DEPARTMENT OF TELECOMMUNICATION ENGINEERING
MEHRAN UNIVERSITY OF ENGINEERING & TECHNOLOGY, JAMSHORO
COMPUTER COMMUNICATION & NETWORKING
(6th Term, 3th Year)
LAB EXPERIMENT # 15/1

Name:_____Roll No: _____

Score:_____Signature of the Lab Tutor:_____Date: _____

OBJECTIVES

#	Topic	#. Of Lectures	CLO	Taxonomy level
1	**Design** the complex network and **configure** the static NAT, dynamic NAT, and PAT (NAT Overload) in the designed network.	3	2	P3

OUTCOME(S)

a. An ability to apply knowledge of math, science, and engineering	**PLO1:** Engineering Knowledge:
k. an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.	**PLO5:** Modern Tool Usage

RUBRICS:

Performance Metric	Exceeds expectation (4-5)	Meets expectations (2-3)	Does not meet expectations (0-1)	Score
Knowledge and application [PLO1]	Applies the appropriate knowledge and concepts to the problem with accuracy and proficiency; shows precise understanding of these knowledge and concepts.	Applies the relevant knowledge and concept to the problem, possibly in a roundabout way; understands the major points of the knowledge, with possible misunderstanding or failure to recall minor points;	Fails to apply relevant knowledge and concepts to the problem; misunderstands or fails to recall critical points.	
Modern Tool Usage [PLO5]	Computer and software are extensively used in the course	Computer and software are somewhat utilized, effort was put into learning new software	Computer and software are not utilized, no attempt was made at learning new software	
			Total Score	

EQUIPMENT

- Two PC
- One server Servers
- Two Routers with console

- Appropriate cables for connections

DISCUSSION & CONFIGURATION

NAT:

Network address translation or commonly known as NAT is a concept which was introduced when the task forces of Telecommunication like, ITU,IEEE figured out that soon ipv4 addresses will run out and there will be scarcity of network addresses, in simple terms the internet was at risk. One solution was already developed that was ipv6 but until ipv6 was adopted as the universal standard for logical addressing another solution or mechanism was needed to provide logical addresses to the overwhelming growth of internet users., hence NAT was introduced.

NAT is used in a way that multiple private networks can have same private ip addresses but they'll be translated as they leave there gateway interface, the translated ip is called public ip address.

NAT has multiple ways to be implemented in a router, like static NAT and Dynamic NAT, there is another method or a way which is called overloading (PAT) which will be discussed later on.

STATIC NAT:

Static NAT is a mechanism of NAT in which each private ip is mapped to a public ip addressed manually, the configuration takes place by the administrator in which

the mapping is done as per requirement and is fixed, every host of the network has a specific private ip and that will be mapped to a specific public ip address and this cannot change until reconfigured manually, hence the name STATIC NAT.

Commands used in STATIC NAT:

a) Defining inside and outside interface on the router.
   The network in a Router before Exit interface or gateway interface is the inside part of your network, and the interface from which the data is travelling to internet is the outside of your network.
   Inside and Outside Interface will be defined by using the commands given below.

   R(Config-if)# ip nat inside (apply this command when you are into the inside interface)
   R(Config-if)#ip nat outsideside (apply this command when you are into the inside interface)
b) R(Config)#ip nat inside source static <private ip of host > <public ip of host>
   Repeat this command for every host in a network

Commands used in DYNAMIC NAT:

a) R(Config-if)# ip nat inside (apply this command when you are into the inside interface)
   R(Config-if)#ip nat outside (apply this command when you are into the inside interface)
b) R(Config)#access-list <number> permit <ip address of network> <wildcard mask of network>
c) Ip nat pool <name of pool> <start of public ip address> <end of public ip address> netmask <subnet mask>
d) Ip nat inside source list <number> pool <name>

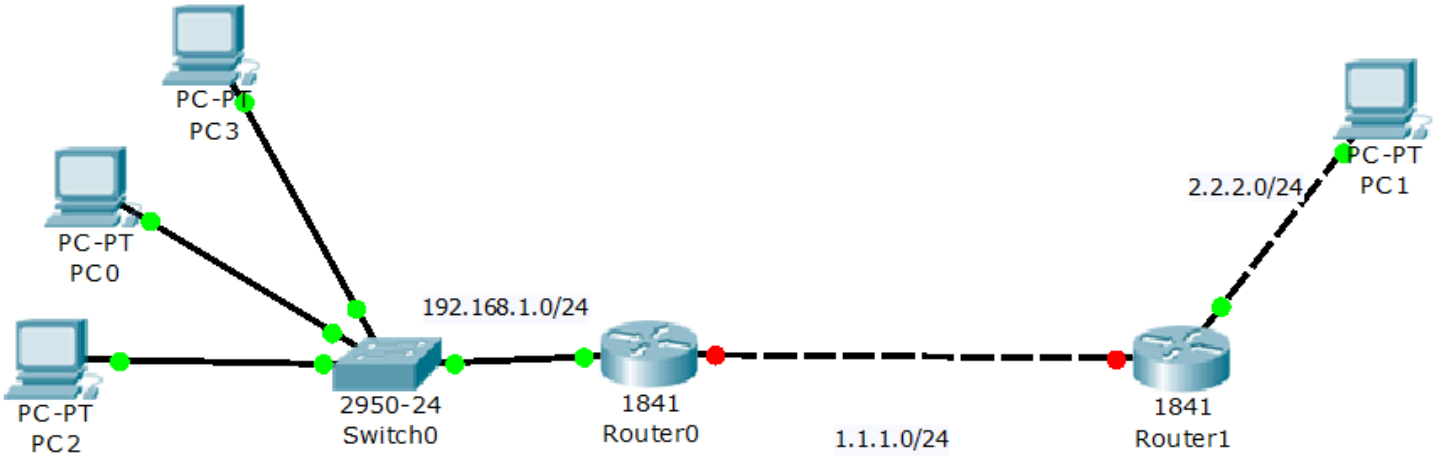
Commands used in OVERLOAD (PAT):

a) R(Config-if)# ip nat inside (apply this command when you are into the inside interface)
   R(Config-if)#ip nat outsideside (apply this command when you are into the inside interface)
b) R(Config)#access-list <number> permit <ip address of network> <wildcard mask of network>
c) Ip nat pool <name of pool> <start of public ip address> <end of public ip address> netmask <subnet mask>
d) Ip nat inside source list <number> pool <name> overload

.

Step 1:

Design the network topology in to the packet tracer



Step 2:

Do the basic configuration on both routers.

Give ip addresses to PC0.PC1,Server0 192.168.1.10,192.168.1.11,192.168.1.12 respectively and give default gateway as 192.168.1.1

1) R0 configuration:

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname R1
R1(config)#int fa0/0
R1(config-if)#ip add 192.168.1.254 255.255.255.0
R1(config-if)#no shut

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

R1(config-if)#int fa0/1
R1(config-if)#ip add 1.1.1.1 255.255.255.0
R1(config-if)#no shut

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
```

On Router1

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname R2
R2(config)#int fa0/0
R2(config-if)#ip add 1.1.1.2 255.255.255.0
R2(config-if)#no shut

R2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up
R2(config)#int fa0/1
R2(config-if)#ip add 2.2.2.254 255.255.255.0
R2(config-if)#no shut
```

2) Configure Static Nat on R1

```
R1(config)#int fa0/0
R1(config-if)#ip nat inside
R1(config-if)#exit
R1(config)#int fa0/1
R1(config-if)#ip nat outside
R1(config-if)#exit
R1(config)#ip nat inside source static 192.168.1.1  1.1.1.1
```

After the basic configuration your topology should be up in all interfaces and static NAT is also configured.
Now to check the configuration.

3) Configure static route on Router1

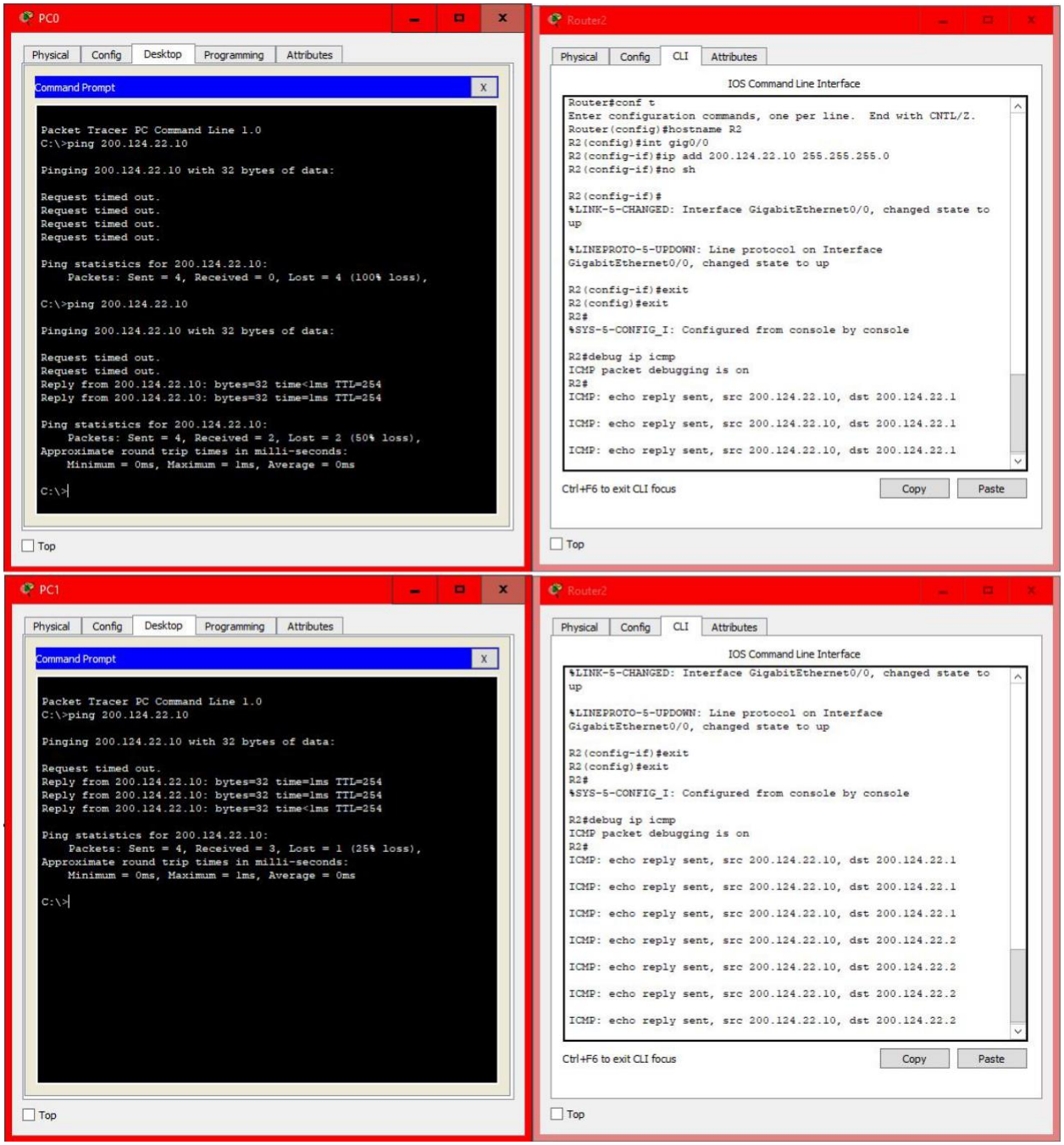
```
R1(config)#ip route 0.0.0.0 0.0.0.0 fa0/1
```

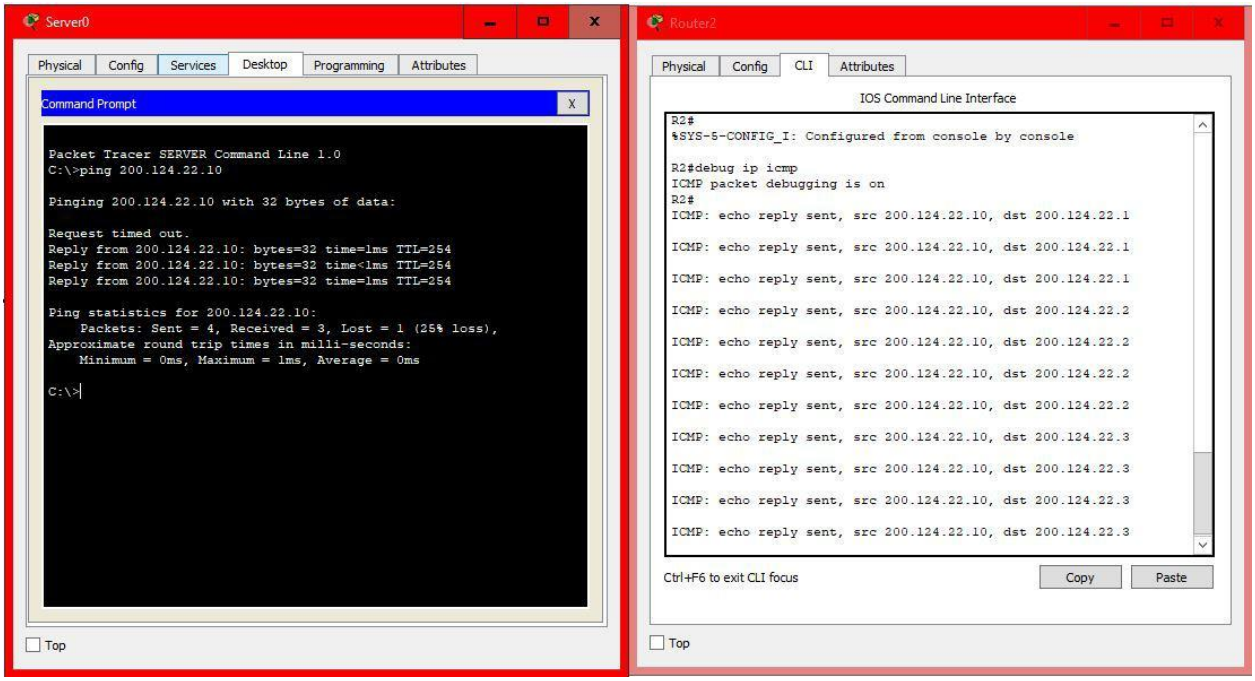
4) Confirm NAT

Enable ip debug in R2 to check the incoming traffic's ip address.

```
R2#debug ip icmp
ICMP packet debugging is on
```

Ping from each host to R2

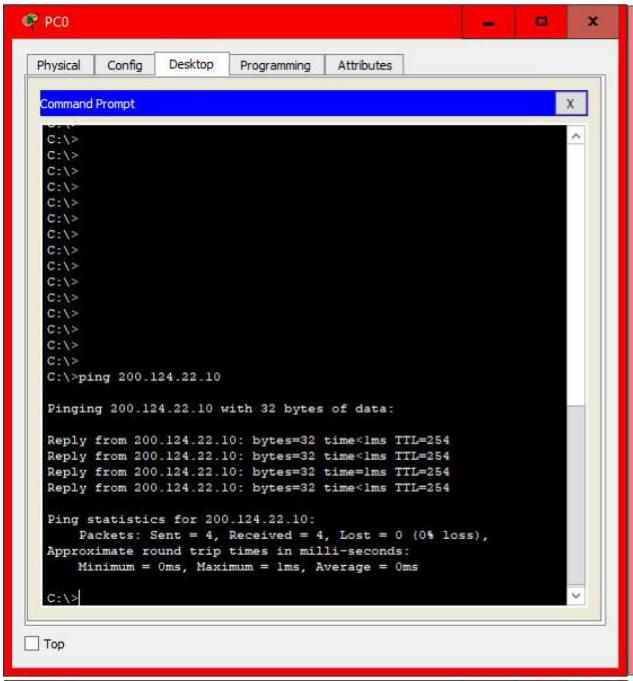

PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Packet Tracer PC Command Line 1.0
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Request timed out.
Request timed out.
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
Top
Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname R2
R2(config)#int gig0/0
R2(config-if)#ip add 200.124.22.10 255.255.255.0
R2(config-if)#no sh
R2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
R2(config-if)#exit
R2(config)#exit
R2#
%SYS-5-CONFIG_I: Configured from console by console
R2#debug ip icmp
ICMP packet debugging is on
R2#
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
Ctrl+F6 to exit CLI focus
Copy
Paste
Top
PC1
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Packet Tracer PC Command Line 1.0
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Request timed out.
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
Top
Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
R2(config-if)#exit
R2(config)#exit
R2#
%SYS-5-CONFIG_I: Configured from console by console
R2#debug ip icmp
ICMP packet debugging is on
R2#
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

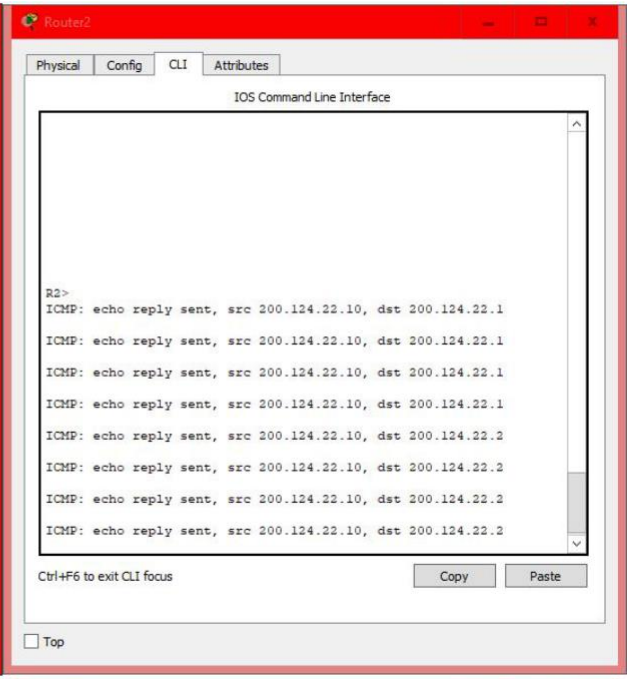


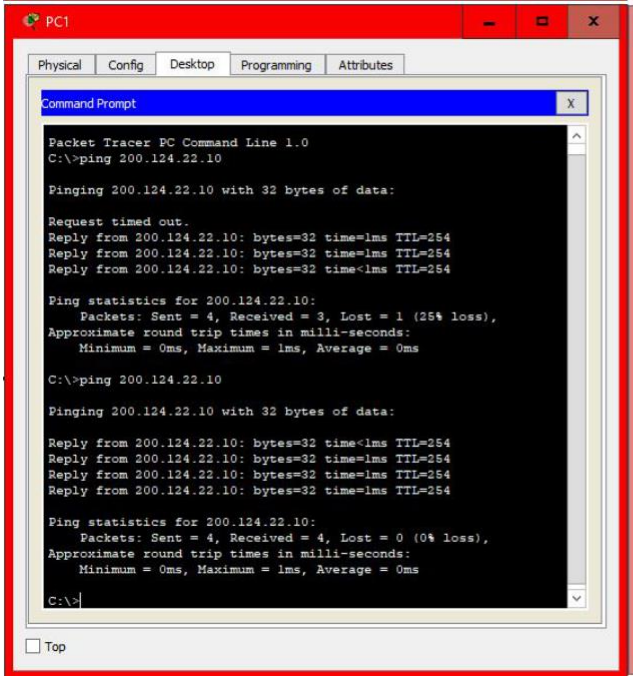
As we can see above the ip address of destination of host is shown as 1.1.1.1 respectively and it is confirmed our private ip's are now translated to public ip addresses.

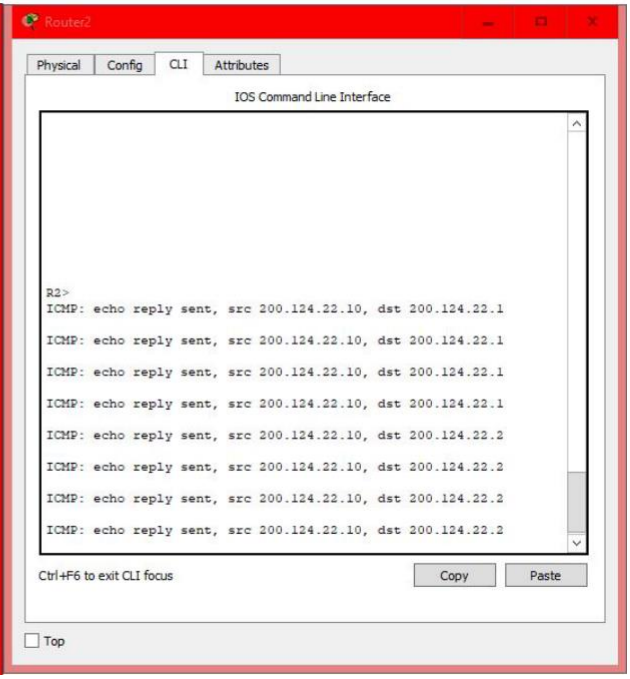
4) Configure DYNAMIC NAT
   Inside and outside interfaces are already configured

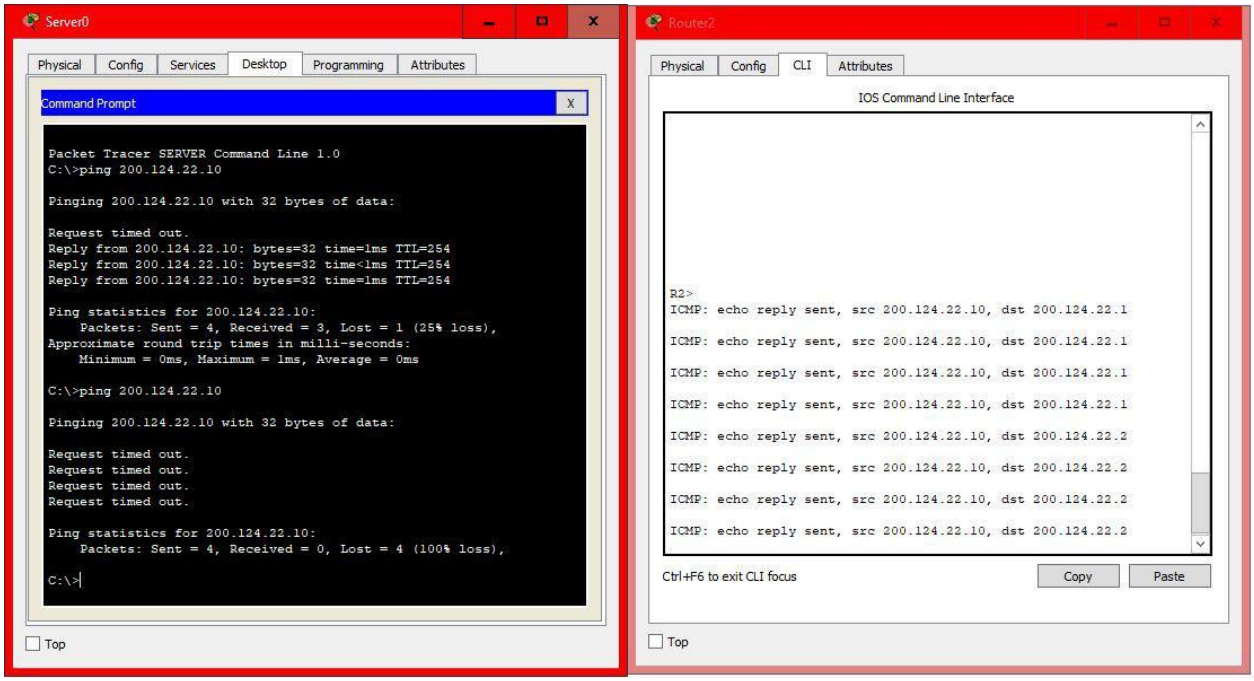
   R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
   R1(config)#ip nat pool dynamic 1.1.1.3 1.1.1.10 netmask 255.255.255.0
   R1(config)#ip nat inside source list 1 pool dynamic


PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
Top


Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
R2>
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
Ctrl+F6 to exit CLI focus
Copy
Paste
Top


PC1
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Packet Tracer PC Command Line 1.0
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Request timed out.
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
Top


Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
R2>
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.2
Ctrl+F6 to exit CLI focus
Copy
Paste
Top



As we can see in above images our dynamic NAT had only two public ip's in the pool which were allocated to pc1 and pc 2 since there was no other public ip left to assign server (3rd host) and the ping was failed from server.

5) Configure Overload (PAT)

Port Address Translation:

PAT is a type of NAT translation, the difference the mapping takes place on unique port numbers instead of different public ip's. The public ip remain same for all hosts.

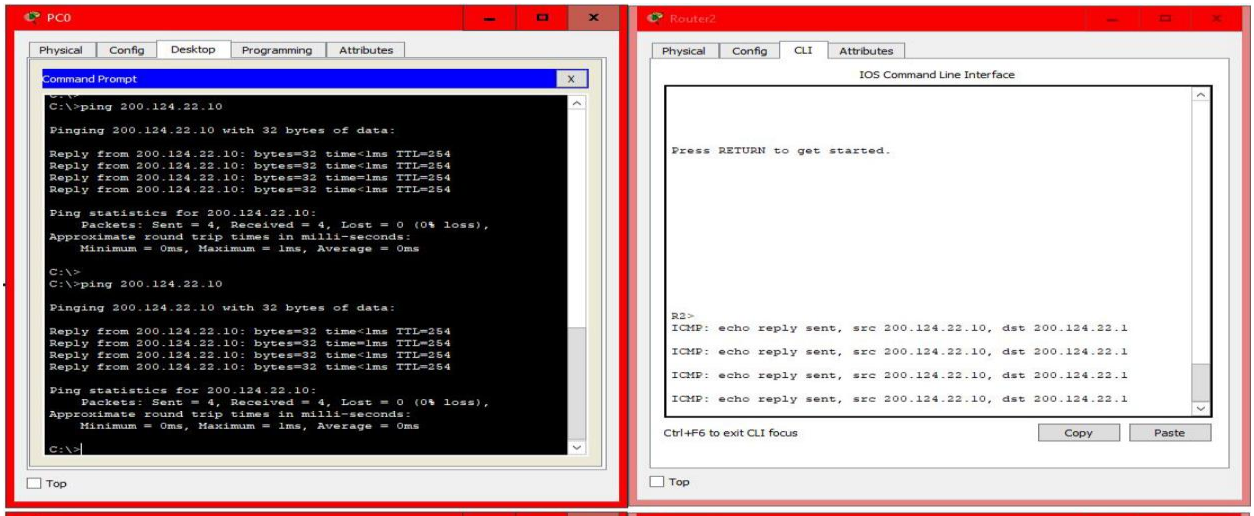
Configuration:

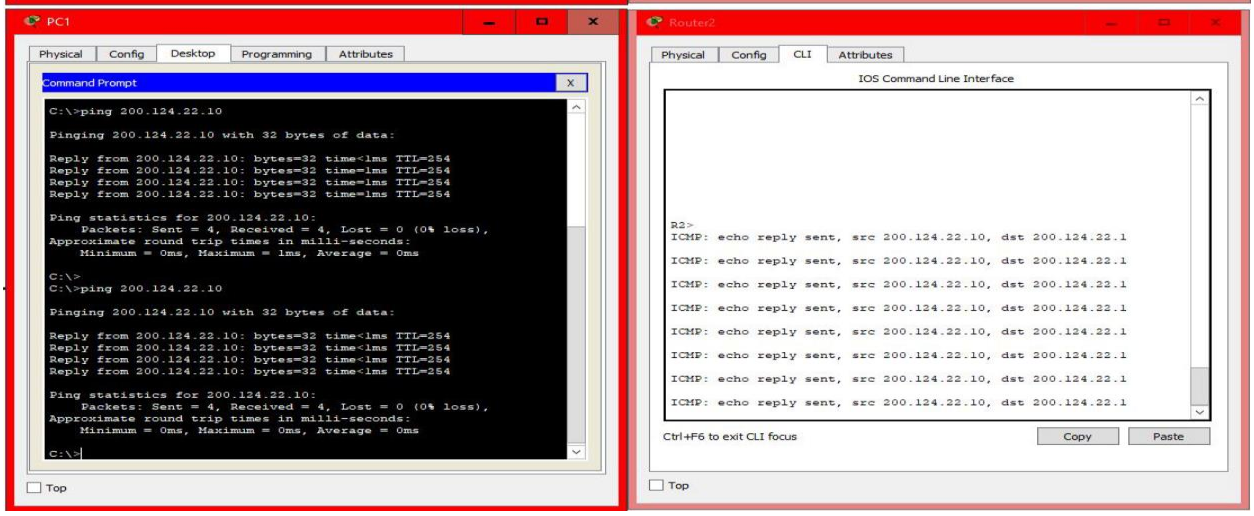
Dynamic PAT

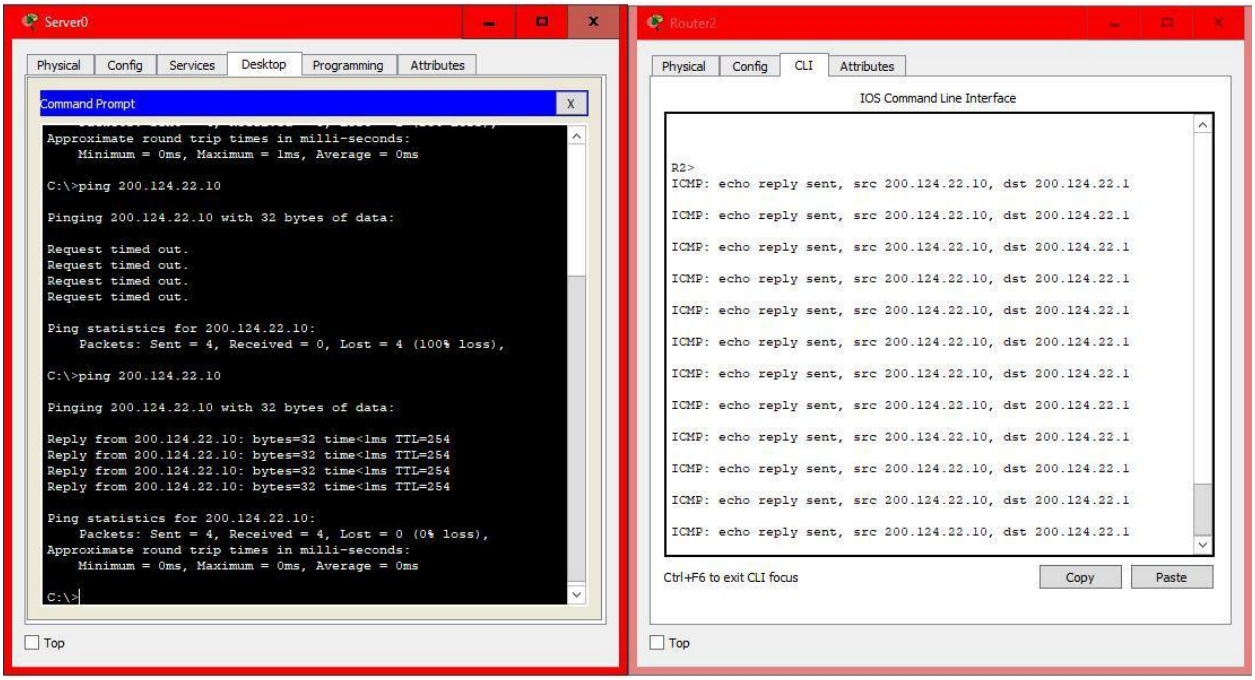
```
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
R1(config)#ip nat inside source list 1 pool dynamic overload
```

Static PAT

```
R1(config)#ip nat inside source list 1 interface fa0/1 overload
```


PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
Top
Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
Press RETURN to get started.
R2>
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
Ctrl+F6 to exit CLI focus
Copy
Paste
Top


PC1
Physical
Config
Desktop
Programming
Attributes
Command Prompt
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Reply from 200.124.22.10: bytes=32 time=1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>
Top
Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
R2>
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
Ctrl+F6 to exit CLI focus
Copy
Paste
Top


Server0
Physical
Config
Services
Desktop
Programming
Attributes
Command Prompt
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>ping 200.124.22.10
Pinging 200.124.22.10 with 32 bytes of data:
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Reply from 200.124.22.10: bytes=32 time<1ms TTL=254
Ping statistics for 200.124.22.10:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>
Top
Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
R2>
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
ICMP: echo reply sent, src 200.124.22.10, dst 200.124.22.1
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

As we can see the ping from three different hosts to R2 were successful and all hosts had same destination address of 200.124.22.1. This is the advantage of PAT that one public ip is enough for entire network to reach internet, this can save a lot of ip addresses and misuse of scarce ipv4 addresses.

FINAL CHECK LIST

1. Return all equipment and materials to their proper storage area.
   Submit your answers to question, before the next laboratory.

Task 1: Answer the following questions.
Q 1) Why was NAT introduced?
Answer: When the task forces of Telecommunication like, ITU,IEEE figured out that soon ipv4 addresses will run out and there will be scarcity of network addresses, in simple terms the internet was at risk. One solution was already developed that was ipv6 but until ipv6 was adopted as the universal standard for logical addressing another solution or mechanism was needed to provide logical addresses to the overwhelming growth of internet users., hence NAT was introduced.
Q2) How many types of Nat available, which Nat mechanism is generally used for commercial purpose ISP's?
Answer: There are 3 types of NAT:

- Static NAT
- Dynamic NAT
- Port Address Translation (PAT)

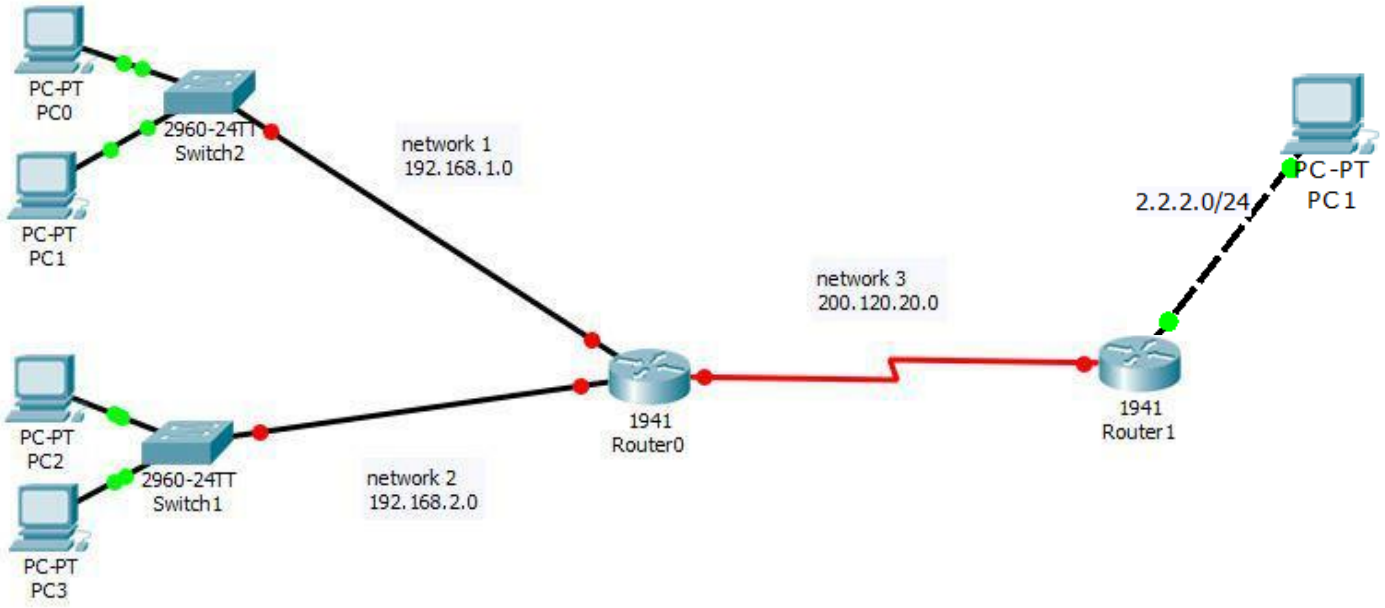
PAT mechanism is generally used for commercial purpose ISP's.
Q3) If there is NAT available why was there a need of PAT mechanism?

Answer: In PAT mechanism, there is just a single inside global IP address providing Internet access to all inside hosts. Whereas in dynamic NAT the number of inside local addresses is greater than the number of inside global addresses that's why PAT is more efficient mechanism.

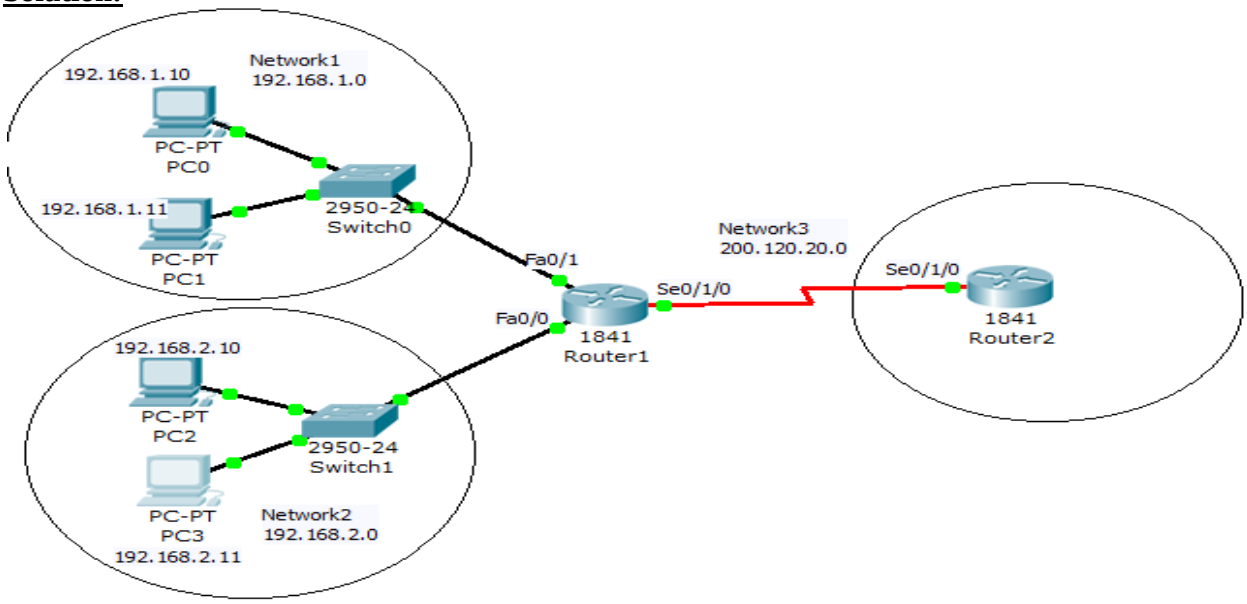
Q 4) What is the difference between Static NAT, Dynamic NAT, Overload?

Static NAT	Dynamic NAT	PAT
Static NAT (Network Address Translation) is one-to-one mapping of a private IP address to a public IP address.	Dynamic NAT can be defined as mapping of a private IP address to a public IP address from a group of public IP addresses called as NAT pool	Port Address Translation (PAT) is another type of dynamic NAT which can map multiple private IP addresses to a single public IP address by using a technology known as Port Address Translation.
Static NAT (Network Address Translation) is useful when a network device inside a private network needs to be accessible from internet.	The public IP address is taken from the pool of IP addresses configured on the end NAT router.	The public IP remain same for all the hosts
	The public to private mapping may vary based on the available public IP address in NAT pool.	

Task 2: Design the following topology and perform the PAT accordingly.



Solution:




```
32-R1(config)#access-list  permit 192.168.1.0 0.0.0.255
                           ^
% Invalid input detected at '^' marker.

32-R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
32-R1(config)#ip nat pool pat 200.120.20.1 200.120.20.1 netmask 255.255.255.0
32-R1(config)#ip nat inside source list 1 pool pat overload
32-R1(config)#exit
32-R1#
%SYS-5-CONFIG_I: Configured from console by console
32-R1#
32-R1#sh ip nat translations
32-R1#sh ip nat translations
```

(NETWORK1)

PC0:

```
PC>ping 200.120.20.10

Pinging 200.120.20.10 with 32 bytes of data:

Reply from 200.120.20.10: bytes=32 time=70ms TTL=254
Reply from 200.120.20.10: bytes=32 time=30ms TTL=254
Reply from 200.120.20.10: bytes=32 time=30ms TTL=254
Reply from 200.120.20.10: bytes=32 time=29ms TTL=254

Ping statistics for 200.120.20.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 29ms, Maximum = 70ms, Average = 39ms
```

```
32-R2>en
32-R2#debug ip icmp
ICMP packet debugging is on
32-R2#
ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1
```

Nat Translation :

```
32-R1#sh ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
icmp 200.120.20.1:13   192.168.1.10:13    200.120.20.10:13   200.120.20.10:13
icmp 200.120.20.1:14   192.168.1.10:14    200.120.20.10:14   200.120.20.10:14
icmp 200.120.20.1:15   192.168.1.10:15    200.120.20.10:15   200.120.20.10:15
icmp 200.120.20.1:16   192.168.1.10:16    200.120.20.10:16   200.120.20.10:16
```

PC1

```
PC>ping 200.120.20.10

Pinging 200.120.20.10 with 32 bytes of data:

Reply from 200.120.20.10: bytes=32 time=30ms TTL=254
Reply from 200.120.20.10: bytes=32 time=25ms TTL=254
Reply from 200.120.20.10: bytes=32 time=30ms TTL=254
Reply from 200.120.20.10: bytes=32 time=25ms TTL=254

Ping statistics for 200.120.20.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 25ms, Maximum = 30ms, Average = 27ms
```

```
ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1

ICMP: echo reply sent, src 200.120.20.10, dst 200.120.20.1
```

Nat Translation :

```
32-R1#sh ip nat translation
Pro  Inside global     Inside local       Outside local      Outside global
icmp 200.120.20.1:13   192.168.1.11:13    200.120.20.10:13   200.120.20.10:13
icmp 200.120.20.1:14   192.168.1.11:14    200.120.20.10:14   200.120.20.10:14
icmp 200.120.20.1:15   192.168.1.11:15    200.120.20.10:15   200.120.20.10:15
icmp 200.120.20.1:16   192.168.1.11:16    200.120.20.10:16   200.120.20.10:16
```