# N A T – P A T

## Basic Concepts of NAT

This assignment explains basic concepts of static NAT, dynamic NAT, PAT, inside local, outside local, inside global and outside global in detail with examples.

### Basic overview of NAT

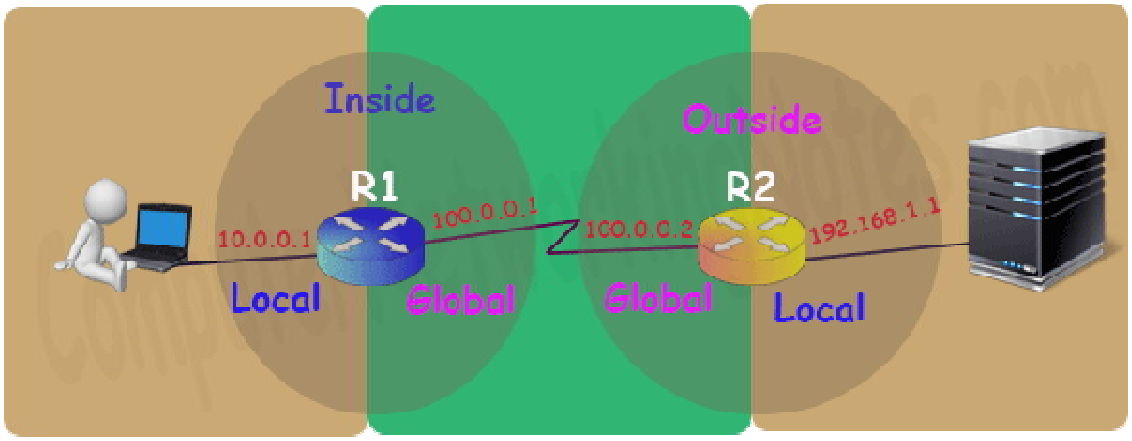
There are several situations where we need address translation such as, a network which do not have sufficient public IP addresses want to connect with the Internet, two networks which have same IP addresses want to merge or due to security reason a network want to hide its internal IP structure from the external world. NAT (Network Address Translation) is the process which translates IP address. NAT can be performed at firewall, server and router. In this assignment we will understand how it is performed at Cisco router.

### NAT Terminology

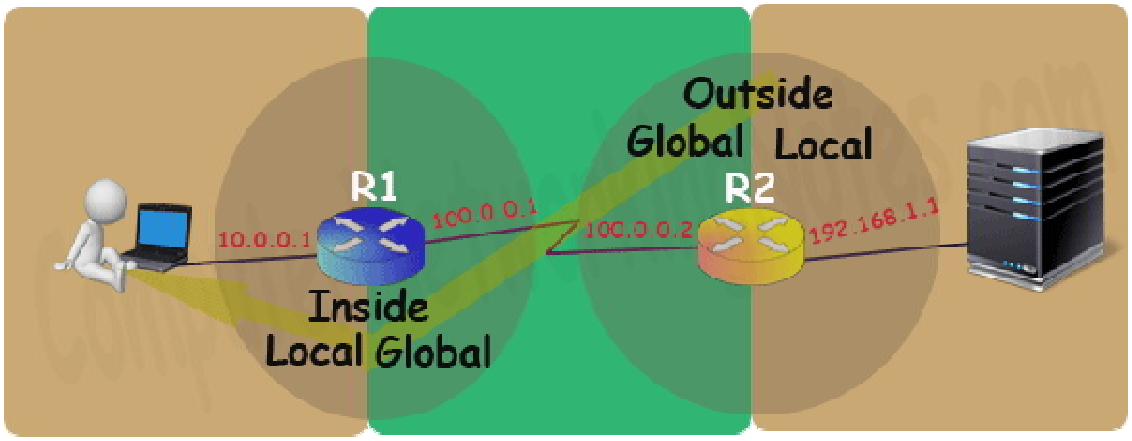
Before we understand NAT in details let's get familiar with four basic terms used in NAT.

| Term | Description |
|---|---|
| Inside Local IP Address | Before translation source IP address located inside the local network. |
| Inside Global IP Address | After translation source IP address located outside the local network. |
| Outside Global IP Address | Before translation destination IP address located outside the remote network. |
| Outside Local IP Address | After translation destination IP address located inside the remote network. |

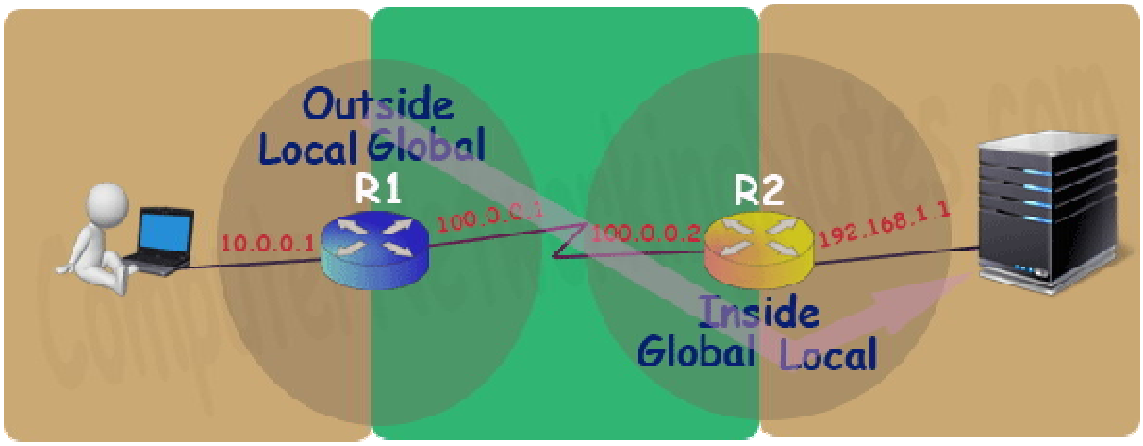
Let's understand these terms with an example. Suppose a user is browsing a website from his home computer. The network which connects his computer with internet is considered as a local network for him. Same as the network which connects the webserver where the website is located with internet is considered as a local network for webserver. The network which connects both networks on internet is considered as a global network.




On router the interface which is connected with local network will be configured with inside local IP address and the interface which is connected with global network will be configured with inside global IP address. Inside and outside depend on where we are standing right now. For example in above network for user router R1 is inside and router R2 is outside.




While for webserver router R2 is inside and router R1 is outside.



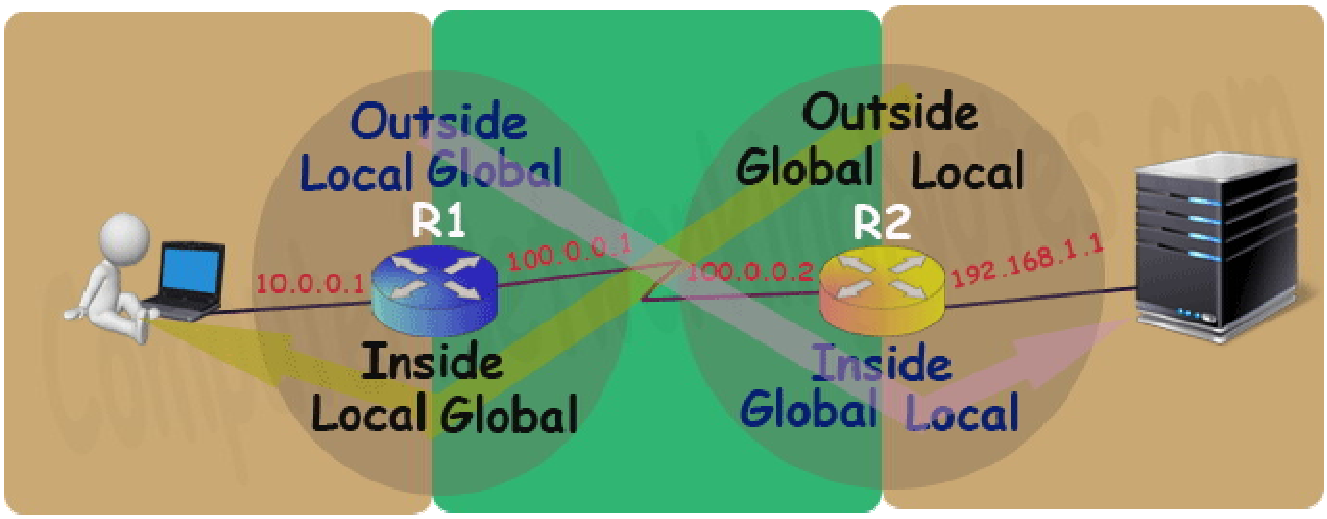
Basically on a NAT enabled router there are two types of interface inside local and inside global.

So, what about outside global and outside local? Well… these terms are used to explain the NAT process theoretically. Practically we never need to configure the outside local and outside global as they sound. For example let's discuss above example once again.

On R1 we will configure inside local address (10.0.0.1) and inside global address (100.0.0.1) which will become outside local address (10.0.0.1) and outside global address (100.0.0.1) for R2 respectively.

Same way on R2 we will configure inside local address (192.168.1.1) and inside global address (100.0.0.2) which will become outside local address (192.168.1.1) and outside global address (100.0.0.2) for R1 respectively.

So practically we only configure inside local and inside global. What is inside for one side is the outside for other side.




**Types of NAT**

There are three types of NAT; Static NAT, Dynamic NAT and PAT. These types define how inside local IP address will be mapped with inside global IP address.

**Static NAT**

In this type we manually map each inside local IP address with inside global IP address. Since this type uses one to one mapping we need exactly same number of IP address on both sides.

**Dynamic NAT**

In this type we create a pool of inside global IP addresses and let the NAT device to map inside local IP address with the available outside global IP address from the pool automatically.

**PAT**

In this type a single inside global IP address is mapped with multiple inside local IP addresses using the source port address. This is also known as PAT (Port Address Translation) or NAT over load.

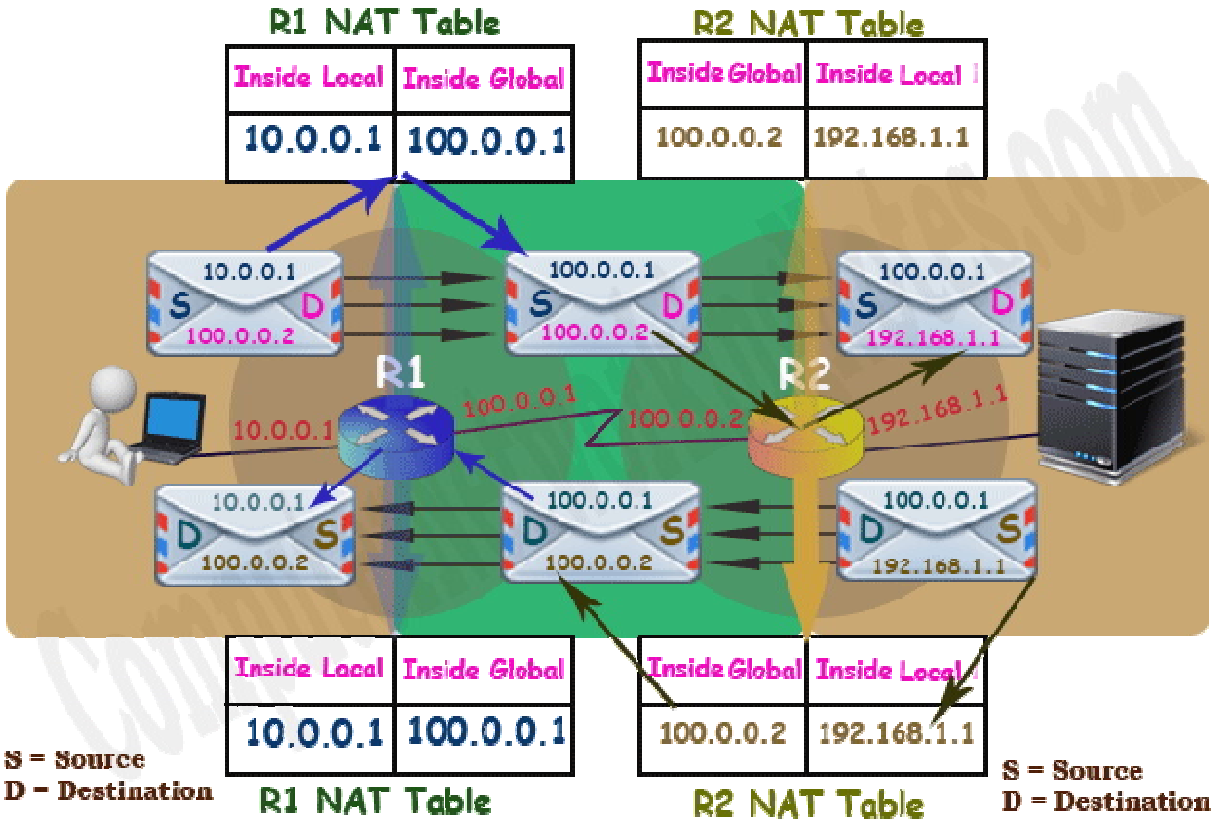
Situations where NAT is used

There are no hard and fast rules about where we should use NAT or where we should not use the NAT. Whether we should use the NAT or not is purely depends on network requirement for example NAT is the best solution in following situations: -

- Our network is built with private IP addresses and we want to connect it with internet. As we know to connect with internet we require public IP address. In this situation we can use NAT device which will map private IP address with public IP address.
- Two networks which are using same IP address scheme want to merge. In this situation NAT device is used to avoid IP overlapping issue.
- We want to connect multiple computers with internet through the single public IP address. In this situation NAT is used to map the multiple IP addresses with single IP address through the port number.

**How NAT Works**

To understand how NAT works, let's take one more example. In this example a user is accessing a web server. User and Webserver both are connected through the NAT devices. Both user and webserver are using private IP addresses which are not routable on the internet. Now let's understand how NAT makes this communication possible.




User generates a data packet for web server. This packet has source address 10.0.0.1 and destination address 100.0.0.2.

*Here source address is the correct address but why the packet has destination address 100.0.0.2 instead of actual destination address 192.168.1.1?*

When a system needs to connect with the website, it uses DNS server to resolve the IP address of the website. DNS server advertises the global IP address of the website. Outsider can connect with the website through the advertised IP address only. In our example the global IP address of web server is 100.0.0.2. For this reason the packet has the destination address 100.0.0.2 instead of 192.168.1.1.

This packet reaches at R1. Since this packet contains private IP address in source filed which is not routable on internet, R1 has to update the private IP address with a routable public IP address before forwarding this packet.

R1 checks NAT table for available public IP addresses. Depending on what type of NAT (Static, Dynamic or PAT) is configured one routable public IP will be picked from NAT table for this packet.

In our example 100.0.0.1 is picked for this packet. Now R1 will replace 10.0.0.1 with 100.0.0.1 in the source filed of the packet and forward it to the R2.

R2 receives this packet and reads the destination IP address. R2 looks in NAT table to find out the actual IP address of the destination. Since the NAT table of R2 has an entry for the address 100.0.0.2 which maps it with the address 192.168.1.1, R2 will replace the destination address 100.0.0.2 with the address 192.168.1.1 and forward it to the web server.

Webserver will process this packet and reply with its own packet. This packet has source address 192.168.1.1 and destination address 100.0.0.1.

*Since webserver received this packet from 100.0.0.1 so it will reply to it instead of 10.0.0.1.*

R2 receives this packet. Before forwarding this packet R2 will replace the source IP address with the mapped IP address in NAT table. In this example 192.168.1.1 will be replaced with 100.0.0.2.

R1 receives this packet and checks its destination address. R1 will perform a query in NAT table to figure out the IP address which is associated with this destination IP address. Since this destination IP address 100.0.0.1 is mapped with 10.0.0.1, R1 will replace this destination IP address 100.0.0.1 with 10.0.0.1 and forward it to the PC.

From user's point of view the IP address of the webserver is 100.0.0.2. While from web server's point of view the IP address of the user is 100.0.0.1. This way both user and webserver will never know to whom they are communicating actually.

**Advantages and disadvantages of NAT**

Nat provides following advantages: -

- NAT solves IP overlapping issue.
- NAT hides internal IP structure from external world.
- NAT allows us to connect with any network without changing IP address.
- NAT allows us to connect multiple computers with internet through the single the public IP address.

**NAT has following disadvantages: -**

- NAT adds additional delay in network.
- Several applications are not compatible with NAT.
- End to end IP traceability will not work with NAT.
- NAT hides actual end device.

That's all for this article. In next part of this assignment we will learn how to configure static NAT and dynamic NAT in Cisco router.

# How to Configure Static NAT in Cisco Router

This assignment explains how to configure static NAT (Network Address Translation) in Cisco Router step by step with Packet Tracer examples.

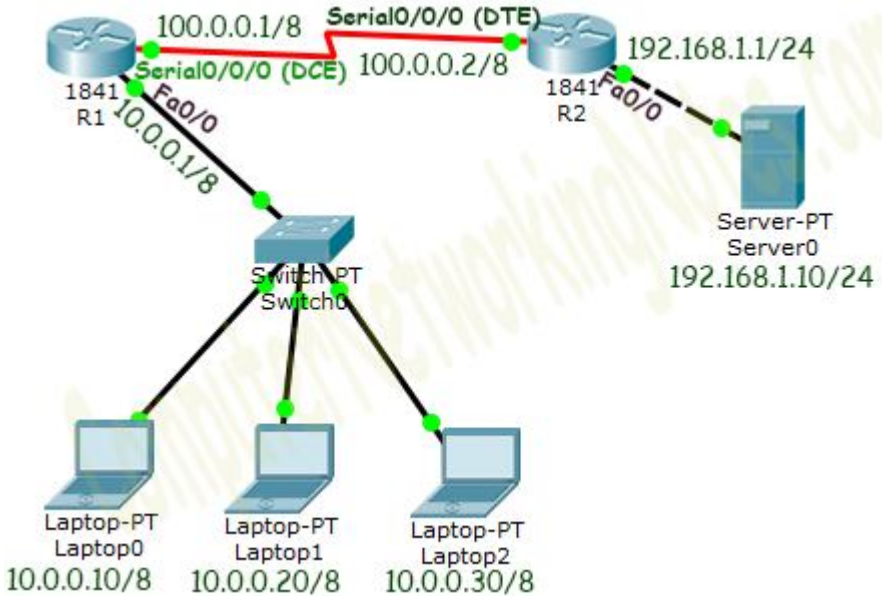
In order to configure NAT we have to understand four basic terms; inside local, inside global, outside local and outside global. These terms define which address will be mapped with which address.

| Term | Description |
|---|---|
| Inside Local IP Address | Before translation source IP address located inside the local network. |
| Inside Global IP Address | After translation source IP address located outside the local network. |
| Outside Global IP Address | Before translation destination IP address located outside the remote network. |
| Outside Local IP Address | After translation destination IP address located inside the remote network. |

### Static NAT Practice LAB Setup

In this assignment I will use Packet Tracer network simulator software for demonstration.

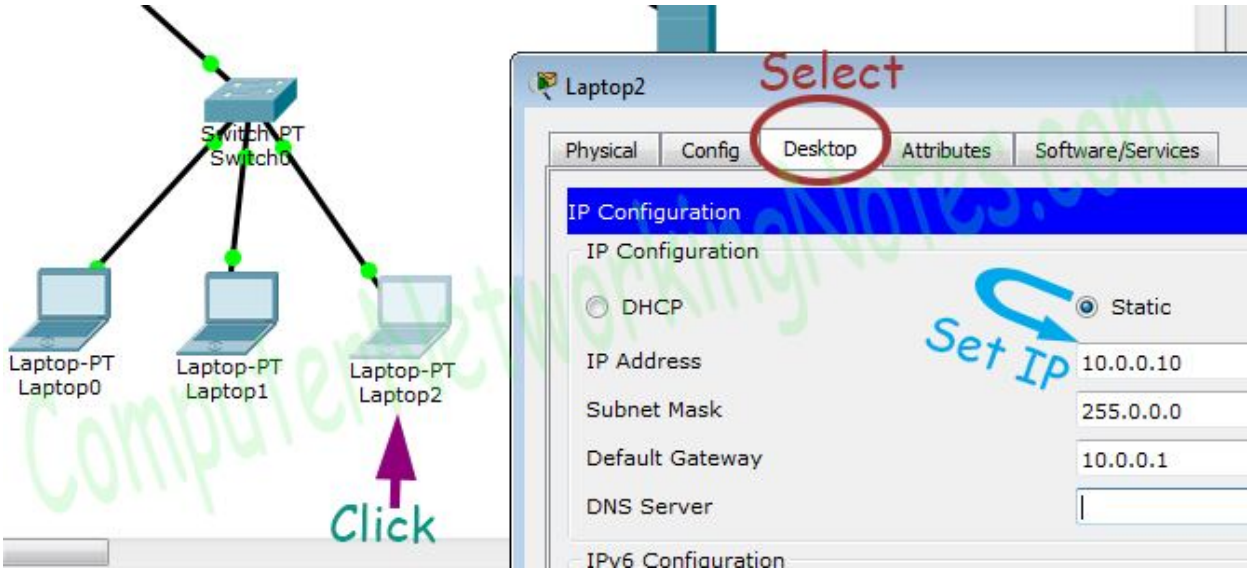
Create a lab as illustrates in following figure.



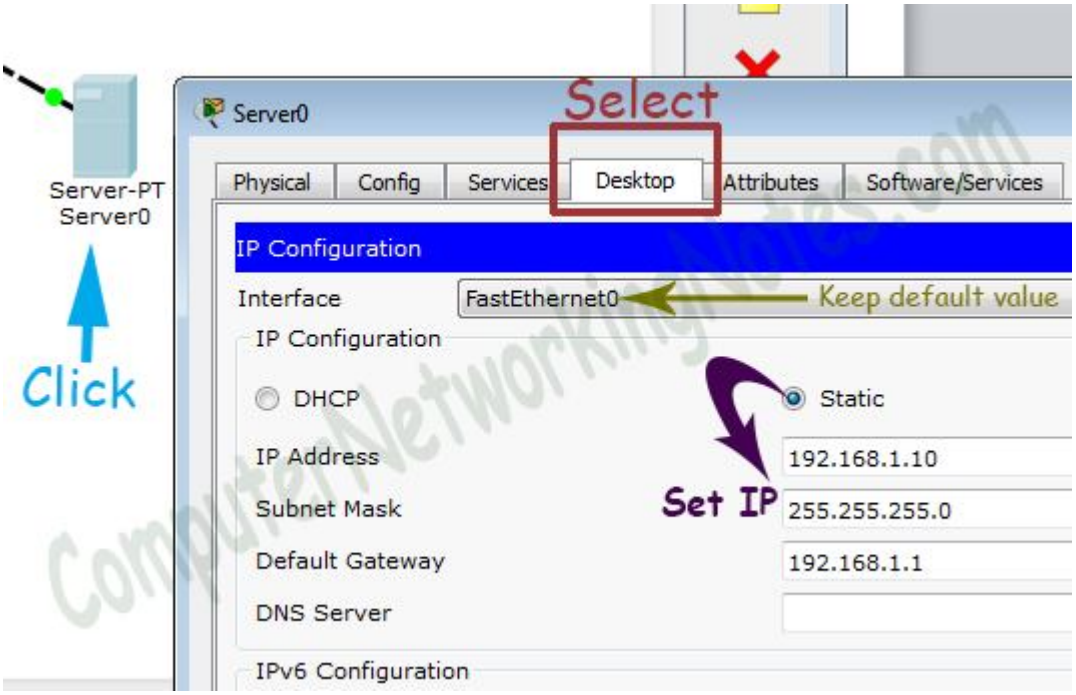
**Initial IP Configuration**

| 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 |

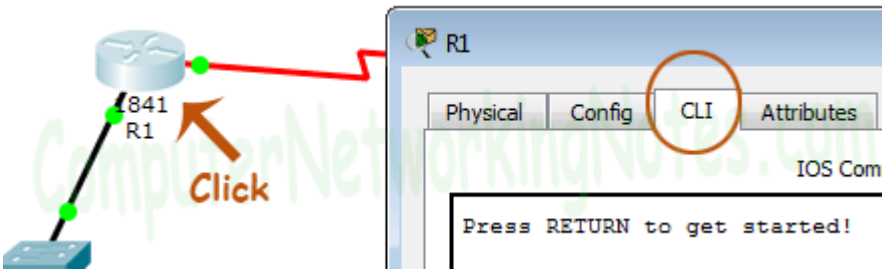
To assign IP address in Laptop click **Laptop** and click **Desktop** and **IP configuration** and Select **Static** and **set IP address** as given in above table.



Following same way configure IP address in Server.




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: -

- Define IP address mapping
- Define inside local interface
- 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.

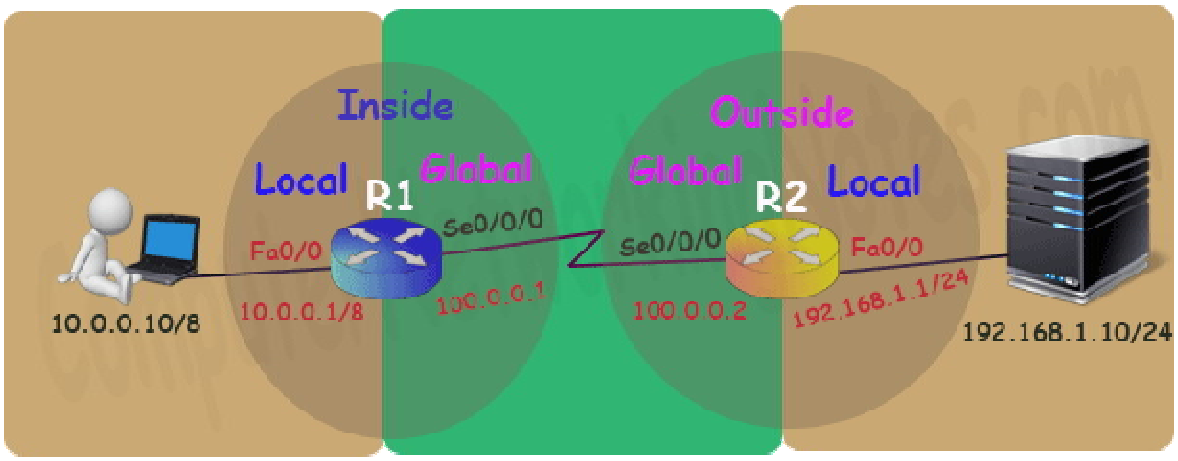
Following command will define interface Fa0/0 as inside local.

```
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 assignment 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
```

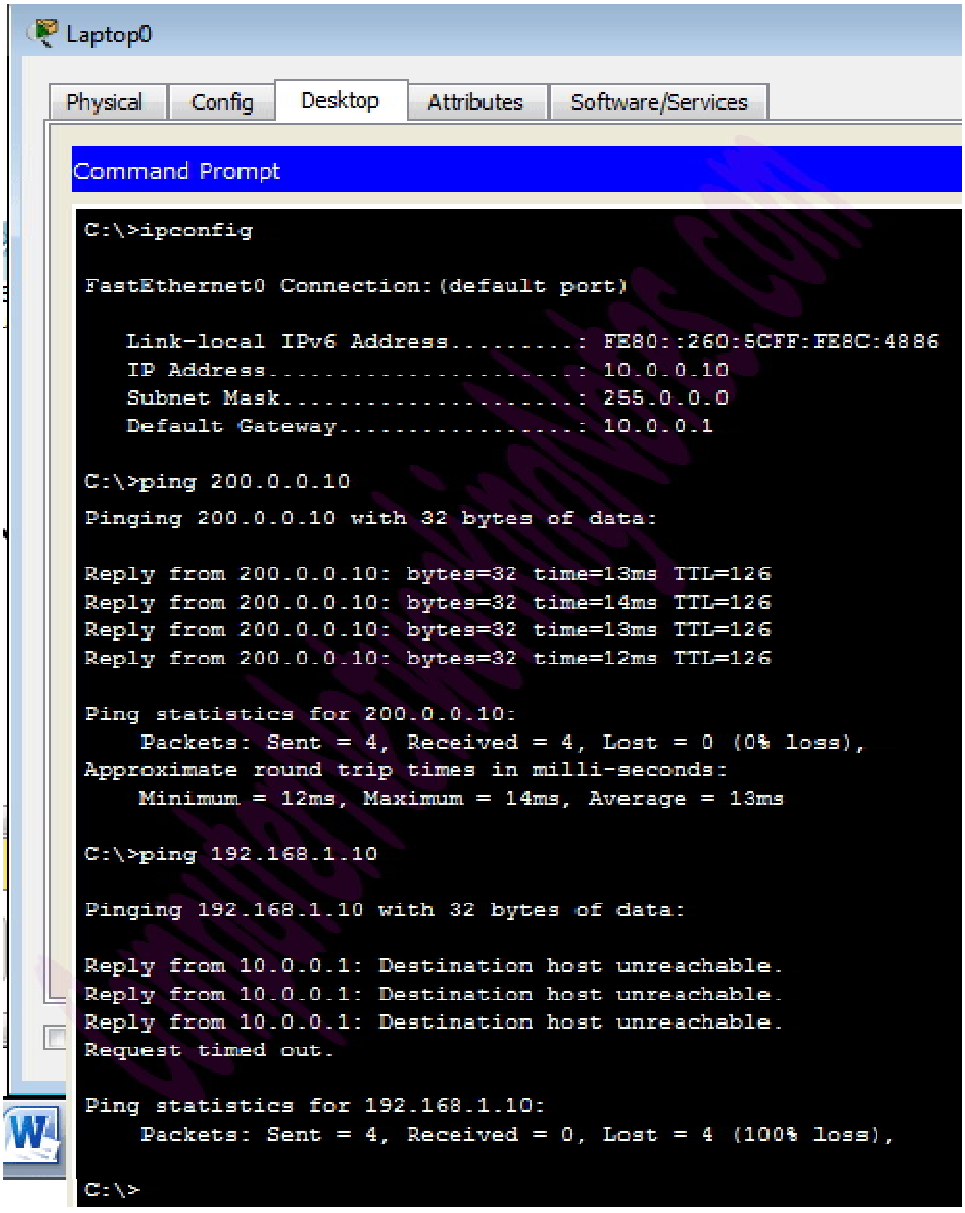
**Testing Static NAT Configuration**

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 | Inside Global IP Address |
|---|---|---|
| Laptop0 | 10.0.0.10 | 50.0.0.10 |
| Server | 192.168.1.10 | 200.0.0.10 |

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.

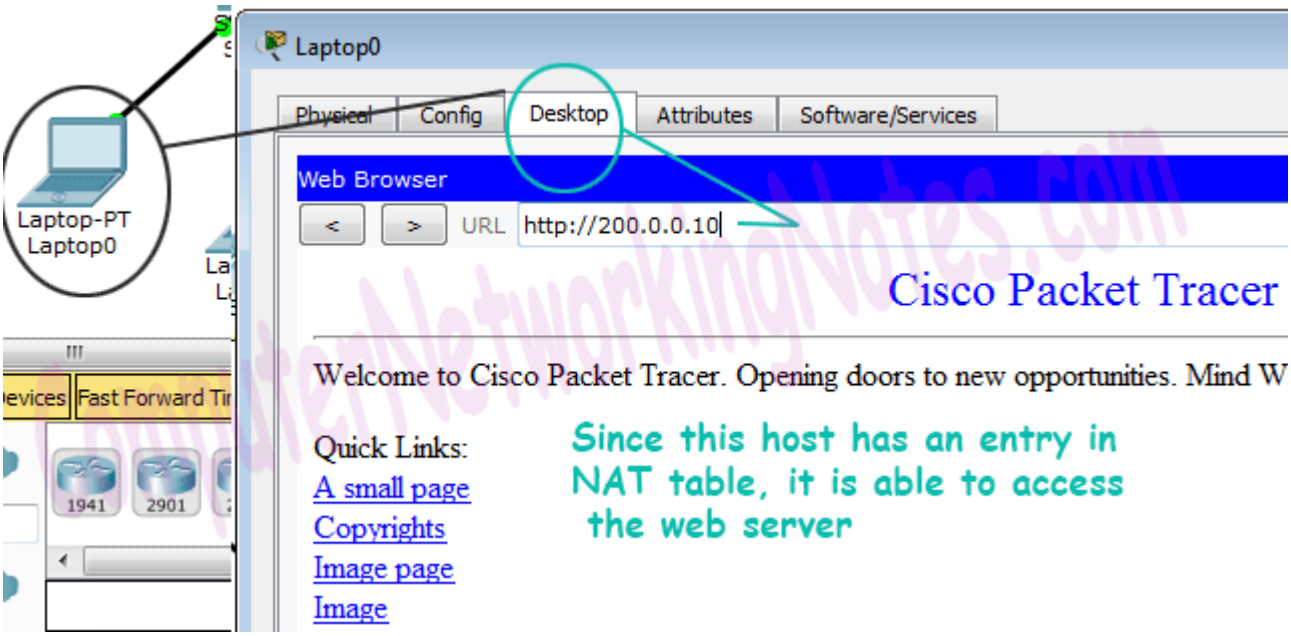



First command verifies that we are testing from correct NAT device.

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.

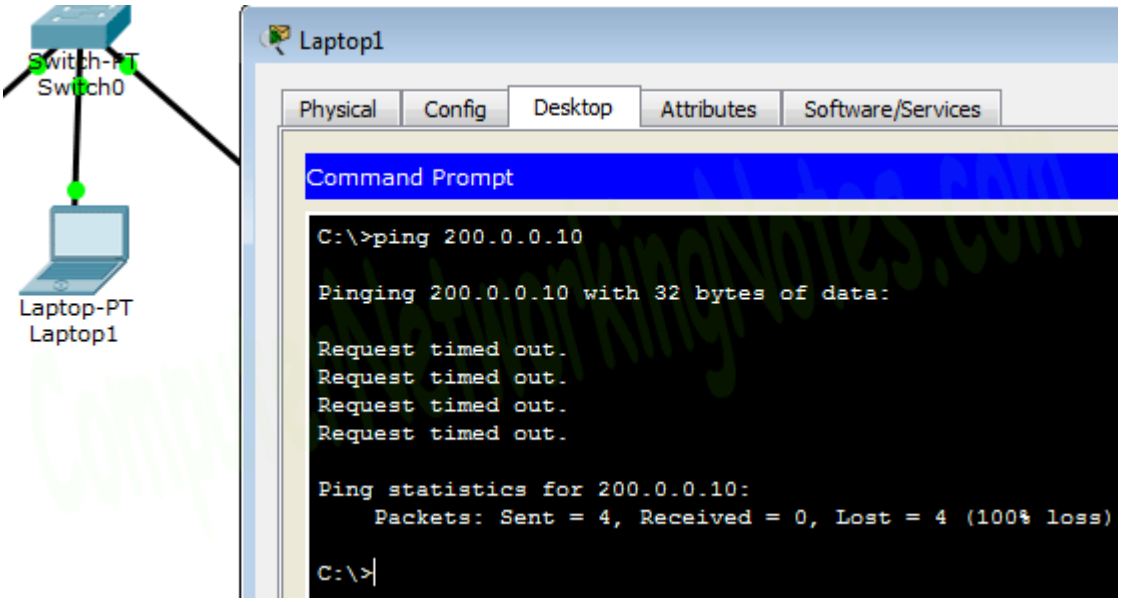
Third command checks whether we are able to access the remote device on its actual IP address or not. A ping error confirms that we are not able to connect with remote device on this IP address.

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




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

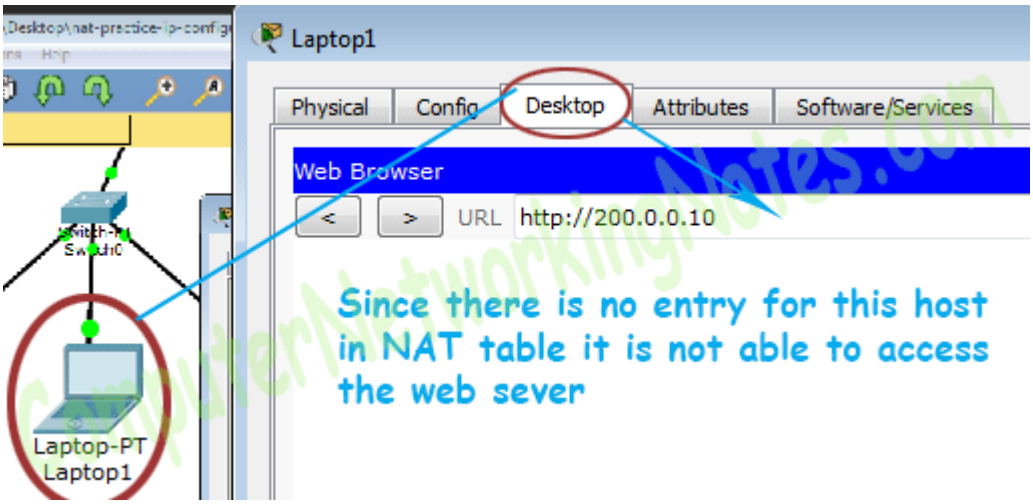
Now run **ping 200.0.0.10** command from Laptop1.




*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 assignment step by step, you should get the same output of testing. Although it's very rare but some time you may get different output. To figure out what went wrong you can use my practice topology with all above configuration. Download my practice topology*

Download NAT Practice LAB with Static NAT configuration

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#
```

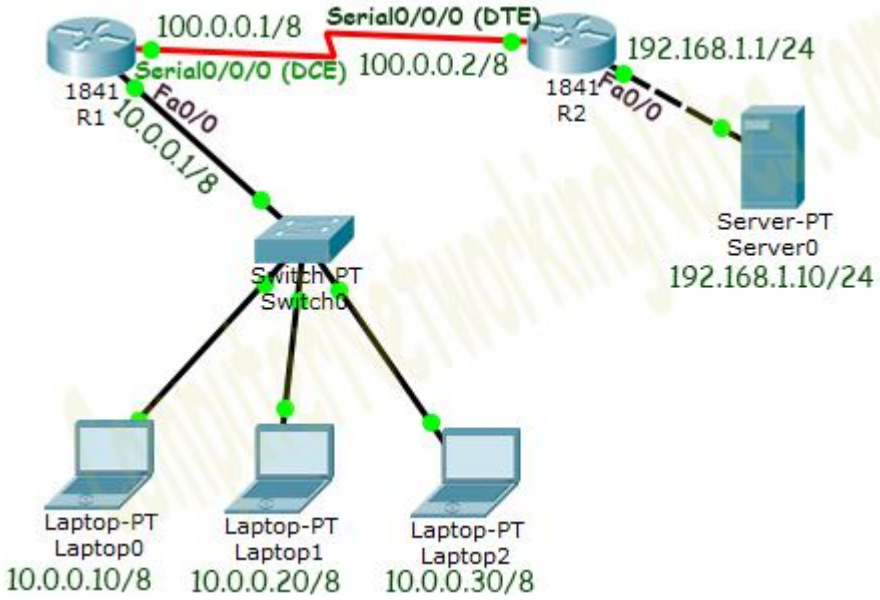
*Pay a little bit extra attention on outside local address filed. Have you noticed one interesting feature of NAT in above output? Why actual outside local IP address is not listed in this filed?*

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.

That's all for this assignment. In next part we will learn dynamic NAT configuration step by step with examples.

## How to Configure Dynamic NAT in Cisco Router



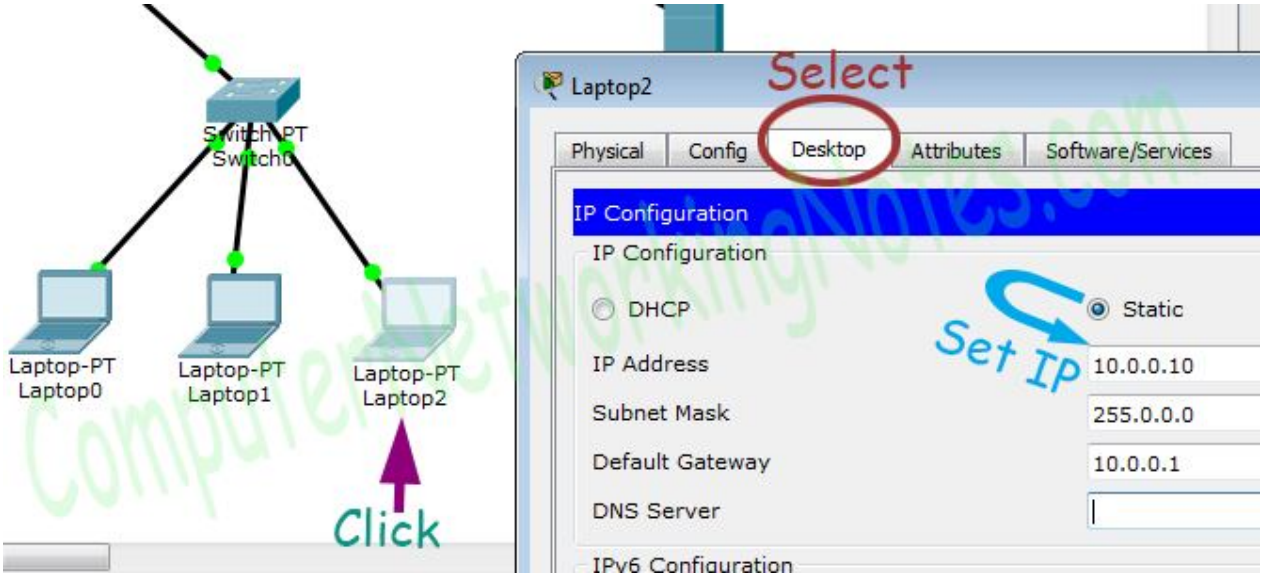

Initial IP Configuration

| 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 |

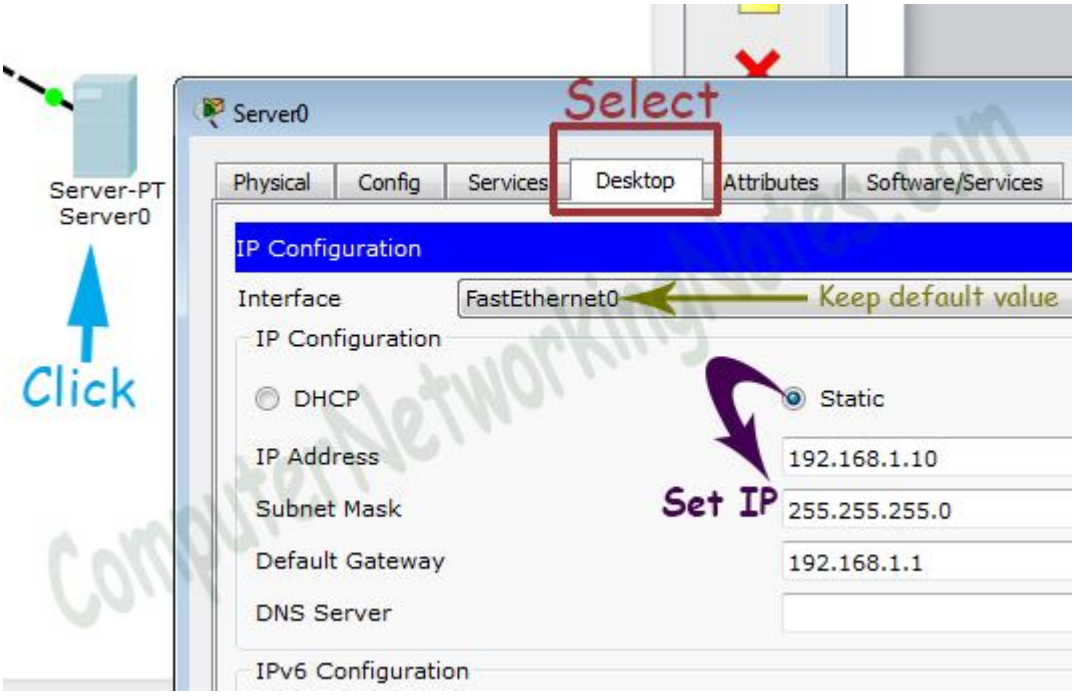
Alternatively you can download my practice topology which is configured with this initial IP configuration.

Download NAT Practice LAB with initial IP configuration

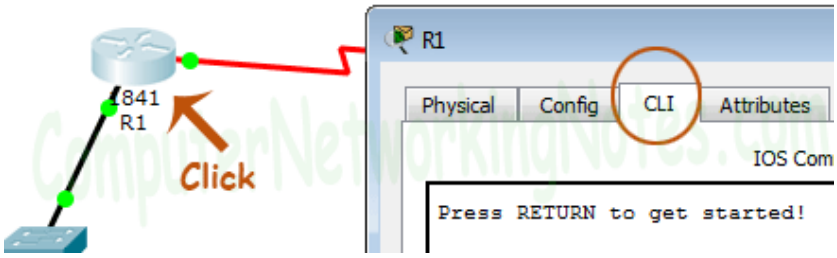
To assign IP address in Laptop click **Laptop** and click **Desktop** and click **IP configuration** and Select **Static** and set **IP address** as given in above table.




Following same way configure IP address in Server.




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




Run following commands to set IP address and hostname.

```
Router>enable
Router# configure terminal
Router(config)#
Router(config)#hostname R1
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
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)#
```

Same way access the command prompt of R2 and run following commands to set IP address and hostname.

```
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 dynamic nat.

Configure Dynamic NAT

Dynamic NAT configuration requires four steps: -

- Create an access list of IP addresses which need translation
- Create a pool of all IP address which are available for translation
- Map access list with pool
- Define inside and outside interfaces

In first step we will create a standard access list which defines which inside local addresses are permitted to map with inside global address.

To create a standard numbered ACL following global configuration mode command is used:-

```
Router(config)# access-list ACL_Identifier_number permit/deny matching-parameters
```

Let's understand this command and its options in detail.

**Router(config)#**

This command prompt indicates that we are in global configuration mode.

**access-list**

Through this parameter we tell router that we are creating or accessing an access list.

**ACL_Identifier_number**

With this parameter we specify the type of access list. We have two types of access list; standard and extended. Both lists have their own unique identifier numbers. Standard ACL uses numbers range 1 to 99 and 1300 to 1999. We can pick any number from this range to tell the router that we are working with standard ACL. This number is used in groping the conditions under a single ACL. This number is also a unique identifier for this ACL in router.

**permit/deny**

An ACL condition has two actions; permit and deny. If we use permit keyword, ACL will allow all packets from the source address specified in next parameter. If we use deny keyword, ACL will drop all packets from the source address specified in next parameter.

**matching-parameters**

This parameter allows us to specify the contents of packet that we want to match. In a standard ACL condition it could be a single source address or a range of addresses. We have three options to specify the source address.

- Any
- host
- A.B.C.D

**Any**

Any keyword is used to match all sources. Every packet compared against this condition would be matched.

**Host**

Host keyword is used to match a specific host. To match a particular host, type the keyword host and then the IP address of host.

**A.B.C.D**

Through this option we can match a single address or a range of addresses. To match a single address, simply type its address. To match a range of addresses, we need to use wildcard mask.

**Wildcard mask**

Just like subnet mask, wildcard mask is also used to draw a boundary in IP address. Where subnet mask is used to separate network address from host address, wildcard mask is used to distinguish the matching portion from the rest. Wildcard mask is the invert of Subnet mask. Wildcard can be calculated in decimal or in binary from subnet mask.

We have three hosts in lab. Let's create a standard access list which allows two hosts and denies one host.

```
R1(config)#access-list 1 permit 10.0.0.10 0.0.0.0
R1(config)#access-list 1 permit 10.0.0.20 0.0.0.0
R1(config)#access-list 1 deny any
```

In second step we define a pool of inside global addresses which are available for translation.

Following command is used to define the NAT pool.

```
Router(config)#ip nat pool [Pool Name] [Start IP address] [End IP address] netmask [Subnet mask]
```

This command accepts four options pool name, start IP address, end IP address and Subnet mask.

**Pool Name**: - This is the name of pool. We can choose any descriptive name here.

**Start IP Address**: - First IP address from the IP range which is available for translation.

**End IP Address**: - Last IP address from the IP range which is available for translation. There is no minimum or maximum criteria for IP range for example we can have a range of single IP address or we can have a range of all IP address from a subnet.

**Subnet Mask**: - Subnet mask of IP range.

Let's create a pool named ccna with an IP range of two addresses.

```
R1(config)#ip nat pool ccna 50.0.0.1 50.0.0.2 netmask 255.0.0.0
```

This pool consist two class A IP address 50.0.0.1 and 50.0.0.2.

In third step we map access list with pool. Following command will map the access list with pool and configure the dynamic NAT.

```
Router(config)#ip nat inside source list [access list name or number] pool [pool name]
```

This command accepts two options.

**Access list name or number**: - Name or number the access list which we created in first step.

**Pool Name**: - Name of pool which we created in second step.

In first step we created a standard access list with number **1** and in second step we created a pool named **ccna**. To configure a dynamic NAT with these options we will use following command.

```
R1(config)#ip nat inside source list 1 pool ccna
```

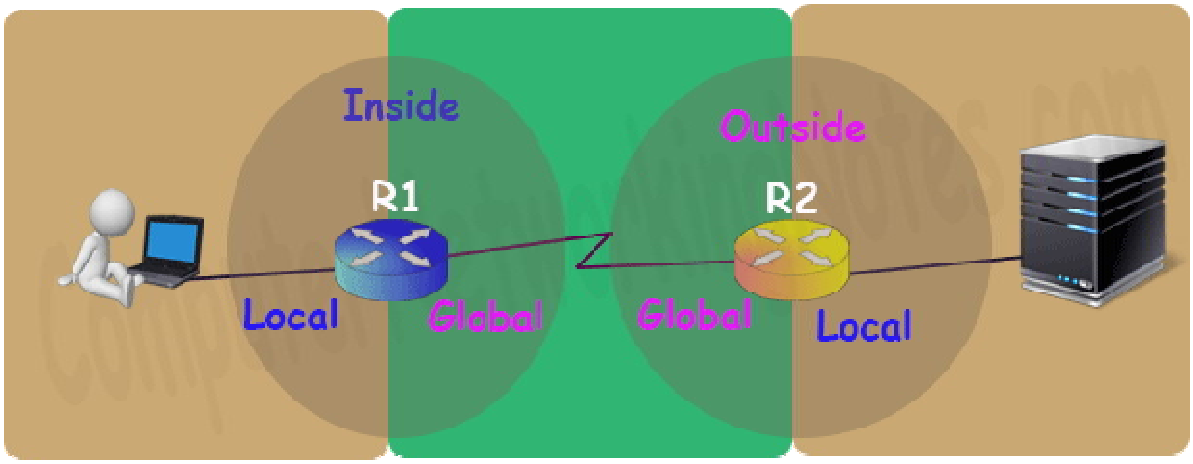
Finally we have to define which interface is connected with local network and which interface is connected with global network.

To define an inside local we use following command

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

Following command defines inside global

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




Let's implement all these commands together and configure the dynamic NAT.

R1 Dynamic NAT Configuration

```
R1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 1 permit 10.0.0.10 0.0.0.0
R1(config)#access-list 1 permit 10.0.0.20 0.0.0.0
R1(config)#access-list 1 deny any
R1(config)#ip nat pool ccna 50.0.0.1 50.0.0.2 netmask 255.0.0.0
R1(config)#ip nat inside source list 1 pool ccna
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#interface Serial0/0/0
R1(config-if)#ip nat outside
```

```
R1(config-if)#exit
R1(config)#
```

For testing purpose I configured dynamic translations for two addresses only.

On R2 we can keep standard configuration or can configure dynamic NAT as we just did in R1 or can configure static NAT as we learnt in pervious part of this article.

Let's do a quick recap of what we learnt in previous part and configure static NAT on R2.

```
R2>enable
R2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip nat inside source static 192.168.1.10 200.0.0.10
R2(config)#interface Serial 0/0/0
R2(config-if)#ip nat outside
R2(config-if)#exit
R2(config)#interface FastEthernet 0/0
R2(config-if)#ip nat inside
R2(config-if)#exit
R2(config)#
```

To understand above commands in detail please see the second part of this assignment.

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 assignment explain routing in detail with examples

Routing Protocols Explained in details

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
```

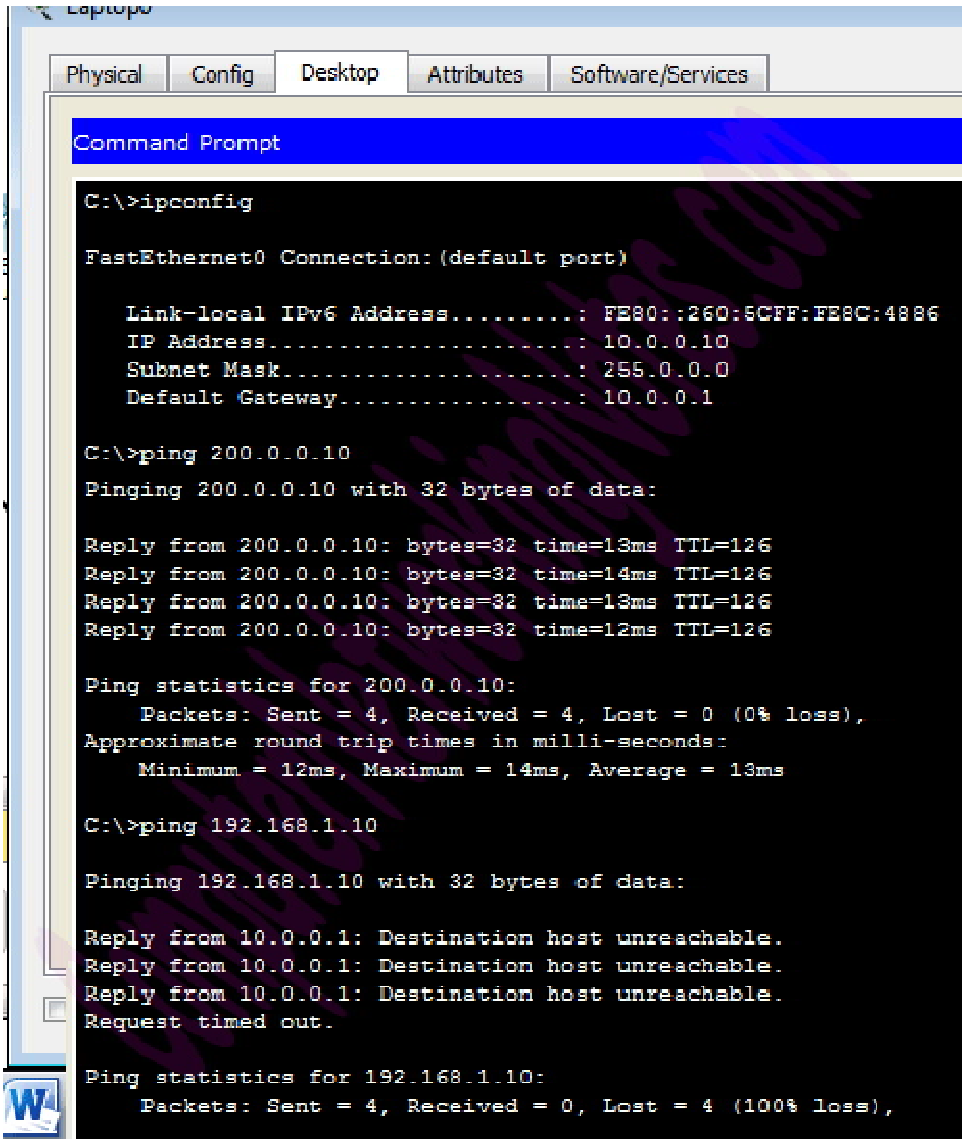
**Testing Dynamic NAT Configuration**

In this lab we configured dynamic NAT on R1for 10.0.0.10 and 10.0.0.20 and static NAT on R2 for 192.168.1.10.

| Device | Inside Local IP Address | Inside Global IP Address |
|---|---|---|
| Laptop0 | 10.0.0.10 | 50.0.0.1 |
| Laptop1 | 10.0.0.20 | 50.0.0.2 |
| Server | 192.168.1.10 | 200.0.0.10 |

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.

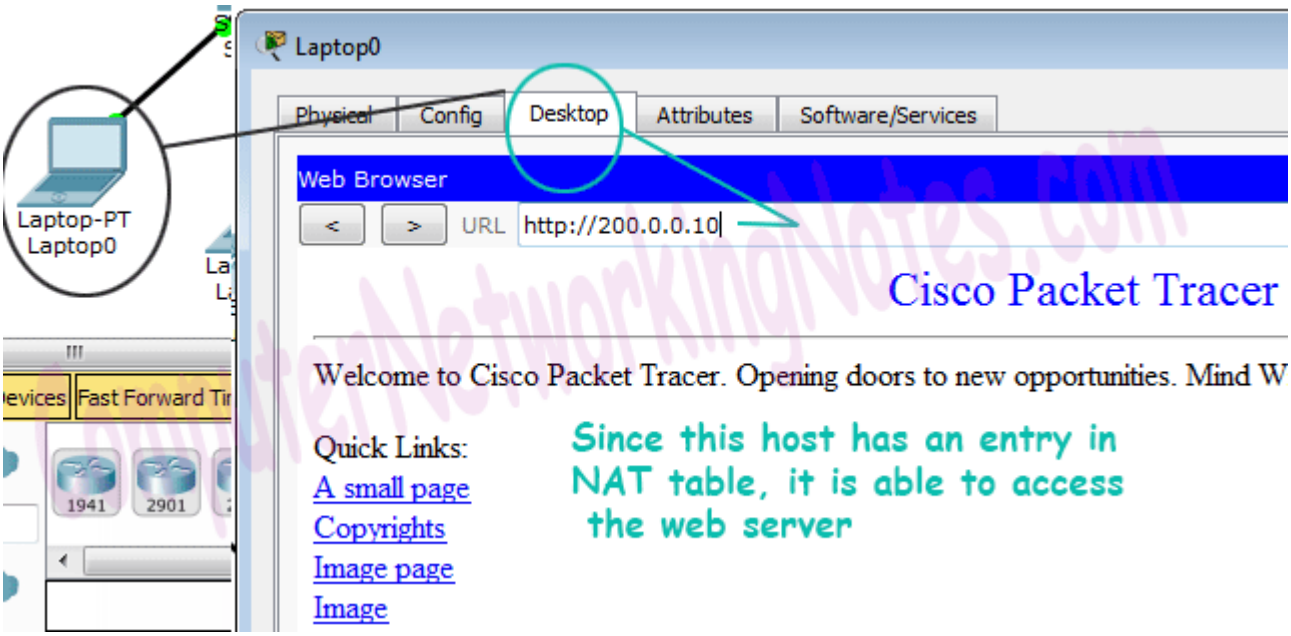



First command verifies that we are testing from correct NAT device.

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.

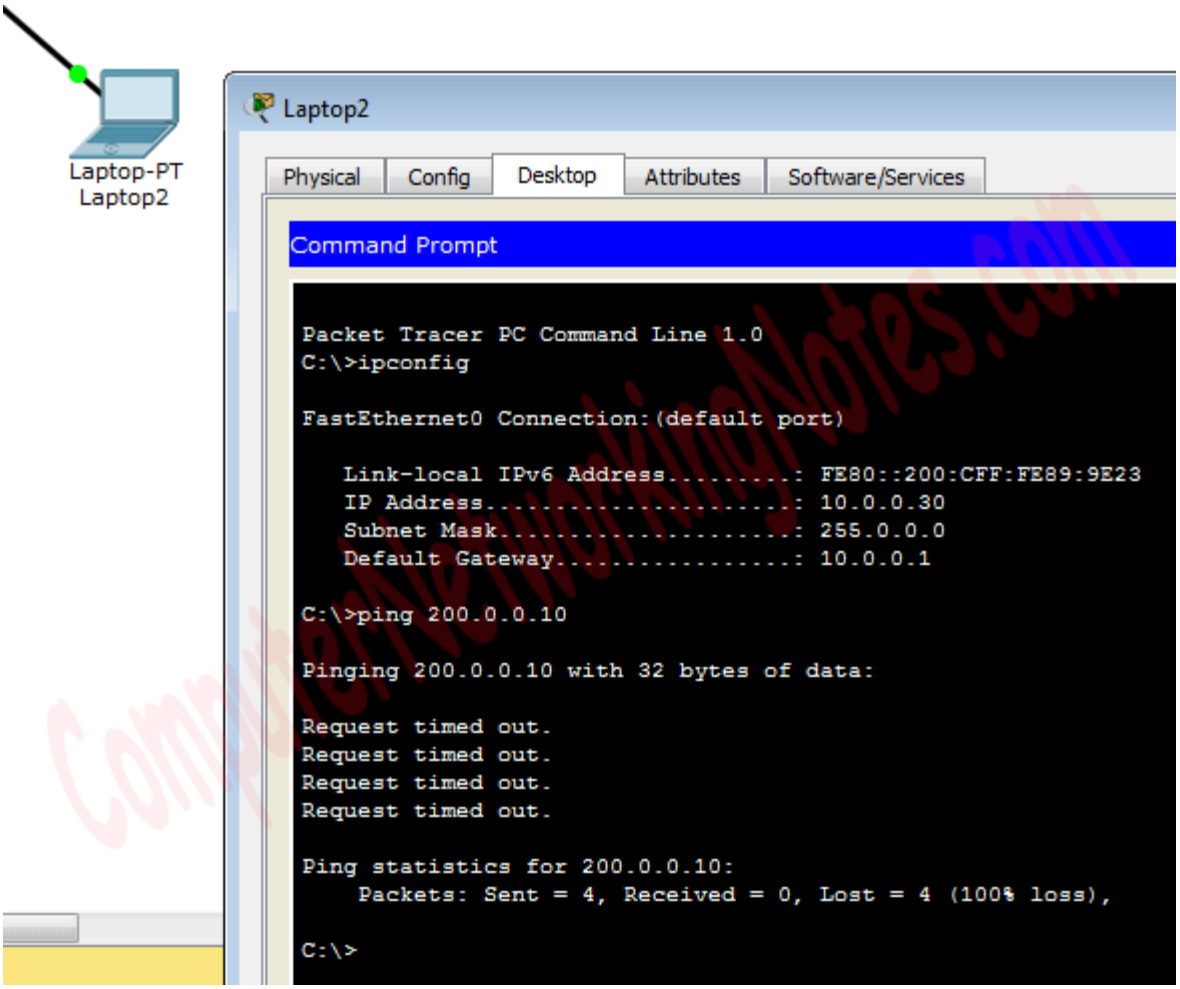
Third command checks whether we are able to access the remote device on its actual IP address or not. A ping error confirms that we are not able to connect with remote device on this IP address.

Let's do one more testing. Close the command prompt and click web server and access 200.0.0.10.

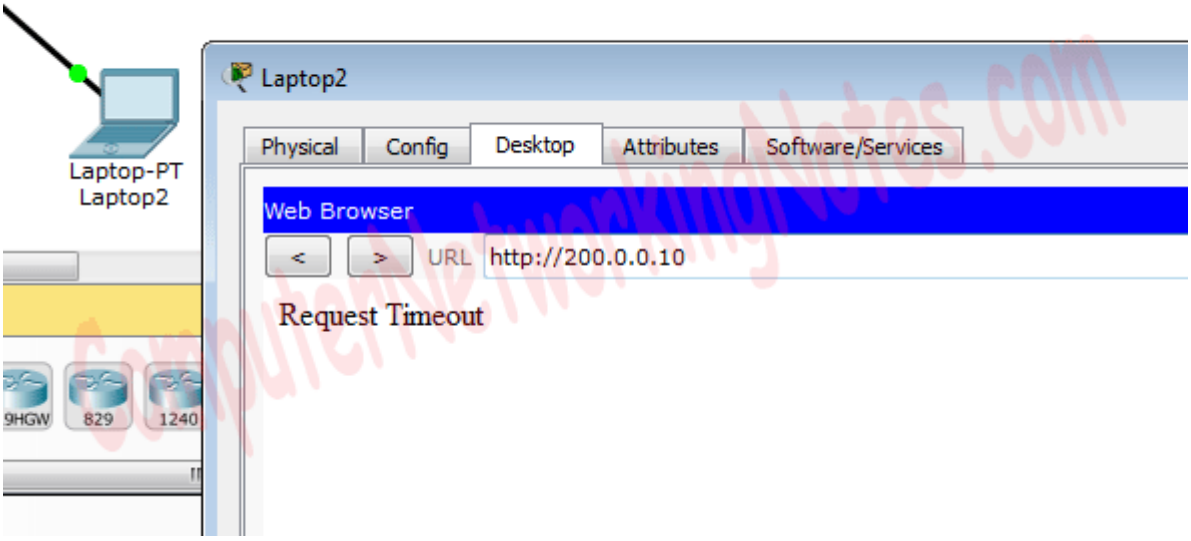



Above figure confirms that host 10.0.0.10 is able to access the 200.0.0.10. You can also do the same testing from Laptop1, result will be same.

Now run ping 200.0.0.10 command from Laptop2.




Close the command prompt and access web server from this host.




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

Because we configured NAT only for two hosts (Laptop0 and Laptop1) which IP addresses are 10.0.0.10 and 10.0.0.20. So only the host 10.0.0.10 and 10.0.0.20 will be able to access the remote device.

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

Following figure illustrates this translation on router R1.

```
R1>en
R1#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
tcp 50.0.0.1:1025      10.0.0.10:1025     200.0.0.10:80      200.0.0.10:80
tcp 50.0.0.2:1025      10.0.0.20:1025     200.0.0.10:80      200.0.0.10:80

R1#
```

We did three tests one from each host, but why only two tests are listed here? Remember in first step we created an access list. Access list filters the unwanted traffic before it reaches to the NAT. We can see how many packets are blocked by ACL with following command

R1#show ip access-lists 1

```
R1#show ip access-lists 1
Standard IP access list 1
    permit host 10.0.0.10 (8 match(es))
    permit host 10.0.0.20 (2 match(es))
    deny any (3 match(es))

R1#
```

Basically it is access list which filters the traffic. NAT does not filter any traffic it only translate the address.

Following figure illustrate NAT translation on router R2

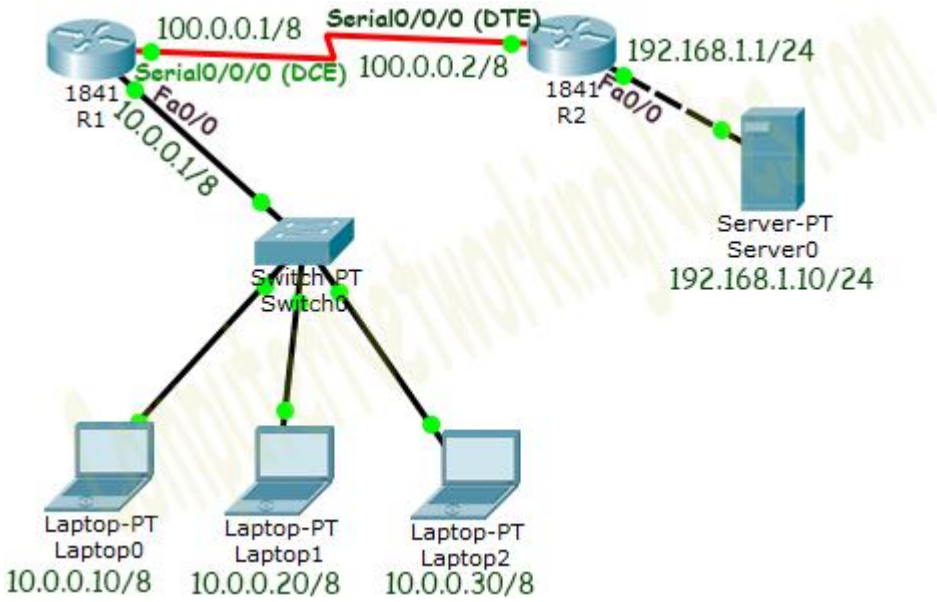
```
R2>enable
R2#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
---  200.0.0.10        192.168.1.10       ---                ---
tcp 200.0.0.10:80      192.168.1.10:80    50.0.0.1:1025      50.0.0.1:1025
tcp 200.0.0.10:80      192.168.1.10:80    50.0.0.2:1025      50.0.0.2:1025

R2#
```

That's all for this assignment. In next part we will learn NAT overload (PAT) configuration step by step with examples.

## Configure PAT in Cisco Router

This assignment explains how to configure PAT (Port Address Translation) also known NAT Overload in Cisco Router.

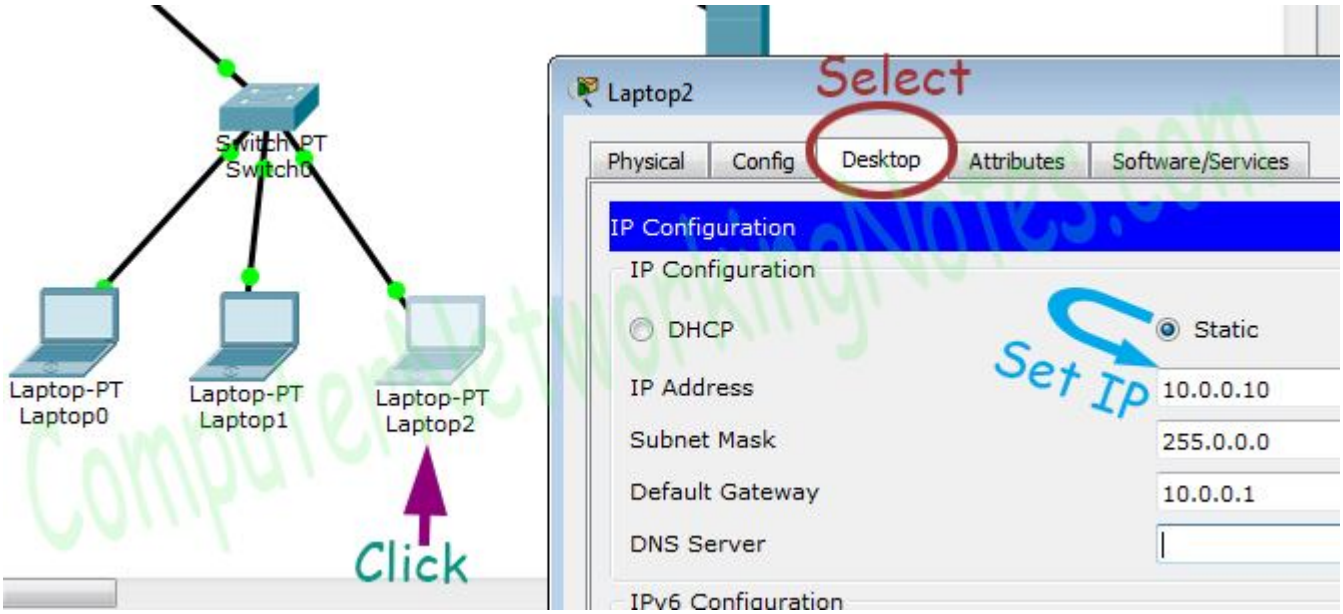



Initial IP Configuration

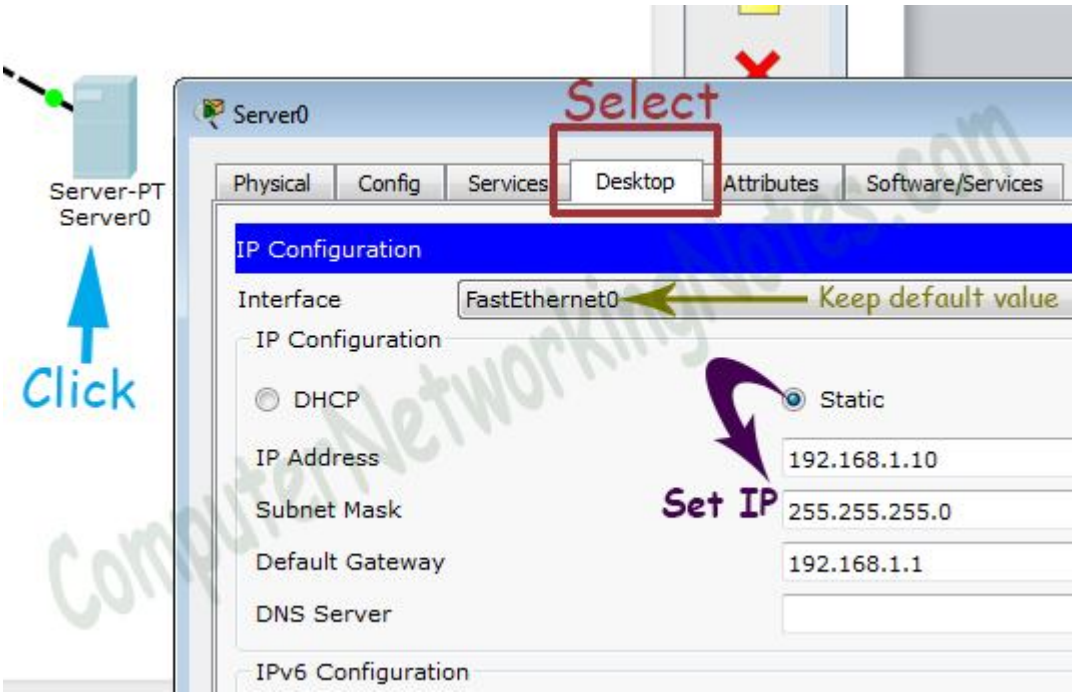
| **Device / Interface** | **IP Address** | **Connected With** |
|---|---|---|
| Laotop0 | 10.0.0.10/8 | Fa0/0 of R0 |
| Laptop1 | 10.0.0.20/8 | Fa0/0 of R0 |
| Laptop2 | 10.0.0.30/8 | Fa0/0 of R0 |
| Server0 | 192.168.1.10/24 | Fa0/0 of R1 |
| Serial 0/0/0 of R1 | 100.0.0.1/8 | Serial 0/0/0 of R2 |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

Alternatively you can download my practice topology which is configured with this initial IP configuration.

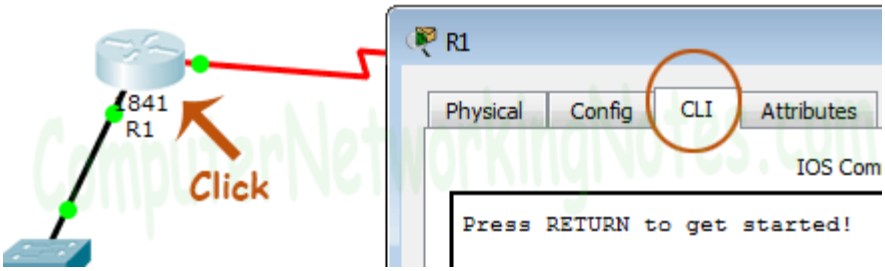
To assign IP address in Laptop click **Laptop** and click **Desktop** and click **IP configuration** and Select **Static** and set **IP address** as given in above table.




Following same way configure IP address in Server.




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



Run following commands to set IP address and hostname.

```
Router>enable
Router# configure terminal
Router(config)#
Router(config)#hostname R1
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
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)#
```

Same way access the command prompt of R2 and run following commands to set IP address and hostname.

```
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 pat.

**Configure PAT (NAT Overload)**

PAT configuration requires four steps: -

- Create an access list of IP addresses which need translation
- Create a pool of all IP address which are available for translation
- Map access list with pool
- Define inside and outside interfaces

In first step we will create a standard access list which defines which inside local addresses are permitted to map with inside global address.

To create a standard numbered ACL following global configuration mode command is used:-

```
Router(config)# access-list ACL_Identifier_number permit/deny matching-parameters
```

Let's understand this command and its options in detail.

**Router(config)#**

This command prompt indicates that we are in global configuration mode.

**access-list**

Through this parameter we tell router that we are creating or accessing an access list.

**ACL_Identifier_number**

With this parameter we specify the type of access list. We have two types of access list; standard and extended. Both lists have their own unique identifier numbers. Standard ACL uses numbers range 1 to 99 and 1300 to 1999. We can pick any number from this range to tell the router that we are working with standard ACL. This number is used in groping the conditions under a single ACL. This number is also a unique identifier for this ACL in router.

**permit/deny**

An ACL condition has two actions; permit and deny. If we use permit keyword, ACL will allow all packets from the source address specified in next parameter. If we use deny keyword, ACL will drop all packets from the source address specified in next parameter.

**matching-parameters**

This parameter allows us to specify the contents of packet that we want to match. In a standard ACL condition it could be a single source address or a range of addresses. We have three options to specify the source address.

- Any
- host
- A.B.C.D

**Any**

Any keyword is used to match all sources. Every packet compared against this condition would be matched.

**Host**

Host keyword is used to match a specific host. To match a particular host, type the keyword host and then the IP address of host.

**A.B.C.D**

Through this option we can match a single address or a range of addresses. To match a single address, simply type its address. To match a range of addresses, we need to use wildcard mask.

**Wildcard mask**

Just like subnet mask, wildcard mask is also used to draw a boundary in IP address. Where subnet mask is used to separate network address from host address, wildcard mask is used to distinguish the matching portion from the rest. Wildcard mask is the invert of Subnet mask. Wildcard can be calculated in decimal or in binary from subnet mask.

We have three hosts in lab. Let's create a standard access list which allows two hosts and denies one host.

```
R1(config)#access-list 1 permit 10.0.0.10 0.0.0.0
R1(config)#access-list 1 permit 10.0.0.20 0.0.0.0
R1(config)#access-list 1 deny any
```

To learn standard ACL in detail you can use following assignment.

Standard ACL Explained with Examples

In second step we define a pool of inside global addresses which are available for translation.

Following command is used to define the NAT pool.

```
Router(config)#ip nat pool [Pool Name] [Start IP address] [End IP address] netmask [Subnet mask]
```

This command accepts four options pool name, start IP address, end IP address and Subnet mask.

**Pool Name**: - This is the name of pool. We can choose any descriptive name here.

**Start IP Address**: - First IP address from the IP range which is available for translation.

**End IP Address**: - Last IP address from the IP range which is available for translation. There is no minimum or maximum criteria for IP range for example we can have a range of single IP address or we can have a range of all IP address from a subnet.

**Subnet Mask**: - Subnet mask of IP range.

Let's create a pool named ccna with a single IP address.

```
R1(config)#ip nat pool ccna 50.0.0.1 50.0.0.1 netmask 255.0.0.0
```

In third step we map access list with pool. Following command will map the access list with pool and configure the PAT.

```
Router(config)#ip nat inside source list [access list name or number] pool [pool name]overload
```

This command accepts two options.

**Access list name or number**: - Name or number the access list which we created in first step.

**Pool Name**: - Name of pool which we created in second step.

In first step we created a standard access list with number 1 and in second step we created a pool named ccna. To configure a PAT with these options we will use following command.

```
R1(config)#ip nat inside source list 1 pool ccna overload
```

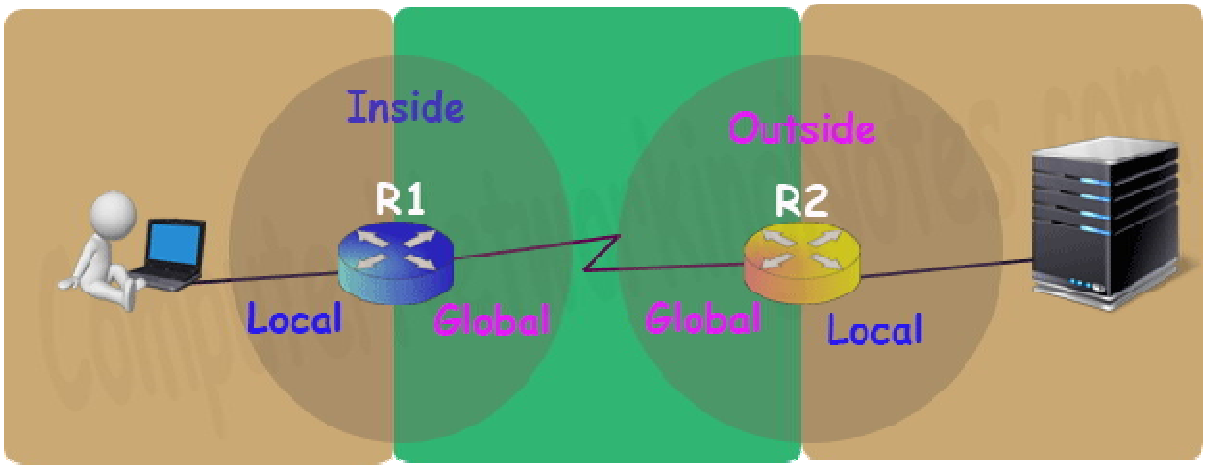
Finally we have to define which interface is connected with local network and which interface is connected with global network.

To define an inside local we use following command

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

Following command defines inside global

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




Let's implement all these commands together and configure the PAT.

R1 PAT (NAT Overload) Configuration

```
R1>enable
R1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 1 permit 10.0.0.10 0.0.0.0
R1(config)#access-list 1 permit 10.0.0.20 0.0.0.0
R1(config)#access-list 1 deny any
R1(config)#ip nat pool ccna 50.0.0.1 50.0.0.1 netmask 255.0.0.0
R1(config)#ip nat inside source list 1 pool ccna overload
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#interface Serial 0/0/0
R1(config-if)#ip nat outside
```

```
R1(config-if)#exit
R1(config)#
```

For testing purpose I configured pat translations for two addresses only.

On R2 we can keep standard configuration or can configure dynamic NAT or can configure static NAT as we learnt in pervious parts of this article.

Let's do a quick recap of what we learnt in previous part and configure static NAT on R2.

```
R2>enable
R2#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#ip nat inside source static 192.168.1.10 200.0.0.10
R2(config)#interface Serial 0/0/0
R2(config-if)#ip nat outside
R2(config-if)#exit
R2(config)#interface FastEthernet 0/0
R2(config-if)#ip nat inside
R2(config-if)#exit
R2(config)#
```

To understand above commands in detail please see the second part of this assignment.

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 assignment explain routing in detail with examples

Routing Protocol Explained

**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
```

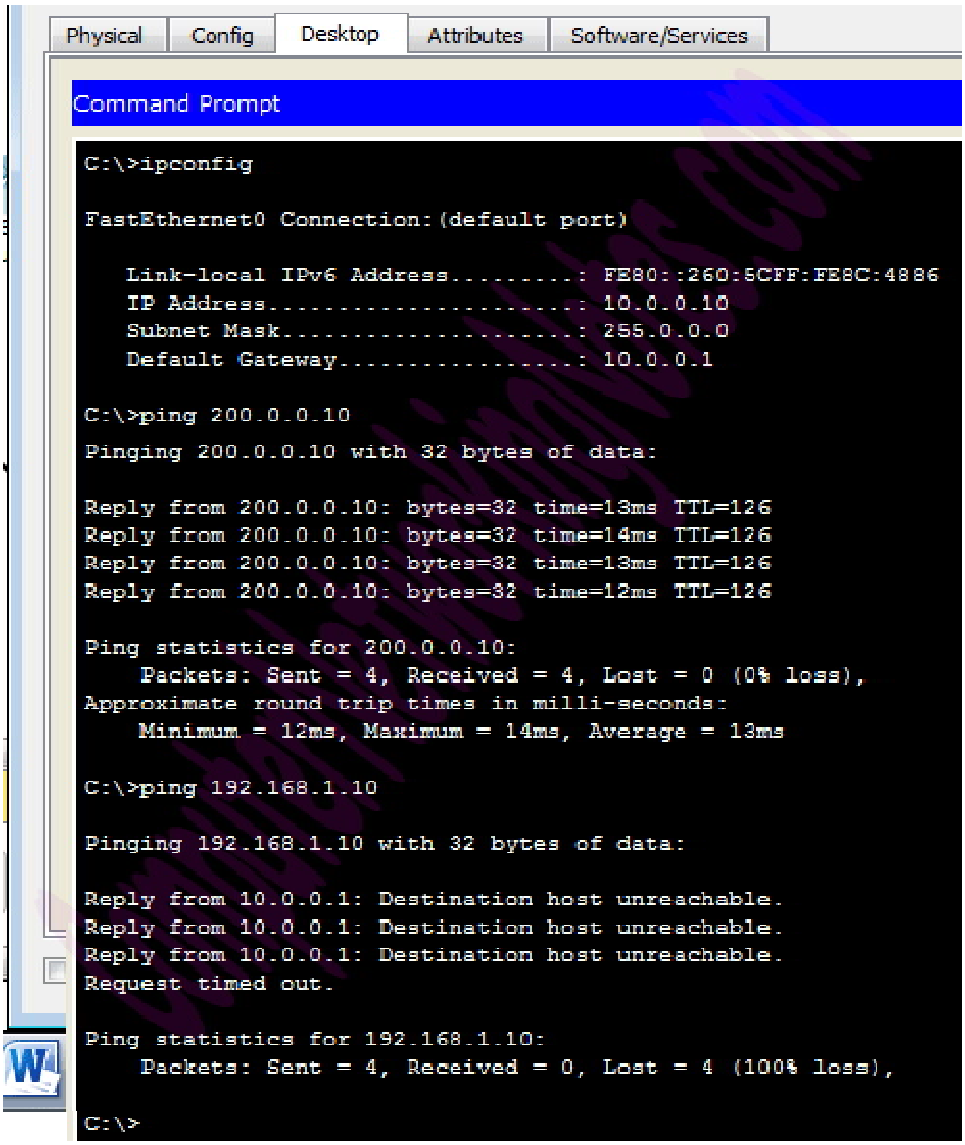
**Testing PAT Configuration**

In this lab we configured PAT on R1for 10.0.0.10 and 10.0.0.20 and static NAT on R2 for 192.168.1.10.

| Device | Inside Local IP Address | Inside Global IP Address |
|---|---|---|
| Laptop0 | 10.0.0.10 | 50.0.0.1 |
| Laptop1 | 10.0.0.20 | 50.0.0.2 |
| Server | 192.168.1.10 | 200.0.0.10 |

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.

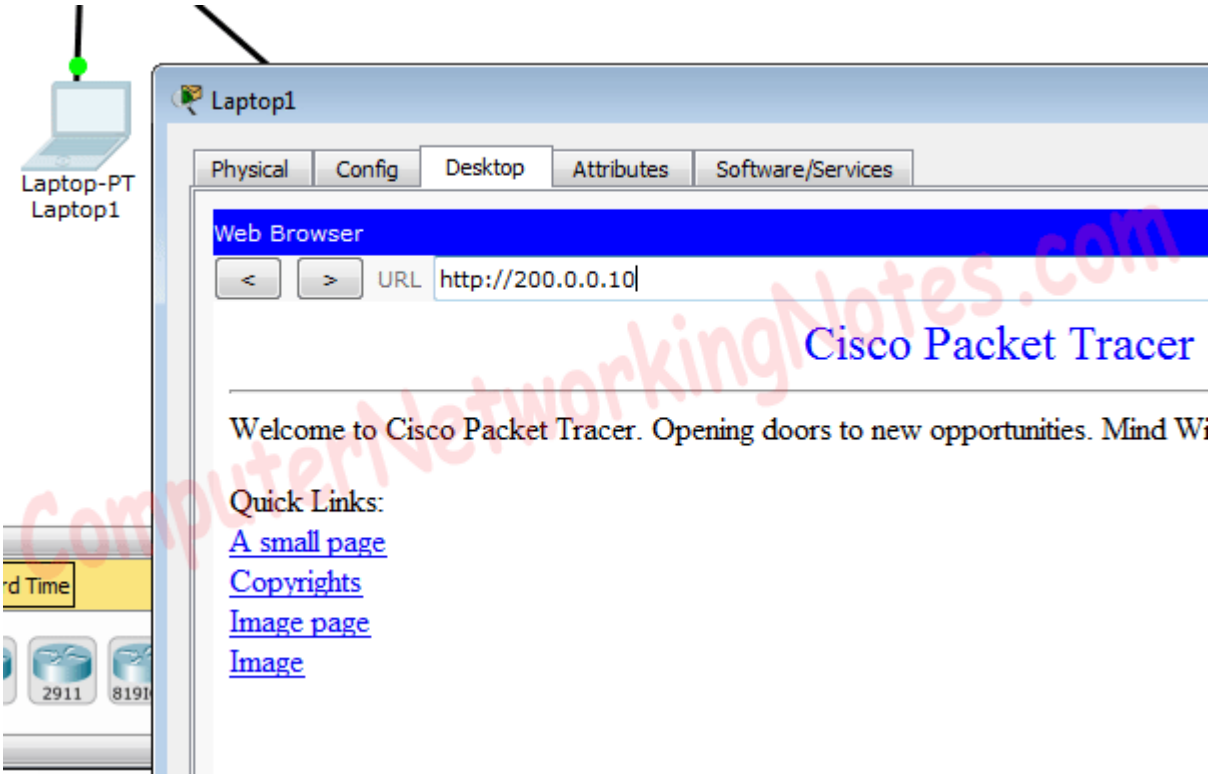



First command verifies that we are testing from correct NAT device.

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.

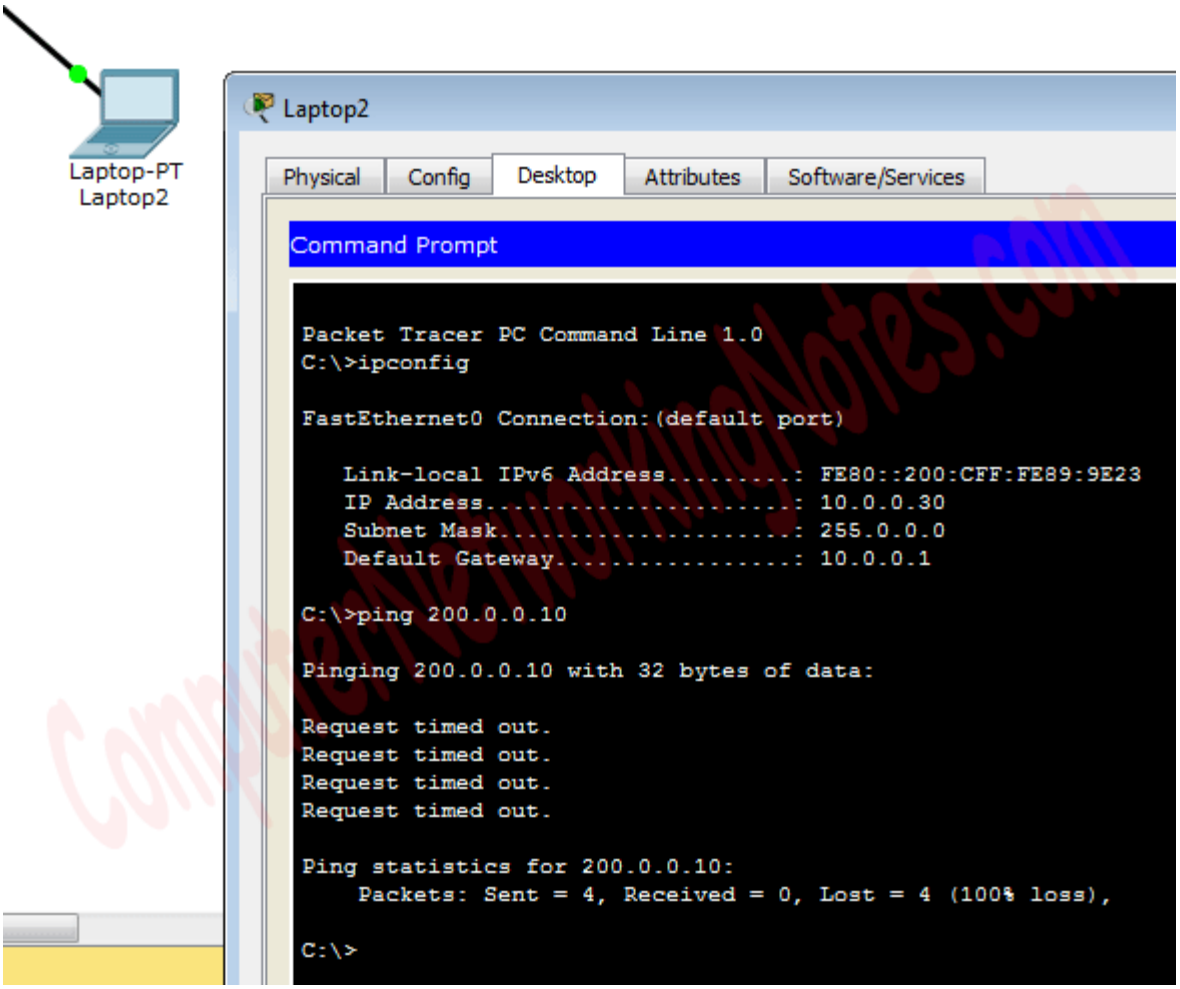
Third command checks whether we are able to access the remote device on its actual IP address or not. A ping error confirms that we are not able to connect with remote device on this IP address.

Let's do one more testing. Close the command prompt and click web server and access 200.0.0.10.

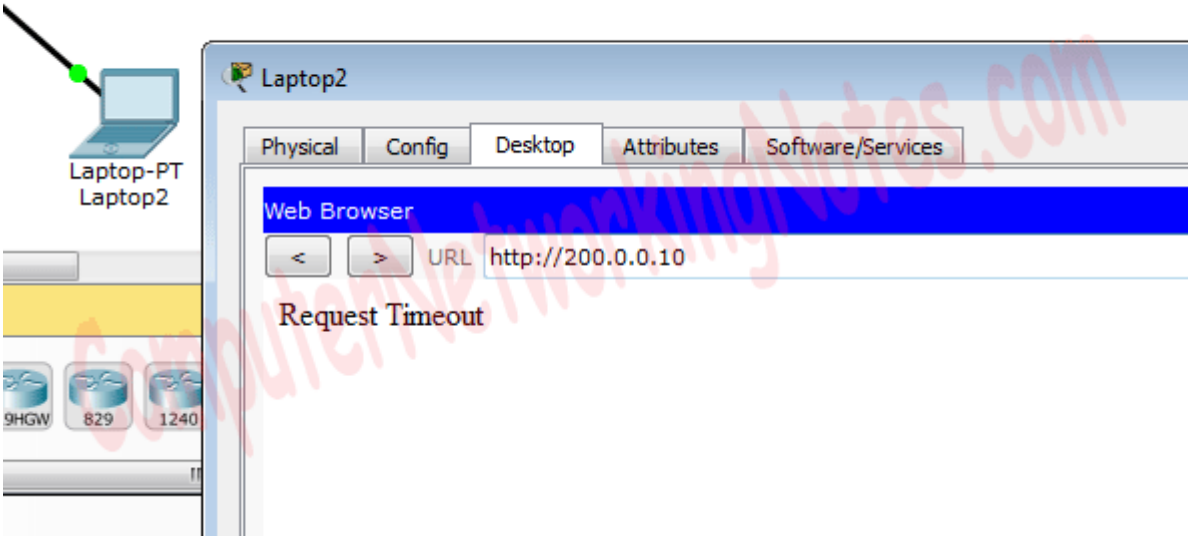



Above figure confirms that host 10.0.0.10 is able to access the 200.0.0.10. You can also do the same testing from Laptop1, result will be same.

Now run **ping 200.0.0.10** command from Laptop2.




Close the command prompt and access web server from this host.




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

Because we configured PAT only for two hosts (Laptop0 and Laptop1) which IP addresses are 10.0.0.10 and 10.0.0.20. So only the host 10.0.0.10 and 10.0.0.20 will be able to access the remote device.

*If you followed this assignment step by step, you should get the same output of testing. Although it's very rare but some time you may get different output. To figure out what went wrong you can use my practice topology with all above configuration. Download my practice topology*

Download NAT Practice LAB with PAT configuration

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 translation
Pro  Inside global     Inside local       Outside local      Outside global
icmp 50.0.0.1:1        10.0.0.20:1        200.0.0.10:1       200.0.0.10:1
icmp 50.0.0.1:2        10.0.0.20:2        200.0.0.10:2       200.0.0.10:2
icmp 50.0.0.1:3        10.0.0.20:3        200.0.0.10:3       200.0.0.10:3
icmp 50.0.0.1:4        10.0.0.20:4        200.0.0.10:4       200.0.0.10:4
tcp 50.0.0.1:1024      10.0.0.10:1025     200.0.0.10:80      200.0.0.10:80
tcp 50.0.0.1:1025      10.0.0.20:1025     200.0.0.10:80      200.0.0.10:80

R1#
```

As we can see in above output same inside global IP address is used to translate all the inside local IP addresses. For each inside local IP address a unique port number is used.

Following figure illustrate NAT translation on router R2

```
R2#show ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
icmp 200.0.0.10:1      192.168.1.10:1     50.0.0.1:1         50.0.0.1:1
icmp 200.0.0.10:2      192.168.1.10:2     50.0.0.1:2         50.0.0.1:2
icmp 200.0.0.10:3      192.168.1.10:3     50.0.0.1:3         50.0.0.1:3
icmp 200.0.0.10:4      192.168.1.10:4     50.0.0.1:4         50.0.0.1:4
---  200.0.0.10        192.168.1.10       ---                ---
tcp 200.0.0.10:80      192.168.1.10:80    50.0.0.1:1024      50.0.0.1:1024
tcp 200.0.0.10:80      192.168.1.10:80    50.0.0.1:1025      50.0.0.1:1025

R2#
```

In above output the Outside global field also confirms that all packets are coming from single IP address.