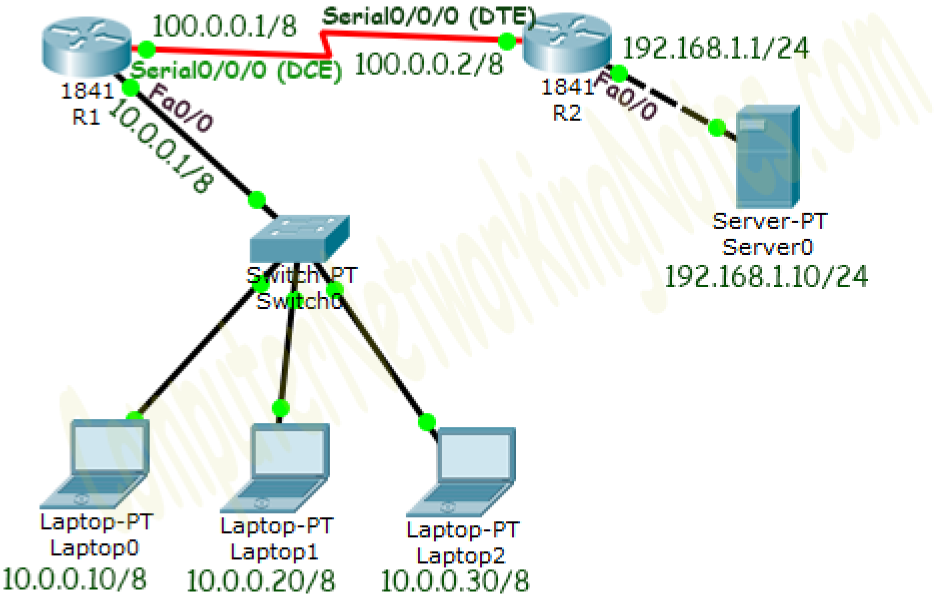
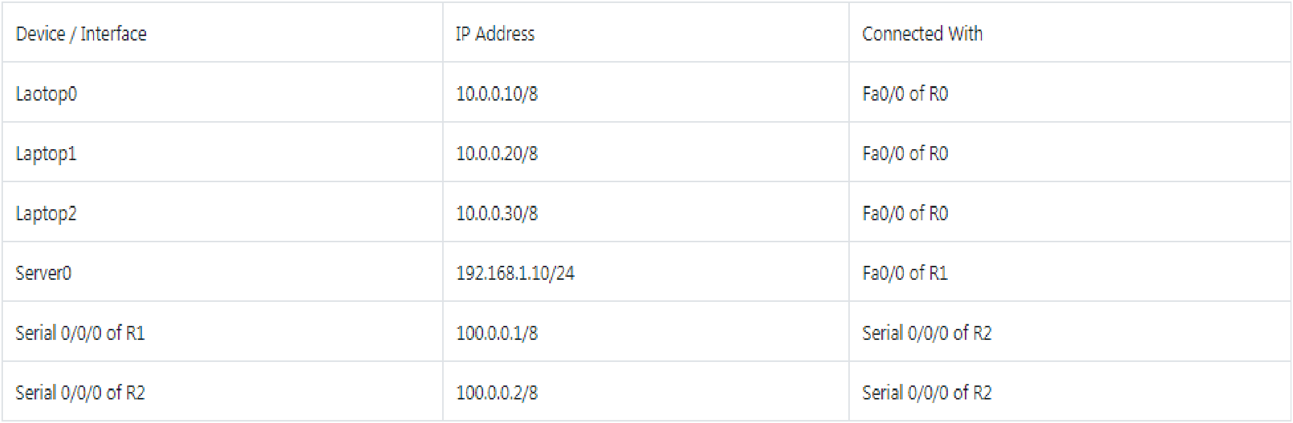
LAB 7A - Configuring Static and Dynamic NAT LAB

Create this network topology lab as shown in following figure

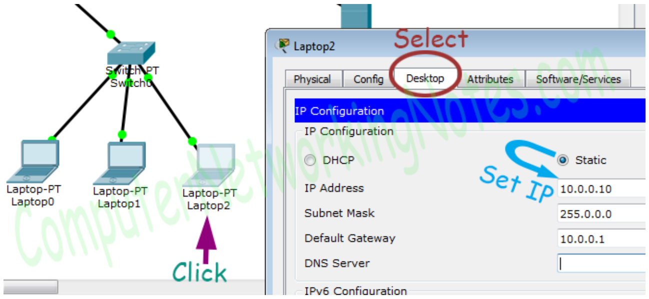


Initial IP Configuration



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.



IP Configuration

○ DHCP ● Static

IP Address 10.0.0.10

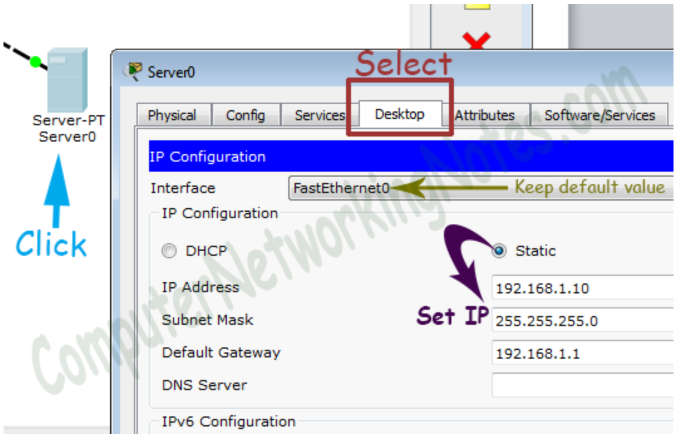
Subnet Mask 255.0.0.0

Default Gateway 10.0.0.1

DNS Server

IPv6 Configuration

Following same way configure IP address in Server.



IP Configuration

Interface FastEthernet0

IP Configuration

○ DHCP ● Static

IP Address 192.168.1.10

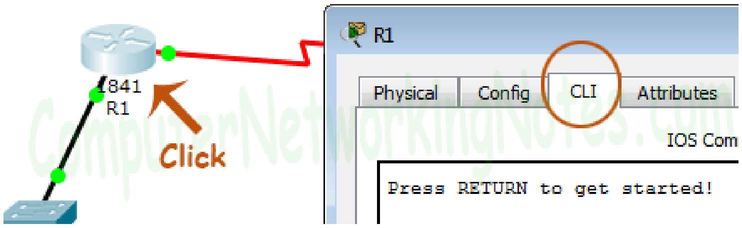
Subnet Mask 255.255.255.0

Default Gateway 192.168.1.1

DNS Server

IPv6 Configuration

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



```
Press RETURN to get started!
```

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.

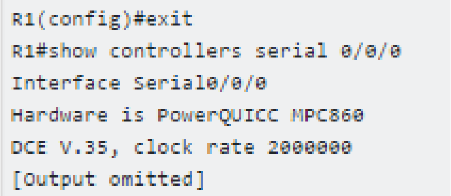
ip address 10.0.0.1 255.0.0.0 command assigns IP address to interface.

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.

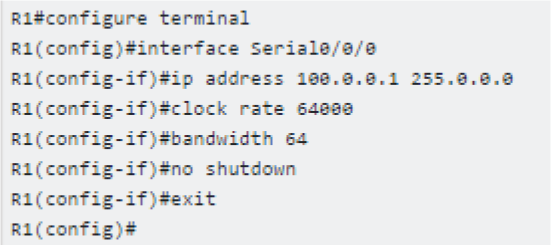
We can use show controllers interface command from privilege mode to check the cable's end.



```
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

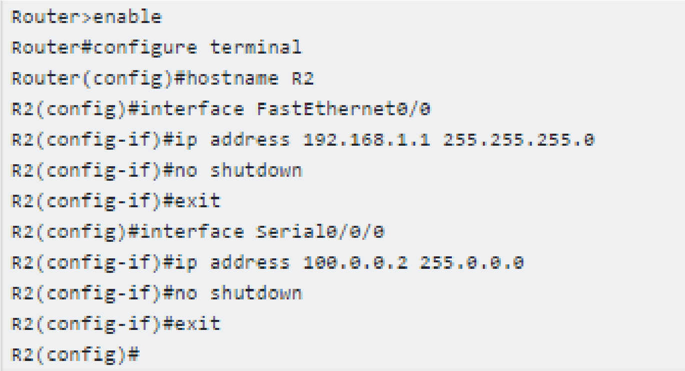
Bandwidth works as an influencer. It is used to influence the metric calculation of EIGRP or any other routing protocol which uses bandwidth parameter in route selection process.

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)#
```

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.

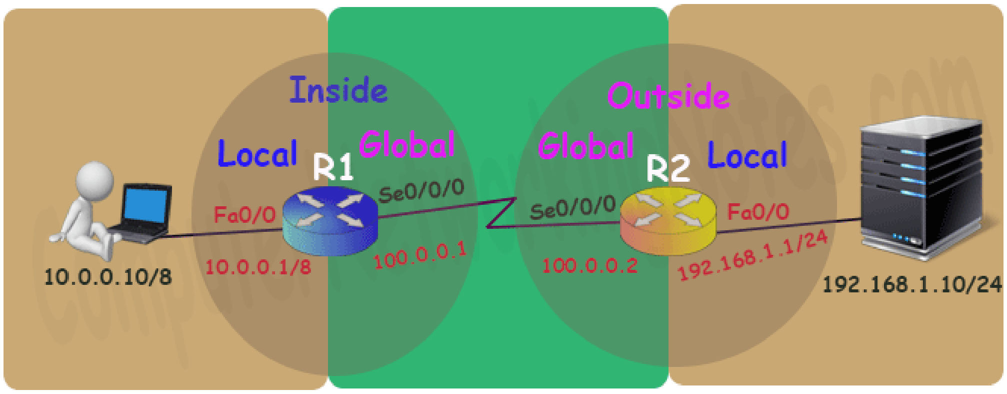
Following command will define interface Fa0/0 as inside local.

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.

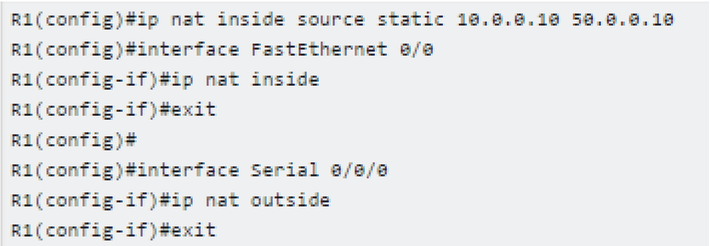
Router(config-if)#ip nat outside

Following figure illustrates these terms.



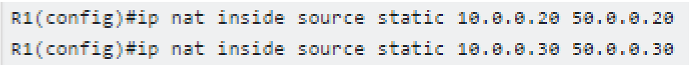
Let’s implement all these commands together and configure the static NAT.

R1 Static NAT Configuration



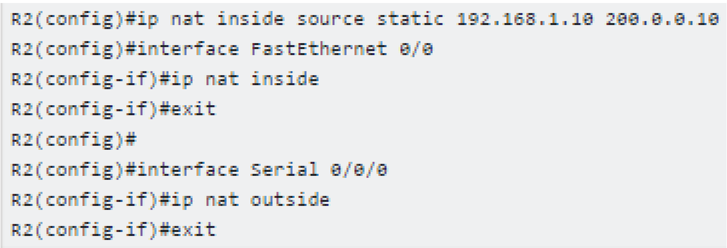
```
R1(config)#ip nat inside source static 10.0.0.10 50.0.0.10
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#
R1(config)#interface Serial 0/0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
```

In similar way,



```
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.

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

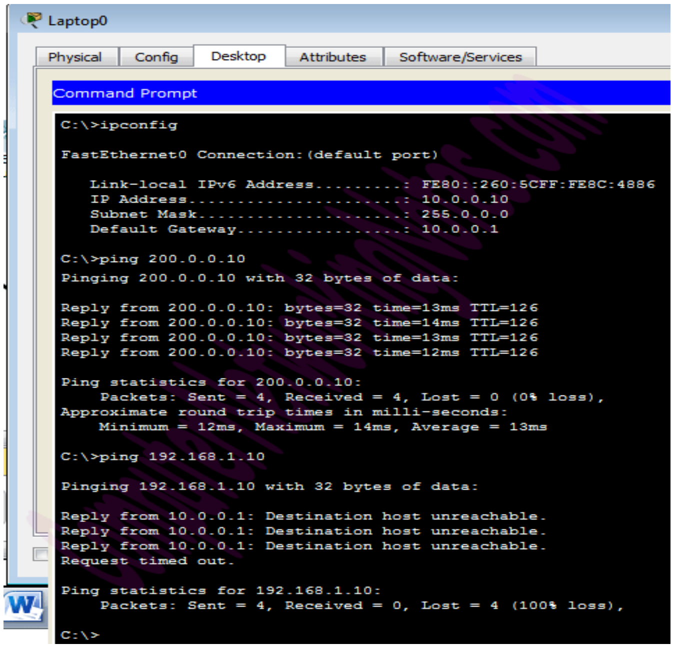
Testing Static NAT Configuration

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.



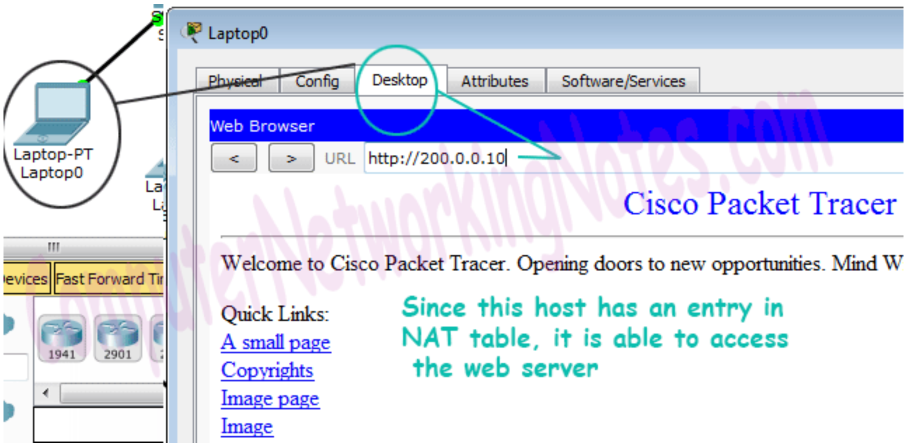
```
C:\>ipconfig

FastEthernet0 Connection:(default port)

   Link-local IPv6 Address.........: FE80::260:5CFF:FE8C:4886
   IP Address......................: 10.0.0.10
   Subnet Mask.....................: 255.0.0.0
   Default Gateway.................: 10.0.0.1

C:\>ping 200.0.0.10
Pinging 200.0.0.10 with 32 bytes of data:

Reply from 200.0.0.10: bytes=32 time=13ms TTL=126
Reply from 200.0.0.10: bytes=32 time=14ms TTL=126
Reply from 200.0.0.10: bytes=32 time=13ms TTL=126
Reply from 200.0.0.10: bytes=32 time=12ms TTL=126

Ping statistics for 200.0.0.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 14ms, Average = 13ms

C:\>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:

Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Request timed out.

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

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.

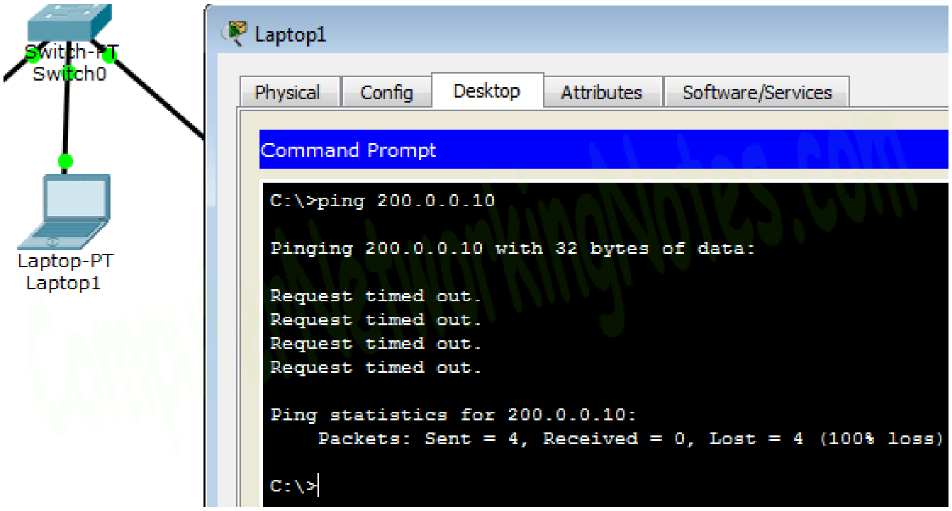
Third command checks whether we are able to access the remote device on its actual IP address or not. A ping error confirms that we are not able to connect with remote device on this IP address.

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.

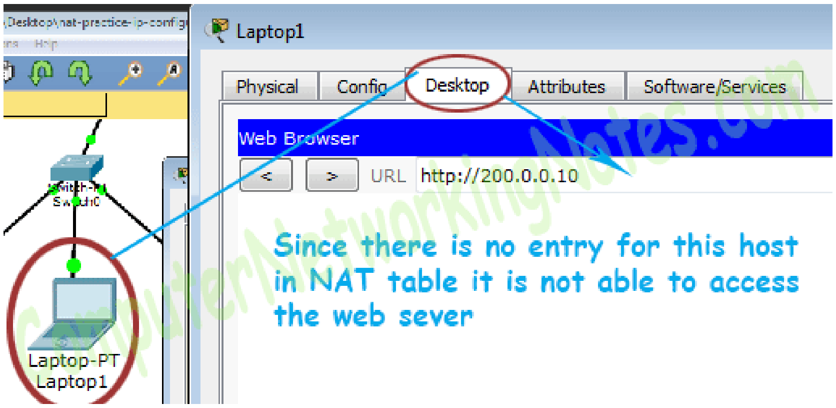


```
C:\>ping 200.0.0.10

Pinging 200.0.0.10 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 200.0.0.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss)

C:\>
```

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.

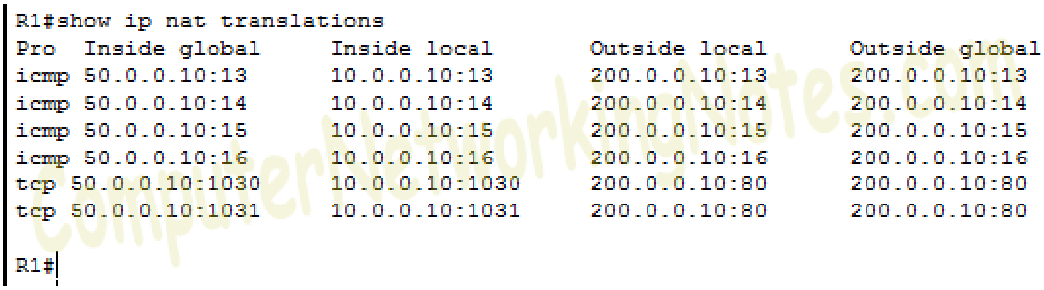
To confirm it again, let’s try to access web service from this host.



Since there is no entry for this host in NAT table it is not able to access the web sever

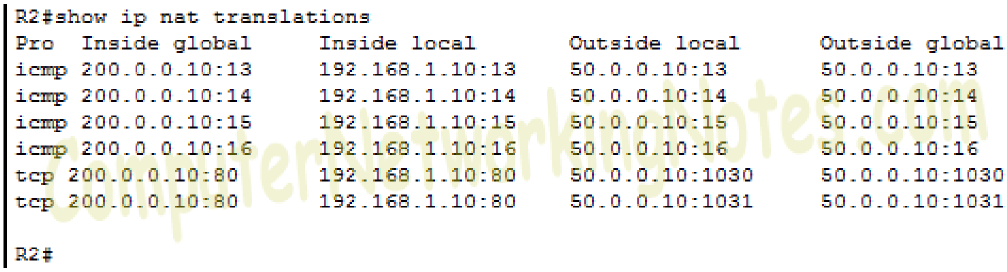
We can also verify this translation on router with ***show ip nat translation*** command.

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#
```

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.

Configure Dynamic NAT

Dynamic NAT configuration requires four steps: -

1. Create an access list of IP addresses which need translation
2. Create a pool of all IP address which are available for translation
3. Map access list with pool
4. 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 this lab. Let's create a standard access list which allows two hosts and denies one host.

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

R1(config)#access-list 1 permit 10.0.0.30 0.0.0.0

R1(config)#access-list 1 deny any

R1(config)#interface fastethernet 0/0

R1(config-if)#ip access-group 1 in

R1(config-if)#exit

R1(config)#
```

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

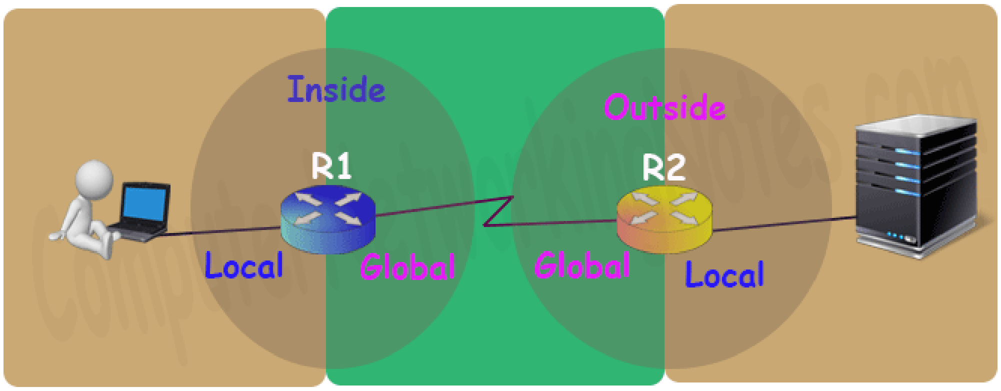
Finally we have to define which interface is connected with local network and which interface is connected with global network.

To define an inside local we use following command

Router(config-if)#ip nat inside

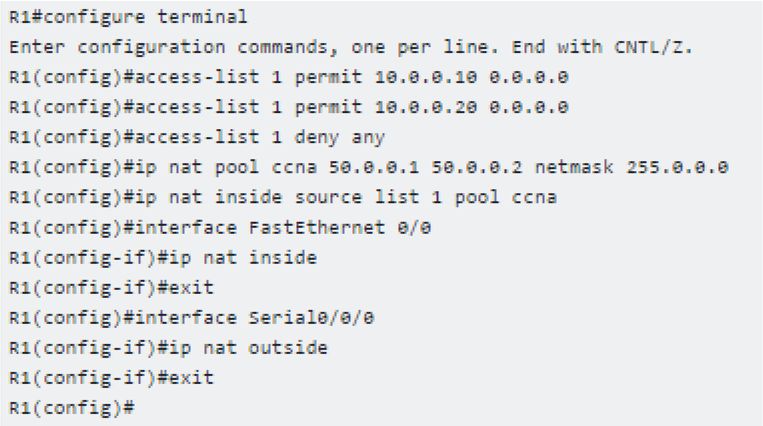
Following command defines inside global

Router(config-if)#ip nat outside



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

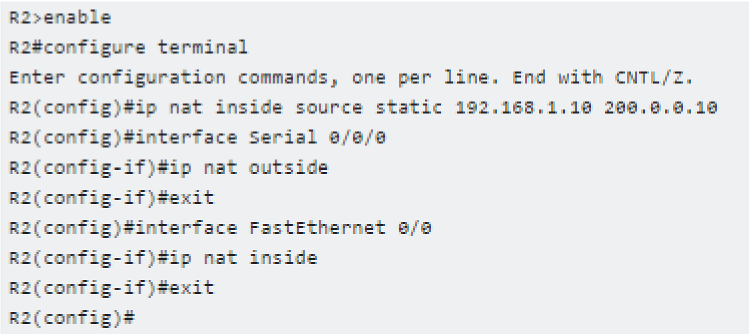
R1 Dynamic NAT Configuration



```
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 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)#
```

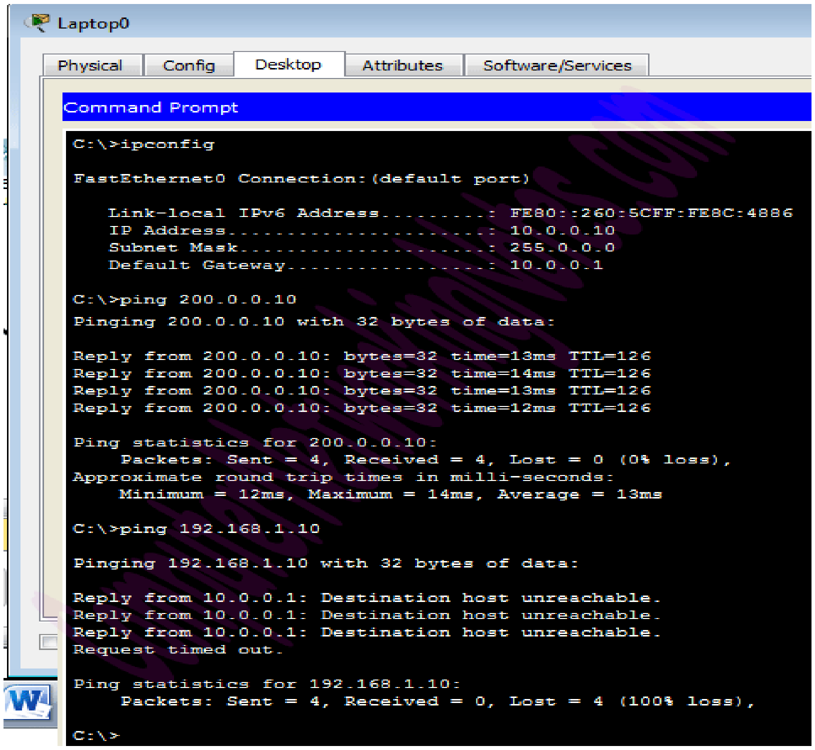
Testing Dynamic NAT Configuration

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 **Laptop1** and **Desktop** and click **Command Prompt**.

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



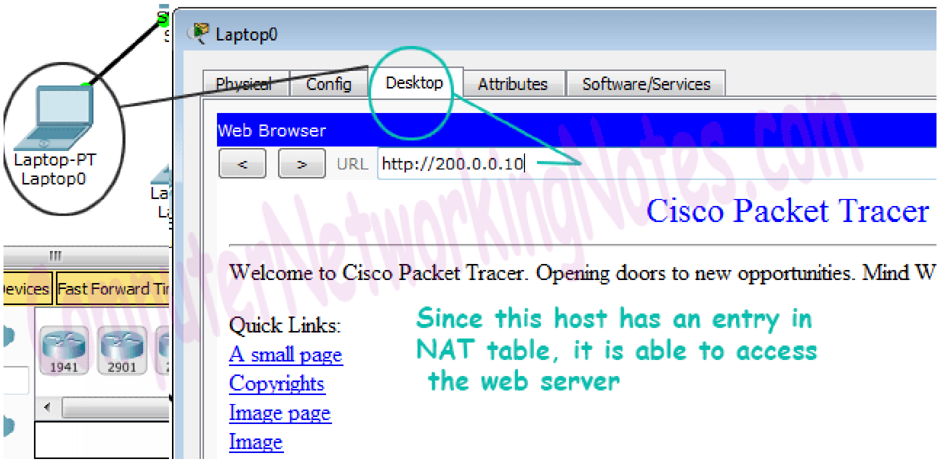
```
C:\>ipconfig

FastEthernet0 Connection:(default port)

   Link-local IPv6 Address.........: FE80::260:5CFF:FE8C:4886
   IP Address......................: 10.0.0.10
   Subnet Mask.....................: 255.0.0.0
   Default Gateway.................: 10.0.0.1

C:\>ping 200.0.0.10
Pinging 200.0.0.10 with 32 bytes of data:

Reply from 200.0.0.10: bytes=32 time=13ms TTL=126
Reply from 200.0.0.10: bytes=32 time=14ms TTL=126
Reply from 200.0.0.10: bytes=32 time=13ms TTL=126
Reply from 200.0.0.10: bytes=32 time=12ms TTL=126

Ping statistics for 200.0.0.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 14ms, Average = 13ms

C:\>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:

Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Request timed out.

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

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.



Cisco Packet Tracer

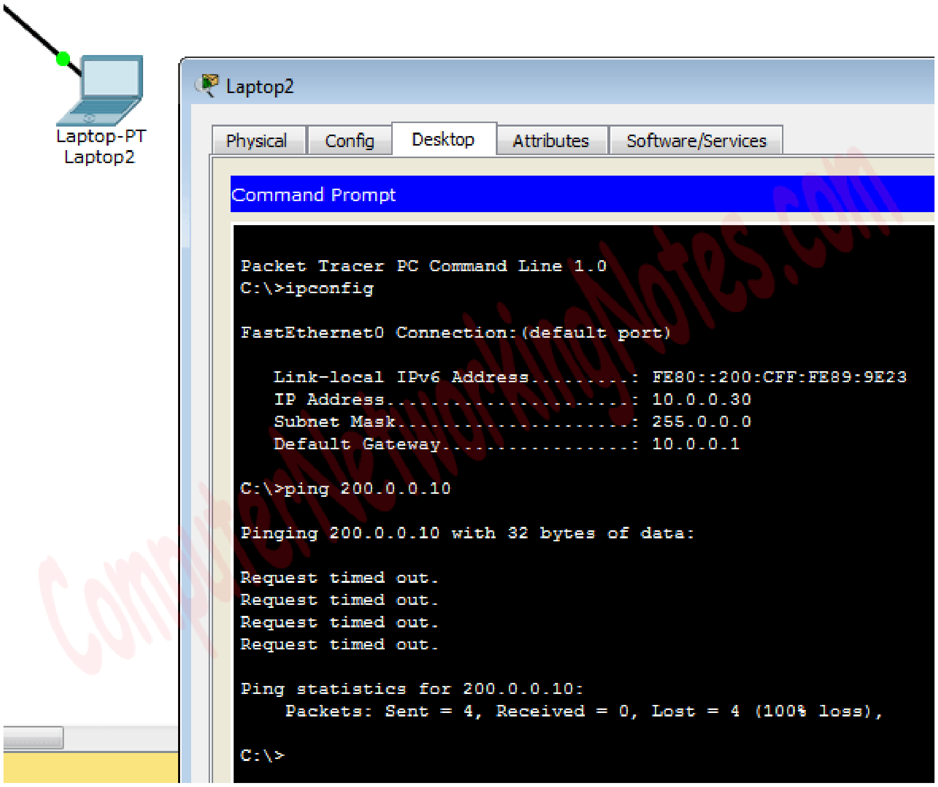
Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind W

Quick Links:
A small page
Copyrights
Image page
Image

Since this host has an entry in NAT table, it is able to access the web server

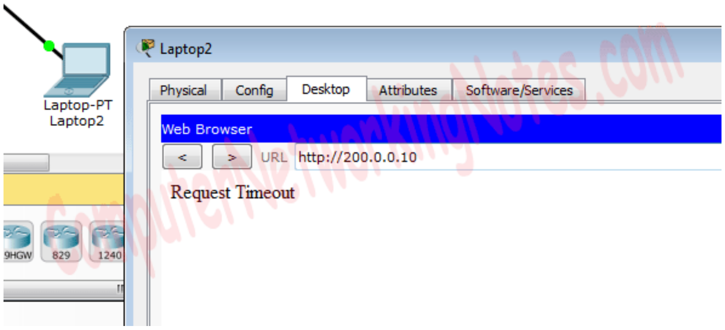
Above figure confirms that host 10.0.0.20 is able to access the 200.0.0.10. You can also do the same testing from Laptop2, result will be same.

Now run ping 200.0.0.10 command from Laptop0.



```
Packet Tracer PC Command Line 1.0
C:\>ipconfig

FastEthernet0 Connection:(default port)

   Link-local IPv6 Address.........: FE80::200:CFF:FE89:9E23
   IP Address......................: 10.0.0.30
   Subnet Mask.....................: 255.0.0.0
   Default Gateway.................: 10.0.0.1

C:\>ping 200.0.0.10

Pinging 200.0.0.10 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 200.0.0.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

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



http://200.0.0.10

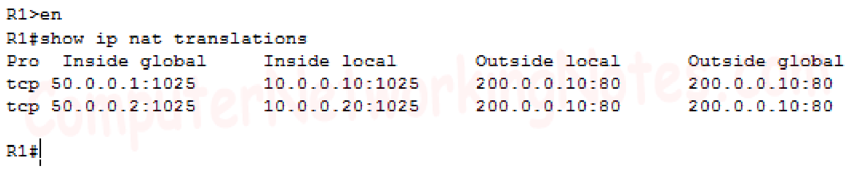
Request Timeout

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

Because we configured NAT only for two hosts (Laptop1 and Laptop2) which IP addresses are 10.0.0.20 and 10.0.0.30. So only the host 10.0.0.20 and 10.0.0.30 will be able to access the remote device.

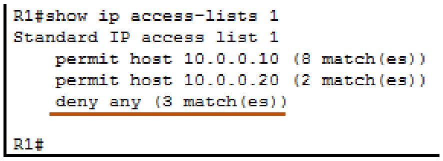
We can also verify this translation on router with ***show ip nat translation*** command.

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#
```

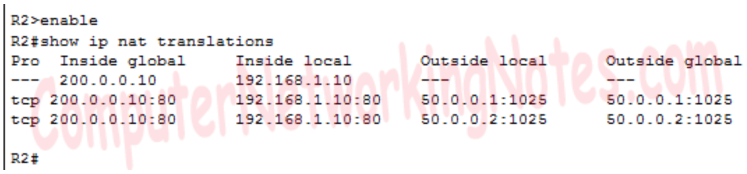
We did three tests one from each host, but why only two tests are listed here? Remember in first step we created an access list. Access list filters the unwanted traffic before it reaches to the NAT. We can see how many packets are blocked by ACL with following command



```
R1#show ip access-lists 1
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.

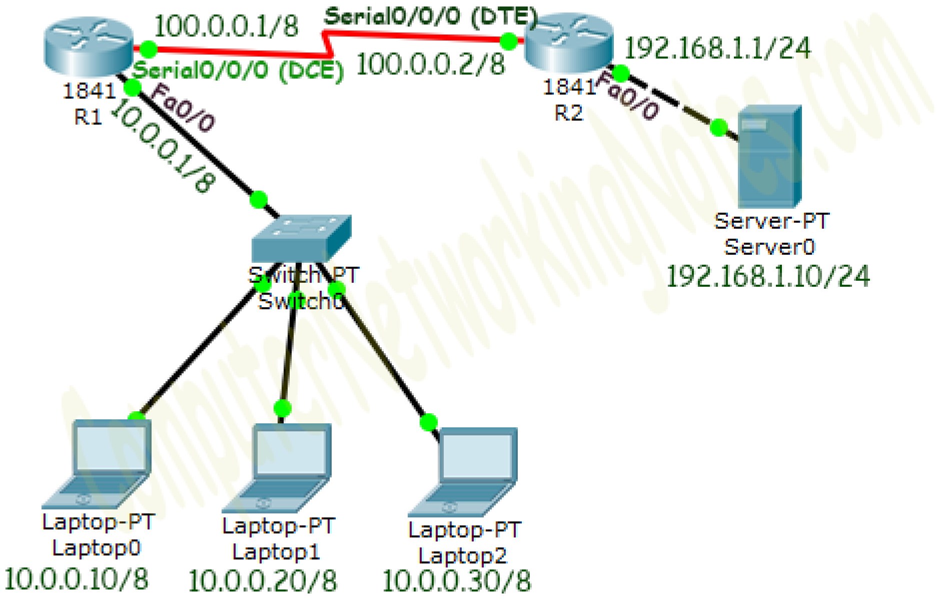
Following figure illustrate NAT translation on router R2



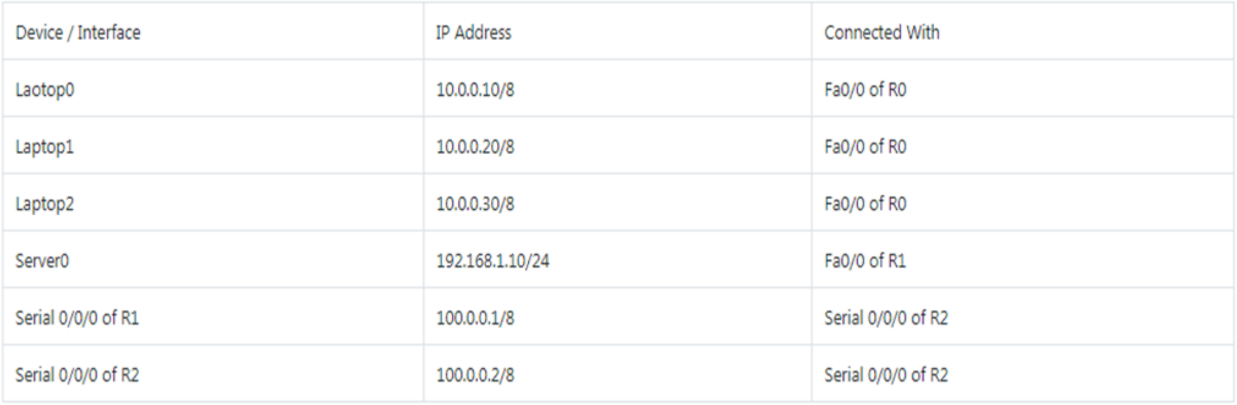
```
R2>enable
R2#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
---  200.0.0.10        192.168.1.10       ---                ---
tcp 200.0.0.10:80     192.168.1.10:80    50.0.0.1:1025      50.0.0.1:1025
tcp 200.0.0.10:80     192.168.1.10:80    50.0.0.2:1025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

R2#
```

LAB 7B- Configuring PAT (NAT – Overload) LAB

Create this network topology lab as shown in following figure

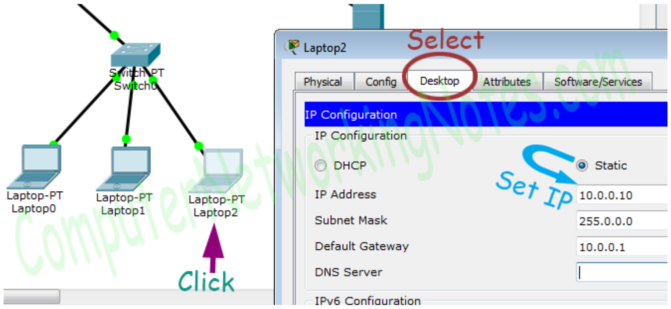


Initial IP Configuration



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.



IP Configuration

IP Configuration

○ DHCP ● Static

IP Address 10.0.0.10

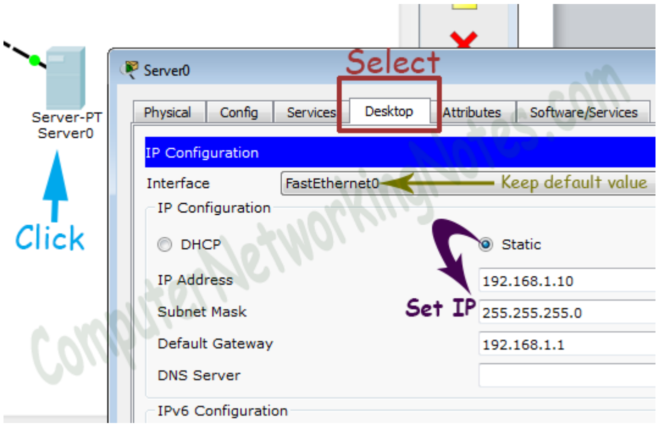
Subnet Mask 255.0.0.0

Default Gateway 10.0.0.1

DNS Server

IPv6 Configuration

Following same way configure IP address in Server.



IP Configuration

Interface FastEthernet0

IP Configuration

○ DHCP ● Static

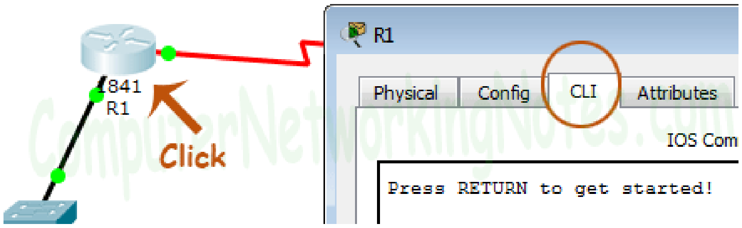
IP Address 192.168.1.10

Subnet Mask 255.255.255.0

Default Gateway 192.168.1.1

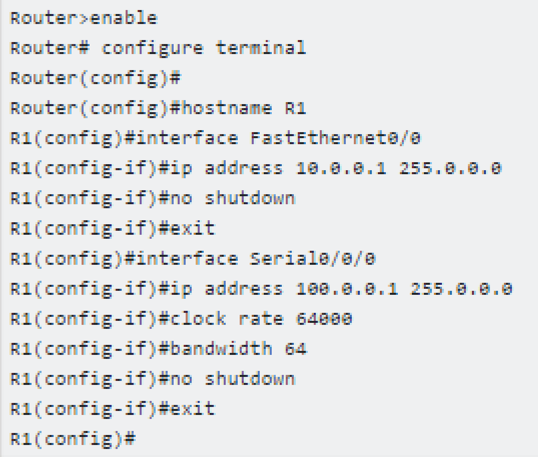
DNS Server

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



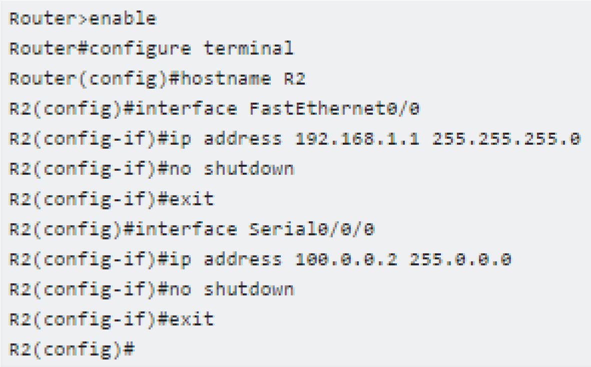
```
Press RETURN to get started!
```

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)#
```

Configure PAT (NAT Overload)

PAT configuration requires four steps: -

1. Create an access list of IP addresses which need translation
2. Create a pool of all IP address which are available for translation
3. Map access list with pool
4. 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.

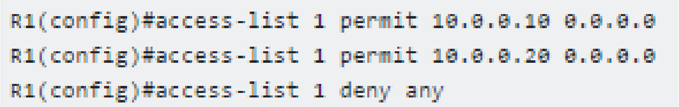
A.B.C.D

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 this 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 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
```

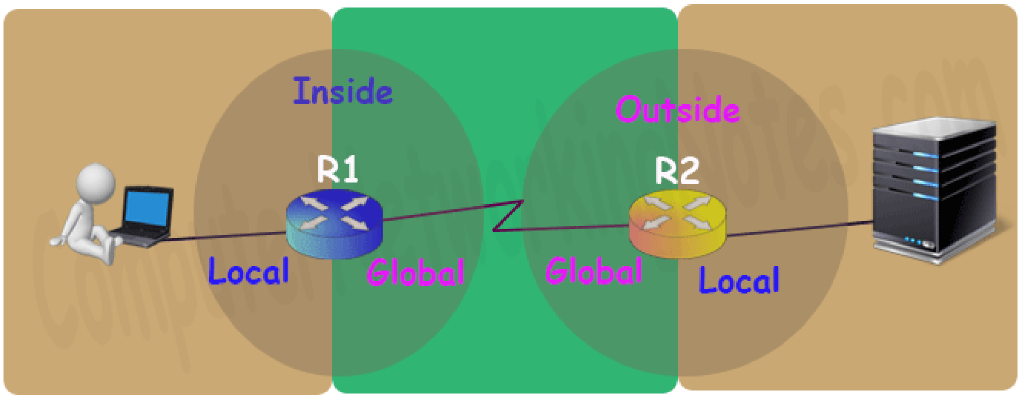
Finally we have to define which interface is connected with local network and which interface is connected with global network.

To define an inside local we use following command

Router(config-if)#ip nat inside

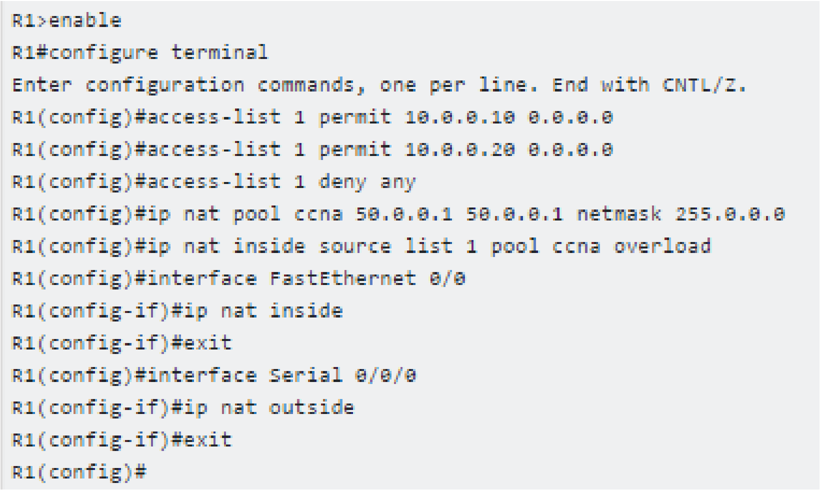
Following command defines inside global

Router(config-if)#ip nat outside



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

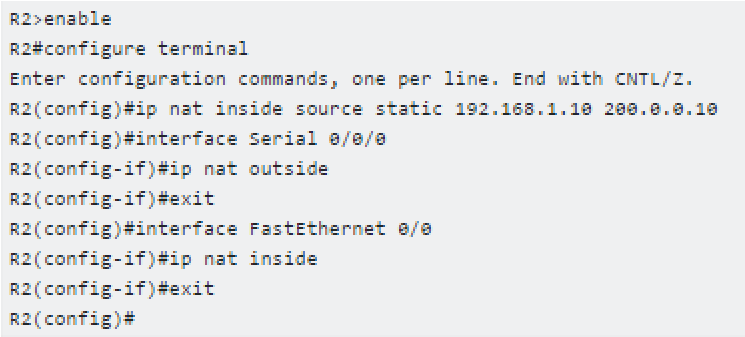
R1 PAT (NAT Overload) Configuration



```
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 we configured pat translations for two addresses only.

On R2 we can keep standard configuration or can 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)#
```

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.

Configure static routing in R1

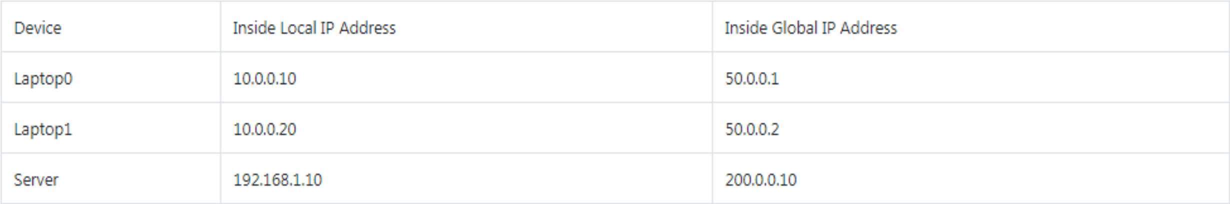
R1(config)#ip route 200.0.0.0 255.255.255.0 100.0.0.2

Configure static routing in R2

R2(config)#ip route 50.0.0.0 255.0.0.0 100.0.0.1

Testing PAT Configuration

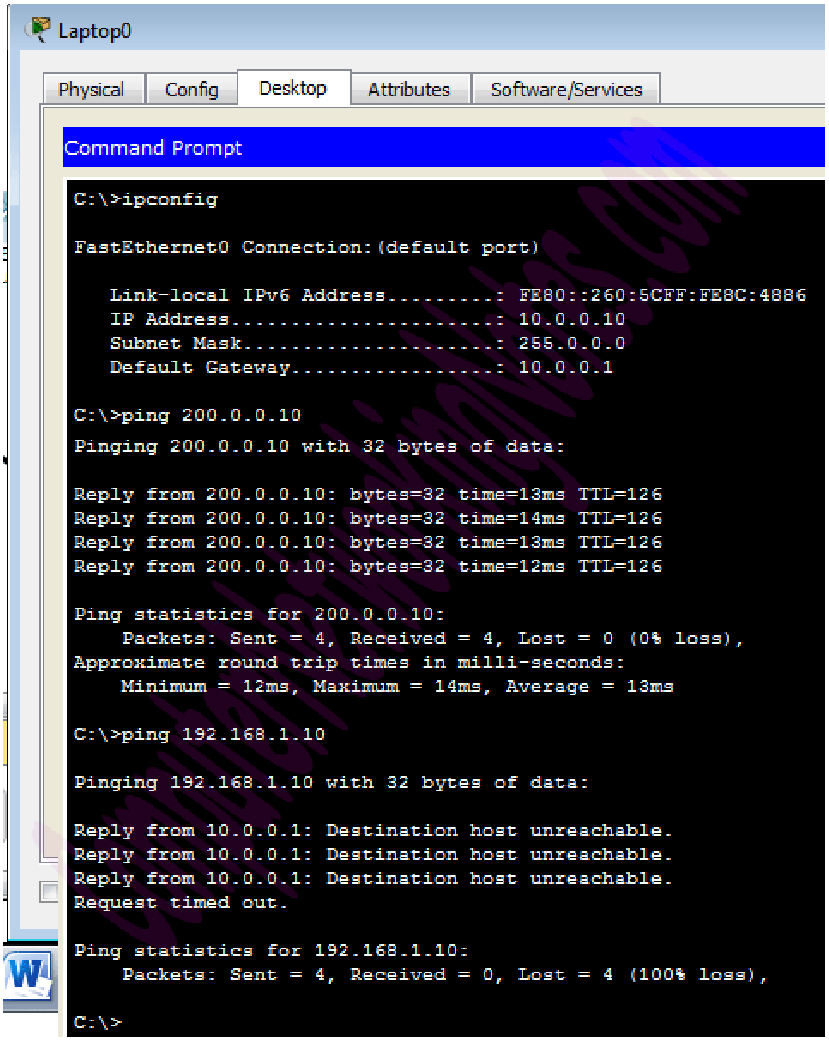
In this lab we configured PAT on R1for 10.0.0.10 and 10.0.0.20 and static NAT on R2 for 192.168.1.10.



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.



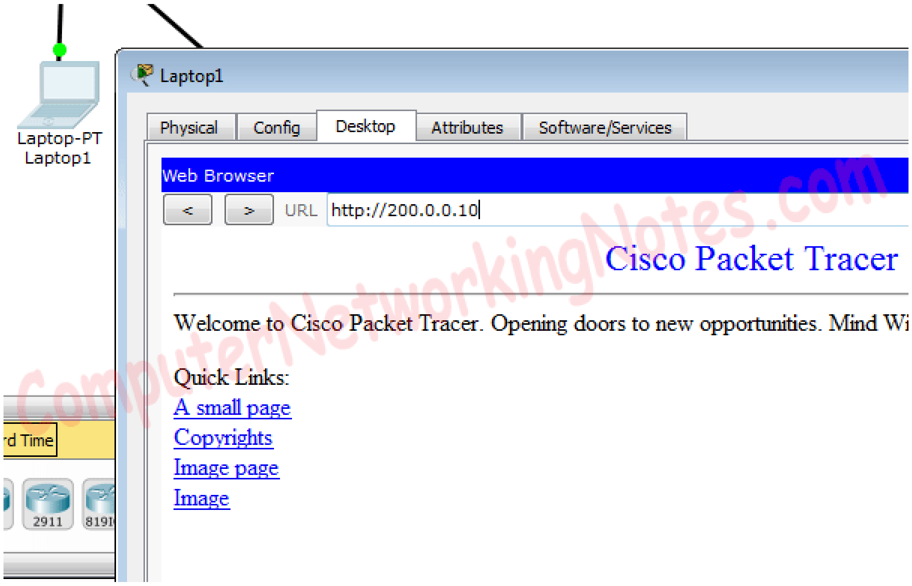
```
C:\>ipconfig

FastEthernet0 Connection:(default port)

   Link-local IPv6 Address.........: FE80::260:5CFF:FE8C:4886
   IP Address......................: 10.0.0.10
   Subnet Mask.....................: 255.0.0.0
   Default Gateway.................: 10.0.0.1

C:\>ping 200.0.0.10
Pinging 200.0.0.10 with 32 bytes of data:

Reply from 200.0.0.10: bytes=32 time=13ms TTL=126
Reply from 200.0.0.10: bytes=32 time=14ms TTL=126
Reply from 200.0.0.10: bytes=32 time=13ms TTL=126
Reply from 200.0.0.10: bytes=32 time=12ms TTL=126

Ping statistics for 200.0.0.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 14ms, Average = 13ms

C:\>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:

Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Request timed out.

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

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.



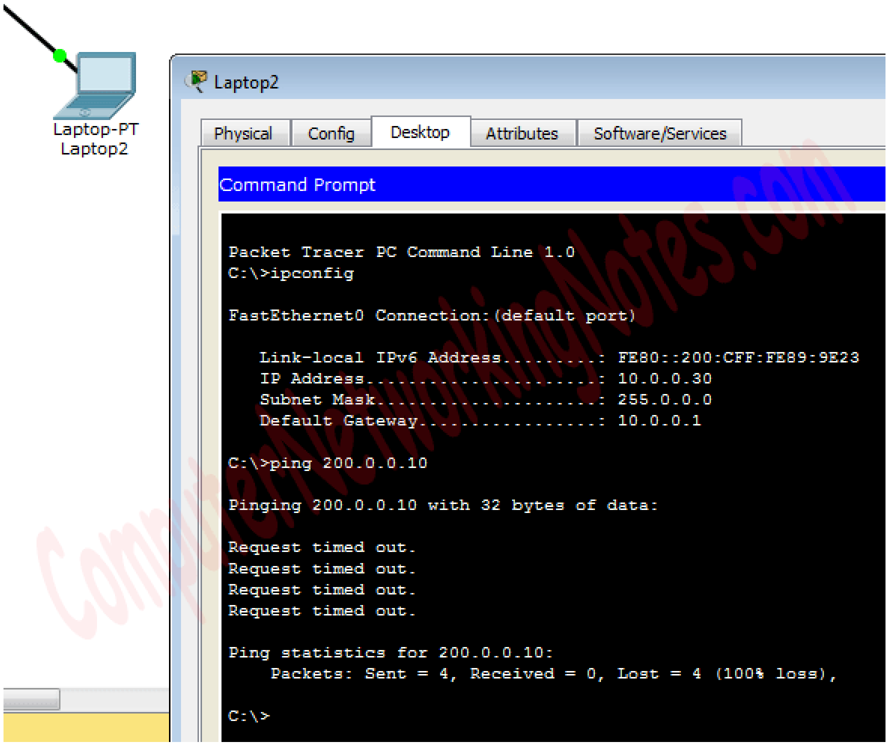
Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wi

Quick Links:
A small page
Copyrights
Image page
Image

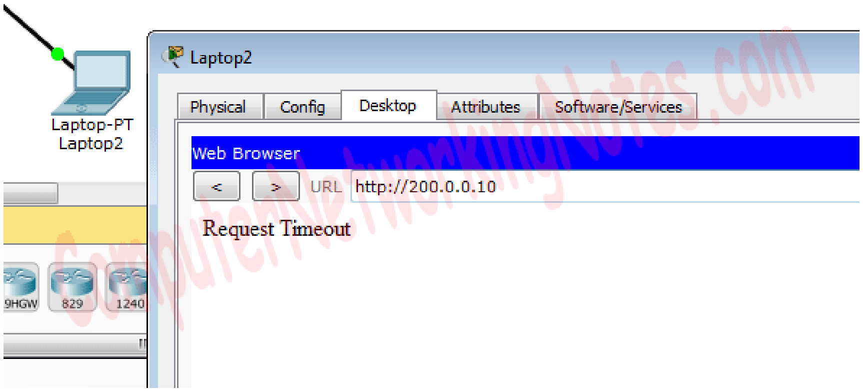
Above figure confirms that host 10.0.0.10 is able to access the 200.0.0.10. You can also do the same testing from Laptop1, result will be same.

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



```
Packet Tracer PC Command Line 1.0
C:\>ipconfig

FastEthernet0 Connection:(default port)

   Link-local IPv6 Address.........: FE80::200:CFF:FE89:9E23
   IP Address......................: 10.0.0.30
   Subnet Mask.....................: 255.0.0.0
   Default Gateway.................: 10.0.0.1

C:\>ping 200.0.0.10

Pinging 200.0.0.10 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 200.0.0.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

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



http://200.0.0.10

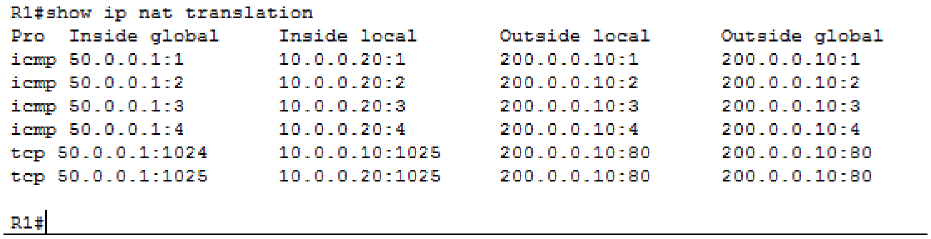
Request Timeout

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.

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

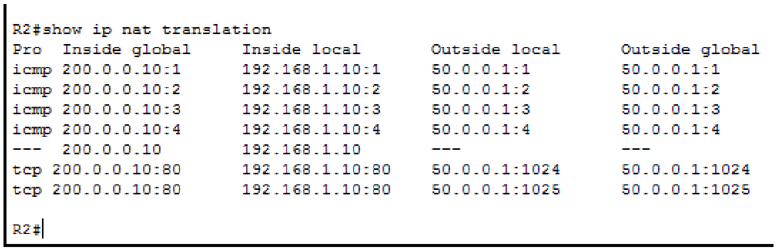
Following figure illustrate this translation on router R1.



```
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.