Name & ID: Basil khowaja bk08432 **Date:** 05/11/2

EE-424L Data Communication & Networking
Fall 2024

Habib University

Dhanani School of Science & Engineering

LAB 09: Configuration of Standard, Extended Access Control List, NAT and PAT

Lab #09 Marks distribution:

		LR2=10	LR4=10	LR5=20	AR4=20
In-Lab Tasks	Task 1	/5	/5	/5	/20
	Task 2	/5	/5	/5	
	Task 3			/10	
Marks Obt.	/60				

Objectives	**The objective of this lab is to configure and verify Standard, Extended Access Control List (ACL) , Network Address Translation (NAT) and Port Address Translation (PAT).**

Introduction

Access Control List (ACL) is a security feature that allows you to filter the network traffic based on configured statements. An ACL can be used to filter either inbound or outbound traffic on an interface. Once you applied an access list on a router, the router examines every packet moving from interface to another interface in the specified direction and takes the appropriate action.

Types of ACL

An ACL can be either of the following two types.

1. **Standard access lists**

A Standard access list can use only the source IP address in an IP packet to filter the network traffic. Standard access lists are typically used permit or deny an entire system or network. They cannot be used to filter individual protocol or services such as FTP and Telnet.

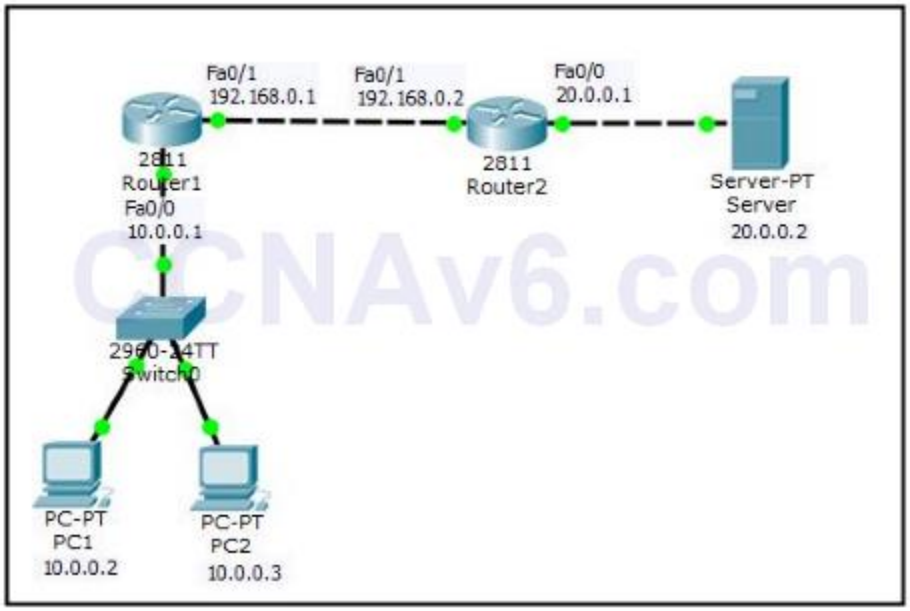
2. **Extended access lists**

Extended access lists use the source and destination IP addresses. They can be used to filter specific protocol or service. An ACL can be configured using either a number or a name. If you decide to use a name to configure an ACL it is referred as Named ACL.

Network Address Translation (NAT) and **Port Address Translation (PAT)** are the two protocols via which we can map the unregistered private (inside local address of an internal network to a registered public (inside global) address of an external network before moving the packet.
The primary distinction is that **NAT** is used to map public IP addresses to private IP addresses in a one-to-one or many-to-one relationships. On the other hand, **PAT** is a sort of **NAT** in which numerous private IP addresses (many-to-one) are mapped into a single public IP address via ports.

Task 1: Configuration of Standard ACL

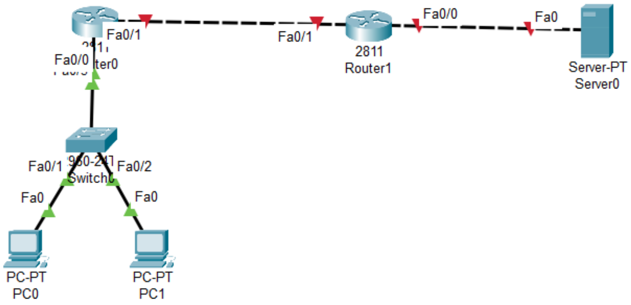
Configure and create the below topology in packet tracer and complete the IP configuration and interface configuration in Table 1 according to your Network Topology. Attach your network topology with labelled IPs, subnet mask and interfaces as shown in below figure.



Device Name	Interface	IPs Assigned to devices
PC1	Fa0/0	10.0.0.2
PC2	Fa0/0	10.0.0.3
Router1	Fa0/0, fa0/1	10.0.0.1, 192.168.0.1
Router2	Fa0/0, fa0/1	20.0.0.1, 192.168.0.2
Server	Fa0/0	20.0.0.2

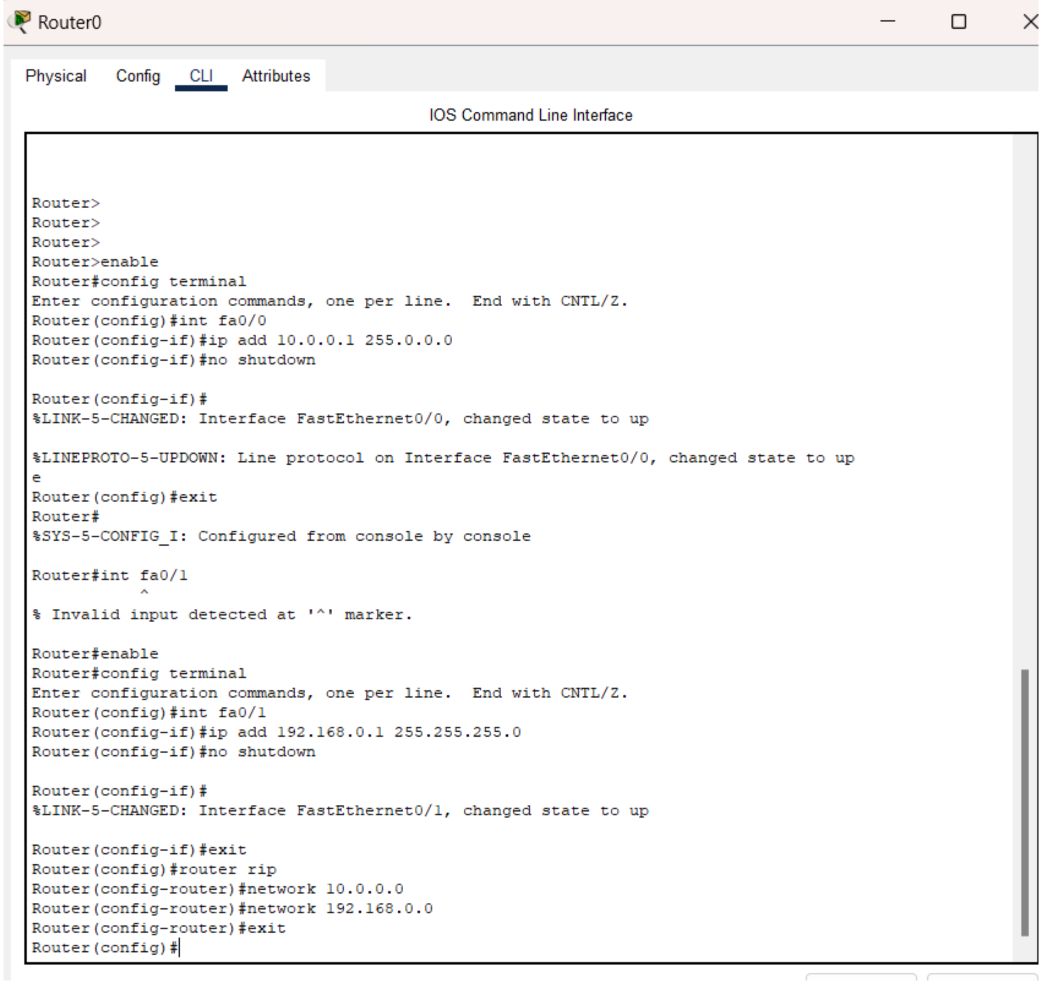
Once you have created the above topology, configure the appropriate IP addresses as mentioned in the topology. To do so, execute the following commands on Router1.

```
Router1( config)# int fa0/ 0
Router1( config-if)# ip add 10.0.0.1 255.0.0.0
Router1( config-if)# no shut
Router1( config-if)# exit

Router1( config)# int fa0/ 1
Router1( config-if)# ip add 192.168.0.1 255.255.255.0
Router1( config-if)# no shut
Router1( config-if)# exit
```



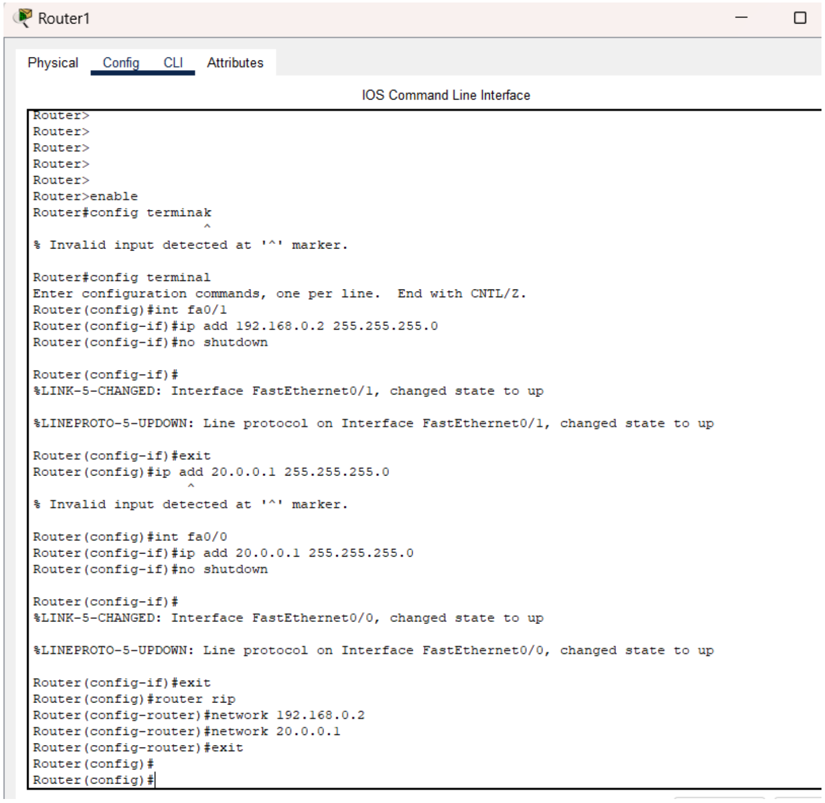
This is my topology^

Once you have configured appropriate IP addresses, use a routing method such as RIP. Write the commands below for RIP on Router 1.



```
Router>
Router>
Router>
Router>enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#ip add 10.0.0.1 255.0.0.0
Router(config-if)#no shutdown

Router(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
e
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#int fa0/1
           ^
% Invalid input detected at '^' marker.

Router#enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/1
Router(config-if)#ip add 192.168.0.1 255.255.255.0
Router(config-if)#no shutdown

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

Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 192.168.0.0
Router(config-router)#exit
Router(config)#
```

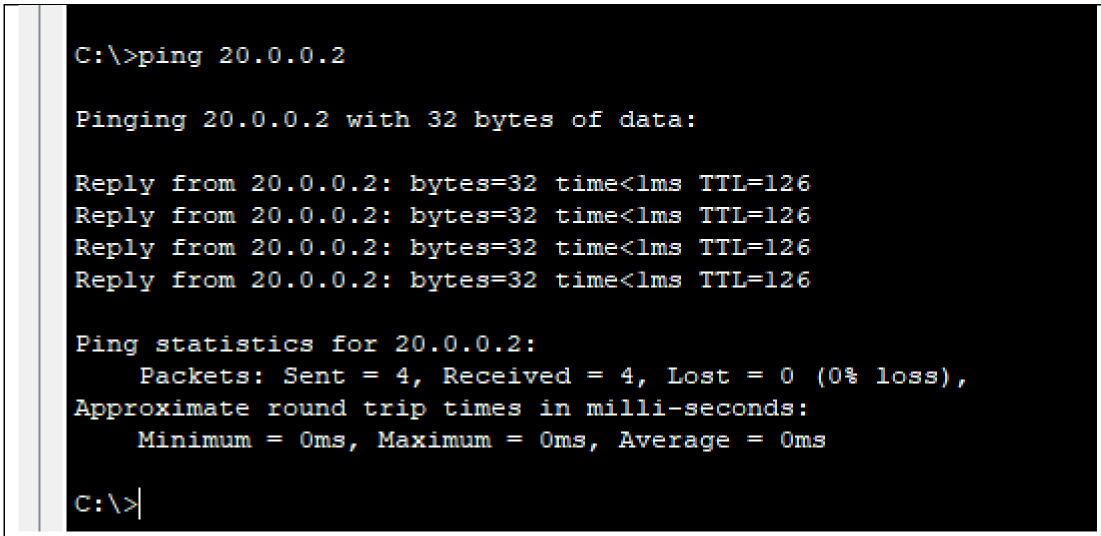
Next, move on to Router2 and configure IP addresses and the RIP routing protocol. Attach the screenshots below.



```
Router>
Router>
Router>
Router>
Router>
Router>enable
Router#config terminak
                     ^
% Invalid input detected at '^' marker.

Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/1
Router(config-if)#ip add 192.168.0.2 255.255.255.0
Router(config-if)#no shutdown

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

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

Router(config-if)#exit
Router(config)#ip add 20.0.0.1 255.255.255.0
                   ^
% Invalid input detected at '^' marker.

Router(config)#int fa0/0
Router(config-if)#ip add 20.0.0.1 255.255.255.0
Router(config-if)#no shutdown

Router(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

Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 192.168.0.2
Router(config-router)#network 20.0.0.1
Router(config-router)#exit
Router(config)#
Router(config)#
```

After configuring IP addresses on routers, configure IP addresses on PC1, PC2, and Server.

Now, open the Command Prompt on PC1 and type ping 20.0.0.2 what is the response?

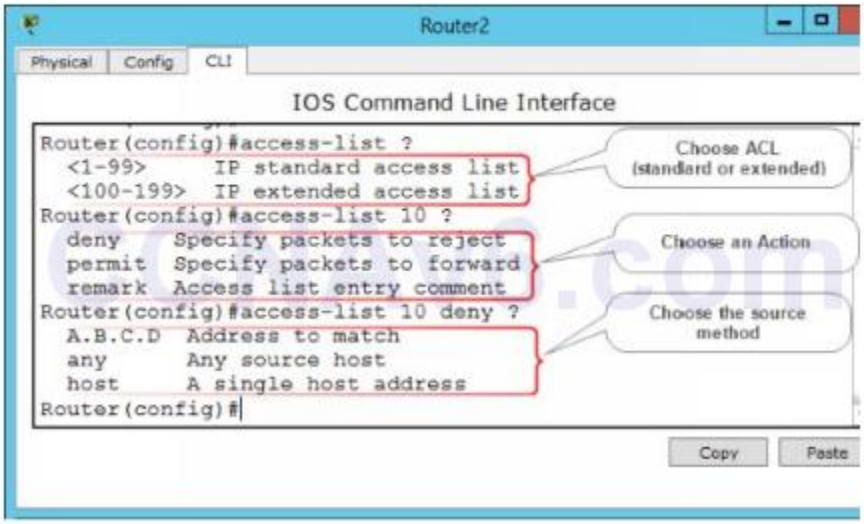


```
C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

it was able to ping and was able to send packets

Now, you have configured the appropriate IP addresses and routing on your network topology.

Before configuring an ACL, we would like to explain the command syntaxes used to configure it. The following figure shows the command syntax used to configure an ACL.



```
Router(config)#access-list ?
  <1-99>      IP standard access list
  <100-199>   IP extended access list
Router(config)#access-list 10 ?
  deny    Specify packets to reject
  permit  Specify packets to forward
  remark  Access list entry comment
Router(config)#access-list 10 deny ?
  A.B.C.D  Address to match
  any      Any source host
  host     A single host address
Router(config)#
```

Configure Standard ACL

In this task, we will restrict host PC 1(10.0.0.2) from accessing Server (20.0.0.2). To do so, perform the following steps:

For our Standard Access-List, we can use the **ACL Number** 1 to 99. These numbers can be **100 to 199**, if you use extended ACLs.

1. First, execute the following command on Router 2 to deny host 10.0.0.2.

Router2(config)# access-list 10 deny host 10.0.0.2

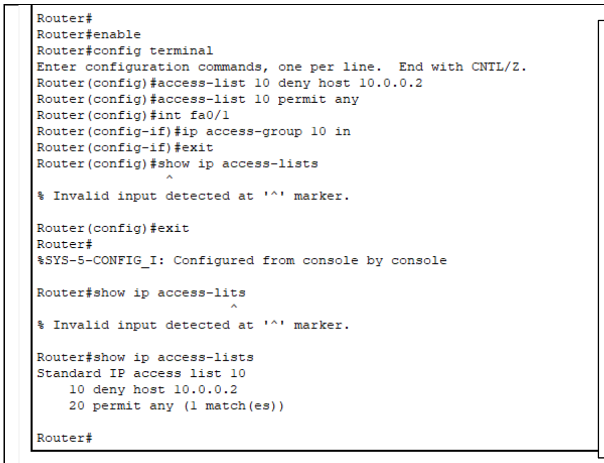
2. Once you deny a host on a router, the router will deny all the hosts until you explicitly define the permitted hosts. In the following command we will permit all the hosts.

Router2(config)# access-list 10 permit any

3. Next, switch to the interface on which you want to apply the ACL, in this case Fa0/ 1, and define the direction (inbound or outbound) of traffic that you want to filter. In this case, we will filter the incoming packets towards Router2. To do so execute the following commands.

Router2(config)# int fa0/1
Router2(config-if)# ip access-group 10 in
Router2(config-if)# exit

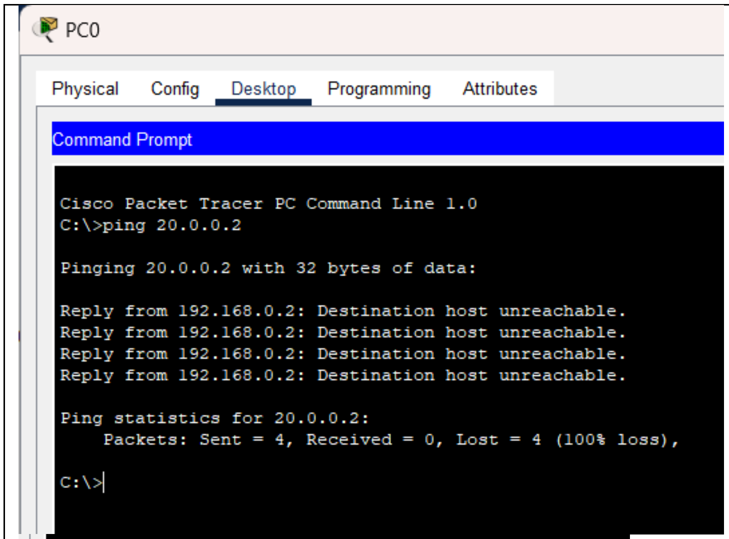
4. Once you applied an ACL on a router, execute the following command to view the applied ACLs.
Run **show ip access-lists** on Router 2 and write your observations below.



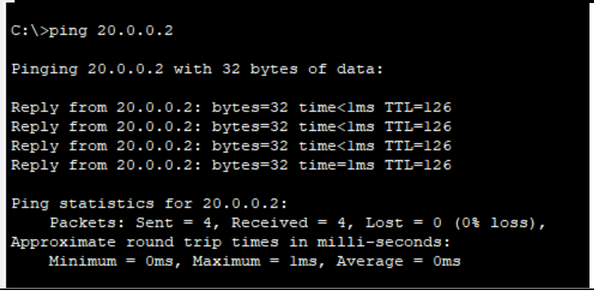
```
Router#
Router#enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#access-list 10 deny host 10.0.0.2
Router(config)#access-list 10 permit any
Router(config)#int fa0/1
Router(config-if)#ip access-group 10 in
Router(config-if)#exit
Router(config)#show ip access-lists
                ^
% Invalid input detected at '^' marker.

Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip access-lits
                        ^
% Invalid input detected at '^' marker.

Router#show ip access-lists
Standard IP access list 10
    10 deny host 10.0.0.2
    20 permit any (1 match(es))

Router#
```

after applying the access control list (ACL) and using the show ip access-lists command on Router2, it is observed that the ACL 10 has been correctly configured with two statements. The first statement denies traffic from the IP address 10.0.0.2, while the second statement permits any other IP addresses. The command output confirms that ACL 10 is active, and the deny and permit rules are functioning, as indicated by the matched packet count in the permit statement.

5. Next, open the Command Prompt of PC1 and PC2 and try to ping server, what is the response?



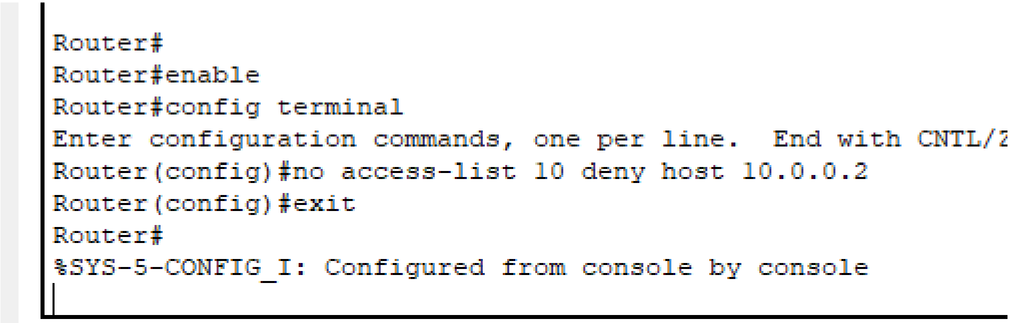
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 192.168.0.2: Destination host unreachable.
Reply from 192.168.0.2: Destination host unreachable.
Reply from 192.168.0.2: Destination host unreachable.
Reply from 192.168.0.2: Destination host unreachable.

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```



```
C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

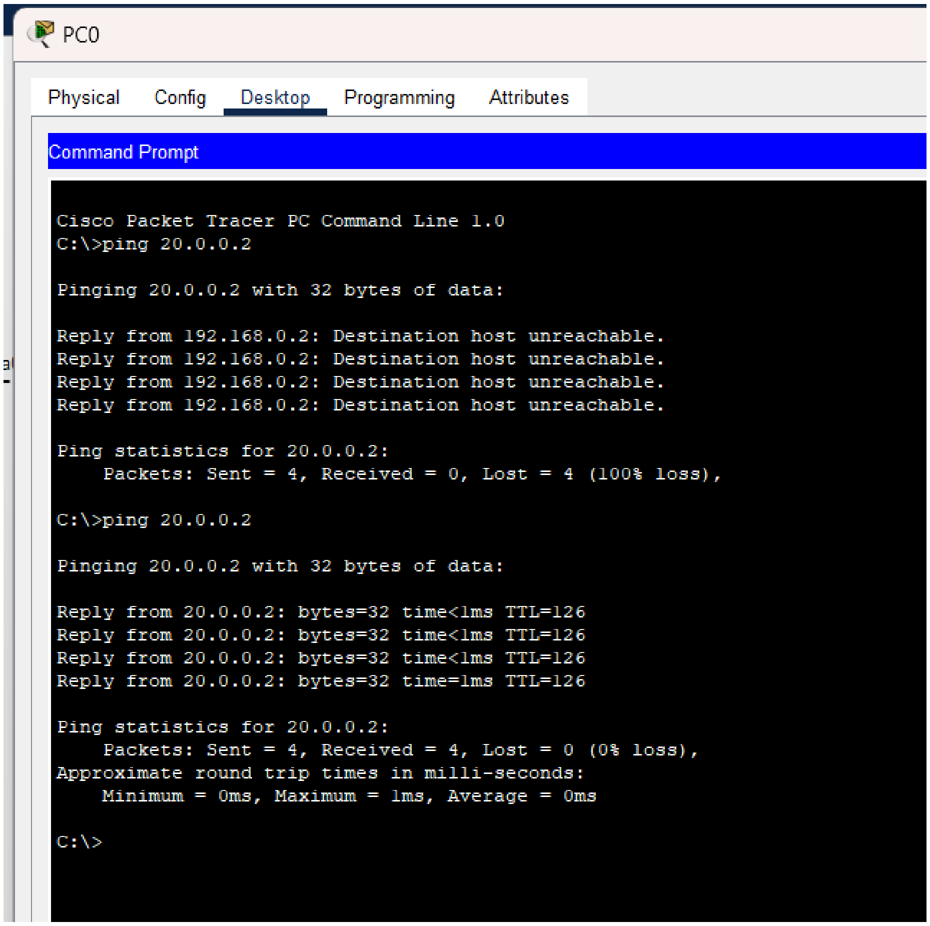
From pc 0 when I pinged the server it was not reachable however from pc 1 it was reachable

6. Now, you have tested your ACL configuration. Now, remove the ACL configuration. To remove the configured ACL, execute the following command on Router2.
Router2(config)# no access-list 10 deny host 10.0.0.2



```
Router#
Router#enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z
Router(config)#no access-list 10 deny host 10.0.0.2
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

7. Try to ping again from PC0 to Router2, this time you should be able ping successfully, because you have removed the applied ACL.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 192.168.0.2: Destination host unreachable.
Reply from 192.168.0.2: Destination host unreachable.
Reply from 192.168.0.2: Destination host unreachable.
Reply from 192.168.0.2: Destination host unreachable.

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time<1ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

In this first ping was with acl and second ping is without acl, it means when I removed acl then pc 0 was able to ping and packets were sent successfully

Task 2: Configuration of Extended ACL

Extended ACL is more precise than standard ACL. Even we can block a particular IP or range of IP address or network address using extended ACL. We can also allow certain hosts and block few as per our requirement using extended ACL. Here in this task we will learn to configure and use Extended access-list. **Extended ACLs** are a little complex if we compare it with **Standard ACLs**. With **Extended ACLs**, we can restrict or allow specific things like **destination, protocol** or **port**.

Configure ACL list on Router 2:
First, we'll create a statement that will permit the PC1 workstation access to Server:

R1(config)#access-list 100 permit ip 10.0.0.2 0.0.0.0 192.168.0.1 0.0.0.0

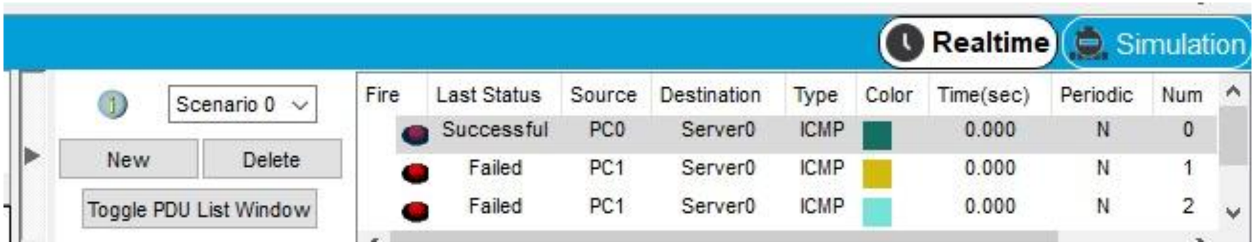
Next, we need to create a statement that will deny the PC2 workstation access to Server:

R1(config)#access-list 100 deny ip 10.0.0.3 0.0.0.0 192.168.0.1 0.0.0.0

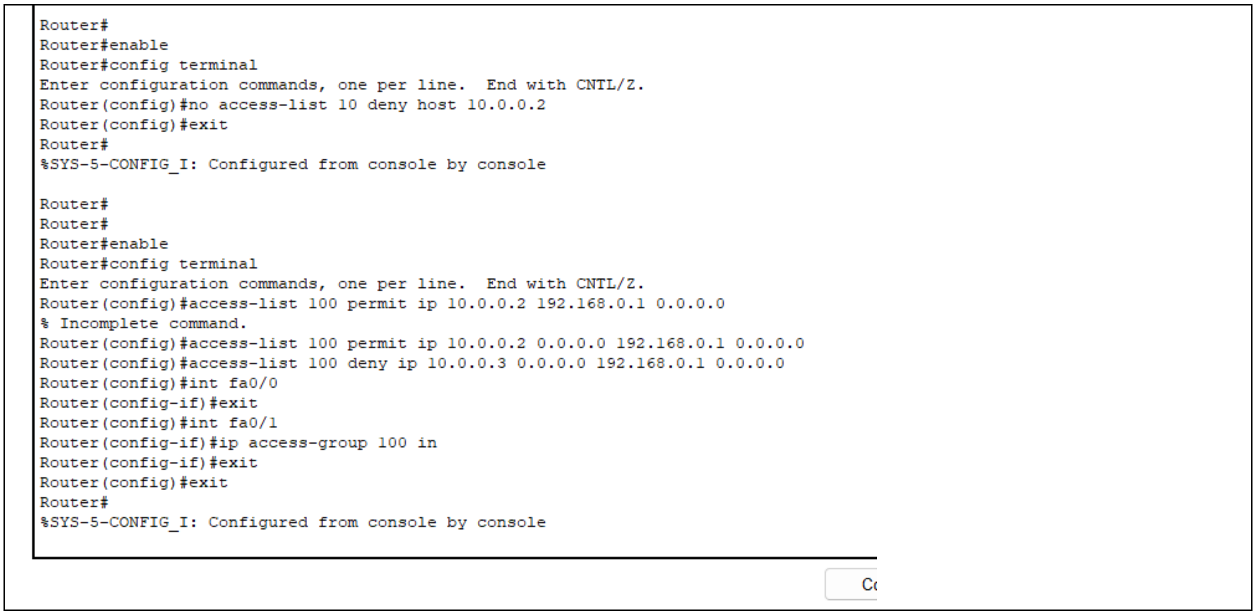
Lastly, we need to apply the access list to the **Fa0/0** interface on R2:

R1(config)#int f0/1
R1(config-if)#ip access-group 100 in

Ping PC1 and PC2 to Server and note down the response below.



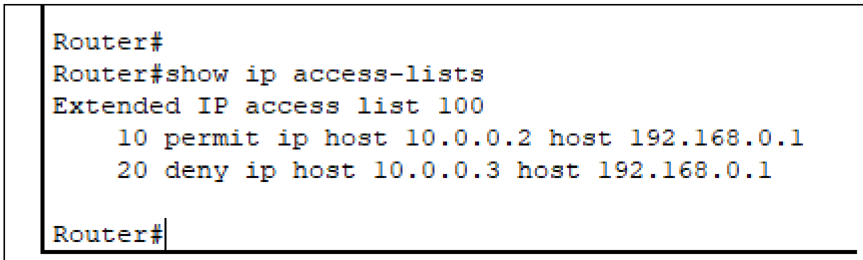
Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num
●	Successful	PC0	Server0	ICMP	🟩	0.000	N	0
●	Failed	PC1	Server0	ICMP	🟨	0.000	N	1
●	Failed	PC1	Server0	ICMP	🟦	0.000	N	2



```
Router#
Router#enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#no access-list 10 deny host 10.0.0.2
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
Router#
Router#enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#access-list 100 permit ip 10.0.0.2 192.168.0.1 0.0.0.0
% Incomplete command.
Router(config)#access-list 100 permit ip 10.0.0.2 0.0.0.0 192.168.0.1 0.0.0.0
Router(config)#access-list 100 deny ip 10.0.0.3 0.0.0.0 192.168.0.1 0.0.0.0
Router(config)#int fa0/0
Router(config-if)#exit
Router(config)#int fa0/1
Router(config-if)#ip access-group 100 in
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Verify the ACL

Router# show access-lists and write down the response below:



```
Router#
Router#show ip access-lists
Extended IP access list 100
    10 permit ip host 10.0.0.2 host 192.168.0.1
    20 deny ip host 10.0.0.3 host 192.168.0.1

Router#
```

Task 3: Static NAT and PAT Configuration

Use the above network topology to configure the commands below.

First delete R1 and R2 and attach new R1 and R2 to avoid any issue encountered previously.

Router 1

```
Router(config)#ip nat inside source static 10.0.0.2 192.168.0.3
Router(config)#ip nat inside source static 10.0.0.3 192.168.0.3
Router(config)#interface fa0/0
Router(config)#ip nat inside
Router(config)#interface fa0/1
Router(config)#ip nat outside
```

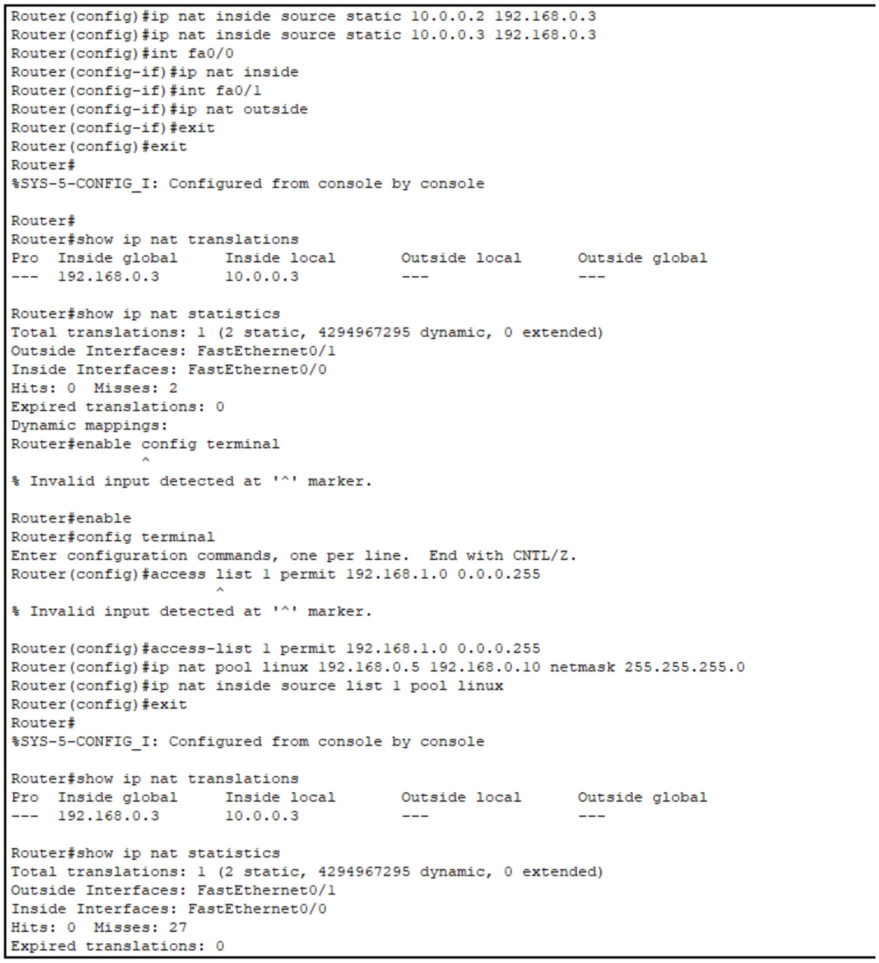
You can verify these by running these two command and attach its screenshot below.

```
Router#show ip nat translations
Router#show ip nat statistics
```

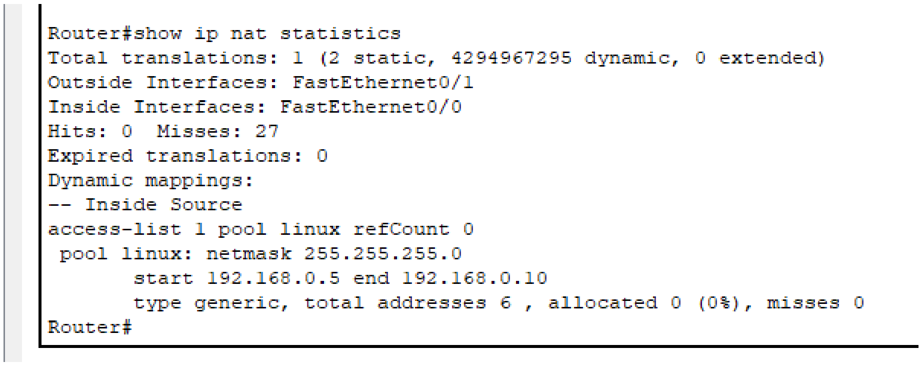
For Dynamic Nat

```
Router(config)# access list 1 permit 192.168.1.0 0.0.0.255
Router(config)#ip nat pool linux 192.168.0.5 192.168.0.10 netmask 255.255.255.0
Router(config)#ip nat inside source list 1 pool linux
```

Attach screenshot of verification of Dynamic NAT.



```
Router(config)#ip nat inside source static 10.0.0.2 192.168.0.3
Router(config)#ip nat inside source static 10.0.0.3 192.168.0.3
Router(config)#int fa0/0
Router(config-if)#ip nat inside
Router(config-if)#int fa0/1
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
Router#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
---  192.168.0.3       10.0.0.3           ---                ---

Router#show ip nat statistics
Total translations: 1 (2 static, 4294967295 dynamic, 0 extended)
Outside Interfaces: FastEthernet0/1
Inside Interfaces: FastEthernet0/0
Hits: 0  Misses: 2
Expired translations: 0
Dynamic mappings:
Router#enable config terminal
              ^
% Invalid input detected at '^' marker.

Router#enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#access list 1 permit 192.168.1.0 0.0.0.255
                      ^
% Invalid input detected at '^' marker.

Router(config)#access-list 1 permit 192.168.1.0 0.0.0.255
Router(config)#ip nat pool linux 192.168.0.5 192.168.0.10 netmask 255.255.255.0
Router(config)#ip nat inside source list 1 pool linux
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip nat translations
Pro  Inside global     Inside local       Outside local      Outside global
---  192.168.0.3       10.0.0.3           ---                ---

Router#show ip nat statistics
Total translations: 1 (2 static, 4294967295 dynamic, 0 extended)
Outside Interfaces: FastEthernet0/1
Inside Interfaces: FastEthernet0/0
Hits: 0  Misses: 27
Expired translations: 0
```



```
Router#show ip nat statistics
Total translations: 1 (2 static, 4294967295 dynamic, 0 extended)
Outside Interfaces: FastEthernet0/1
Inside Interfaces: FastEthernet0/0
Hits: 0  Misses: 27
Expired translations: 0
Dynamic mappings:
-- Inside Source
access-list 1 pool linux refCount 0
 pool linux: netmask 255.255.255.0
       start 192.168.0.5 end 192.168.0.10
       type generic, total addresses 6 , allocated 0 (0%), misses 0
Router#
```

Lab Evaluation Assessment Rubric

EE-424 Lab 09

#	Assessment Elements	Level 1: Unsatisfactory Points 0-1	Level 2: Developing Points 2	Level 3: Good Points 3	Level 4: Exemplary Points 4
LR2	**Program/Code/ Simulation Model/ Network Model**	Program/code/simulation model/network model does not implement the required functionality and has several errors. The student is not able to utilize even the basic tools of the software.	Program/code/simulation model/network model has some errors and does not produce completely accurate results. Student has limited command on the basic tools of the software.	Program/code/simulation model/network model gives correct output but not efficiently implemented or implemented by computationally complex routine.	Program/code/simulation /network model is efficiently implemented and gives correct output. Student has full command on the basic tools of the software.
LR4	**Data Collection**	Measurements are incomplete, inaccurate and imprecise. Observations are incomplete or not included. Symbols, units and significant figures are not included.	Measurements are somewhat inaccurate and imprecise. Observations are incomplete or vague. Major errors are there in using symbols, units and significant digits.	Measurements are mostly accurate. Observations are generally complete. Minor errors are present in using symbols, units and significant digits.	Measurements are both accurate and precise. Data collection is systematic. Observations are very thorough and include appropriate symbols, units and significant digits and task completed in due time.

LR5	**Results & Plots**	Figures/ graphs / tables are not developed or are poorly constructed with erroneous results. Titles, captions, units are not mentioned. Data is presented in an obscure manner.	Figures, graphs and tables are drawn but contain errors. Titles, captions, units are not accurate. Data presentation is not too clear.	All figures, graphs, tables are correctly drawn but contain minor errors or some of the details are missing.	Figures / graphs / tables are correctly drawn and appropriate titles/captions and proper units are mentioned. Data presentation is systematic.
AR4	***Report Submission**	Late submission after 1 week and in between 2 weeks.	Late submission after 2 days and within a week.	Late submission after the lab timing and within 2 days of the due date.	Timely submission of the report and in the lab time.

***Report:** Report will not be accepted after due date