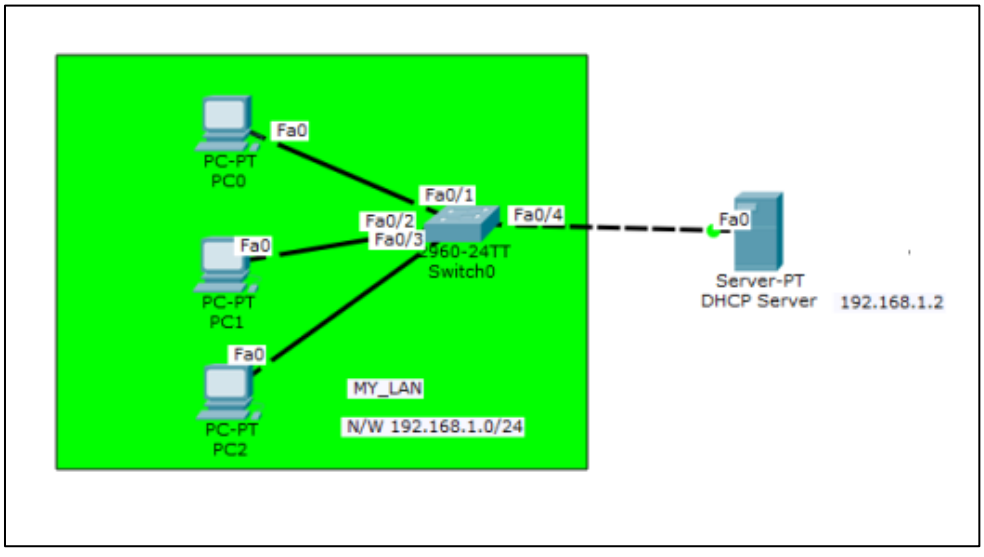
You can test the configuration by pinging PC2 from PC1. Ping should succeed.

It's that simple!

Now let's do the same thing using a Generic server in place of a router:

## Configuring DHCP service  on a generic server in Packet Tracer.

1. Build the network topology in packet tracer.



2. Configure static IP address on the server (192.168.1.2/24).

3. Now configure DHCP service on the generic server.

To do this, click on the server, then click on **Services tab**. You will pick **DHCP** on the menu. Then proceed to define the DHCP network parameters as follows:

**Pool name**: MY_LAN

**Default Gateway:** 192.168.1.1

**DNS Server:** 192.168.1.2

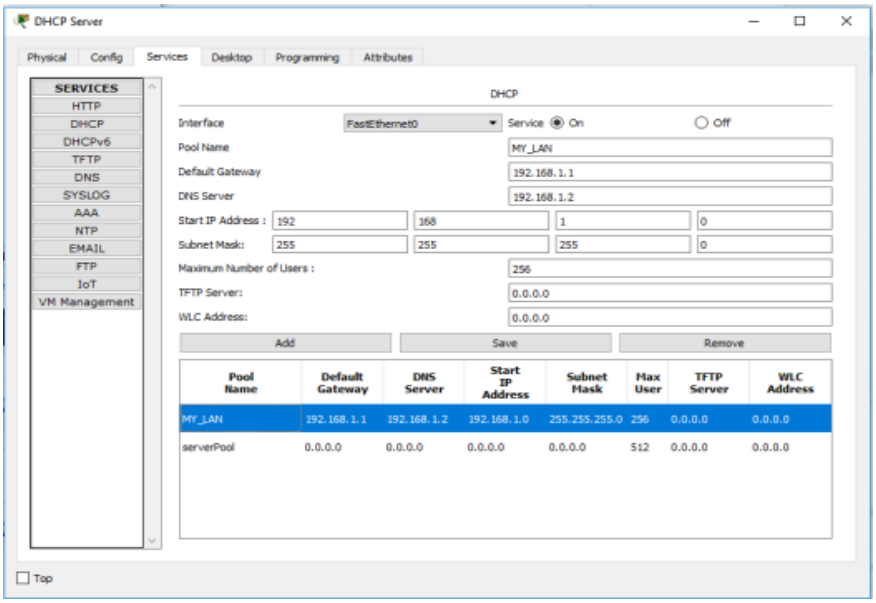
**Start IP Address:** 192.168.1.0

**Subnet Mask:** 255.255.255.0

**Maximum Number of users:** 256

Click on **add** then **Save.**The DHCP entry is included in the list.

Here are the configurations on the server:




Once you've configured everything, turn **ON** the DHCP service.

4. Finally, enable DHCP configuration on each PC. The three PCs should get automatically configured.

As an example, here is the DHCP configuration on PC1:

*Addendum:* You can define a DHCP server on one broadcast domain to serve hosts in a **different** broadcast domain. If you want to do this, then you should consider using *ip helper-address* command. To learn more about this, you can read my article on IP helper address configuration.

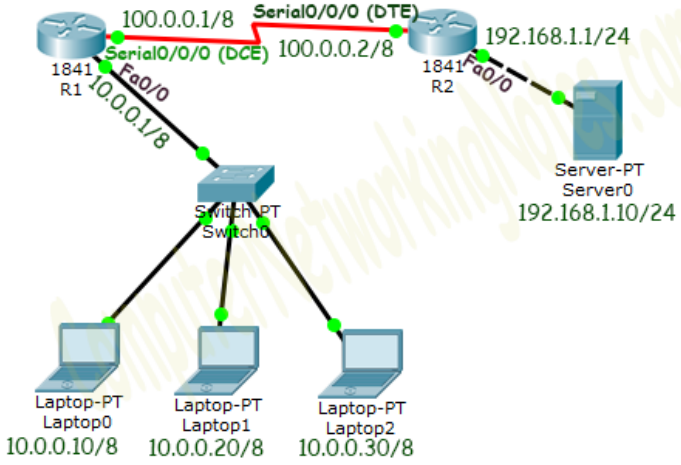
# Network Address Translation (NAT)

NAT (Network Address Translation) is used for security by reusing IP-addresses. The router on which Network Address Translation is configure translates traffic which is accessing internet or coming back to local network.

## Static NAT Practice LAB Setup

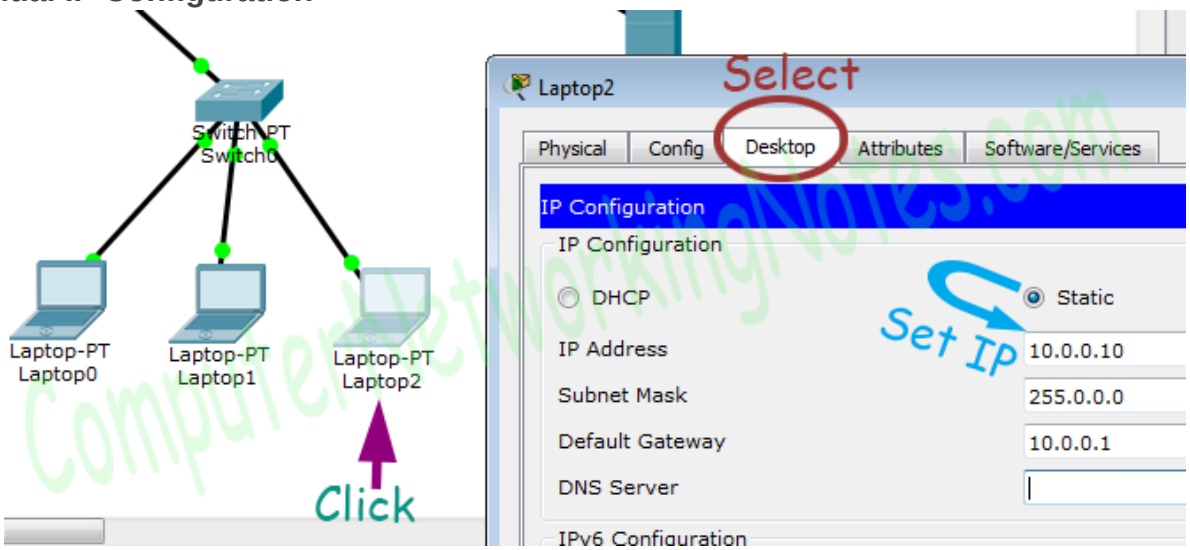
To explain Static NAT Configuration, I will use packet tracer network simulator software. You can use any network simulator software or can use real Cisco devices to follow this guide. There is no difference in output as long as your selected software contains the commands explained in this lecture.

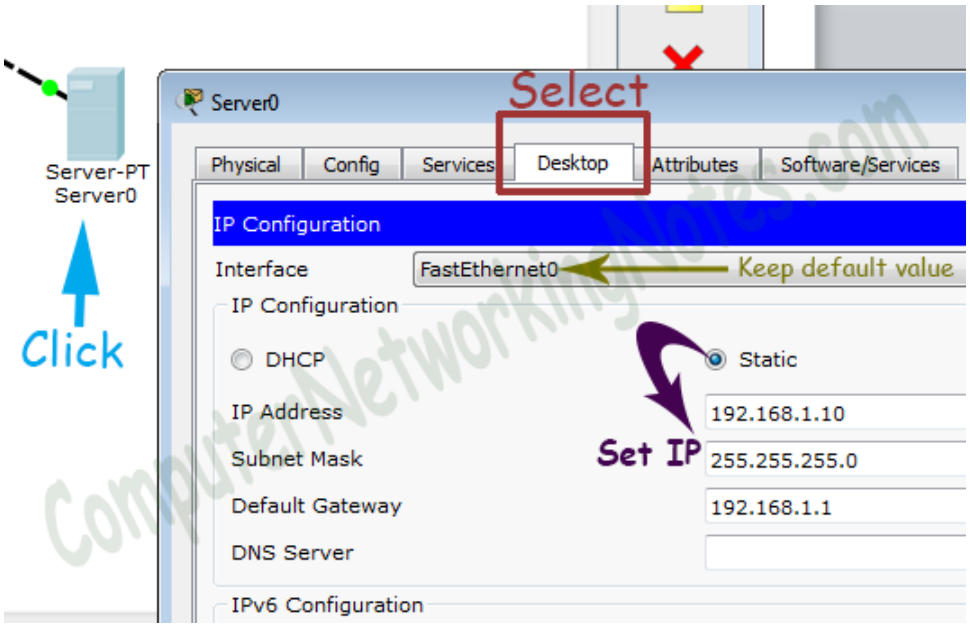
Create a practice lab as shown in following figure.



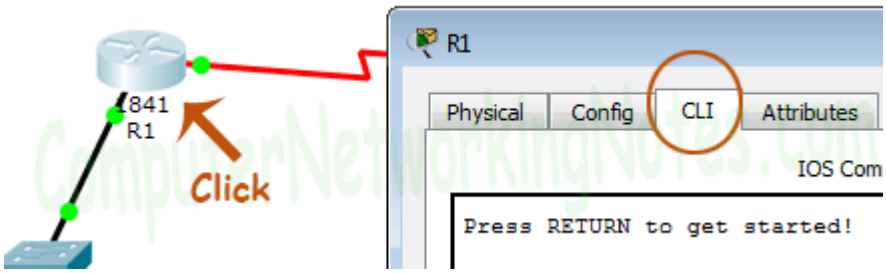

| Device / Interface | IP Address | Connected With |
|---|---|---|
| **Laotop0** | 10.0.0.10/8 | Fa0/0 of R0 |
| **Laptop1** | 10.0.0.20/8 | Fa0/0 of R0 |
| **Laptop2** | 10.0.0.30/8 | Fa0/0 of R0 |
| **Server0** | 192.168.1.10/24 | Fa0/0 of R1 |
| **Serial 0/0/0 of R1** | 100.0.0.1/8 | Serial 0/0/0 of R2 |
| **Serial 0/0/0 of R2** | 100.0.0.2/8 | Serial 0/0/0 of R2 |

## Initial IP Configuration




To configure IP address in Router1 click **Router1** and select **CLI** and press **Enter key**.




Two interfaces of Router1 are used in topology; FastEthernet0/0 and Serial 0/0/0.

By default interfaces on router are remain administratively down during the start up. We need to configure IP address and other parameters on interfaces before we could actually use them for routing. Interface mode is used to assign the IP address and other parameters. Interface mode can be accessed from global configuration mode. Following commands are used to access the global configuration mode.

```
Router>enable
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
```

Before we configure IP address in interfaces let's assign a unique descriptive name to router.

```
Router(config)#hostname R1
R1#
```

Now execute the following commands to set IP address in FastEthernet 0/0 interface.

```
R1(config)#interface FastEthernet0/0
R1(config-if)#ip address 10.0.0.1 255.0.0.0
R1(config-if)#no shutdown
R1(config-if)#exit
```

***interface FastEthernet 0/0*** command is used to enter in interface mode.

***ip address 10.0.0.1 255.0.0.0*** command assigns IP address to interface.

***no shutdown*** command is used to bring the interface up.

***exit*** command is used to return in global configuration mode.

Serial interface needs two additional parameters clock rate and bandwidth. Every serial cable has two ends DTE and DCE. These parameters are always configured at DCE end.

We can use show controllers interface command from privilege mode to check the cable's end.

```
R1(config)#exit
R1#show controllers serial 0/0/0
Interface Serial0/0/0
Hardware is PowerQUICC MPC860
DCE V.35, clock rate 2000000
[Output omitted]
```

Fourth line of output confirms that DCE end of serial cable is attached. If you see DTE here instead of DCE skip these parameters.

Now we have necessary information let's assign IP address to serial interface.

```
R1#configure terminal
R1(config)#interface Serial0/0/0
```

```
R1(config-if)#ip address 100.0.0.1 255.0.0.0
R1(config-if)#clock rate 64000
R1(config-if)#bandwidth 64
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#
```

Router#configure terminal Command is used to enter in global configuration mode.

***Router(config)#interface serial 0/0/0*** Command is used to enter in interface mode.

***Router(config-if)#ip address 100.0.0.1 255.0.0.0*** Command assigns IP address to interface.

**Router(config-if)#clock rate 64000**

In real life environment this parameter controls the data flow between serial links and need to be set at service provider's end. In lab environment we need not to worry about this value. We can use any valid rate here.

**Router(config-if)#bandwidth 64**

Bandwidth works as an influencer. It is used to influence the metric calculation of EIGRP or any other routing protocol which uses bandwidth parameter in route selection process.

***Router(config-if)#no shutdown*** Command brings interface up.

***Router(config-if)#exit*** Command is used to return in global configuration mode.

We will use same commands to assign IP addresses on interfaces of Router2. We need to provided clock rate and bandwidth only on DCE side of serial interface. Following command will assign IP addresses on interface of Router2.

***Initial IP configuration in R2***

```
Router>enable
Router#configure terminal
Router(config)#hostname R2
R2(config)#interface FastEthernet0/0
R2(config-if)#ip address 192.168.1.1 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#interface Serial0/0/0
R2(config-if)#ip address 100.0.0.2 255.0.0.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#
```

That's all initial IP configuration we need. Now this topology is ready for the practice of static nat.

# Configure Static NAT

Static NAT configuration requires three steps: -

1. Define IP address mapping
2. Define inside local interface
3. Define inside global interface

Since static NAT use manual translation, we have to map each inside local IP address (which needs a translation) with inside global IP address. Following command is used to map the inside local IP address with inside global IP address.

```
Router(config)#ip nat inside source static [inside local ip address] [inside global IP address]
```

For example in our lab Laptop1 is configured with IP address 10.0.0.10. To map it with 50.0.0.10 IP address we will use following command

```
Router(config)#ip nat inside source static 10.0.0.10 50.0.0.10
```

In second step we have to define which interface is connected with local the network. On both routers interface Fa0/0 is connected with the local network which need IP translation.

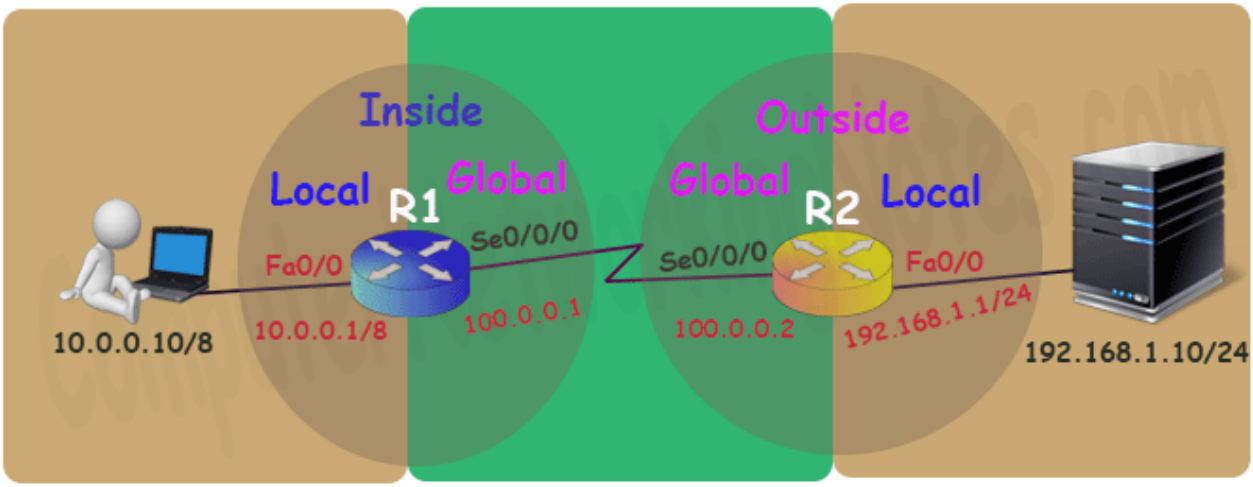
Following command will define interface Fa0/0 as inside local.

```
Router(config-if)#ip nat inside
```

In third step we have to define which interface is connected with the global network. On both routers serial 0/0/0 interface is connected with the global network. Following command will define interface Serial0/0/0 as inside global.

```
Router(config-if)#ip nat outside
```

Following figure illustrates these terms.



Let’s implement all these commands together and configure the static NAT.

# R1 Static NAT Configuration

```
R1(config)#ip nat inside source static 10.0.0.10 50.0.0.10
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#
R1(config)#interface Serial 0/0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

For testing purpose I configured only one static translation. You may use following commands to configure the translation for remaining address.

```
R1(config)#ip nat inside source static 10.0.0.20 50.0.0.20
R1(config)#ip nat inside source static 10.0.0.30 50.0.0.30
```

**R2 Static NAT Configuration**

```
R2(config)#ip nat inside source static 192.168.1.10 200.0.0.10
R2(config)#interface FastEthernet 0/0
R2(config-if)#ip nat inside
R2(config-if)#exit
R2(config)#
R2(config)#interface Serial 0/0/0
R2(config-if)#ip nat outside
R2(config-if)#exit
```

Before we test this lab we need to configure the IP routing. IP routing is the process which allows router to route the packet between different networks. Following lecture explain routing in detail with examples

Routing concepts Explained with Examples

### Configure static routing in R1

```
R1(config)#ip route 200.0.0.0 255.255.255.0 100.0.0.2
```

***Configure static routing in R2***

```
R2(config)#ip route 50.0.0.0 255.0.0.0 100.0.0.1
```

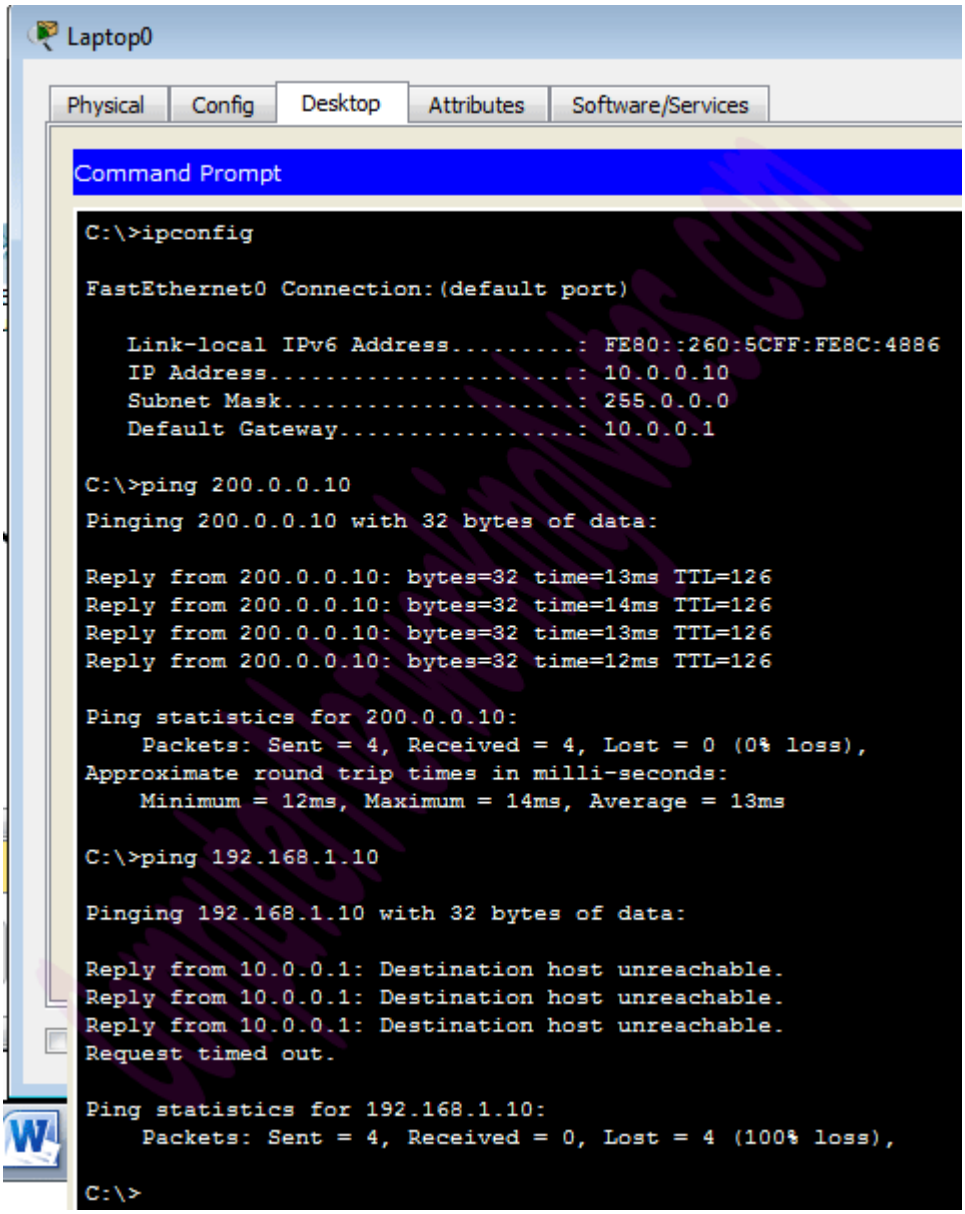
### Testing Static NAT Configuration

In this lab we configured static NAT on R1 and R2. On R1 we mapped inside local IP address 10.0.0.10 with inside global address 50.0.0.10 while on R2 we mapped inside local IP address 192.168.1.10 with inside global IP address 200.0.0.10.

| Device | Inside Local IP Address |
|---|---|
| Laptop0 | 10.0.0.10 |
| Server | 192.168.1.10 |

To test this setup click Laptop0 and Desktop and click Command Prompt.

- Run **ipconfig** command.
- Run **ping 200.0.0.10** command.
- Run **ping 192.168.1.10** command.

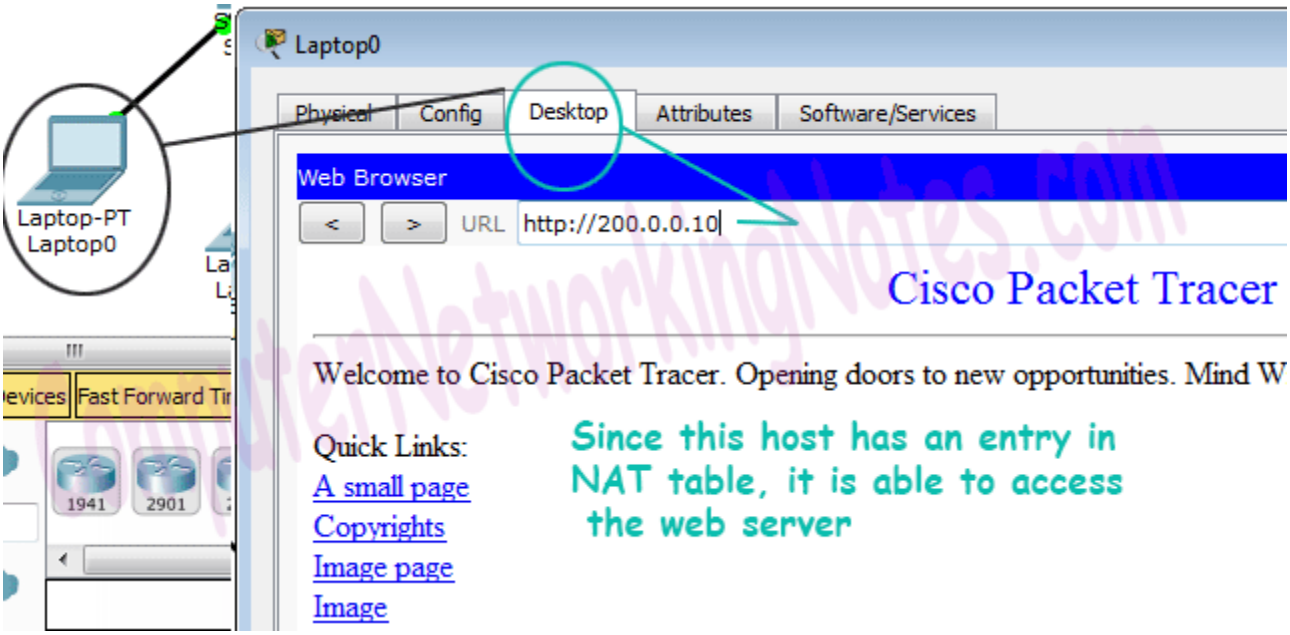



First command verifies that we are testing from correct NAT device.

Second command checks whether we are able to access the remote device or not. A ping reply confirms that we are able to connect with remote device on this IP address.

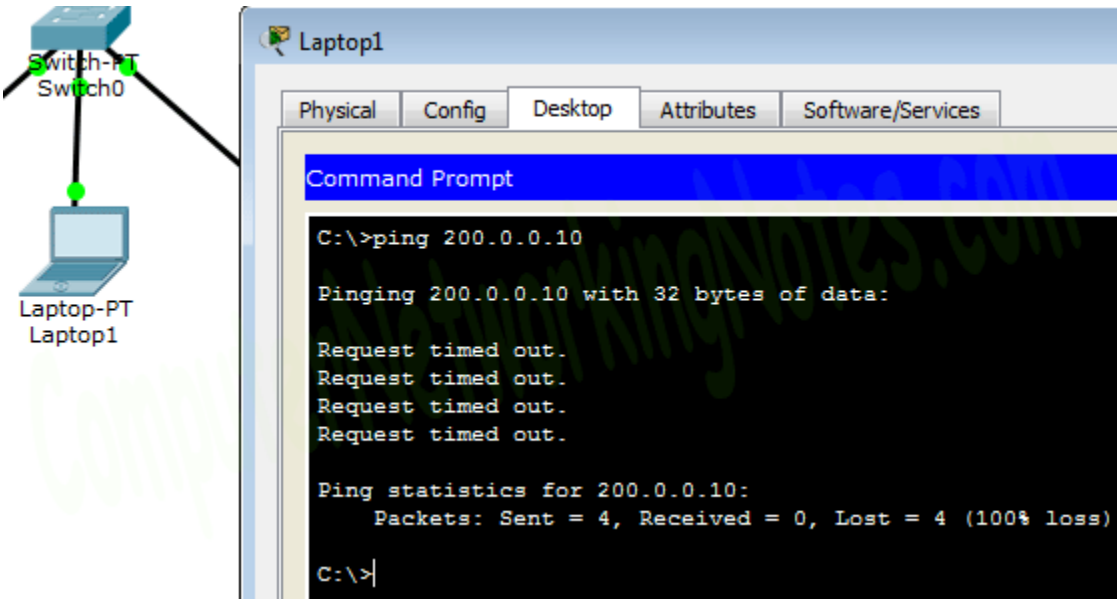
Third command checks whether we are able to access the remote device on its actual IP address or not. A ping error confirms that we are not able to connect with remote device on this IP address.

Let's do one more testing. Click **Laptop0** and click **Desktop** and click **Web Browser** and access 200.0.0.10.




Above figure confirms that host 10.0.0.10 is able to access the 200.0.0.10.

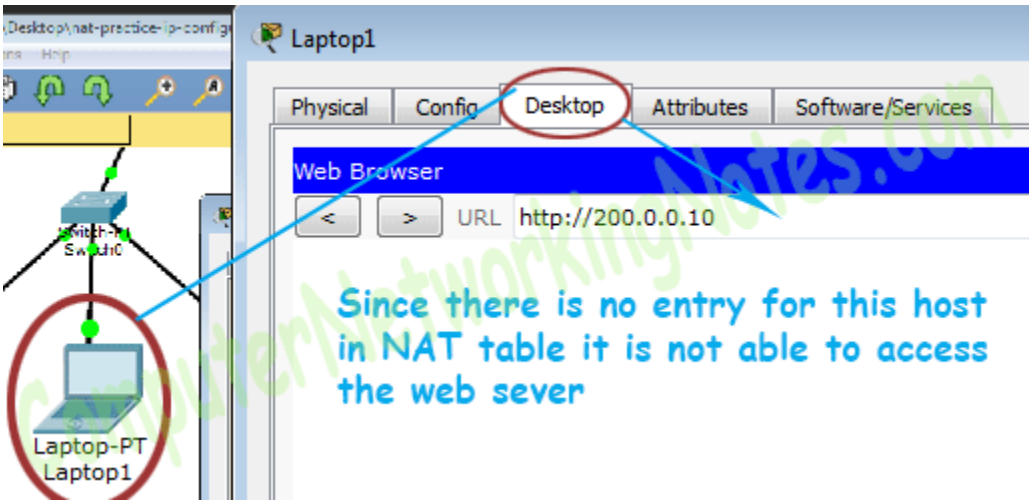
Now run **ping 200.0.0.10** command from Laptop1.




*Why we are not able to connect with the remote device from this host?*

Because we configured NAT only for one host (Laptop0) which IP address is 10.0.0.10. So only the host 10.0.0.10 will be able to access the remote device.

To confirm it again, let's try to access web service from this host.




*If you followed this lecture step by step, you should get the same output of testing. Although it's very rare but some time you may get different output.*

We can also verify this translation on router with ***show ip nat translation*** command.

Following figure illustrate this translation on router R1.

```
R1#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
icmp 50.0.0.10:13      10.0.0.10:13       200.0.0.10:13      200.0.0.10:13
icmp 50.0.0.10:14      10.0.0.10:14       200.0.0.10:14      200.0.0.10:14
icmp 50.0.0.10:15      10.0.0.10:15       200.0.0.10:15      200.0.0.10:15
icmp 50.0.0.10:16      10.0.0.10:16       200.0.0.10:16      200.0.0.10:16
tcp 50.0.0.10:1030     10.0.0.10:1030     200.0.0.10:80      200.0.0.10:80
tcp 50.0.0.10:1031     10.0.0.10:1031     200.0.0.10:80      200.0.0.10:80

R1#
```

Following figure illustrate this translation on router R2

```
R2#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
icmp 200.0.0.10:13     192.168.1.10:13    50.0.0.10:13       50.0.0.10:13
icmp 200.0.0.10:14     192.168.1.10:14    50.0.0.10:14       50.0.0.10:14
icmp 200.0.0.10:15     192.168.1.10:15    50.0.0.10:15       50.0.0.10:15
icmp 200.0.0.10:16     192.168.1.10:16    50.0.0.10:16       50.0.0.10:16
tcp 200.0.0.10:80      192.168.1.10:80    50.0.0.10:1030     50.0.0.10:1030
tcp 200.0.0.10:80      192.168.1.10:80    50.0.0.10:1031     50.0.0.10:1031

R2#
```

*Pay a little bit extra attention on outside local address filed. Have you noticed one interesting feature of NAT in above output? Why actual outside local IP address is not listed in this filed?*

The actual IP address is not listed here because router is receiving packets after the translation. From R1's point of view remote device's IP address is 200.0.0.10 while from R2's point of view end device's IP address is 50.0.0.10.

This way if NAT is enabled we would not be able to trace the actual end device.