Computer Networks - Lab 11

OBJECTIVES

After these Lab students shall be able to perform

- **Understanding with some networking terms**
    - **Network Address**
    - **Broadcast Address**
    - **CIDR (Classless Inter-Domain Routing)**
    - **Prefix and Suffix in IPv4**
    - **Subnet Mask (Binary, decimal)**
    - **Total Host**
    - **Total Network**
- **Understanding of NAT(D-NAT, P-NAT).**
- **Implementation of D-NAT and P-NAT in cisco packet Tracer**
    - **Repeat Static NAT**
    - **Dynamics NAT**
    - **Port NAT**

PRE-LAB READING ASSIGNMENT

Remember the delivered lecture carefully.

Table of Contents

Network Address (Network bit are represented by 1)

A network address is also known as the numerical **network part of an IP address**. ... For example, in the IP address 192.168. 1.0, the network address is 192.168.

Class	IP address range (1ˢᵗ Octet)	Network Mask	Prefix	Number of Networks	Number of Hosts
A	1. - 127.	255.0.0.0	/8	125	16,777,214
B	128. - 191.	255.255.0.0	/16	16,382	65,534
C	192. - 223.	255.255.255.0	/24	2,097,150	254
D	224. - 239.	Multicast addresses			
E	240. - 254.	Restricted/Experimental			

What is a CIDR example?
It is simply a count of the number of network bits (bits that are set to 1) in the subnet mask. ... The CIDR number is typically preceded by a slash "/" and follows the IP address.

- **The IP Address Hierarchy**

    o IP addresses are hierarchical. They have two parts: a *prefix* and a *suffix*.
    o A prefix is assigned by an authority to one, and only, one network.
    o All hosts on a given network have the same prefix.
    o Example: the prefix for the csustan.edu network is 130.17. All computers on our campus network have IP numbers that start with 130.17.
    o It is very important to NOT assign the same IP address to two different NIC's. Since the prefixes of two computers on the same network have to be the same, that means the suffixes have to be different.
    o Example: cs.csustan.edu: 130.17.70.80
    o Example: www.cs.csustan.edu: 130.17.70.35
    o

- 21.7 **Original Classes of IP Addresses**

    o In the original IPv4 addressing scheme, there were five different classes of address:
        ▪ Class A: first bit is 0, prefix is 8 bits, suffix is 24 bits - provides addresses for about 128 big networks with many hosts.
        ▪ Class B: first two bits are 10, prefix is 16 bits, suffix is 16 bits - provides addresses for about 16,000 medium-sized networks with up to about 64,000 hosts

- Class C: first three bits are 110, prefix is 24 bits, suffix is 8 bits - provides addresses for about two million small networks with up to about 256 hosts.
- Class D: first four bits are 1110 - used for multicast addresses
- Class E: first four bits are 1111 - reserved (not assigned)
  - Classes A through D are in use, although Class D addresses (for multicast) are used only locally, within individual networks.
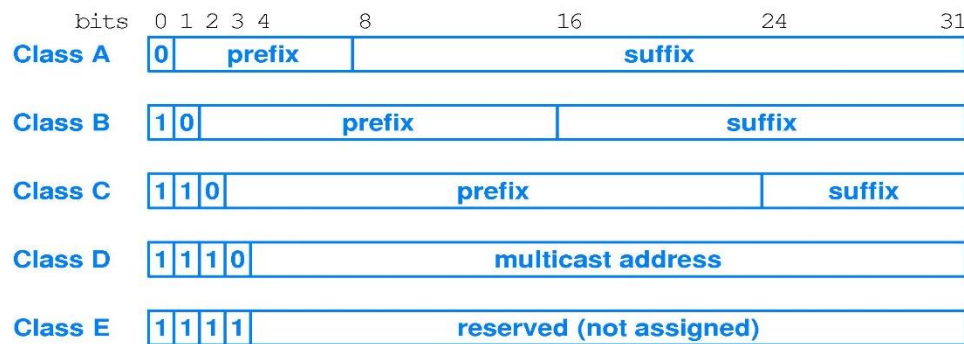

Figure 21.2 The five classes of IPv4 addresses in the original classful scheme.

In Simple: Prefix represent Network bit and Suffix represent Host bit

Total Host (Host bit are represented by 0)

The host's formula will tell you how many hosts will be allowed on a network that has a certain subnet mask. The host's formula is $2h - 2$. The h represents the number of 0s in the subnet mask, if the subnet mask were converted to binary. The first and last addresses are reserved: the first to identify the network and the last to be used as the broadcast address.

Subnet Mask

A subnet mask is **a 32-bit number created by setting host bits to all 0s and setting network bits to all 1s**. In this way, the subnet mask separates the IP address into the network and host addresses. The "255" address is always assigned to a broadcast address, and the "0" address is always assigned to a network address.
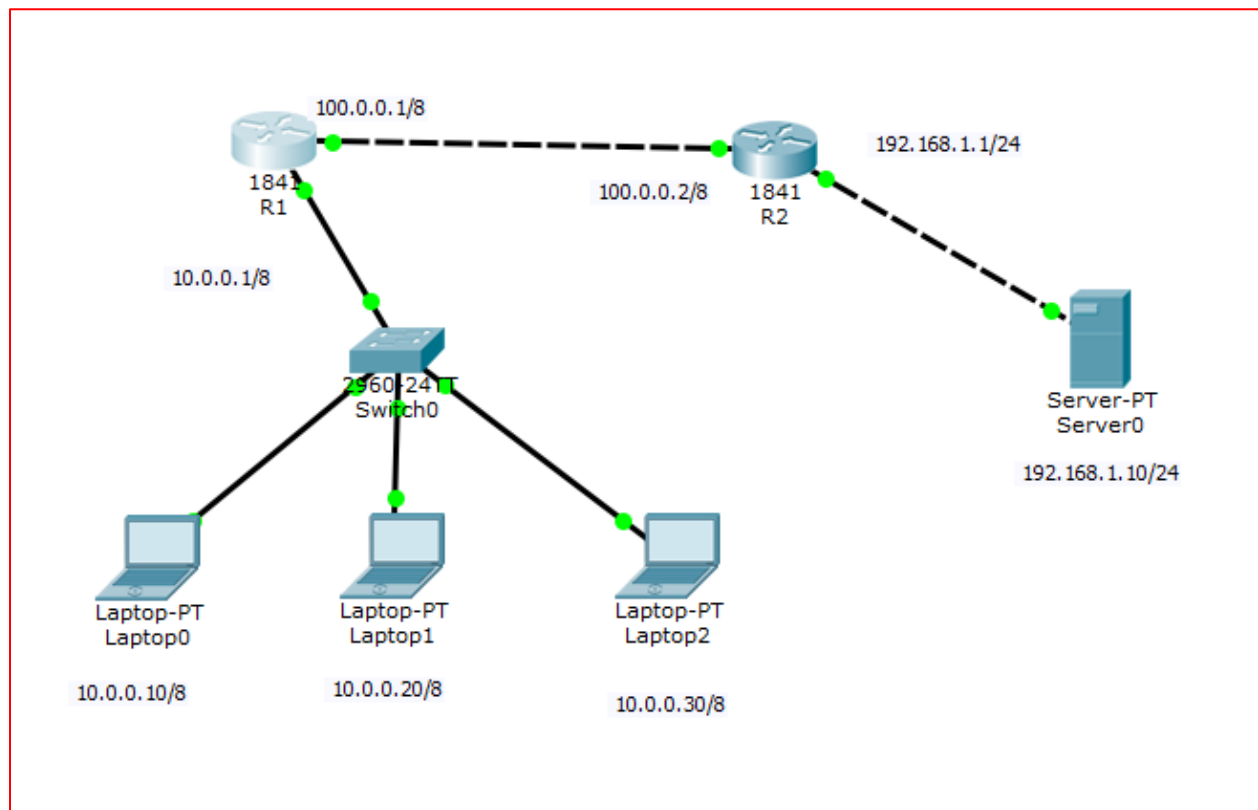
Broadcast Address

A broadcast address is **an IP address that is used to target all systems on a specific subnet network instead of** single hosts. In other words broadcast address allows information to be sent to all machines on a given subnet rather than to a specific machine.

NAT configuration

How to Configure Dynamic NAT in Cisco Router

Dynamic NAT configuration (creating an access list of IP addresses which need translation, creating a pool of available IP address, mapping access list with pool and defining inside and outside interfaces) in detail. Learn how to configure, manage, verify and debug dynamic NAT step by step with packet tracer examples.
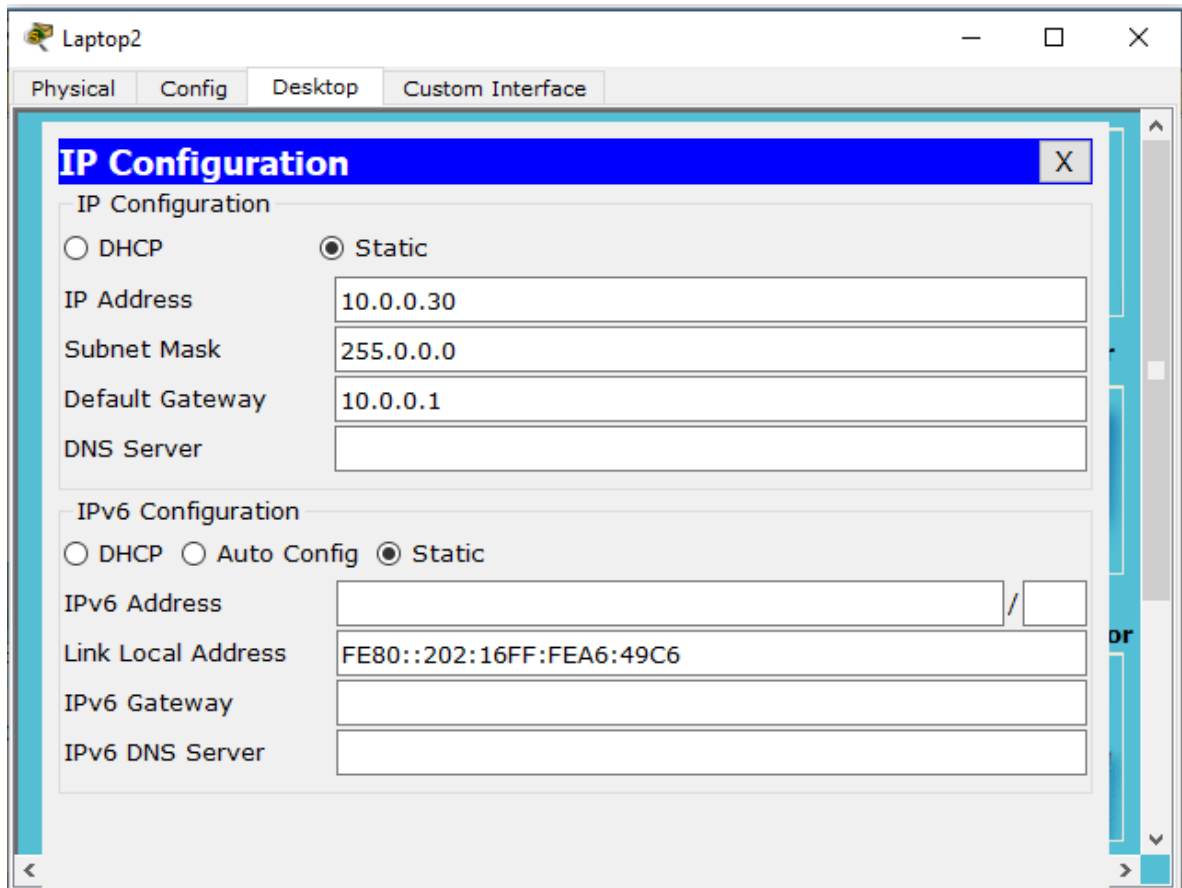
To explain Dynamic NAT configuration, I will use packet tracer network simulator software.

Initial IP Configuration

Device / Interface	IP Address	Connected With
Laotop0	10.0.0.10/8	Fa0/1 of R0
Laptop1	**10.0.0.20/8**	**Fa0/1 of R0**
Laptop2	10.0.0.30/8	Fa0/1 of R0
Server0	**192.168.1.10/24**	**Fa0/1 of R2**
F0/0 of R1	100.0.0.1/8	F0/0 of R2
F0/0 of R2	100.0.0.2/8	F0/0 of R1

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.

Laptop1 — □ ✕

Physical　Config　Desktop　Custom Interface

IP Configuration [X]

IP Configuration

○ DHCP　　　◉ Static

IP Address　　　10.0.0.20

Subnet Mask　　255.0.0.0

Default Gateway　10.0.0.1

DNS Server

IPv6 Configuration

○ DHCP　○ Auto Config　◉ Static

IPv6 Address　　　　　　　　　　／

Link Local Address　FE80::20A:F3FF:FE00:2BC5

IPv6 Gateway

IPv6 DNS Server

Laptop0 — □ ✕

Physical　Config　Desktop　Custom Interface

IP Configuration [X]

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
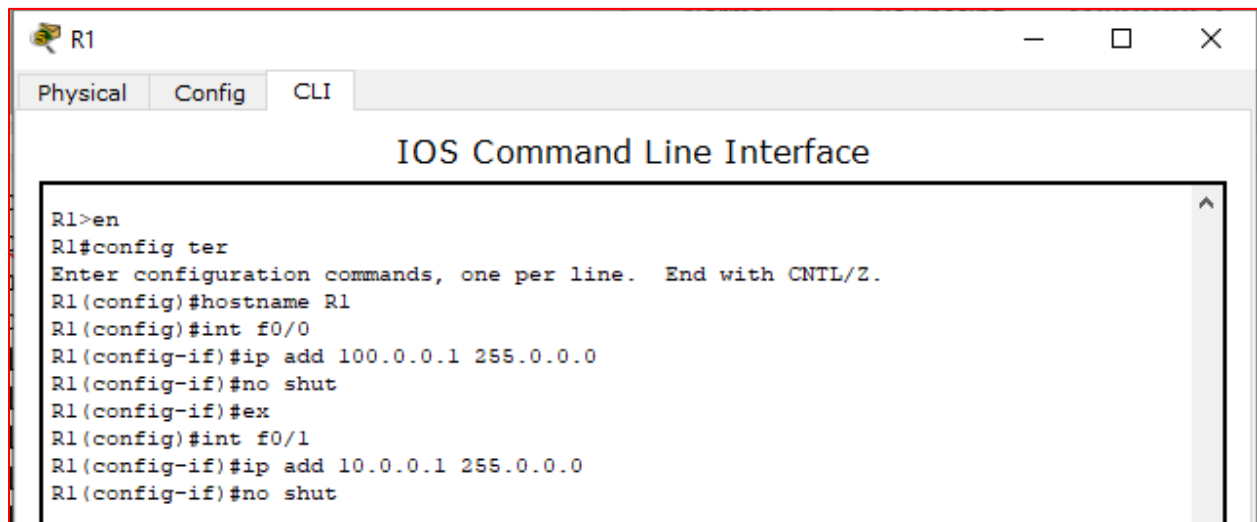
○ DHCP　○ Auto Config　◉ Static

IPv6 Address　　　　　　　　　　／

Link Local Address　FE80::204:9AFF:FE75:CB65
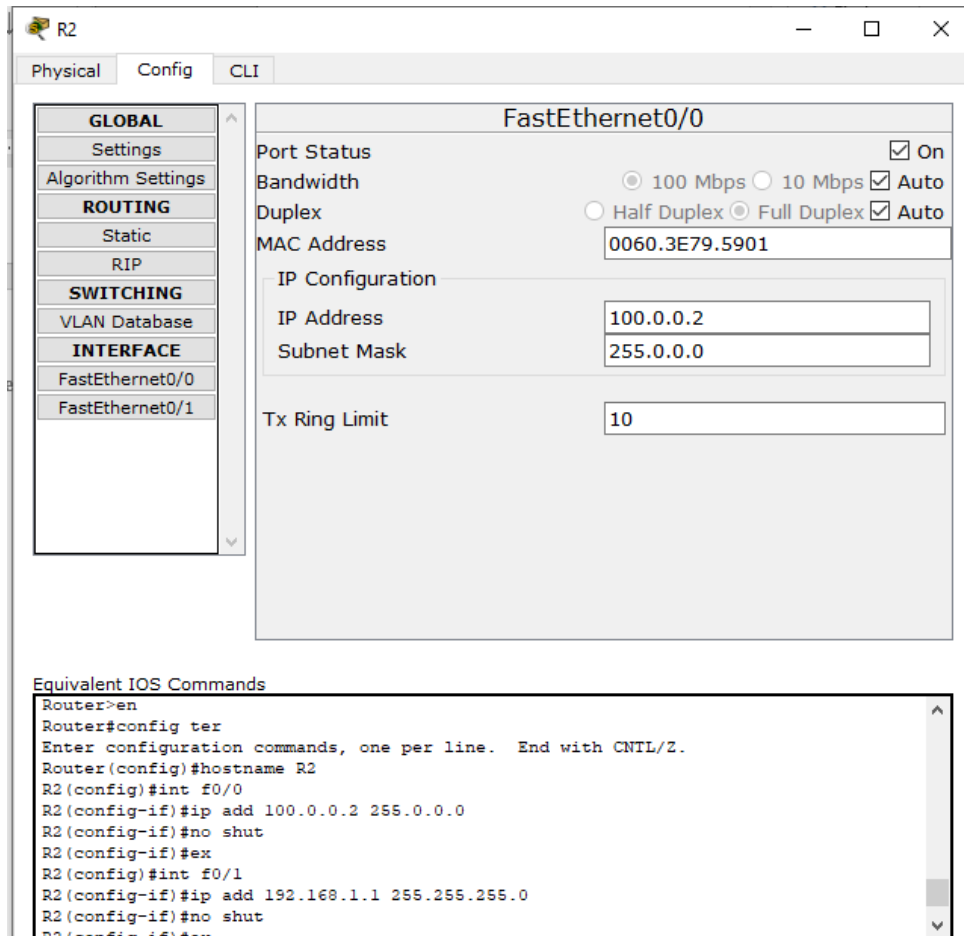
IPv6 Gateway

IPv6 DNS Server

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

> Router>en
> Router#config ter
> Enter configuration commands, one per line. End with CNTL/Z.
> Router(config)#hostname R1
> R1(config)#int f0/0
> R1(config-if)#ip add 100.0.0.1 255.0.0.0
> R1(config-if)#no shut
> R1(config-if)#ex
> R1(config)#int f0/1
> R1(config-if)#ip add 10.0.0.1 255.0.0.0
> R1(config-if)#no shut

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

> Router>en
> Router#config ter
> Enter configuration commands, one per line. End with CNTL/Z.
> Router(config)#hostname R2
> R2(config)#int f0/0
> R2(config-if)#ip add 100.0.0.2 255.0.0.0
> R2(config-if)#no shut
> R2(config-if)#ex
> R2(config)#int f0/1
> R2(config-if)#ip add 192.168.1.1 255.255.255.0
> R2(config-if)#no shut
> R2(config-if)#ex
> 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: -

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.

*An **Access Control List** (ACL) is a set of rules that is usually used to filter network traffic. ACLs can be configured on network devices with packet filtering compatibilities, such as routers and firewalls.*

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

1   Any

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

2   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.

3   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.

4   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 any hosts.

```
R1(config)#access-list 1 permit 10.0.0.0 0.255.255.255
```

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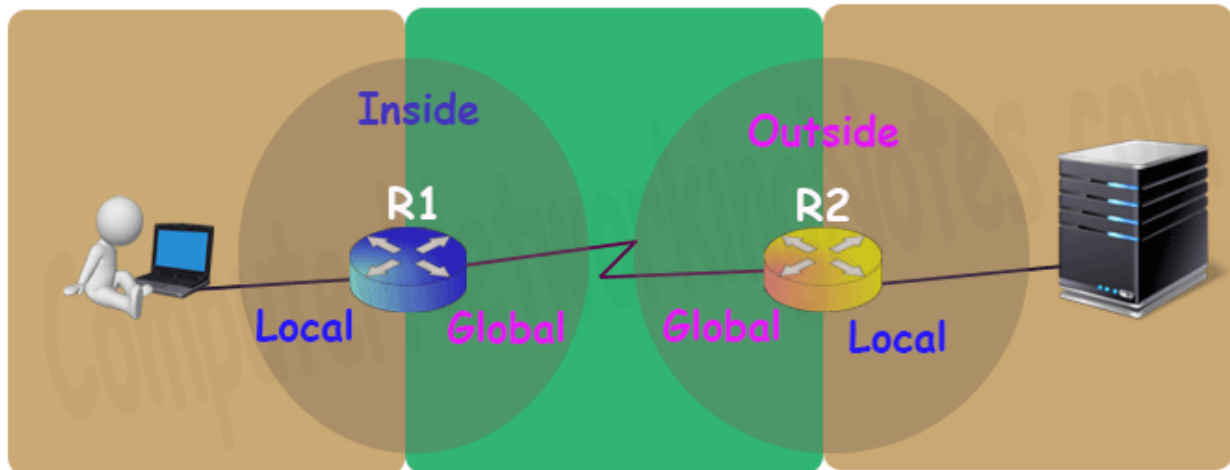
Finally we have to define which interface is connected with local network and which interface is connected with global network.

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 10.0.0.0 0.255.255.255
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 0/1
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
R1(config)#
```
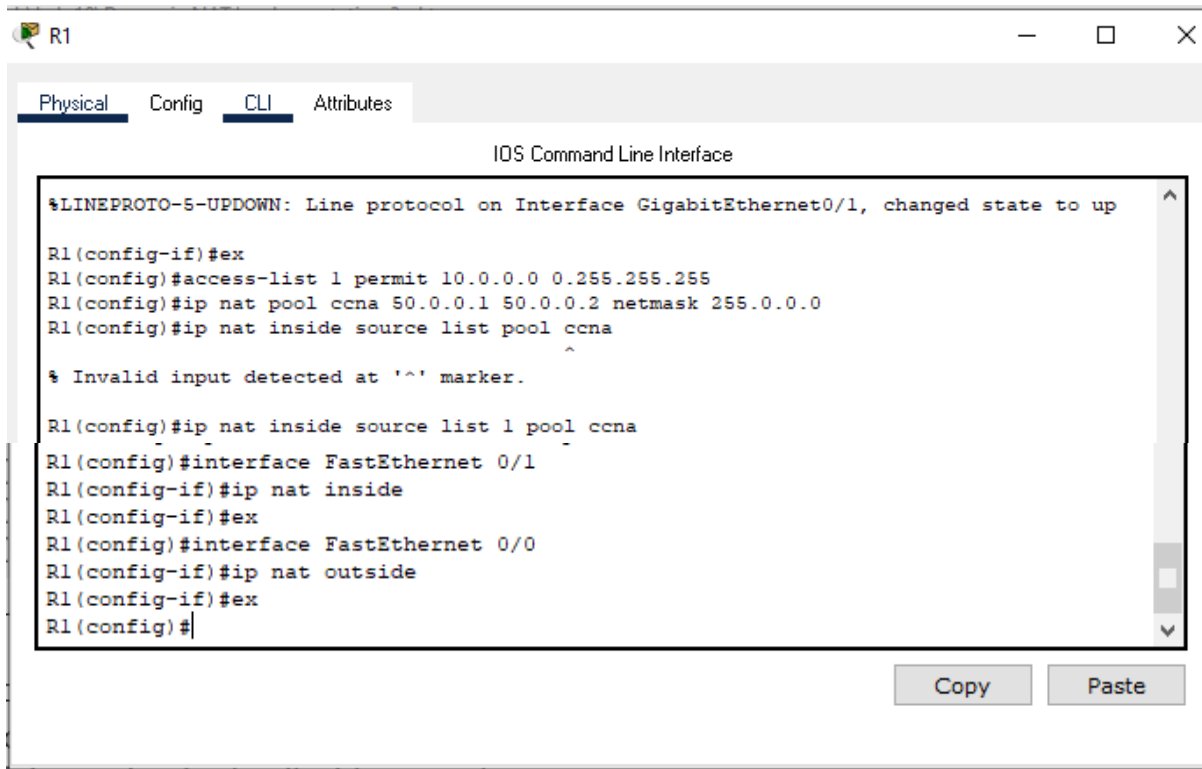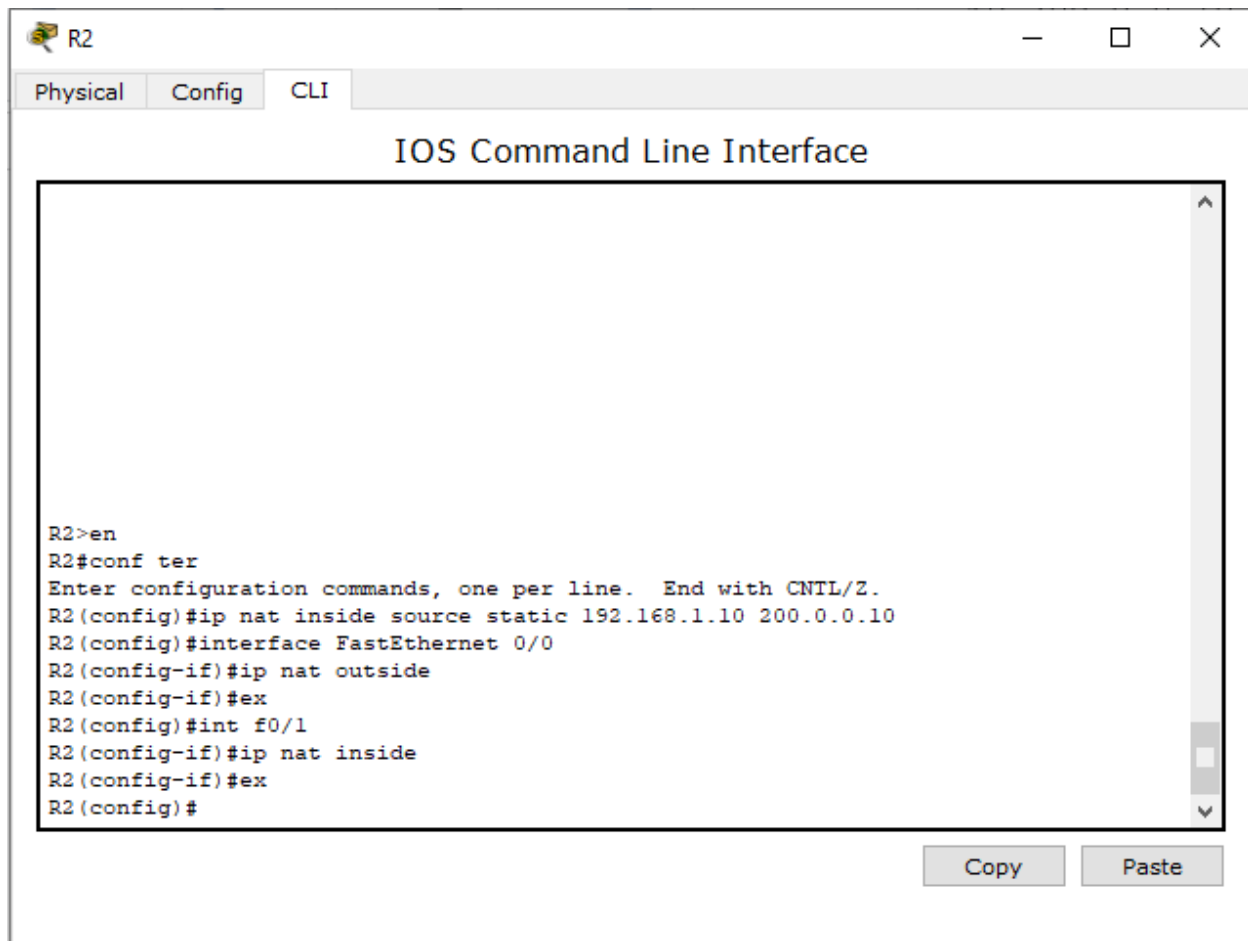
For testing purpose I configured dynamic translations for two addresses only.

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 Lab.

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

```
R2>en
R2#conf ter
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 FastEthernet 0/0
R2(config-if)#ip nat outside
R2(config-if)#ex
R2(config)#int f0/1
R2(config-if)#ip nat inside
R2(config-if)#ex
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
R1(config)# ip route 200.0.0.0 255.255.255.252 100.0.0.2
```

Configure static routing in R2

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

R1#show ip nat statistics

show ip nat translations

clear ip nat translation *
config terminal
no ip nat pool *old pool name*

ip nat pool *new pool*

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



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

Configure PAT in Cisco Router with Examples

How to configure port address translation (PAT) in router step by step with examples. Learn how to connect multiple devices with remote network from single IP address through PAT or NAT Overload, verify and troubleshoot PAT configuration view PAT address translation from show commands.

Create a practice lab as shown in following figure
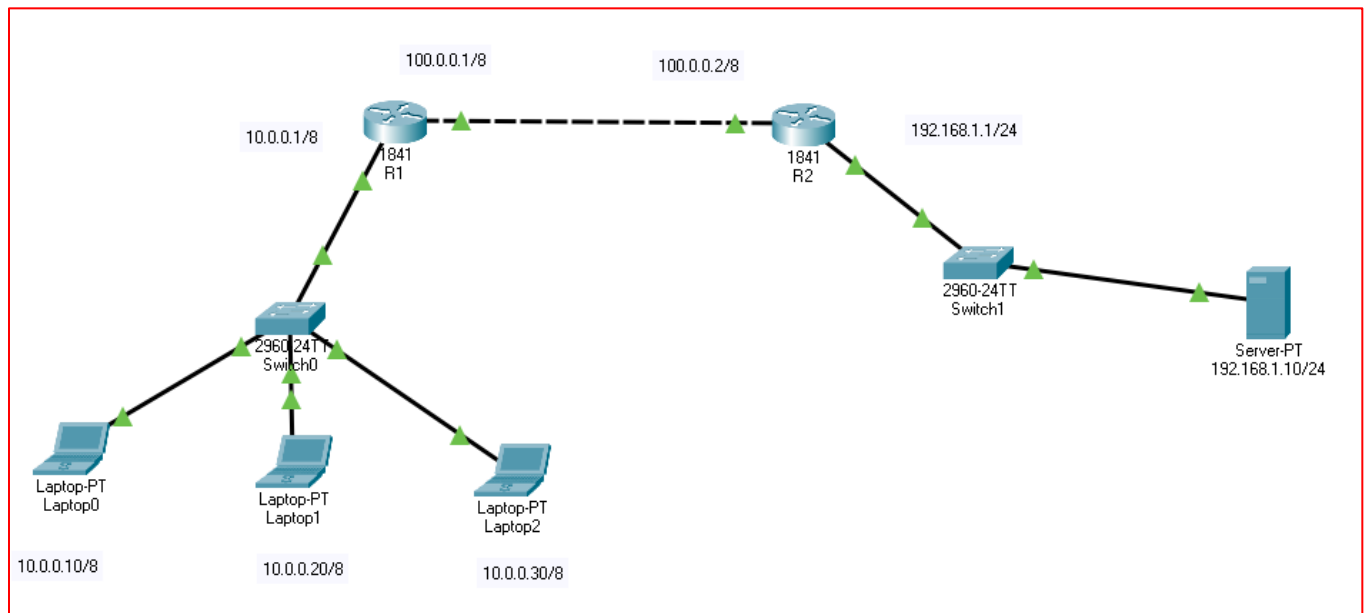
To assign IP address in Laptop click **Laptop** and click **Desktop** and click **IP configuration** and Select **Static** and set **IP address** as given in above table.

Initial IP Configuration

Device / Interface	IP Address	Connected With
Laotop0	10.0.0.10/8	Fa0/1 of R0
Laptop1	**10.0.0.20/8**	**Fa0/1 of R0**
Laptop2	10.0.0.30/8	Fa0/1 of R0
Server0	**192.168.1.10/24**	**Fa0/1 of R2**
F0/0 of R1	100.0.0.1/8	F0/0 of R2
F0/0 of R2	100.0.0.2/8	F0/0 of R1

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.

Laptop1 — □ ✕

Physical | Config | Desktop | Custom Interface

IP Configuration [X]

IP Configuration

○ DHCP   ◉ Static

IP Address: 10.0.0.20

Subnet Mask: 255.0.0.0

Default Gateway: 10.0.0.1

DNS Server:

IPv6 Configuration

○ DHCP   ○ Auto Config   ◉ Static

IPv6 Address: _____ / ___

Link Local Address: FE80::20A:F3FF:FE00:2BC5

IPv6 Gateway:

IPv6 DNS Server:

Laptop0 — □ ✕

Physical | Config | Desktop | Custom Interface

IP Configuration [X]

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

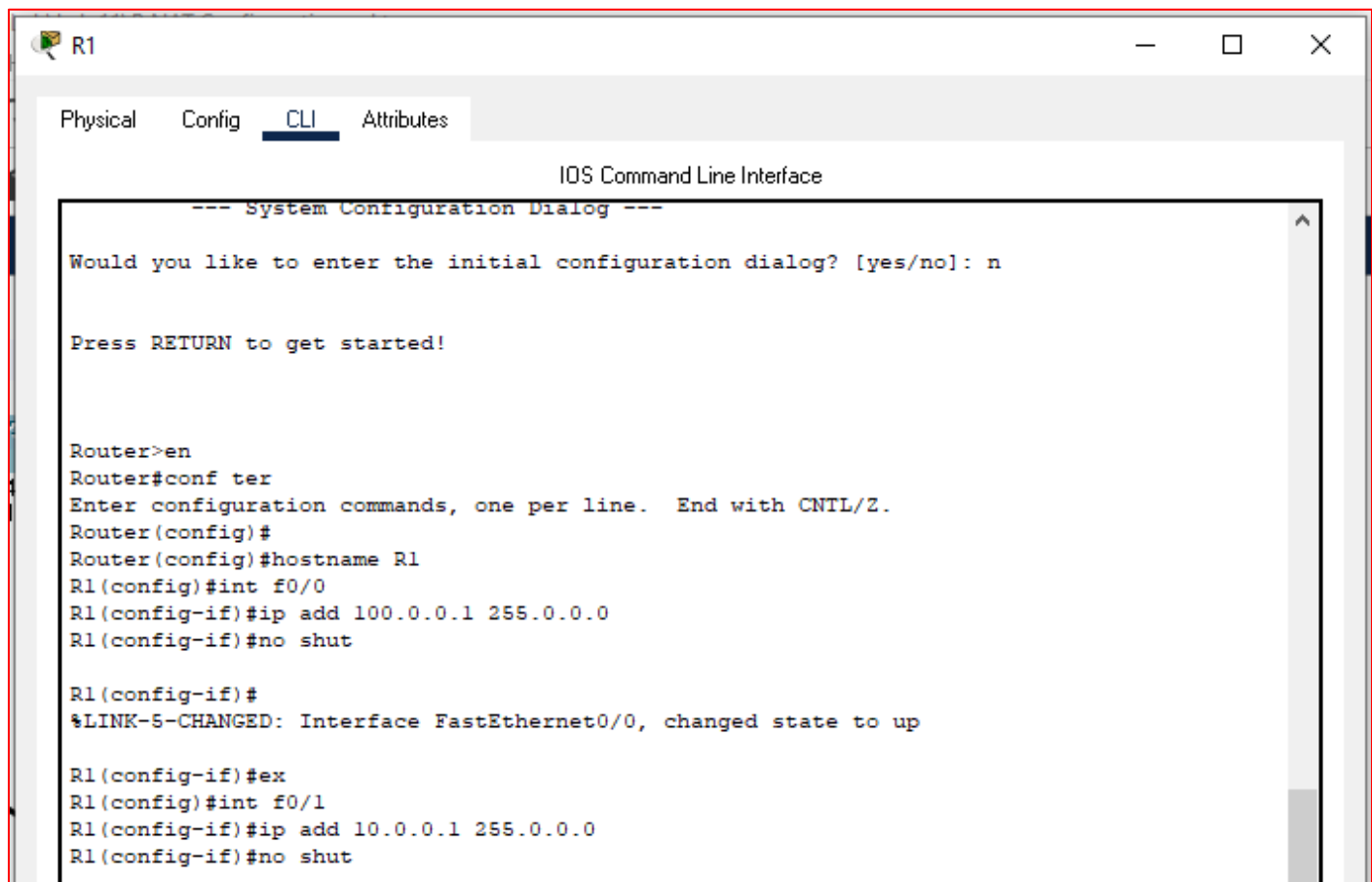○ DHCP   ○ Auto Config   ◉ Static

IPv6 Address: _____ / ___

Link Local Address: FE80::204:9AFF:FE75:CB65

IPv6 Gateway:

IPv6 DNS Server:
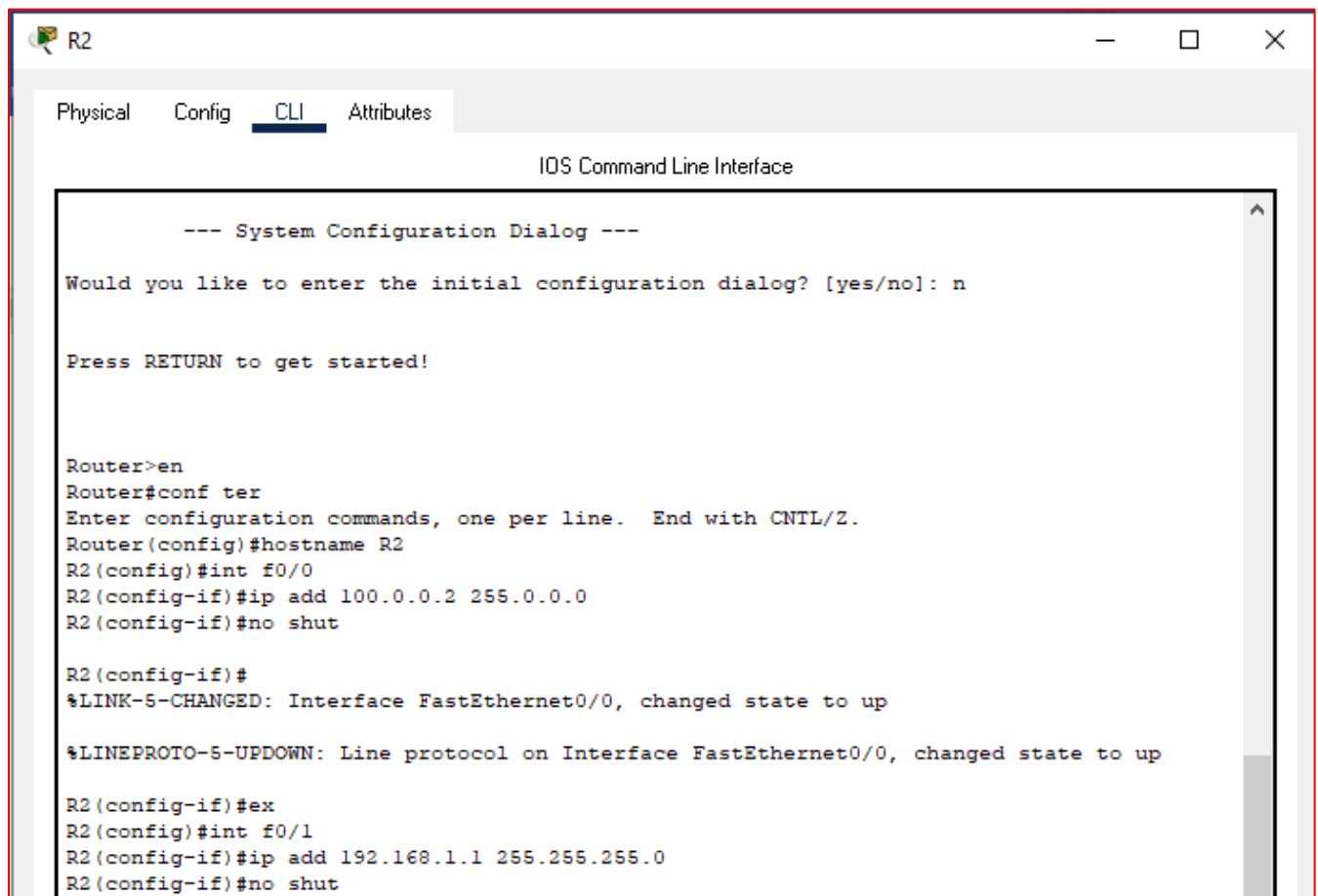
To configure IP address in Router1 click **Router1** and select **CLI** and press **Enter key**.

Router>en
Router#config ter
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname R1
R1(config)#int f0/0
R1(config-if)#ip add 100.0.0.1 255.0.0.0
R1(config-if)#no shut
R1(config-if)#ex
R1(config)#int f0/1
R1(config-if)#ip add 10.0.0.1 255.0.0.0
R1(config-if)#no shut

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

```
Router>en
Router#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#int f0/0
R2(config-if)#ip add 100.0.0.2 255.0.0.0
R2(config-if)#no shut
R2(config-if)#ex
R2(config)#int f0/1
R2(config-if)#ip add 192.168.1.1 255.255.255.0
R2(config-if)#no shut
R2(config-if)#ex
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: -

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

5   Any

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

6   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.

7   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.

8   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 10.0.0.0 0.255.255.255
```

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.

What is the purpose of the overload keyword in the ip nat inside source list 1 pool NAT_POOL overload command?

- It allows many inside hosts to share one or a few inside global addresses.
- It allows a pool of inside global addresses to be used by internal hosts.
- It allows external hosts to initiate sessions with internal hosts.
- It allows a list of internal hosts to communicate with a specific group of external hosts.

**Answers Explanation & Hints:**

Dynamic NAT uses a pool of inside global addresses that are assigned to outgoing sessions. If there are more internal hosts than public addresses in the pool, then an administrator can enable port address translation with the addition of the overload keyword. With port address translation, many internal hosts can share a single inside global address because the NAT device will track the individual sessions by Layer 4 port number.

To define an inside local we use following command

```
Router(config-if)#ip nat inside
```
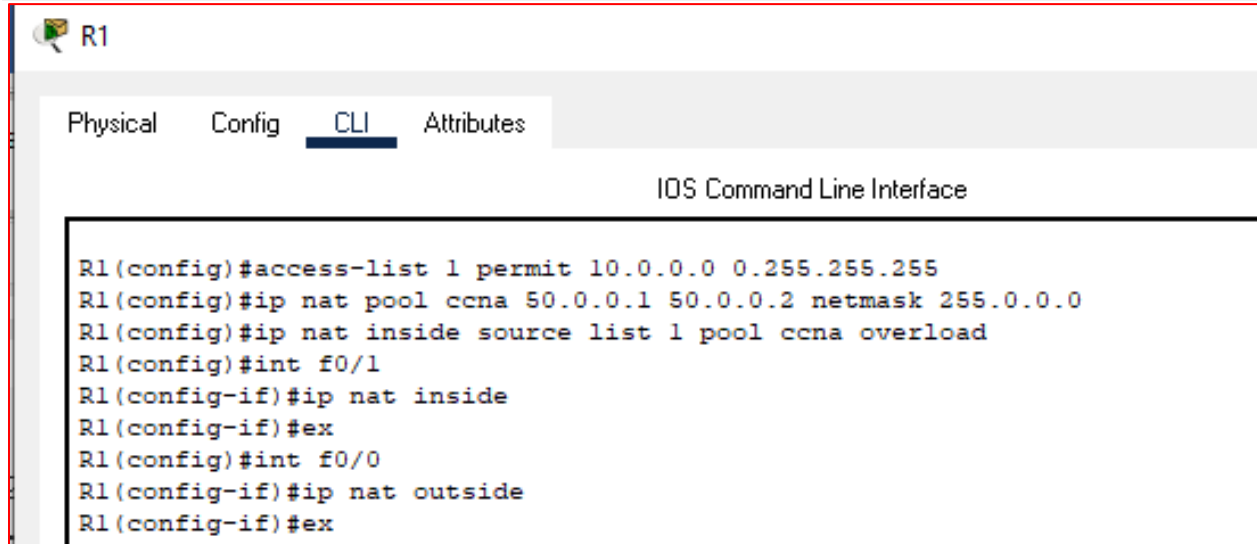
Following command defines inside global

```
Router(config-if)#ip nat outside
```
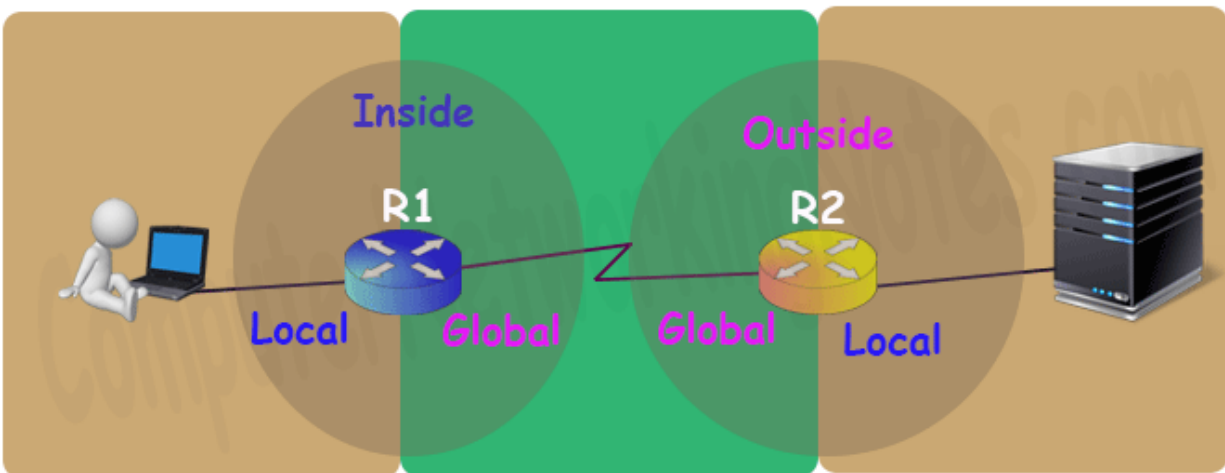
Let's implement all these commands together and configure the PAT.

R1 P NAT Configuration

```
R1#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#access-list 1 permit 10.0.0.0 0.255.255.255
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 overload
R1(config)#interface FastEthernet 0/1
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#interface FastEthernet 0/0
R1(config-if)#ip nat outside
R1(config-if)#exit
R1(config)#
```


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

```
R2>en
R2#conf ter
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)#int f0/0
R2(config-if)#ip nat outside
R2(config-if)#ex
R2(config)#int f0/1
R2(config-if)#in nat inside
                      ^
% Invalid input detected at '^' marker.

R2(config-if)#ip nat inside
R2(config-if)#ex
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
R1(config)# ip route 200.0.0.0 255.255.255.252 100.0.0.2
```

Configure static routing in R2

```
R2(config)#ip route 50.0.0.0 255.0.0.0 100.0.0.1
```
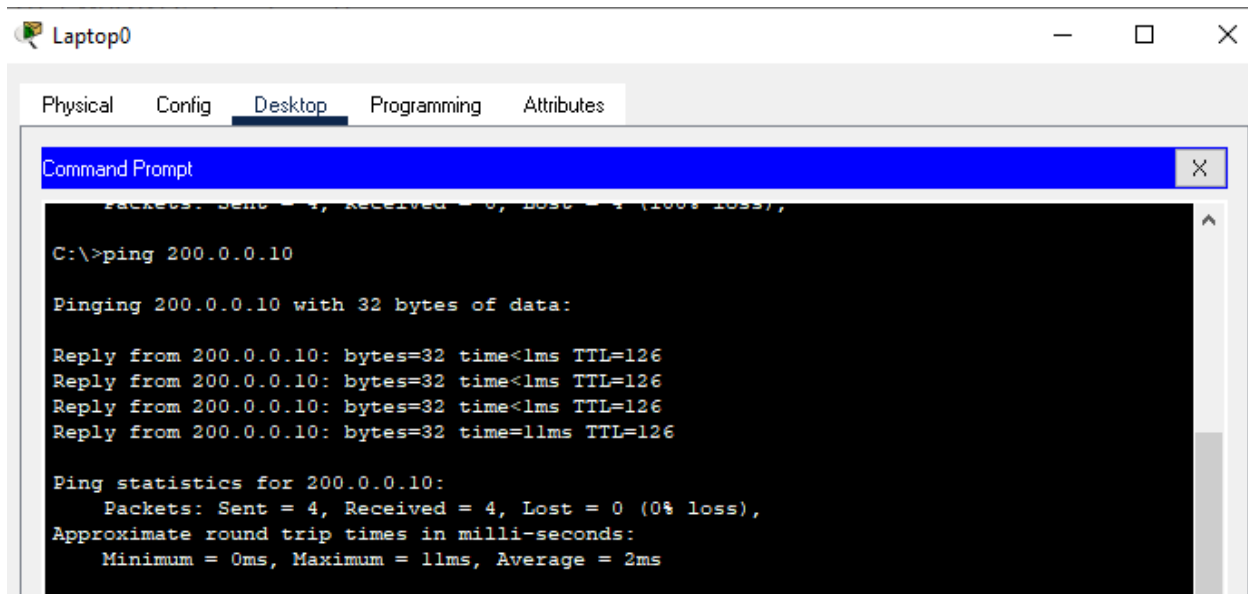
Testing P NAT Configuration

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.

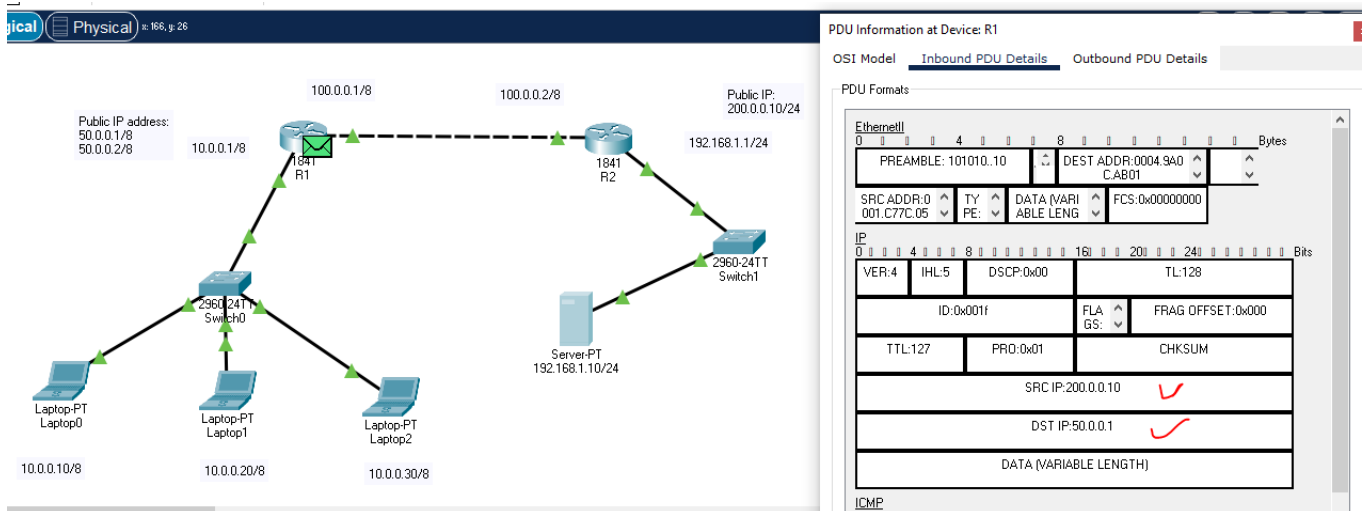| Device	Inside Local IP Address	Inside Global IP Address
Laptop0	10.0.0.10	50.0.0.1
Laptop1	10.0.0.20	50.0.0.2
Server	192.168.1.10	200.0.0.10

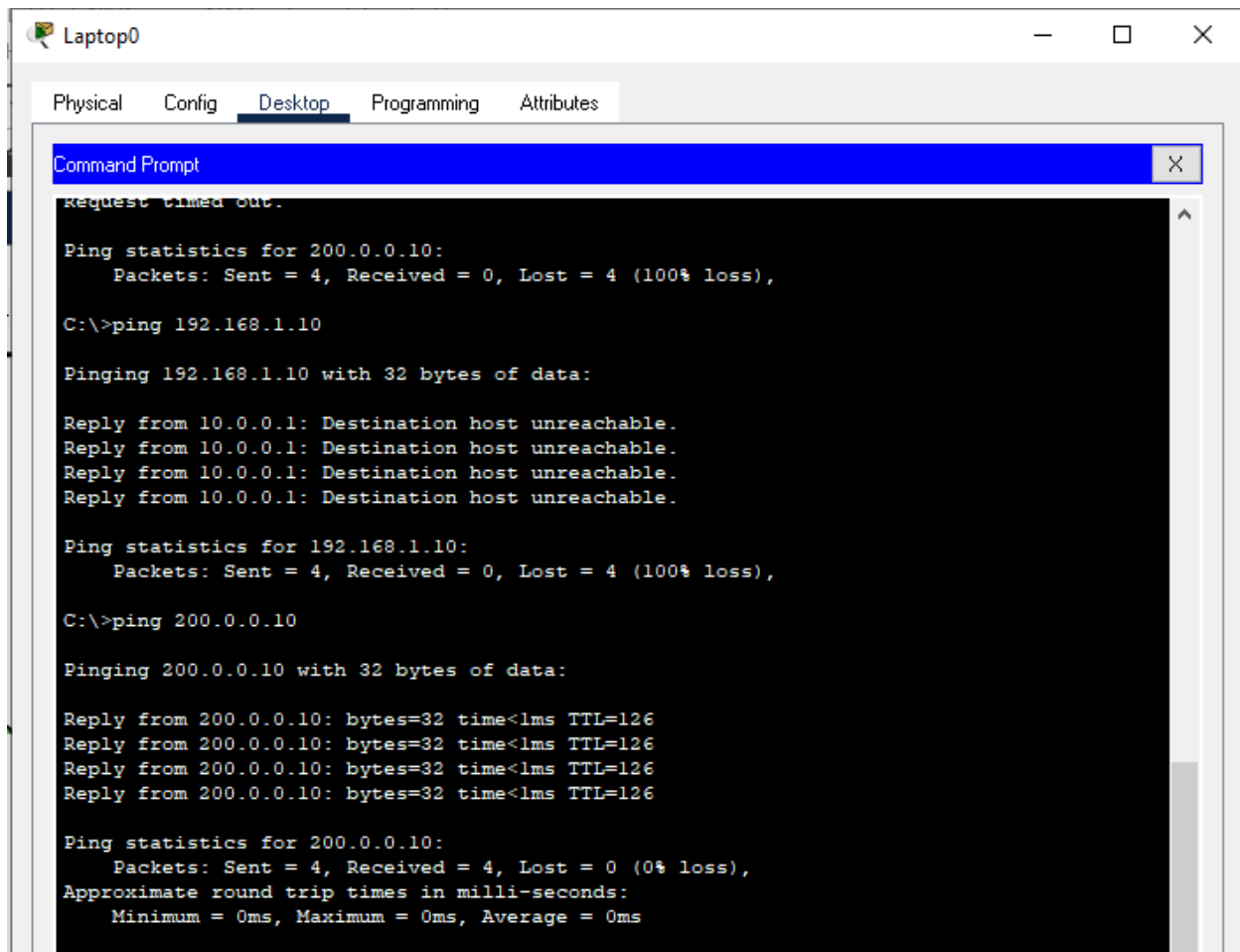To test this setup click **Laptop0** and **Desktop** and click **Command Prompt**.

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

Run **ping 200.0.0.10** command.

S

```
Laptop0                                              —    □    ✕

 Physical    Config    Desktop    Programming    Attributes

 Command Prompt                                              ✕

 Request timed out.

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

 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.
 Reply from 10.0.0.1: Destination host unreachable.

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

 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<1ms TTL=126
 Reply from 200.0.0.10: bytes=32 time<1ms TTL=126
 Reply from 200.0.0.10: bytes=32 time<1ms TTL=126
 Reply from 200.0.0.10: bytes=32 time<1ms 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 = 0ms, Maximum = 0ms, Average = 0ms
```

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.
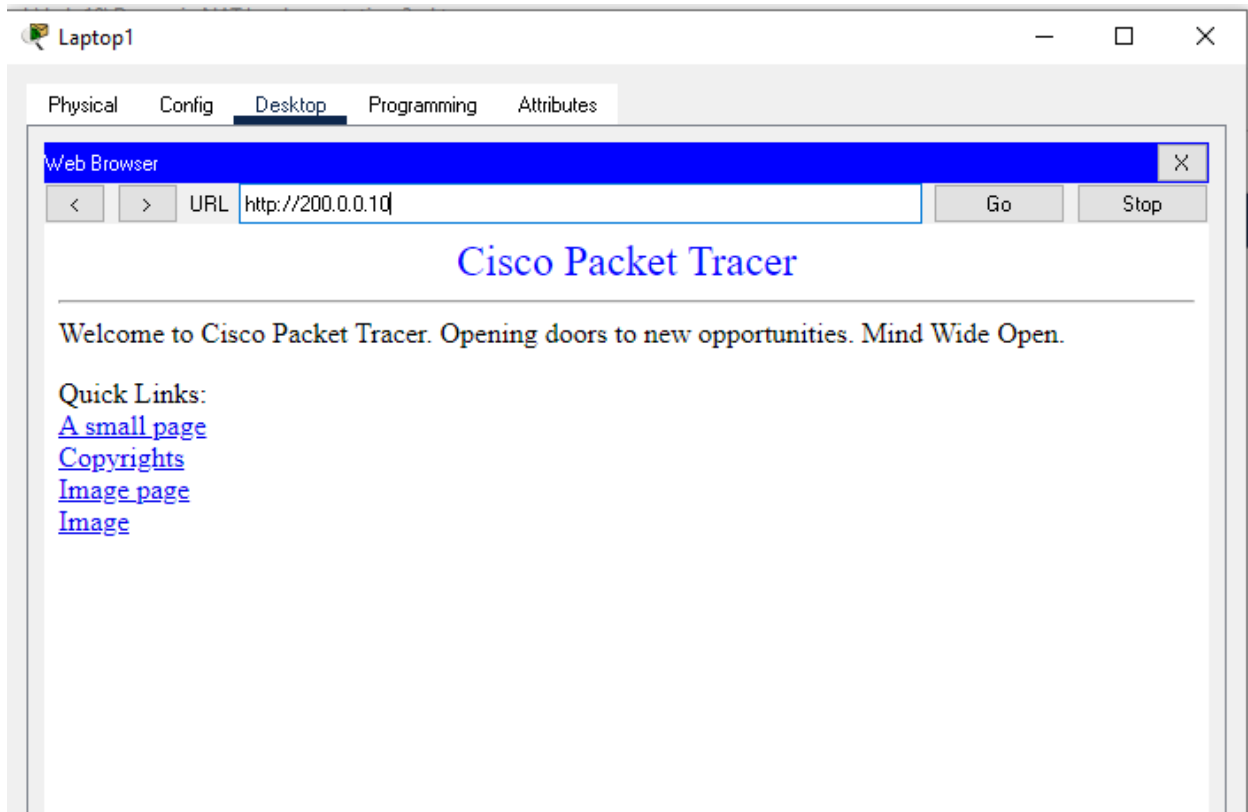
R1#show ip access-lists

Tasks for students:

- **Implement the S-NAT for for web server of (flex and slate) and P-NAT for Client Systems in a single topology.(Use routers and switches).**