VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Computer Networks

Submitted by

SARAAG (1BM21CS190)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
June-2023 to September-2023

B. M. S. College of Engineering,

Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled "LAB COURSE **Computer Networks**" carried out by **SARAAG (1BM21CS190),** who is a bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2023. The Lab report has been approved as it satisfies the academic requirements in respect of **Computer Networks - (22CS4PCCON)** work prescribed for the said degree.

Dr. Shyamala G
Assistant Professor
Department of CSE
BMSCE,Bengaluru

Dr. Jyothi S Nayak
Professor and Head
Department Of CSE
BMSCE,Bengaluru

Index

Sl. No.	Date	Experiment Title	Pg. No.
		CYCLE 1	
1.	16/06	Create a topology with 3 or more end devices using i)Hub ii)Switch iii)Hub & Switch Hybrid	1
2.	23/06	Configure IP address to routers & explore the following messages: ping responses, destination unreachable, request timed out, reply.	11
3.	30/06	Configure default route, static route to the Router	26
4.	14/07	Configure DHCP within a LAN and outside LAN.	36
5.	21/07	Configure Web Server, DNS within a LAN.	57
6.	21/07	Configure RIP routing Protocol in Routers	62
7.	04/08	Configure OSPF routing protocol	69
8.	11/08	To construct simple LAN & understand the concept and operation of ARP	79
9.	11/08	To construct a VLAN & make the PC's communicate among a VLAN	85
10.	18/08	To construct a WLAN & make the nodes communicate wirelessly	94
11.	18/08	To understand the operation of TELNET by accessing the router in the server room from a PC in the IT office.	104
12.	18/08	Demonstrate the TTL/ Life of a Packet	110

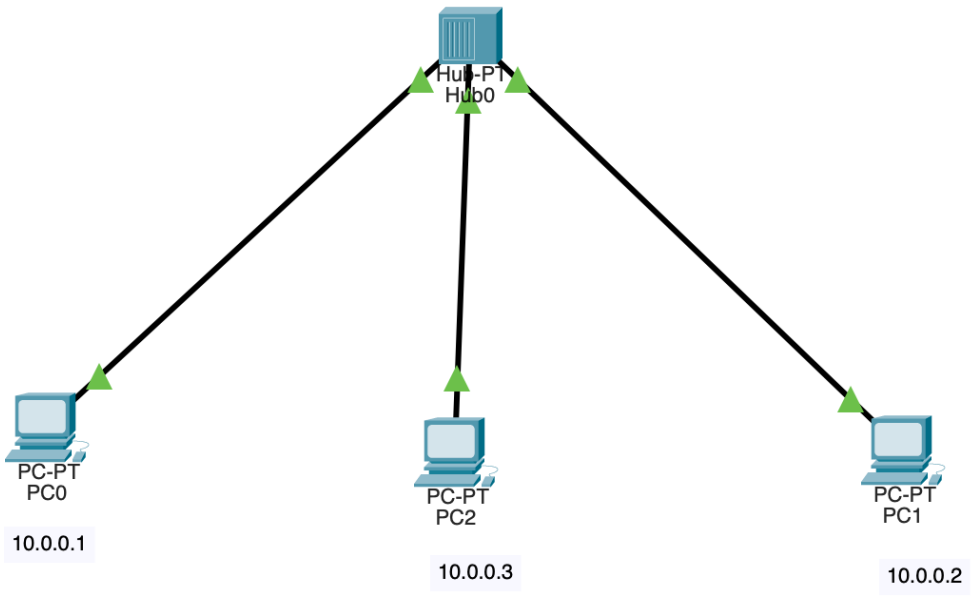
		CYCLE 2	
1.	01/09	Write a program for error detecting code using CRC-CCITT (16-bits).	117
2.	01/09	Write a program for congestion control using Leaky bucket algorithm.	124
3.	01/09	Socket programming using TCP/IP sockets	127
4.	01/09	Socket programming using UDP sockets	131

CYCLE 1

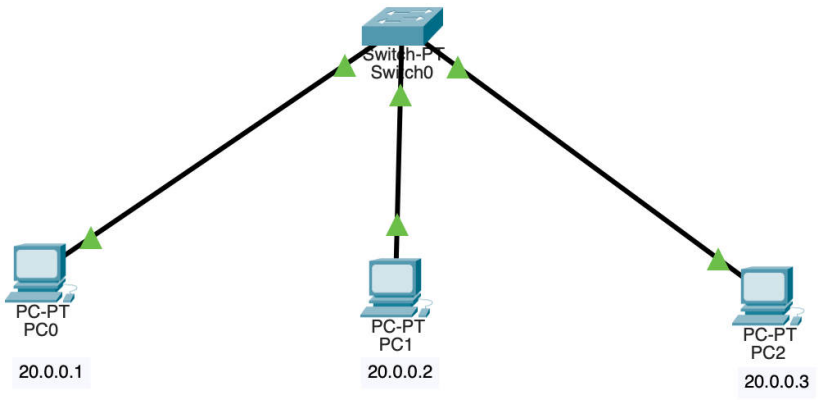
Experiment No. 1

Title:
Create a topology with 3 or more end devices using -
i)Hub ii)Switch iii)Hub & Switch Hybrid

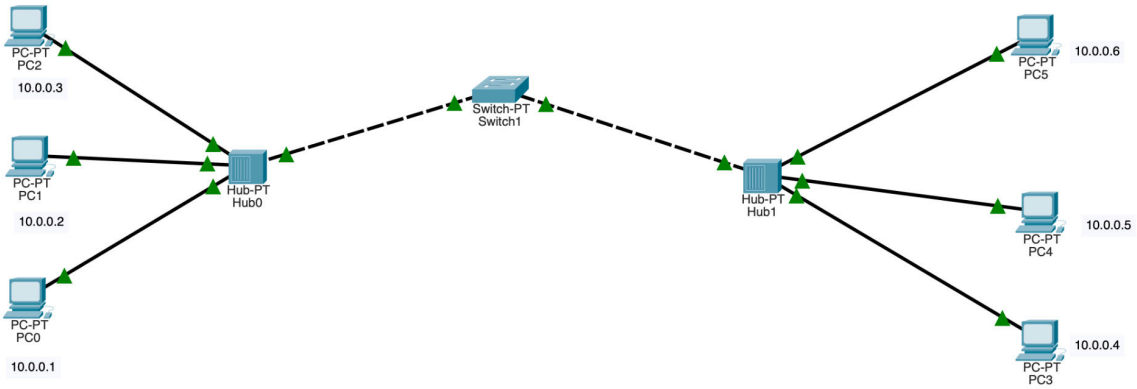
Topology:



i)Hub topology



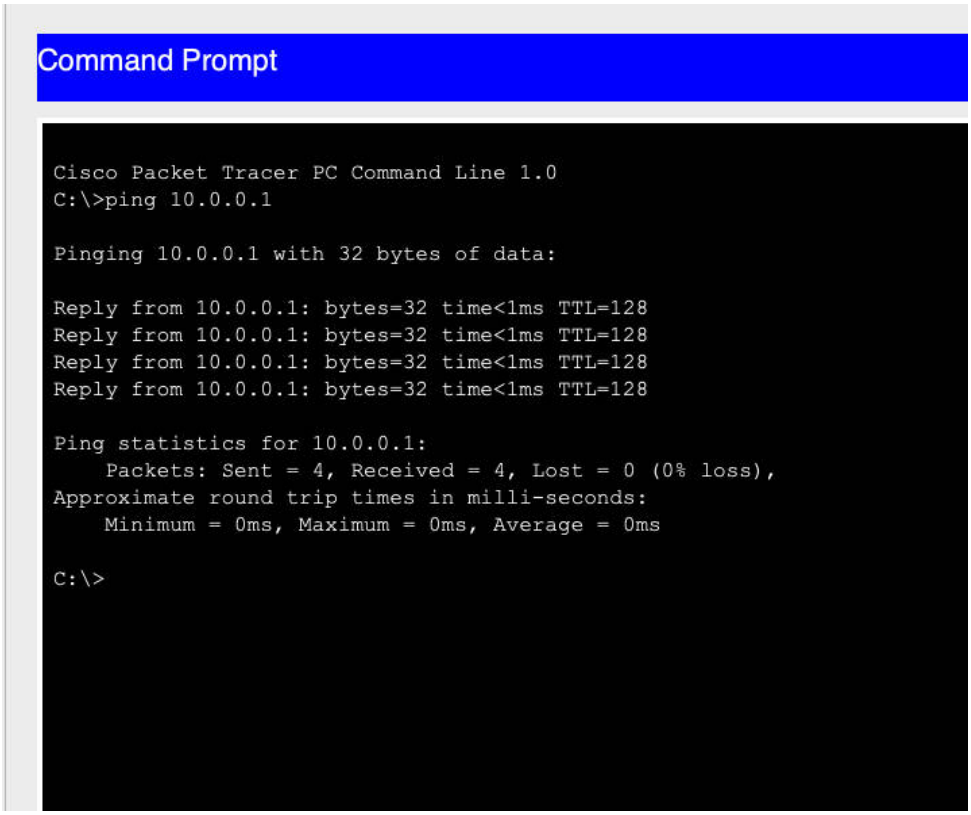
ii)Switch topology

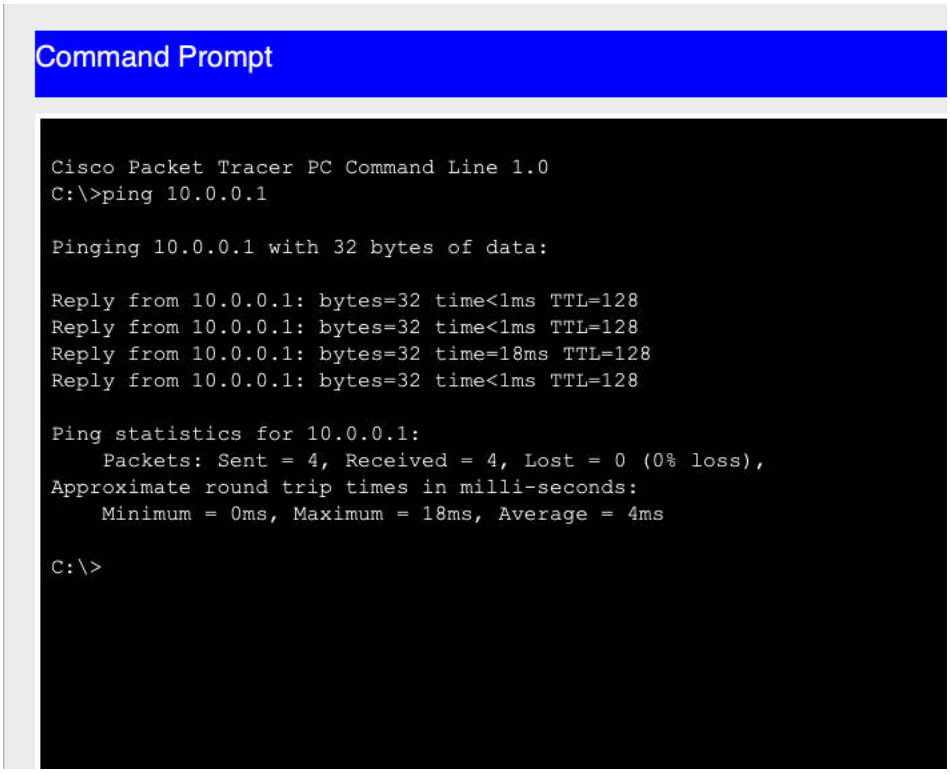


iii)Hub&Switch Hybrid topology

Pinging end devices:

i)





ii)

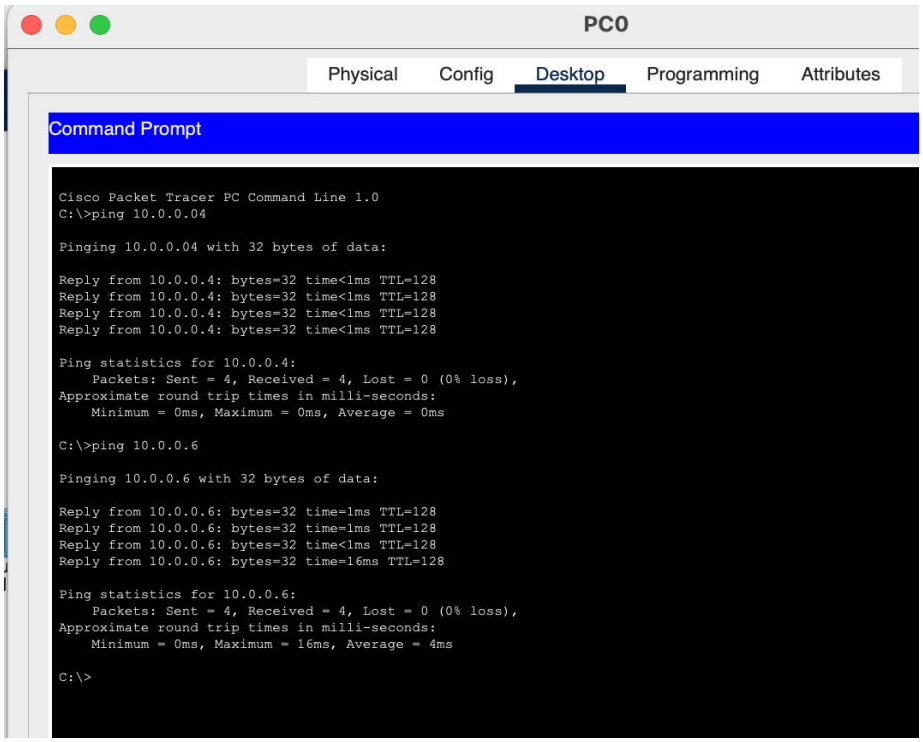
Command Prompt

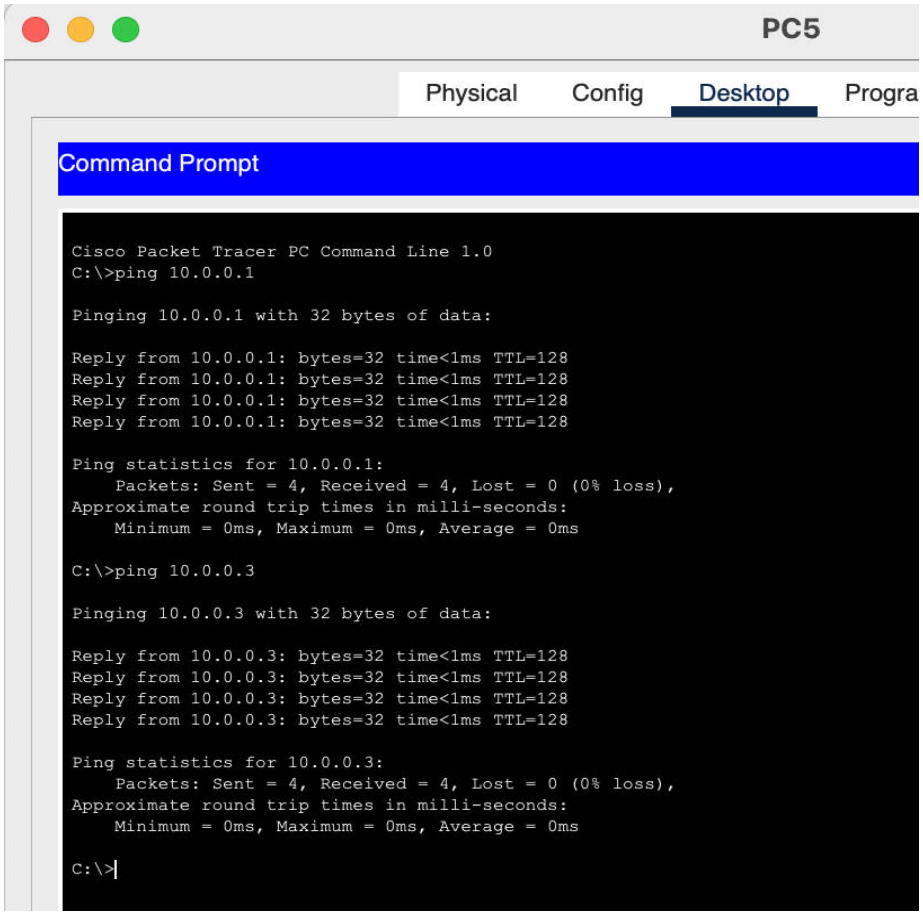
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 20.0.0.3

Pinging 20.0.0.3 with 32 bytes of data:

Reply from 20.0.0.3: bytes=32 time<1ms TTL=128
Reply from 20.0.0.3: bytes=32 time<1ms TTL=128
Reply from 20.0.0.3: bytes=32 time<1ms TTL=128
Reply from 20.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 20.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

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=128
Reply from 20.0.0.2: bytes=32 time=14ms TTL=128
Reply from 20.0.0.2: bytes=32 time<1ms TTL=128
Reply from 20.0.0.2: bytes=32 time<1ms TTL=128

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 = 14ms, Average = 3ms
```

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 20.0.0.1

Pinging 20.0.0.1 with 32 bytes of data:

Reply from 20.0.0.1: bytes=32 time<1ms TTL=128
Reply from 20.0.0.1: bytes=32 time<1ms TTL=128
Reply from 20.0.0.1: bytes=32 time<1ms TTL=128
Reply from 20.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 20.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 20.0.0.3

Pinging 20.0.0.3 with 32 bytes of data:

Reply from 20.0.0.3: bytes=32 time<1ms TTL=128
Reply from 20.0.0.3: bytes=32 time<1ms TTL=128
Reply from 20.0.0.3: bytes=32 time<1ms TTL=128
Reply from 20.0.0.3: bytes=32 time<1ms TTL=128

Ping statistics for 20.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

iii)


PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.04
Pinging 10.0.0.04 with 32 bytes of data:
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Ping statistics for 10.0.0.4:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 10.0.0.6
Pinging 10.0.0.6 with 32 bytes of data:
Reply from 10.0.0.6: bytes=32 time=1ms TTL=128
Reply from 10.0.0.6: bytes=32 time=1ms TTL=128
Reply from 10.0.0.6: bytes=32 time<1ms TTL=128
Reply from 10.0.0.6: bytes=32 time=16ms TTL=128
Ping statistics for 10.0.0.6:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 16ms, Average = 4ms
C:\>


PC5
Physical
Config
Desktop
Progra
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>

Friday

Cycle - 1

16th June, 2023.

Title:-

(1) Create a topology consisting of 3 (or) more ~~Connecting~~ devices with help of a -

(i) Hub (ii) Switch (iii) Hub & Switch.

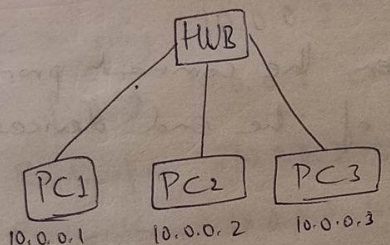
~~Demonstrate~~ Simulate sending a simple PDU from source to destination & demonstrate a ping message.

Aim:-

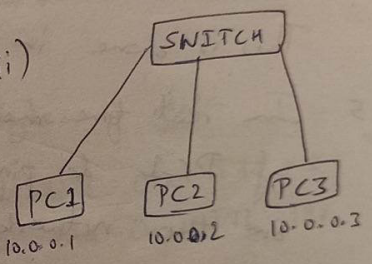
To understand different ping messages when they are caused in the given topology.

Topology:-

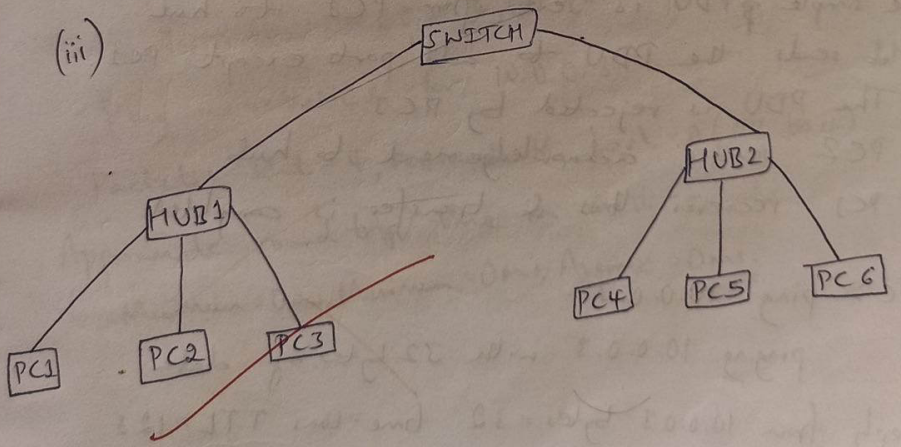
(i)



(ii)



(iii)



Procedure:-

1. Place the required no. of devices (PCs)
2. Add a generic HUB for (i), generic switch (ii) & both for (iii) and connect them with the devices.
3. In exp. (iii) 2 Hubs are used, which are connected to the switch.
4. Send a simple PDU message from source PC to destination PC & observe the simulation. This is done in simulation mode
5. In real time mode, open the command prompt of PC1 & ping one of the end devices on the network.

Result:-

(i) 1. The simple PDU is sent from PC1 to hub.
2. Hub sends the PDU to all ports except PC1.
3. The PDU is rejected by PC3.
4. PC2 sends acknowledgement to hub
5. PC1 recieves this & transfer is completed.

1. PC> ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data

Reply from 10.0.0.3 bytes=32 time=0ms TTL=128
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128

Ping statistics for 10.0.0.3

Packets: sent = 4, recieved = 4, lost = 0 (0% loss)

Approximate round trip time in milli-seconds

Minimum = 0ms, Maximum = 0ms, Average = 0ms

(ii) ~~PDU~~ PDU sent from PC1 to PC3

1. The PDU is sent from PC1 to switch.
2. Switch broadcasts the PDU to all outport except the input port.
3. PC3 acknowledges & sends, the switch transfers this to PC1 without broadcast

PC > ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data

Reply from 10.0.0.3 bytes = 32 time = 0ms TTL = 128

Reply from 10.0.0.3 bytes = 32 time = 0ms TTL = 128

Reply from 10.0.0.3 bytes = 32 time = 0ms TTL = 128

Reply from 10.0.0.3 bytes = 32 time = 0ms TTL = 128

Ping statistics for 10.0.0.3

Packets: sent = 4, recieved = 4, lost = 0 (0% loss)

Approximate round trip time in milli-seconds

Minimum = 0ms, Maximum = 0ms, Average = 0ms.

(iii) PDU sent from PC1 to PC6

1. PDU is sent from PC1 to HUB1
2. Hub sends copy of PDU to PC2 & PC3 & switch
3. The switch forwards message to HUB2, which intern sends it to PC4, PC5, PC6.
4. PC6 recievs message & acknowlegdes
5. The switch forwards the acknowledgement to HUB1.
6. The HUB1 broadcasts it to PC1, PC2, PC3.
7. PC1 recieves the acknowlegdgement & transfer is completed

```
PC > ping 10.0.0.6

   pinging 10.0.0.6 with 32 bytes of data

Reply from 10.0.0.6 bytes=32 time=0ms TTL=128
Reply from 10.0.0.6 bytes=32 time=0ms TTL=128
Reply from 10.0.0.6 bytes=32 time=0ms TTL=128
Reply from 10.0.0.6 bytes=32 time=0ms TTL=128

~~Ping statistics~~
Ping statistics for 10.0.0.6
   Packets sent=4, Recieved=4, Lost=0 (0% loss)
Approximate round trip time in milli-seconds
Minimum=0ms Maximum=0ms Average=0ms
```

Observation:-

(i) Hub doesn't store any data & all incoming packets are broadcasted to all ports except the input port. The reciever acknowledges the packet when recieved.

(ii) Switch takes time for new connection to be established. In the first transfer switch broadcasts the packet to all end devices. In following transfer, the switch transfers the packet to destination device based on mac address.

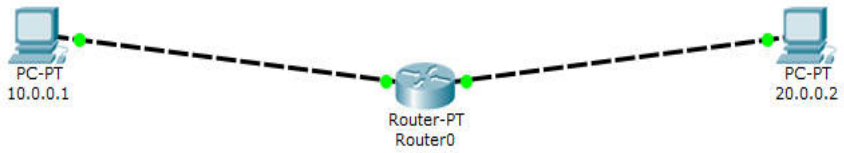
(iii) Switch & hub connected end devices can communicate as long as they are in the same network.

30/6/23

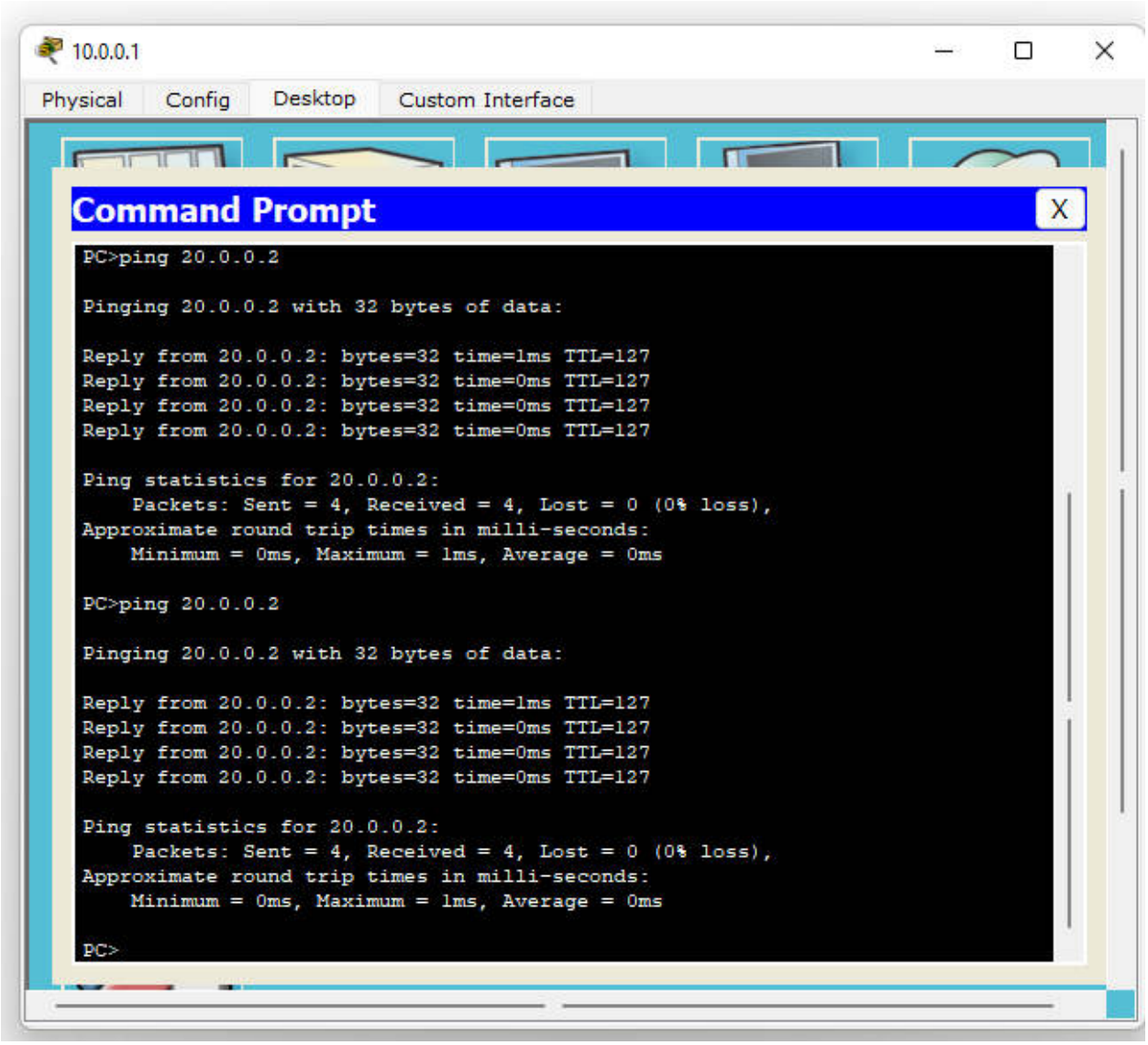
Experiment No. 2

Title:
Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply

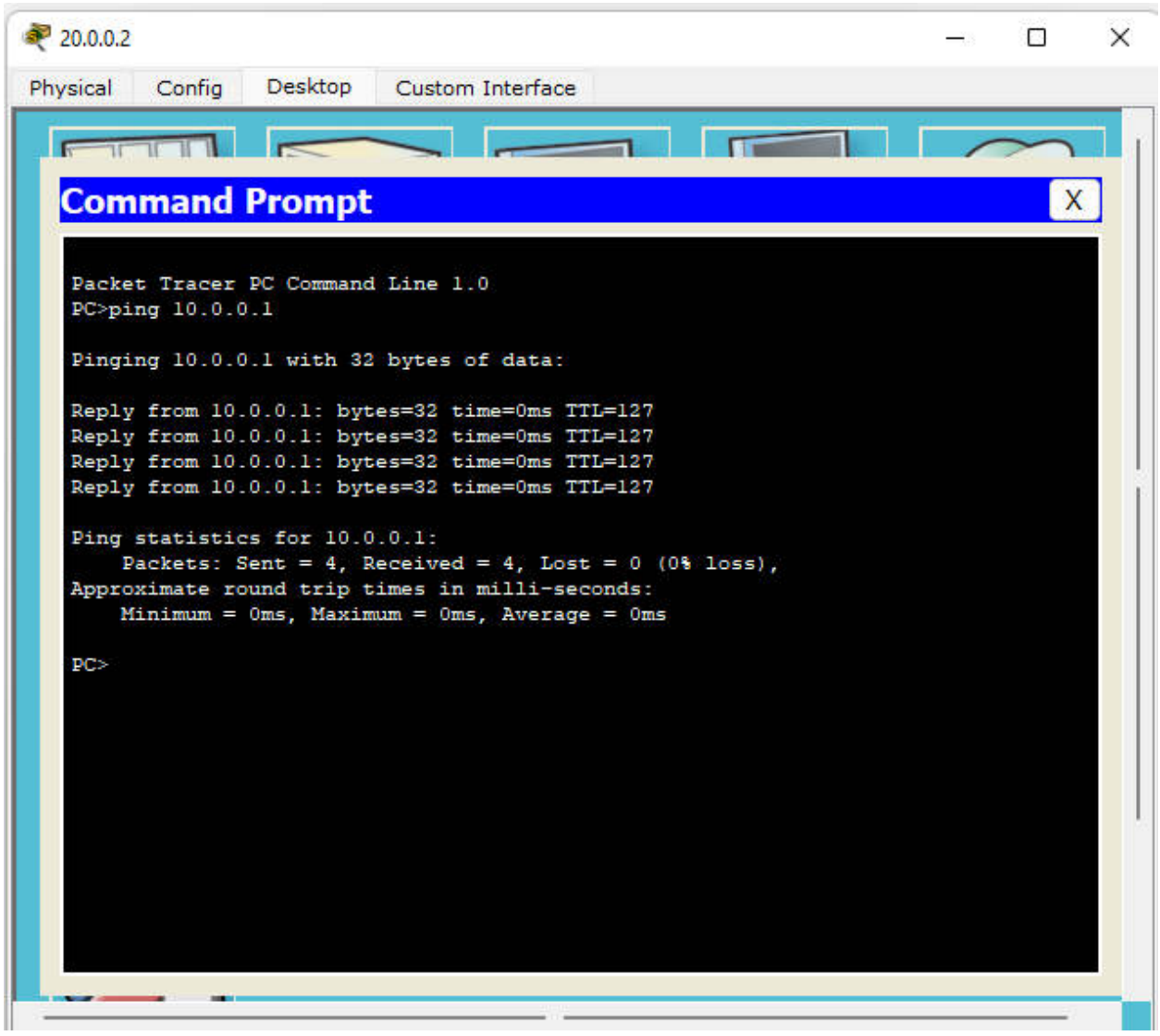
Topology: with single router



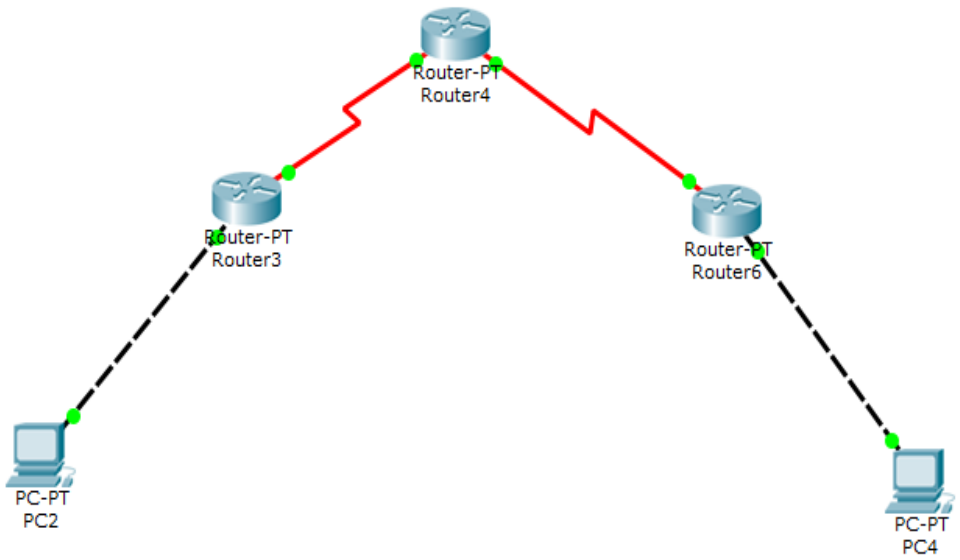
Pinging PC2 from PC1:



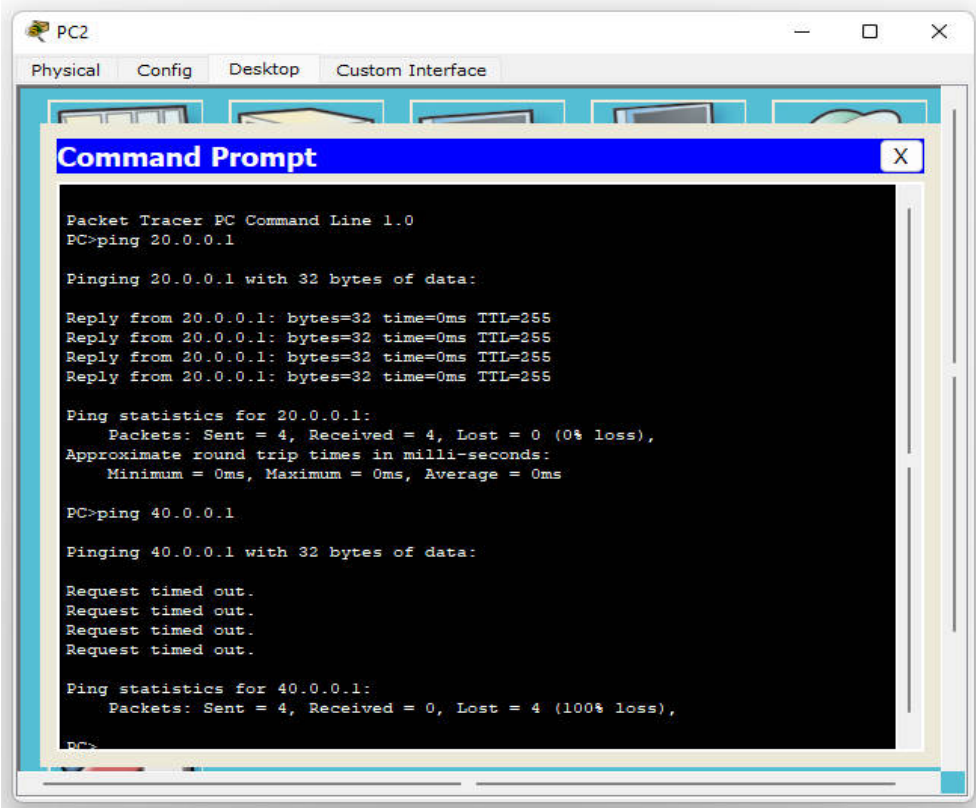
Pinging PC1 from PC2:



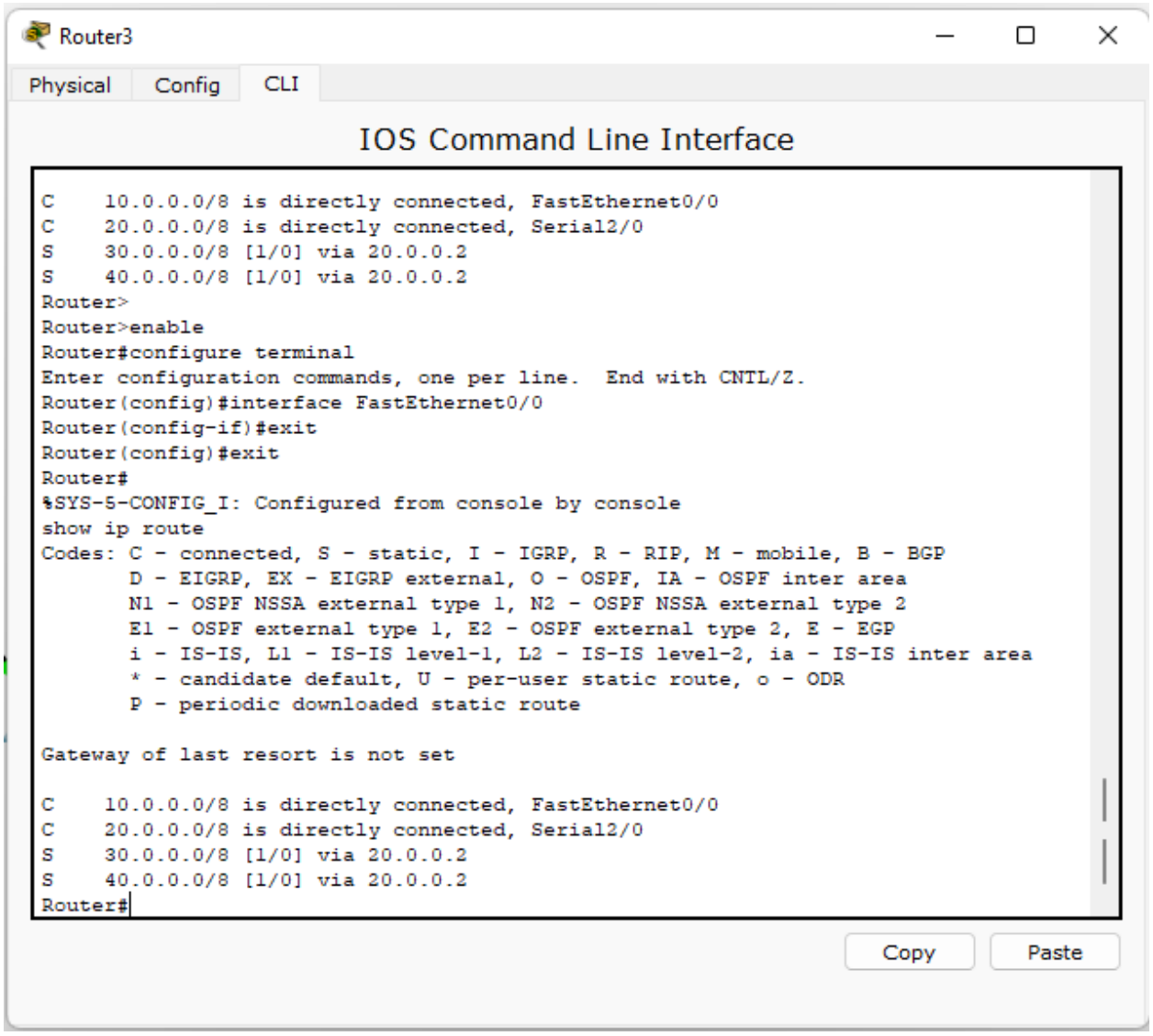
Topology: with 3 Routers

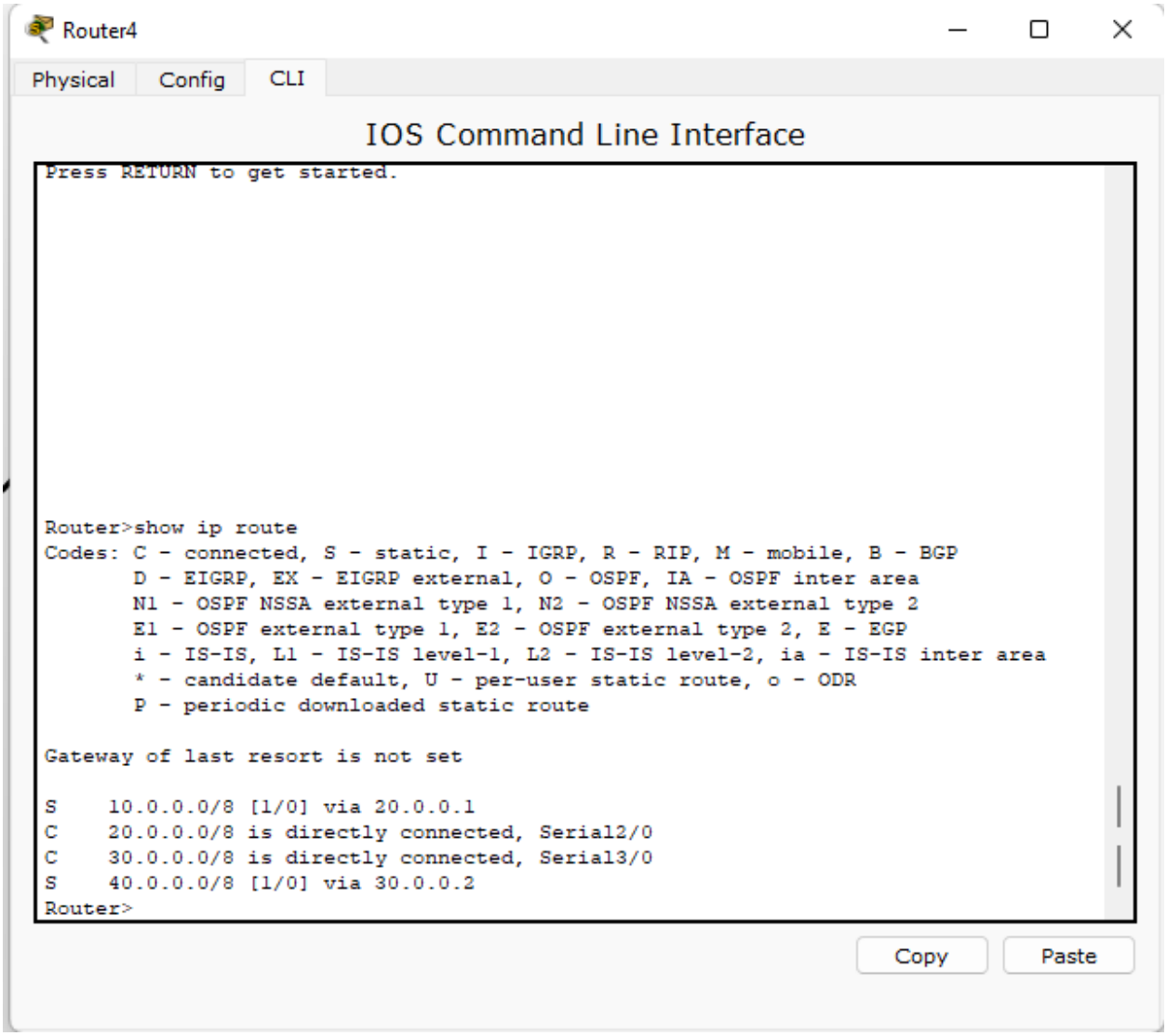


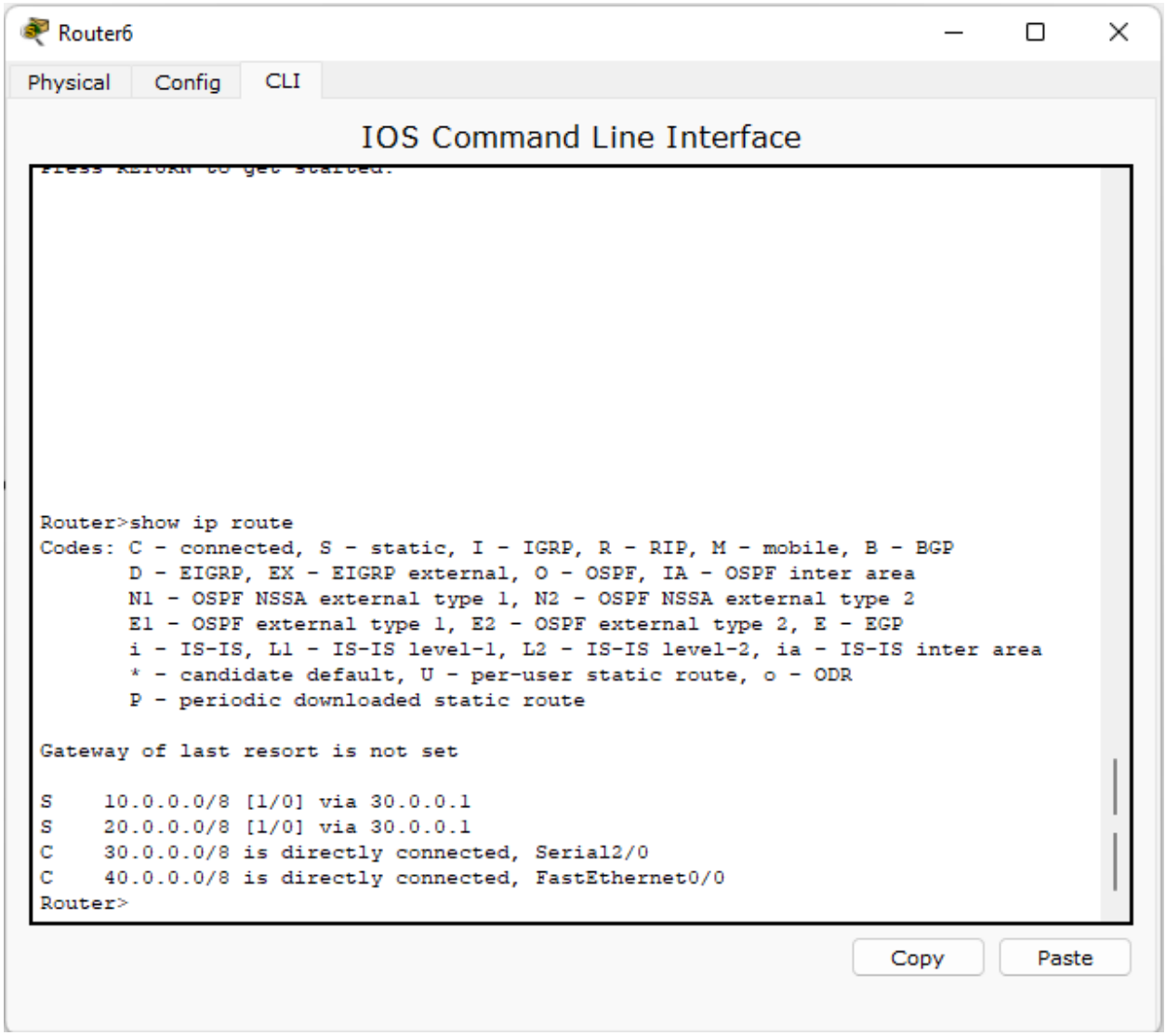
Pinging end device - in different network before setting IP route



IP route - for all routers

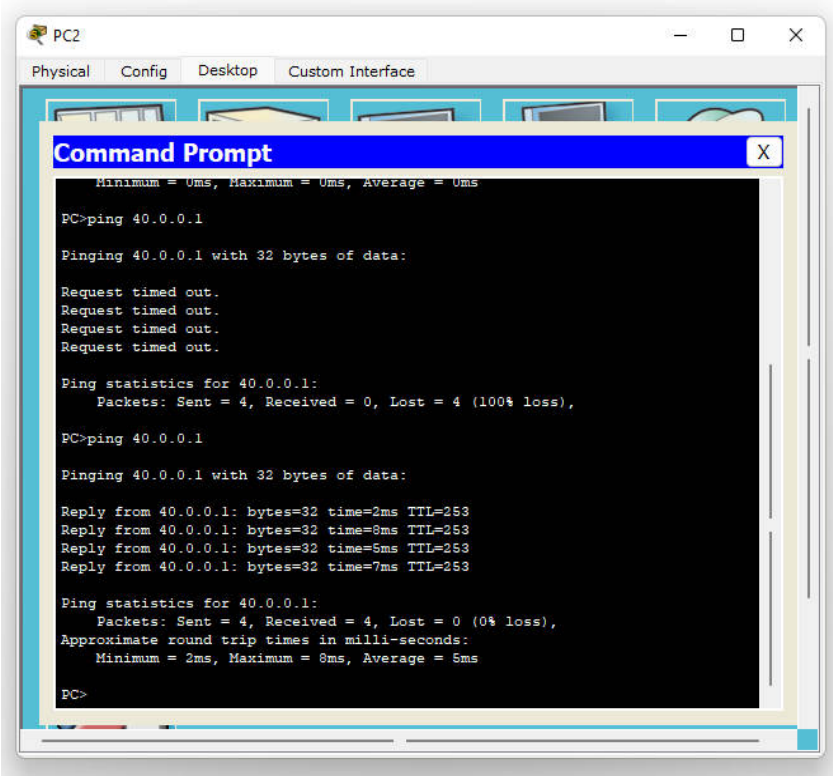



Router4
Physical
Config
CLI
IOS Command Line Interface
Press RETURN to get started.
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
S 10.0.0.0/8 [1/0] via 20.0.0.1
C 20.0.0.0/8 is directly connected, Serial2/0
C 30.0.0.0/8 is directly connected, Serial3/0
S 40.0.0.0/8 [1/0] via 30.0.0.2
Router>
Copy
Paste

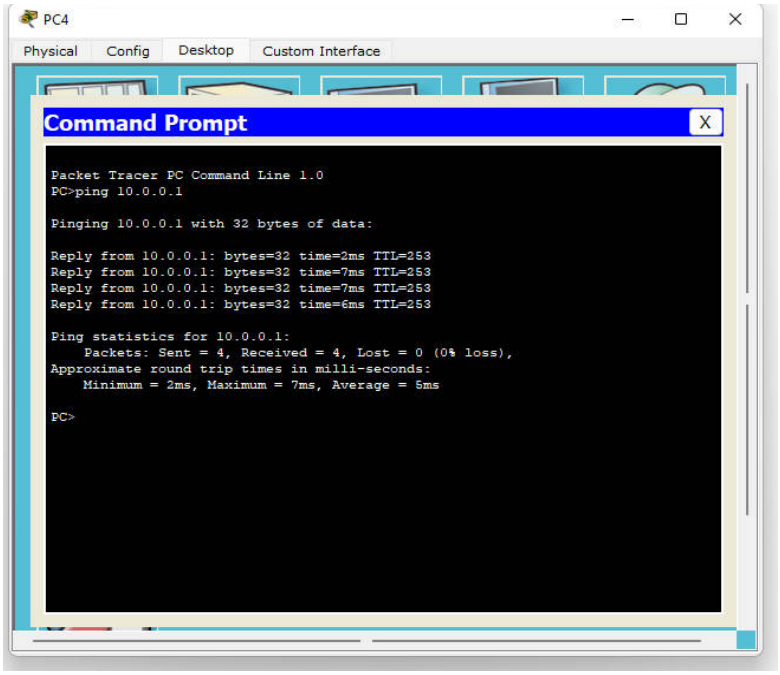

Router6
Physical
Config
CLI
IOS Command Line Interface
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
S 10.0.0.0/8 [1/0] via 30.0.0.1
S 20.0.0.0/8 [1/0] via 30.0.0.1
C 30.0.0.0/8 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
Router>
Copy
Paste

After setting IP route

Pinging PC4 from PC2



Pinging PC2 from PC4



Friday 23rd June, 2023.

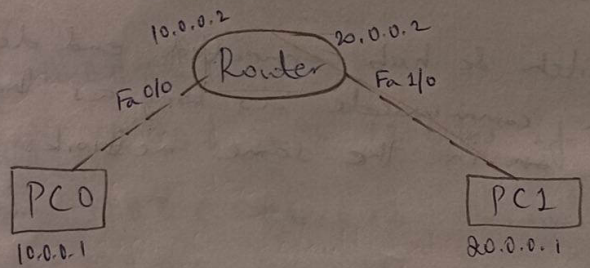
② Title:-

Configure IP address to routers in packet tracer. ~~Exploere~~ Explore the following messages: ping responses, destination unreachable, request timed out, reply.

Aim:-

To understand different ping messages & when they are caused.

Topology:-



Procedure:-

1. Use a generic router & connect two devices to it.
2. Set the IP address of the 2 PC's as 10.0.0.1 & 20.0.0.1
3. In the router go to command line interface & enter no ~~for~~ to continue with configuration dialogue.
4. Using the CLI put IP address same as respective gateway address for ~~end~~ device
5. Select PC0 & open command prompt. Ping the PC1 using its IP address.

CLI for router -

→ Enable
→ Configure terminal
→ Interface Fa 0/0
→ IP address 10.0.0.2 255.0.0.0
→ No shutdown
→ Exit

Result:-

The PC0 & PC1 of different networks are connected using router with their respective gateways.

Observation:

Before setting the default gateway

ping 20.0.0.1
Request timed out
Request timed out
Request timed out
Request timed out

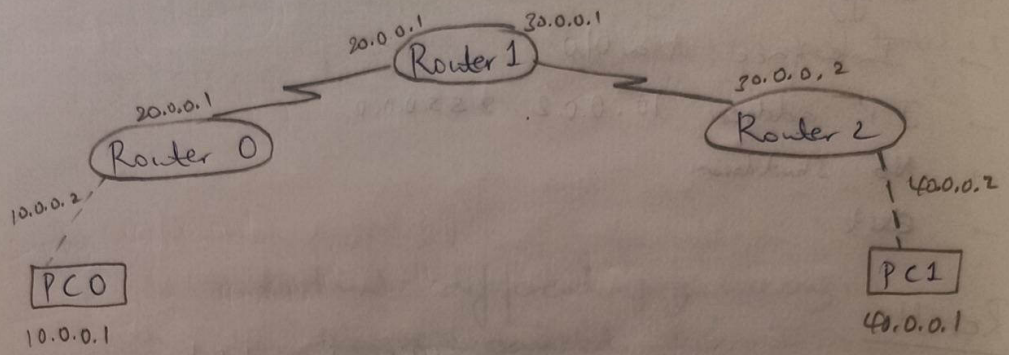
Ping statistics for 10.0.0.2
Packets : sent = 4, recieved = 0, lost = 4 (100% loss)

After setting default gateway

ping 20.0.0.1
Reply from 20.0.0.1 time = 0ms TTL = 128
Reply from 20.0.0.1 time = 0ms TTL = 128
Reply from 20.0.0.1 time = 0ms TTL = 128
Reply from 20.0.0.1 time = 0ms TTL = 128

Ping statistics for 10.0.0.2
Packets; Sent: 4, Recieved = 4, Loss = 0 (0% loss)

Topology:-



Procedure:-

1. Take 3 generic routers & 2 end devices
2. Connect end devices to routers & used suitable wire to the routers
3. Set IP address & default gateway to each end device of different networks.
4. Configure the router using CLI. Put the IP address same as respective gateway address for end device.
5. Select PC0 & open the command prompt Ping PC1 with its IP address.

Result:-

The intermediate router is connected with other two routers properly with green signal between the 3 routers & end device.

Observation:-

Pinging PC1 from PC0 works

* Three routers & 2 PCs are connected as shown in topology.
* Pinging PC1 from PC0 shows destination host unreachable.
* Adding static routers to routers using IP router <dest< subnet mask> <next hop> in enable. config terminal mode

PC0 to Router 1.

1> enable
config terminal
(config)# interface Fa 0/0
(config)# IP address 10.0.0.2 255.0.0.0
(config)# No shutdown

Initial Ping:

> ping 40.0.0.1

Pinging 40.0.0.1 with 32 byte 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

Packets: Sent: 4, Recieved = 0, lost = 4 (100% Loss)

→ Adding Static Router:

Router 0:

Router(config) # IP router 30.0.0.0
255.0.0.0 20.0.0.2

Router(config) # IP route ~~40.0.0.0~~ 40.0.0.2 255.0.0.0
20.0.0.2

>ping 40.0.0.1

pinging 40.0.0.1 with 32 bytes of data

Reply from 40.0.0.1 bytes=32 times=12ms

Reply from 40.0.0.1 bytes=32 times=9ms

Reply from 40.0.0.1 bytes=32 times=5ms

Reply from 40.0.0.1 bytes=32 times=3ms

Packets Sent=4 Recieved=4 lost=0 (100% loss)

Approx round trip time

Minimum=2ms Maximum=12ms Avg=7ms

30/6/23

Friday

30th June, 2023.

→ Setting IP Route: In Router 0

Router# show IP route
C 10.0.0.0/8 is directly connected, Fa0/0
C 20.0.0.0/8 is directly connected, Se2/0

Router # configure terminal
Router (config)# ip route 30.0.0.0 255.0.0.0 20.0.0.2
Router (config)# ip route 40.0.0.0 255.0.0.0 20.0.0.2

Router (config)# exit

Router# show ip route
C 10.0.0.0/8 is directly connected, Fa0/0
C 20.0.0.0/8 is directly connected, Se2/0
S 30.0.0.0/8 [1/0] via 20.0.0.2
S 40.0.0.0/8 [1/0] via 20.0.0.2

→ Ping PC0 from PC1

PC > ping 10.0.0.1

Reply from 10.0.0.1: bytes =32 time =2ms TTL=253
Reply from 10.0.0.1: bytes =32 time =7ms TTL=253
Reply from 10.0.0.1: bytes =32 time =7ms TTL=253
Reply from 10.0.0.1: bytes=32 time=6ms TTL=253

Ping statistics for 10.0.0.1
Packets: Sent = 4, Received = 4, Lost = 0 (0% Loss)

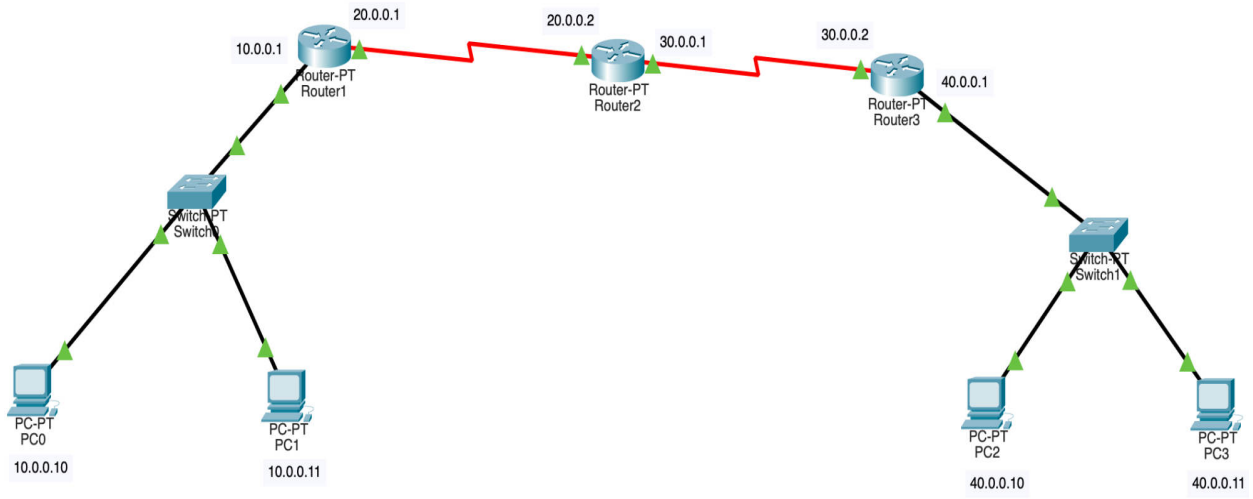
Approx round trip
Min =2ms, Max =7ms, Average =5ms

Experiment No. 3

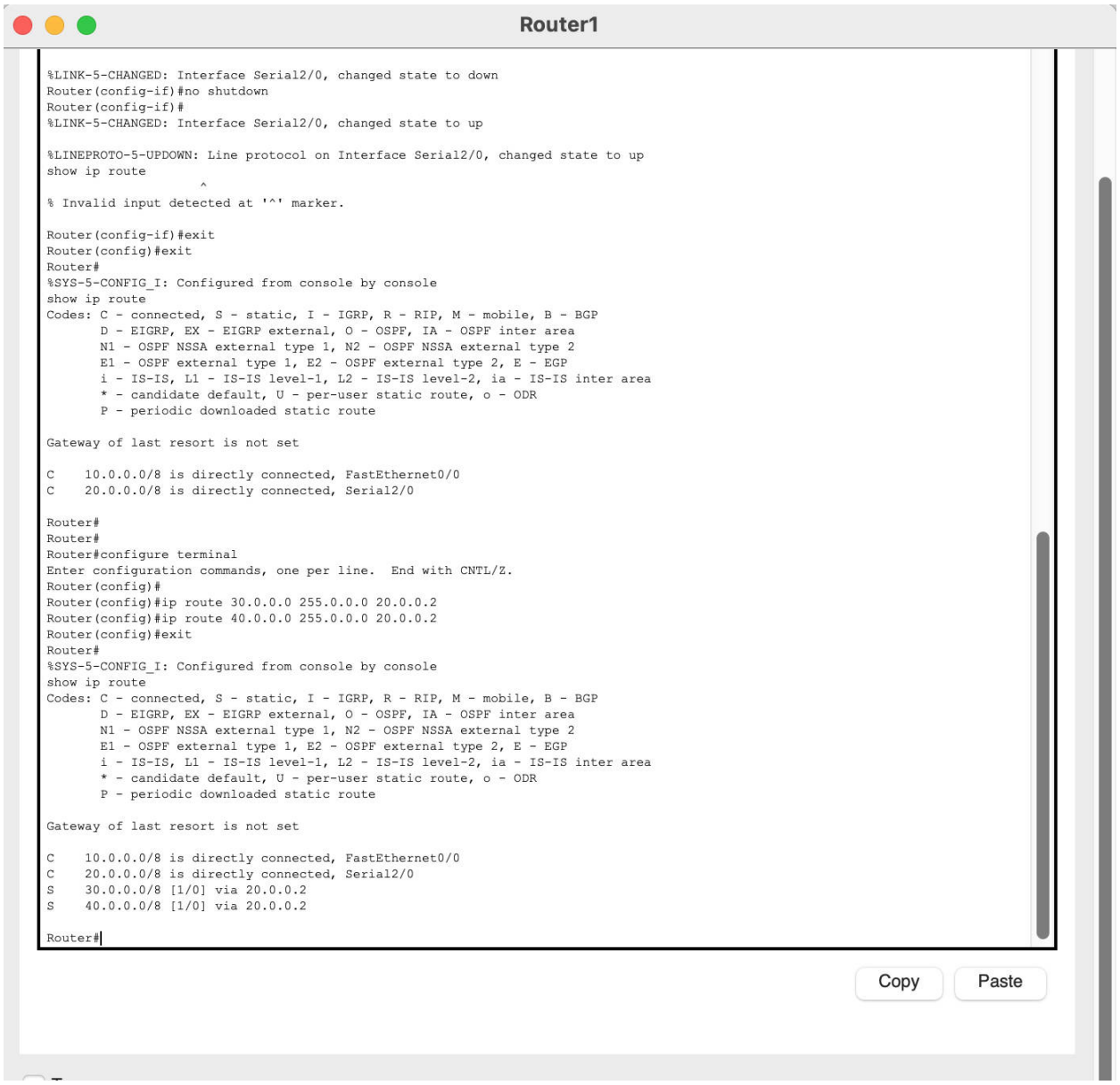
Title:

Configure default route, static route to the Router

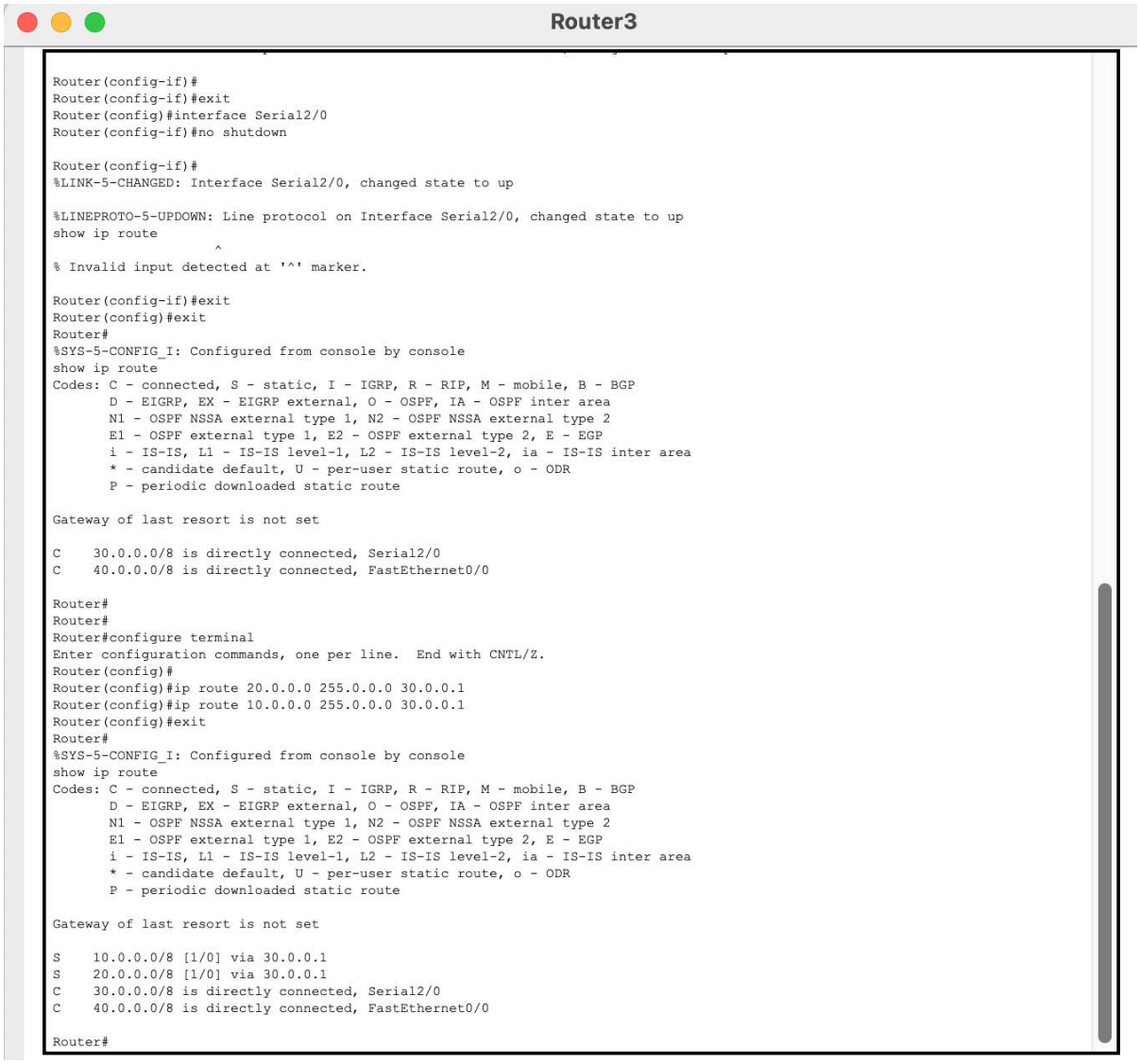
Topology:



IP Route for all routers:

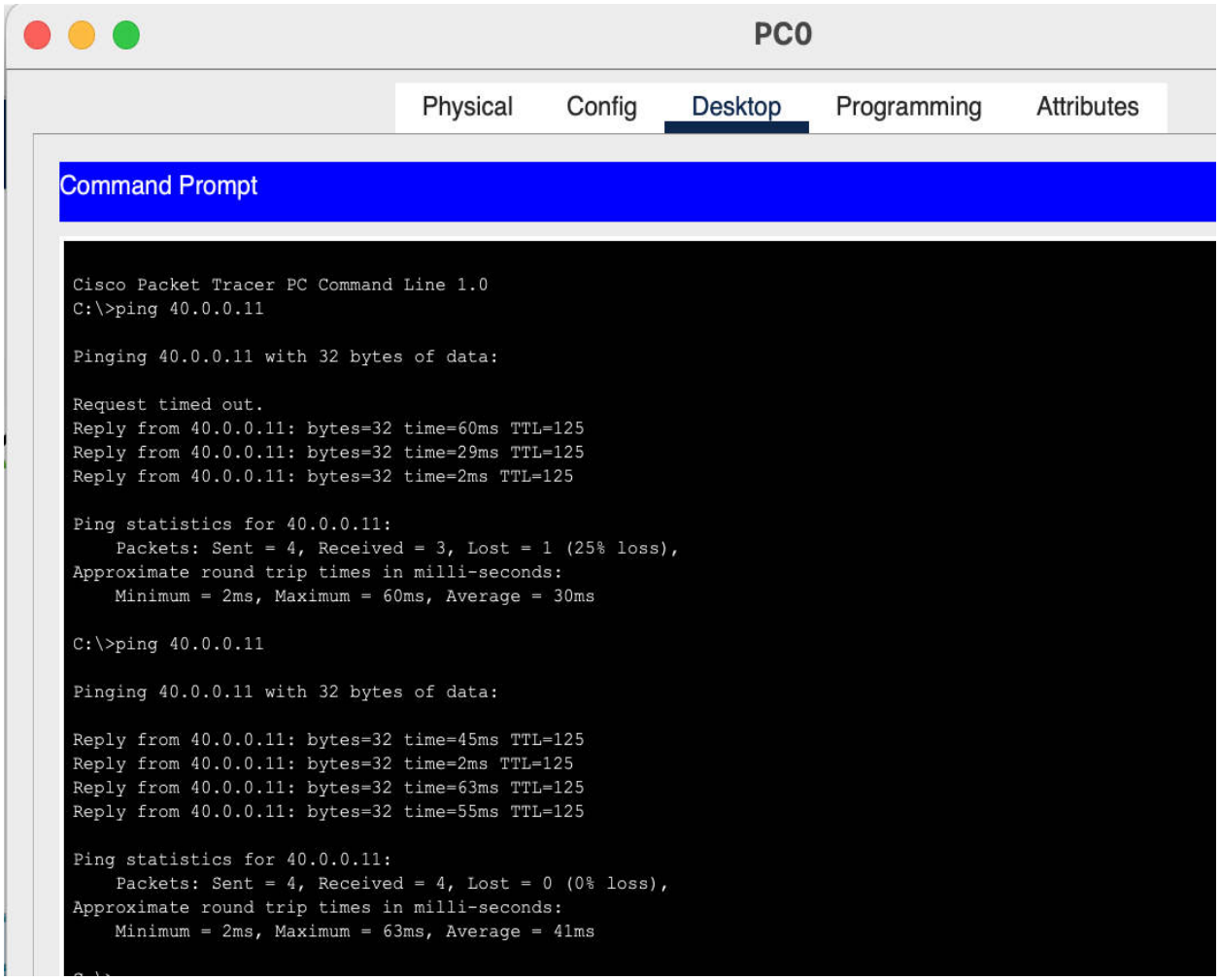



Router2
Physical
Config
CLI
Attributes
IOS Command Line Interface
Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial3/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
show ip route
^
% Invalid input detected at '^' marker.
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
C 20.0.0.0/8 is directly connected, Serial3/0
C 30.0.0.0/8 is directly connected, Serial2/0
Router#
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.1
Router(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.2
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
S 10.0.0.0/8 [1/0] via 20.0.0.1
C 20.0.0.0/8 is directly connected, Serial3/0
C 30.0.0.0/8 is directly connected, Serial2/0
S 40.0.0.0/8 [1/0] via 30.0.0.2
Router#

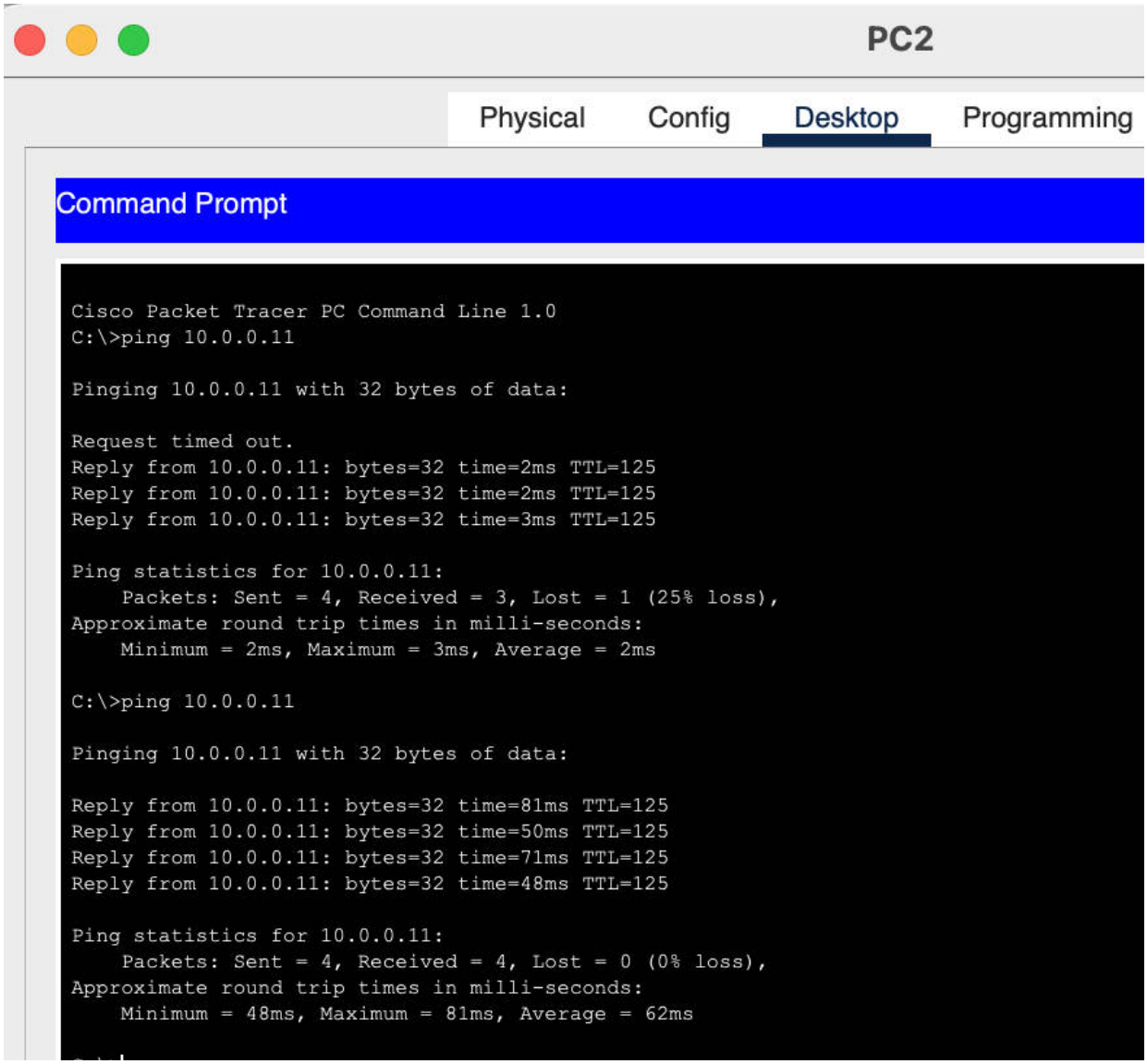

Router3
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
show ip route
^
% Invalid input detected at '^' marker.
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
C 30.0.0.0/8 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
Router#
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#ip route 20.0.0.0 255.0.0.0 30.0.0.1
Router(config)#ip route 10.0.0.0 255.0.0.0 30.0.0.1
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
S 10.0.0.0/8 [1/0] via 30.0.0.1
S 20.0.0.0/8 [1/0] via 30.0.0.1
C 30.0.0.0/8 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
Router#

Pinging end devices in different network:

Ping PC3 from PC0



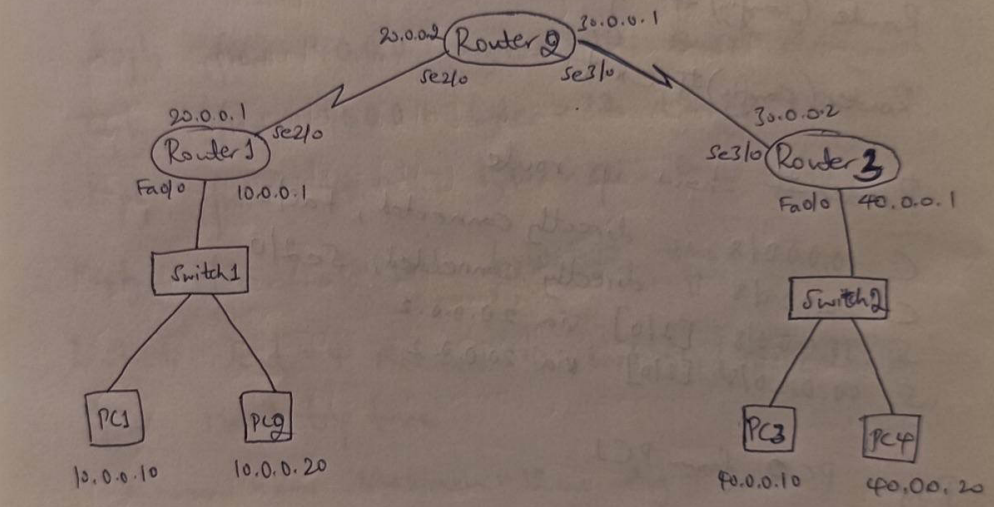
Ping PC1 from PC2



③ Title:-
Configure default route, static route to the Router.

Aim:-
To understand how to configure static ip route & default route using CLI

Topology:-



Procedure:-

1. Connect to end devices to a switch which intern is connected to a router.
2. These 2 routers are ~~connected~~ connected to Router 2.
3. The end devices PC1 & PC2 are set to IP address 10.0.0.10 & 10.0.0.20 statically. Similarly PC3 & PC4 are set to IP 40.0.0.10 & 40.0.0.20.
4. The end devices & switches are connected with copper-straight through wire. The routers are connected using serial ports.

5. The ip is set for Router 1 interfaces using CLI.

Router(config)# Interface Fa0/0
Router(config-if)# ip address 10.0.0.1 2550.0.0
Router(config-if)# no shutdown
Router(config-if)# exit

6. Step 5 is repeated for serial interface Se 2/0 of Router 1 with IP set as 20.0.0.1
Router 2's serial interface Se 2/0 with 20.0.0.2
Se 3/0 with ip address of 30.0.0.1
Router 3's serial interface Se 3/0 with 30.0.0.2
Router 3's Fast ethernet port Fa0/0 with 40.0.0.1

7. Display ip router in router 1 using,
Router > show ip route

8. Default route is set for Router 1
Router > enable
Router # configure terminal
Router(config)# ip route 0.0.0.0 0.0.0.0 20.0.0.2
Router(config)# exit

9. Step 8 is repeated for Router 2 with
Router(config)# ip route 0.0.0.0 0.0.0.0 30.0.0.1

10. Display ip route of router 1 after setting default ip route

11. Ip route is set for Router 2 statically
Router(config)# ip route 40.0.0.0 2550.0.0 30.0.0.2
Router(config)# ip route 10.0.0.0 255.0.0.0 20.0.0.1

Result:-

IP route before setting default route, for Router 1

Gateway of last resort not set

C 10.0.0.0/8 is directly connected, Fa0/0

C 20.0.0.0/8 is directly connected, Se2/0

After setting default route & static route

C 10.0.0.0/8 is directly connected, Fa0/0

C 20.0.0.0/8 is directly connected, Se2/0

S* 0.0.0.0/0 [1/0] via 20.0.0.2

S 40.0.0.0/8 [1/0] via 20.0.0.2

IP route before setting ~~default~~ ip route, for Router 2

C 20.0.0.0/8 is directly connected, ~~Se2~~/0

C 30.0.0.0/8 is directly connected, Se3/0

After setting ~~default & static route~~ ip route,

C 20.0.0.0/8 is directly connected, Se2/0

C 30.0.0.0/8 is directly connected, Se3/0

S 10.0.0.0/8 [1/0] via 20.0.0.1

S 40.0.0.0/8 [1/0] via 30.0.0.2

IP route for Router 3 after setting IP route

C 30.0.0.0/8 is directly connected, Se2/0

C 40.0.0.0/8 is directly connected, Fa0/0

S* 0.0.0.0/0 [1/0] via 30.0.0.1

Result:

Pinging PC2 from PC4

PC4> ping 10.0.0.20

Reply from 10.0.0.20 bytes=32 time=10ms TTL=125

Reply from 10.0.0.20 bytes=32 time=6ms TTL=125

Reply from 10.0.0.20 bytes=32 time=8ms TTL=125

Reply from 10.0.0.20 bytes=32 time=5ms TTL=125

Packets Sent=4, Received=4, Loss=0 (0% loss)

Approx round trip time

Min=5ms Max=10ms, Average=7ms

Observation:-

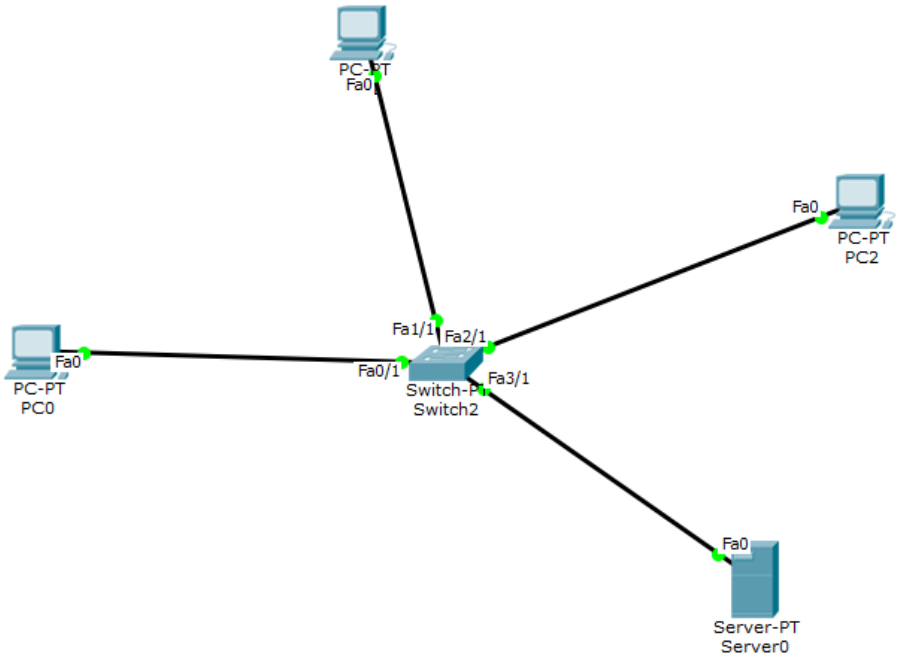
Static & Default ip route can be configured for a router connected to a switch, using CLI. After executing 'show ip route' command, different ip routes for that router are shown. C denotes connected, S denotes static & S* denotes default route.

Experiment No. 4

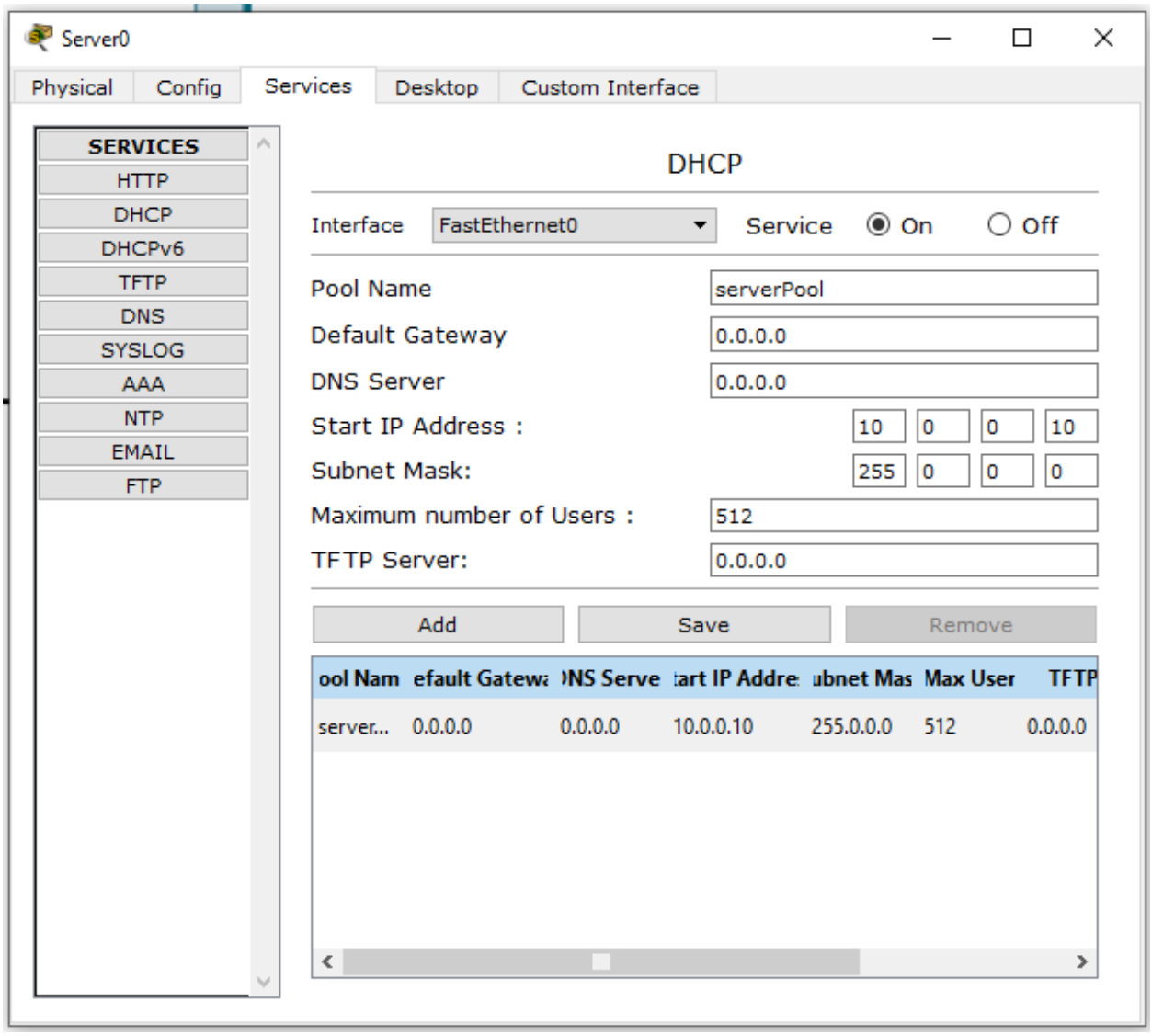
Title:
Configure DHCP within a LAN and outside LAN.

a)Within LAN

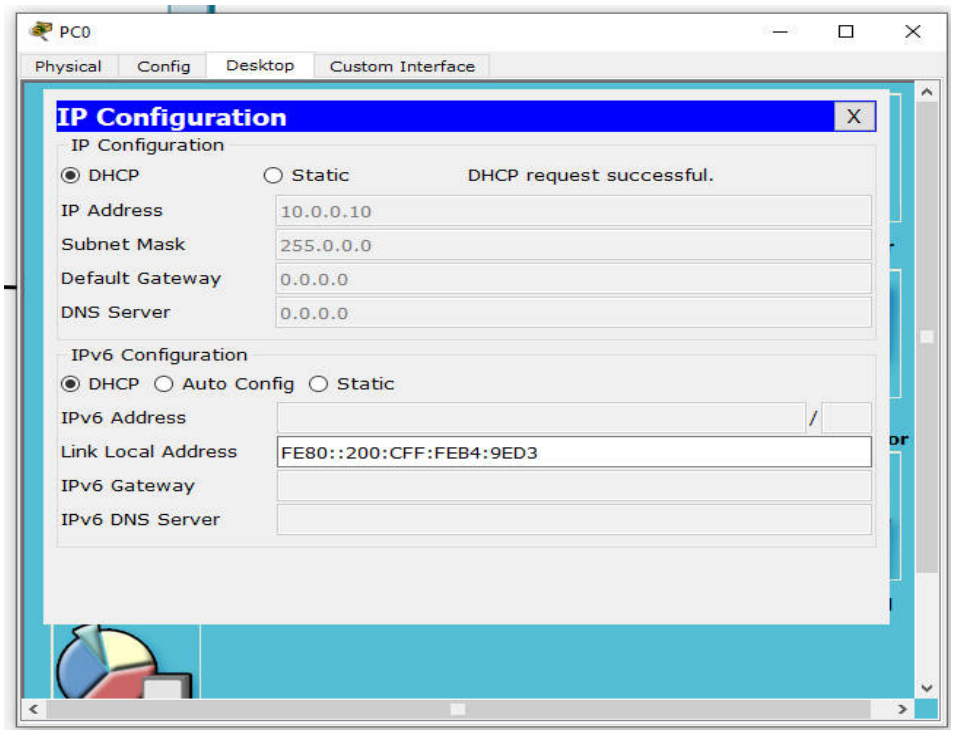
Topology:

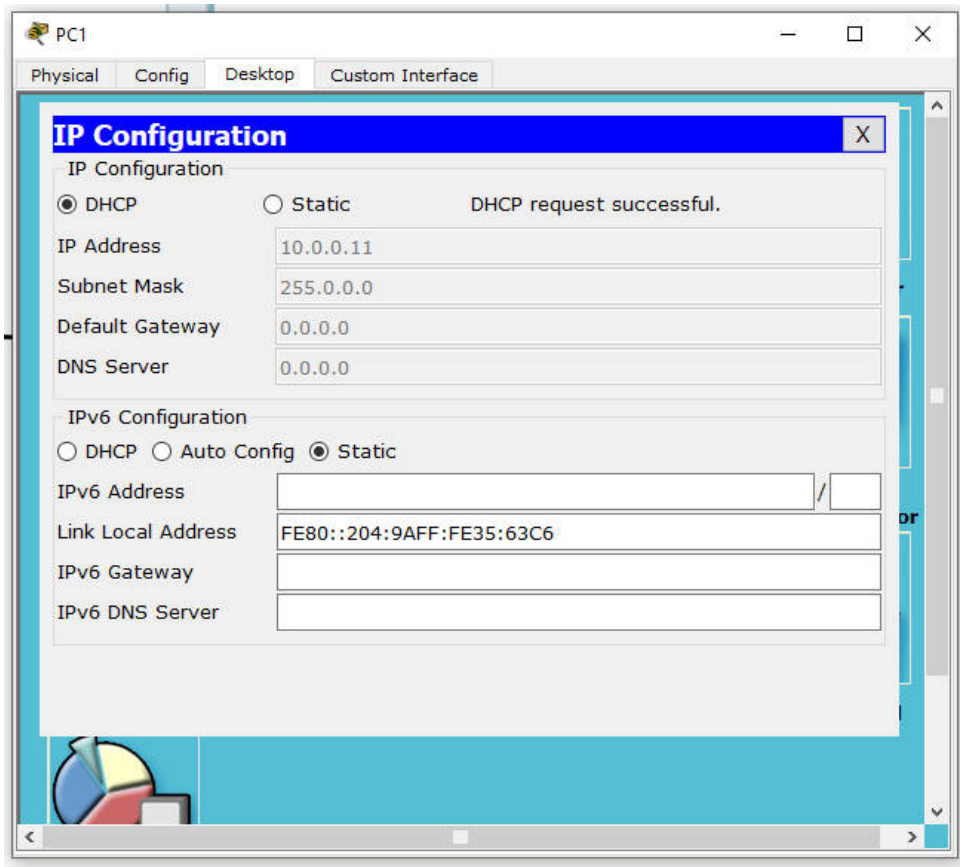


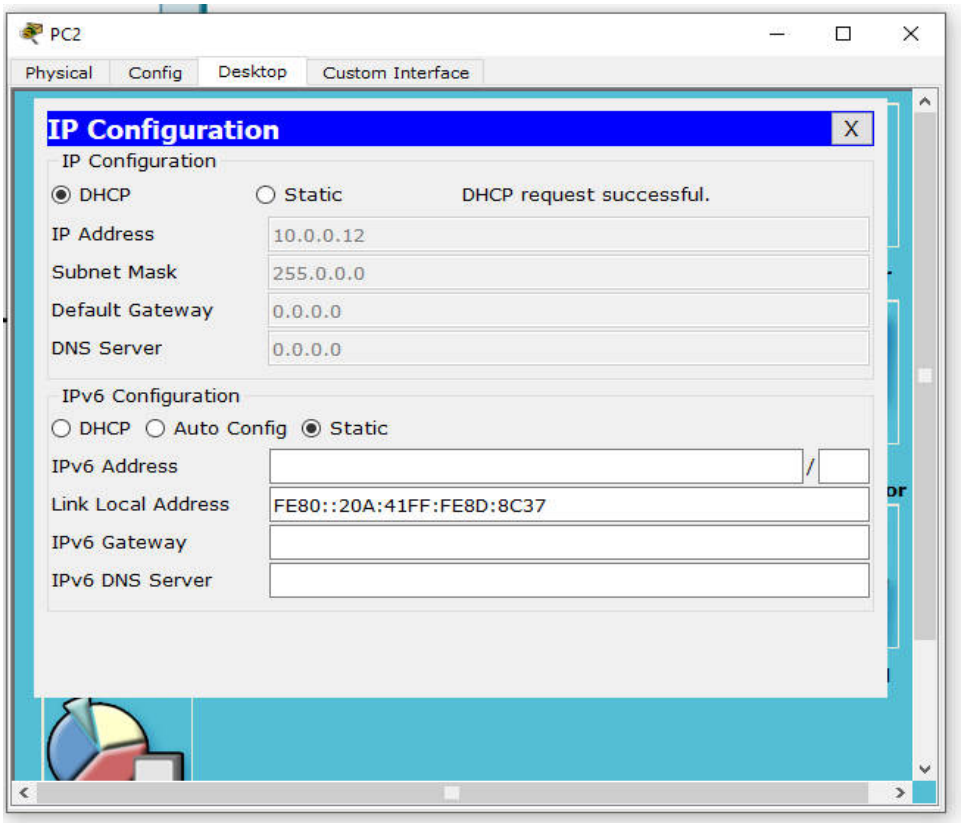
Server:


Server0
Physical
Config
Services
Desktop
Custom Interface
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
DHCP
Interface
FastEthernet0
Service
On
Off
Pool Name
serverPool
Default Gateway
0.0.0.0
DNS Server
0.0.0.0
Start IP Address :
10 0 0 10
Subnet Mask:
255 0 0 0
Maximum number of Users :
512
TFTP Server:
0.0.0.0
Add
Save
Remove
ool Nam efault Gatew DNS Serve tart IP Addre ubnet Mas Max User TFTP
server... 0.0.0.0 0.0.0.0 10.0.0.10 255.0.0.0 512 0.0.0.0

Obtaining IP:



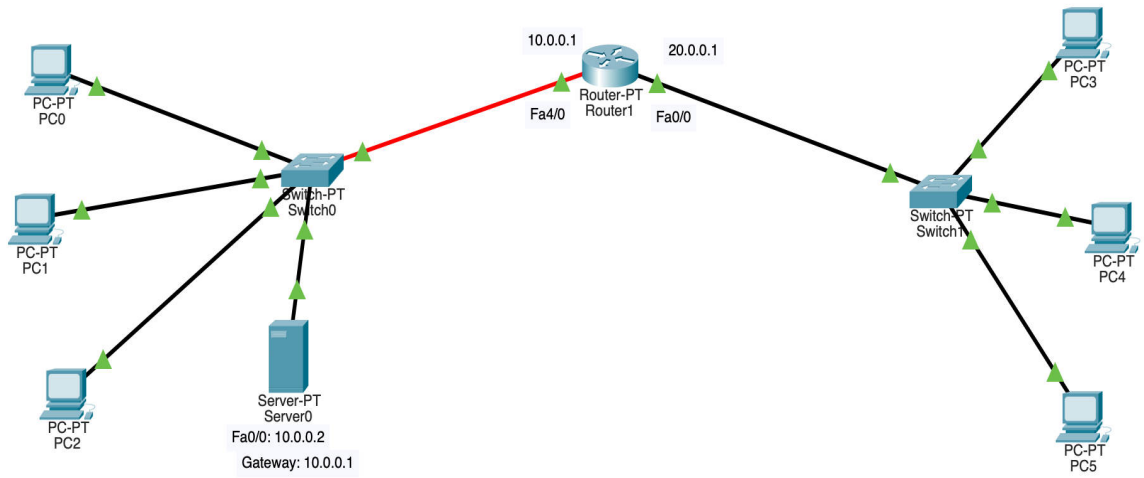




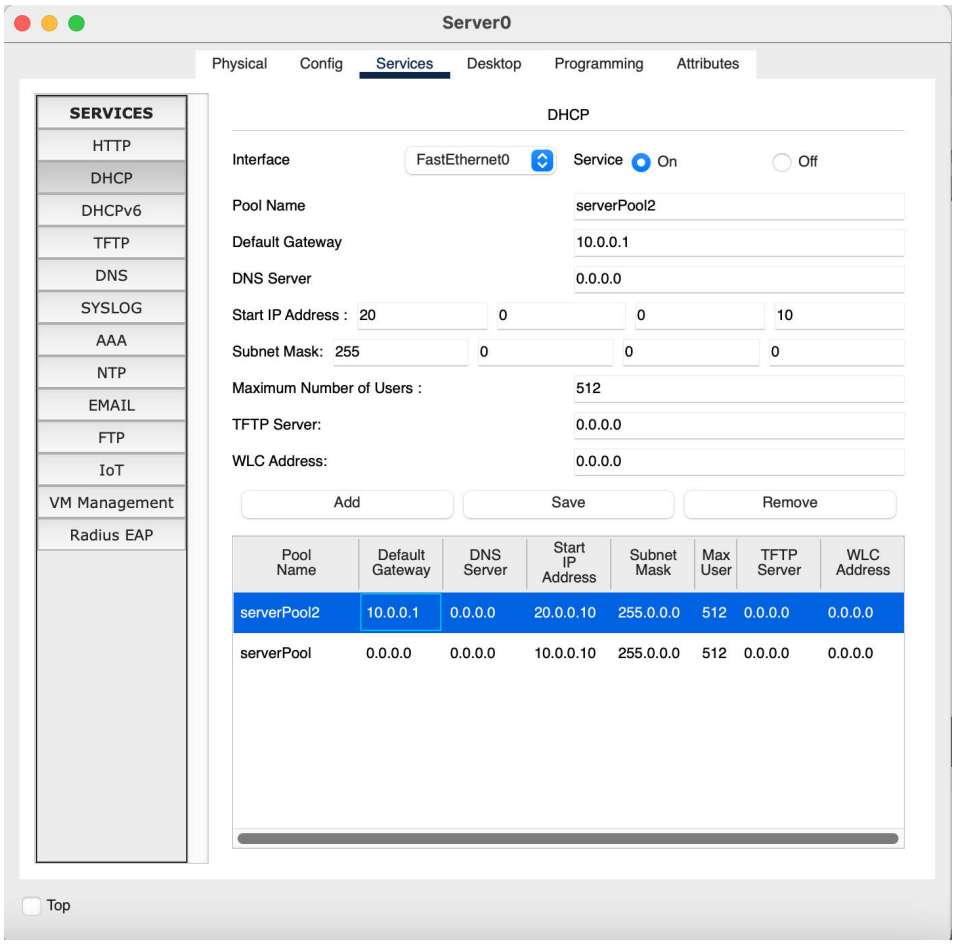
b)Outside LAN

I. Single Router

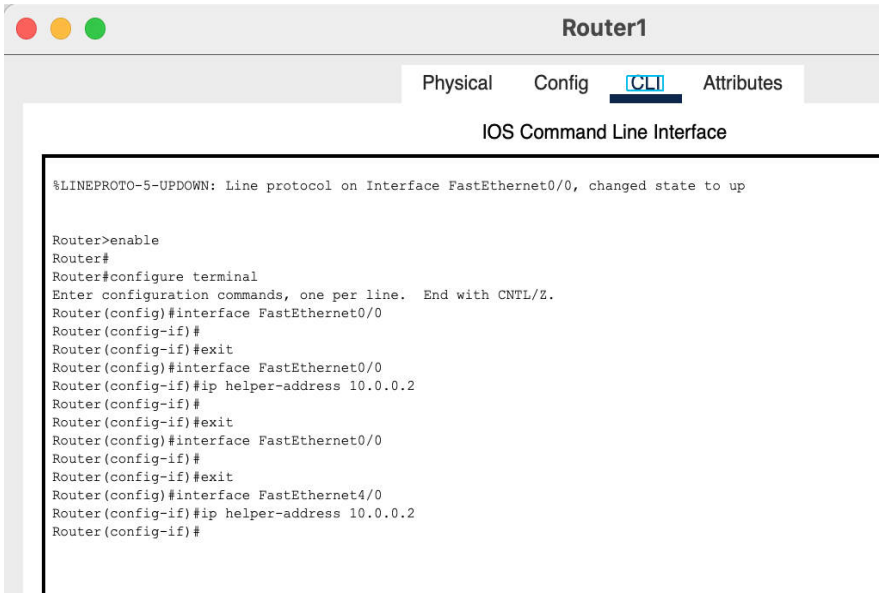
Topology:



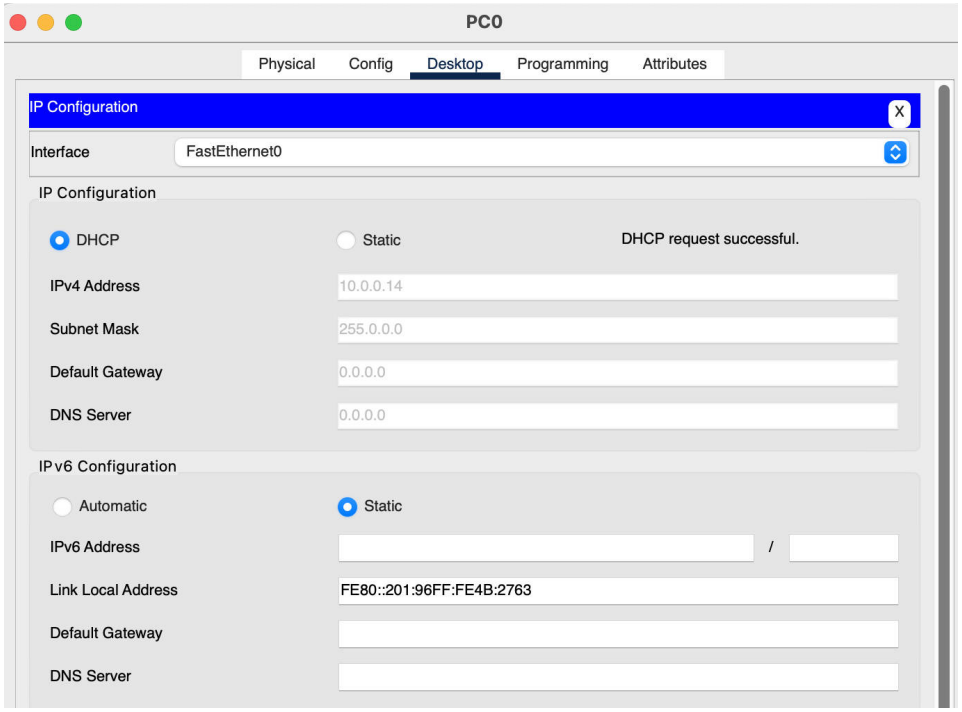
Server:

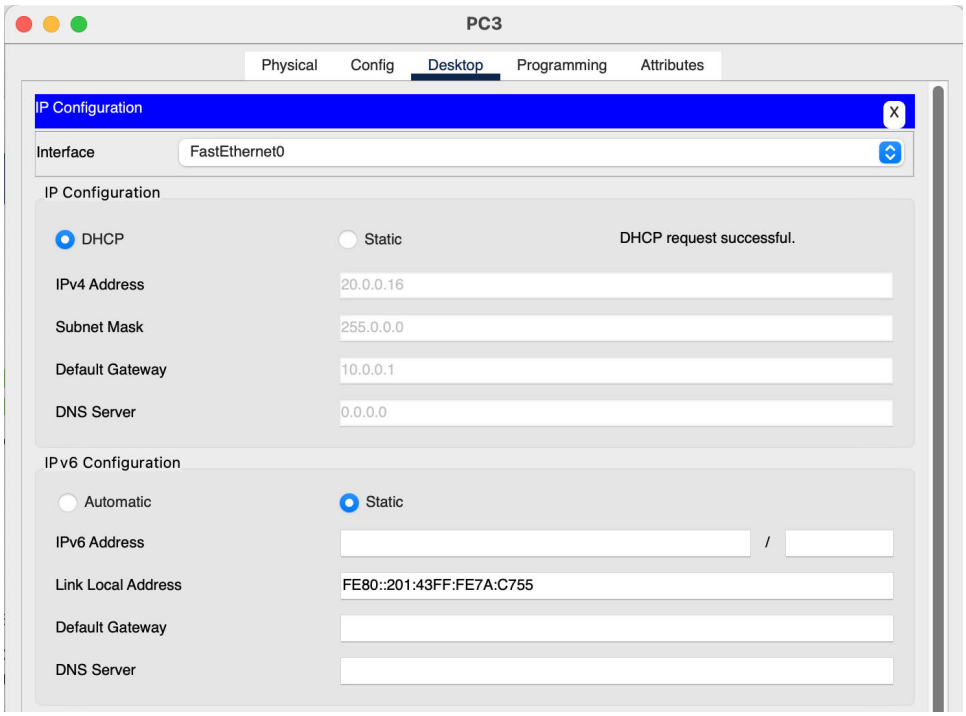


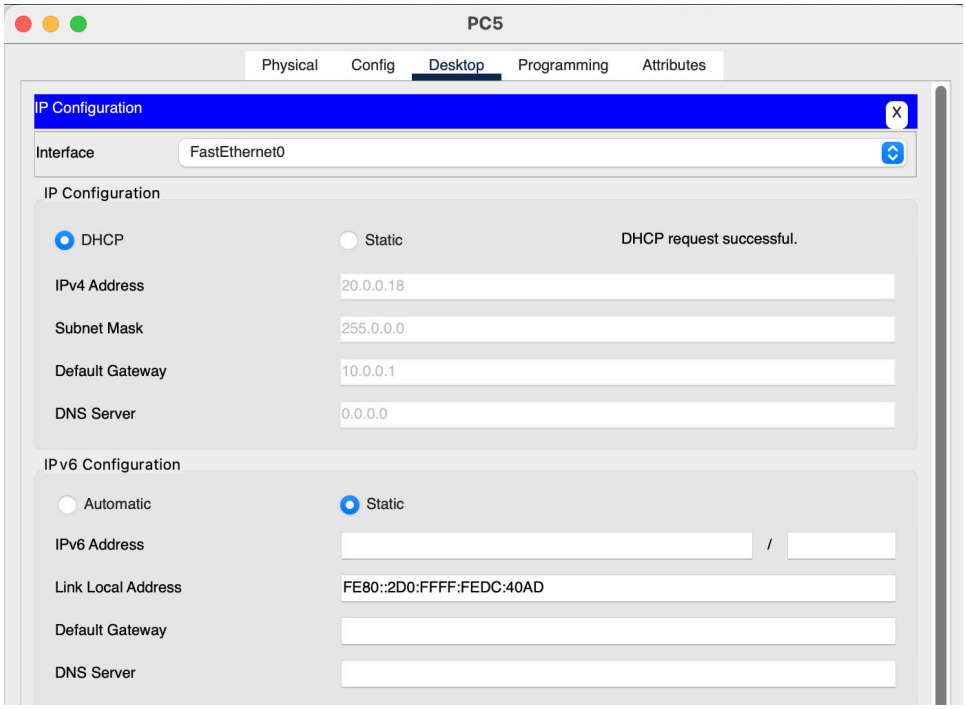
Router:



Obtaining IP:

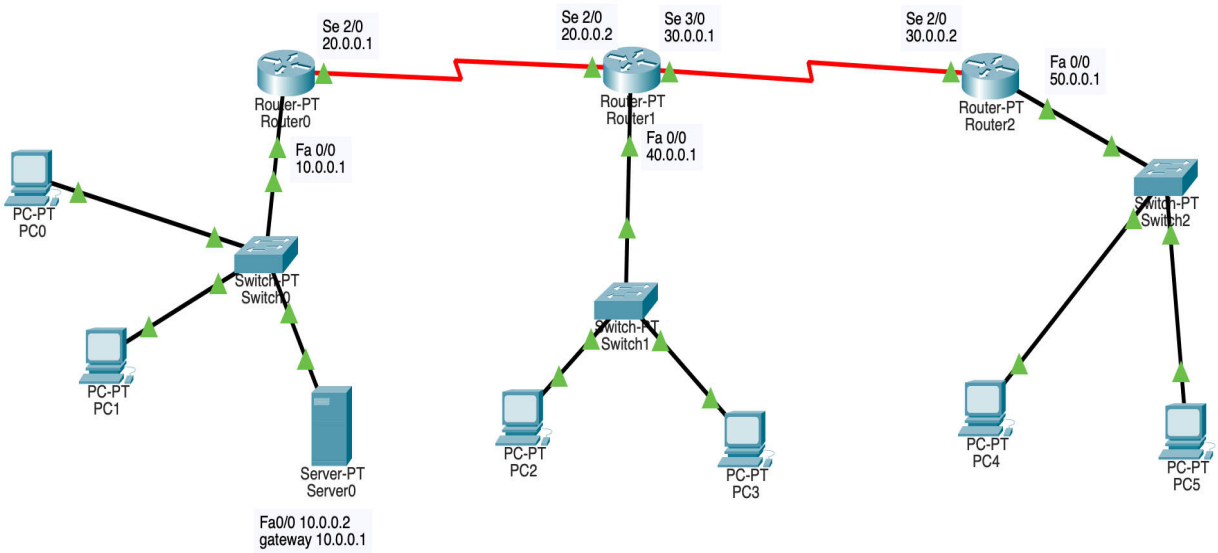




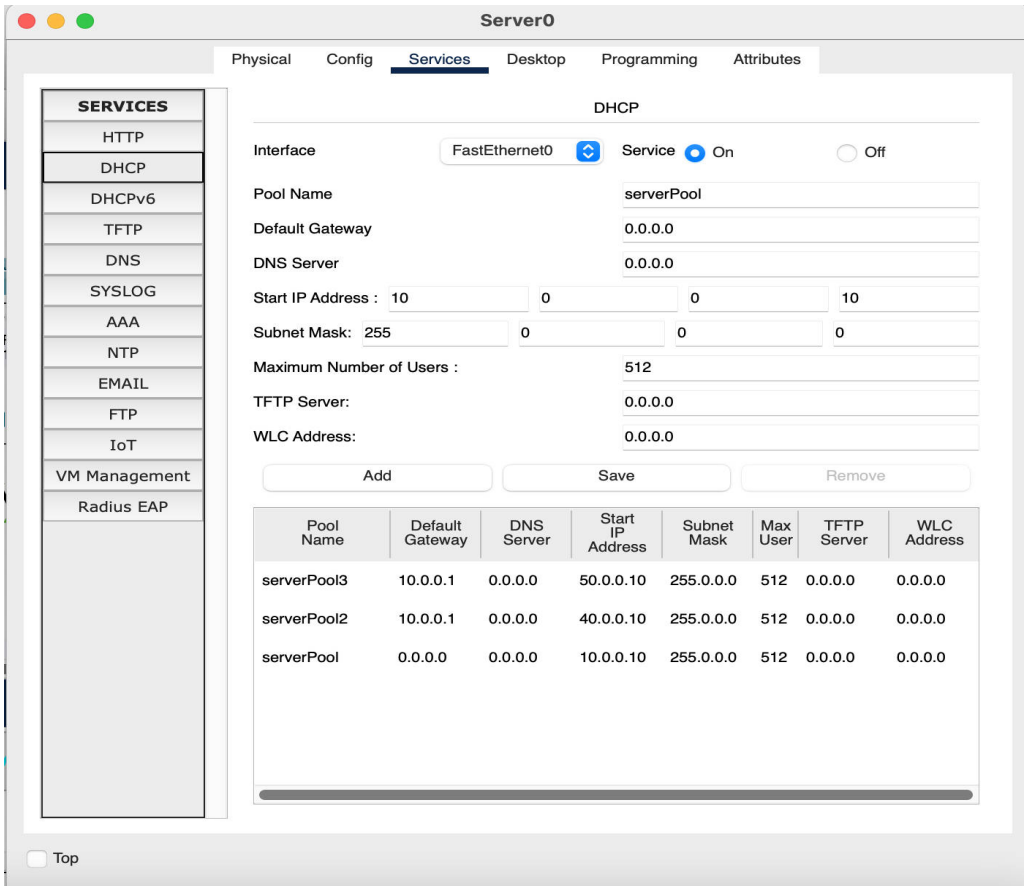


II. Multiple Routers

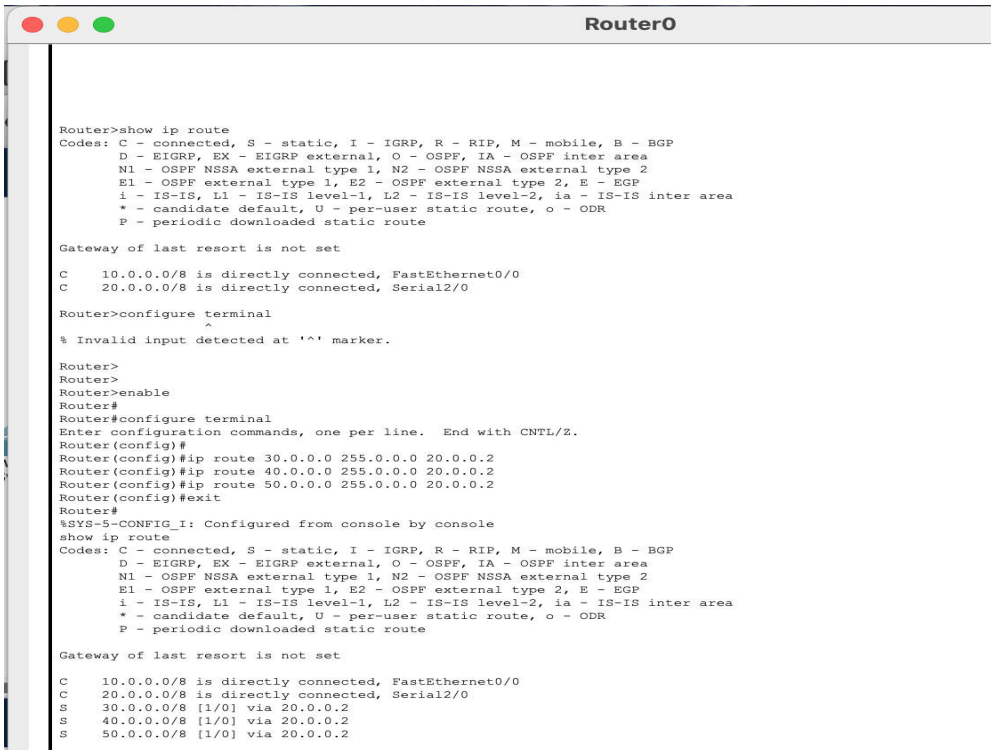
Topology:

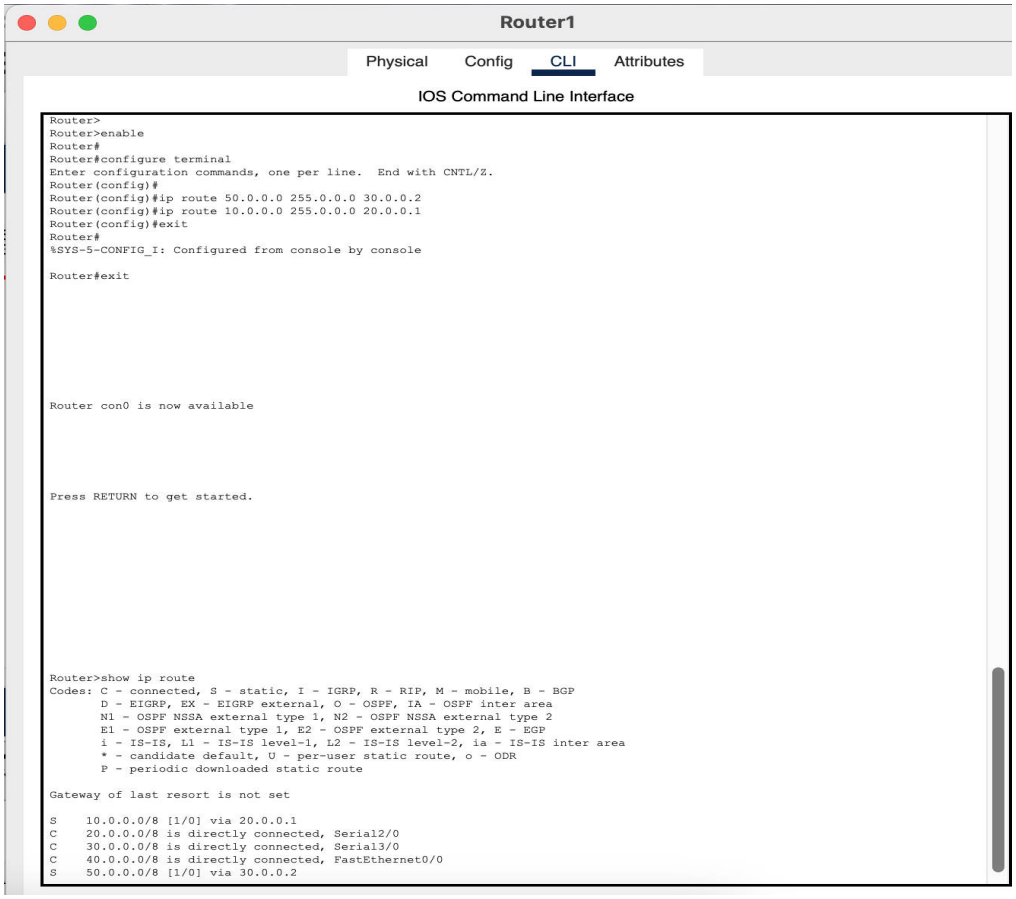


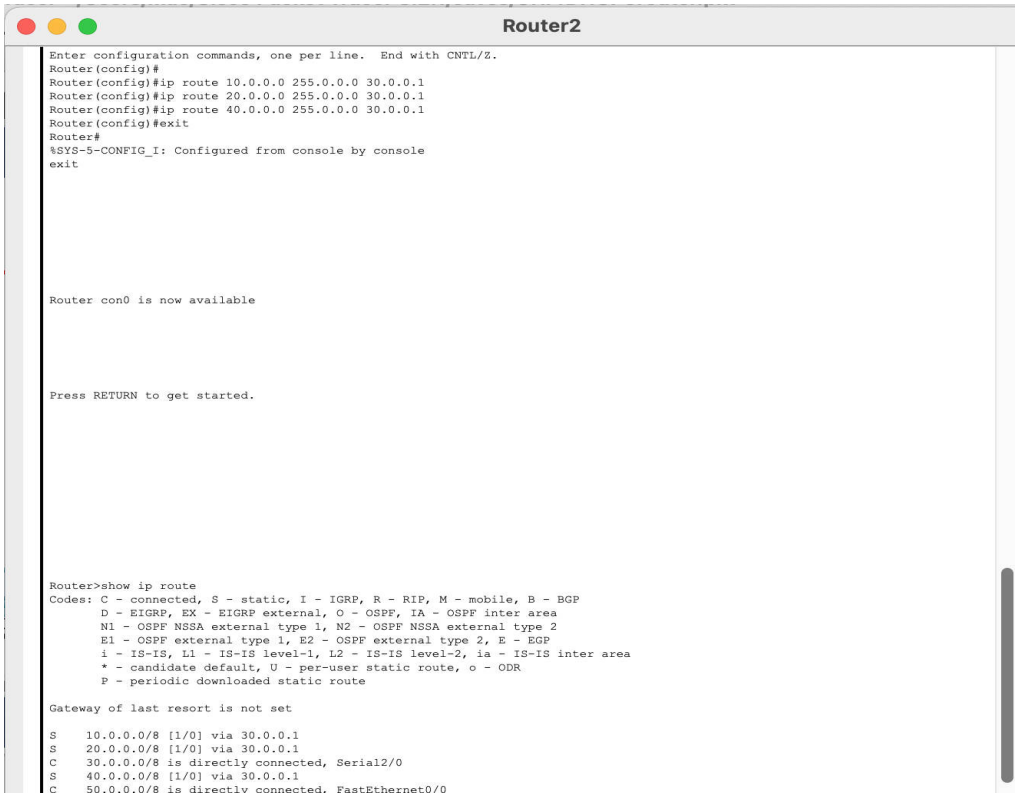
Server:



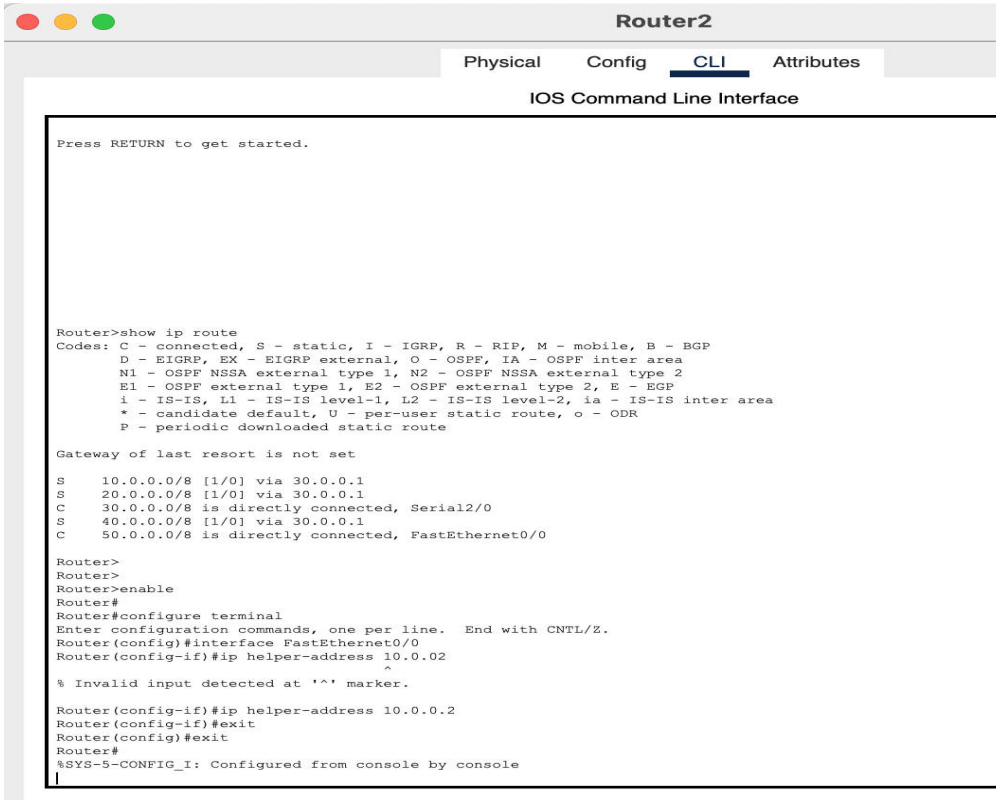
Router: setting IP route



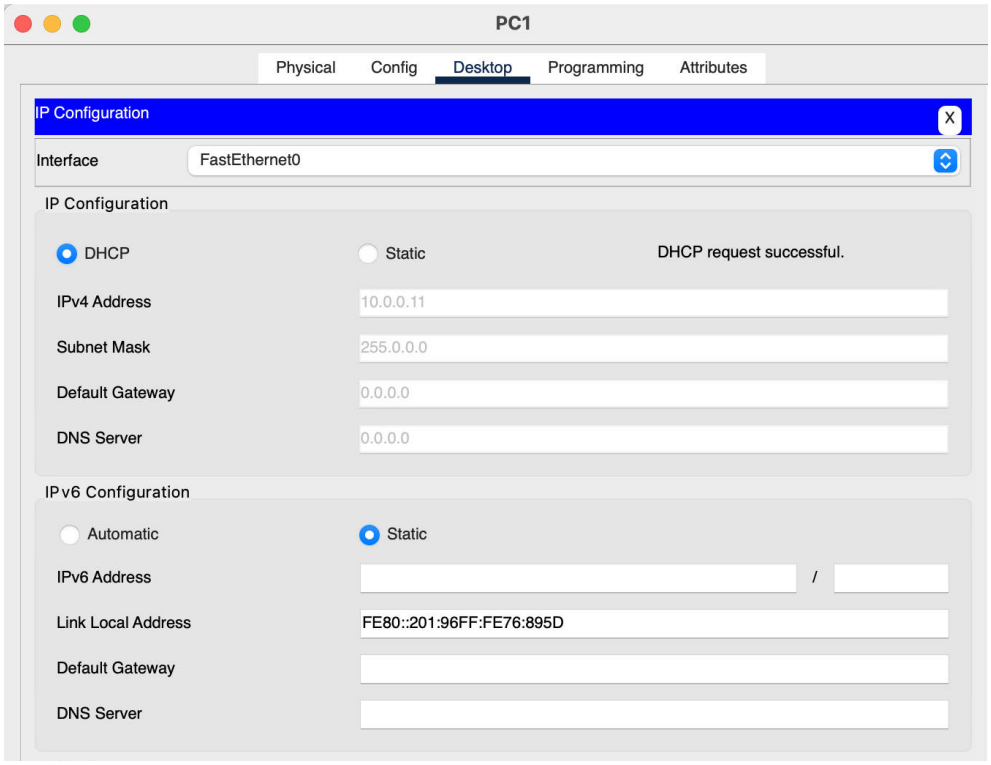

Router1
Physical
Config
CLI
Attributes
IOS Command Line Interface
Router>
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#ip route 50.0.0.0 255.0.0.0 30.0.0.2
Router(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.1
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#exit
Router con0 is now available
Press RETURN to get started.
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
S 10.0.0.0/8 [1/0] via 20.0.0.1
C 20.0.0.0/8 is directly connected, Serial2/0
C 30.0.0.0/8 is directly connected, Serial3/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
S 50.0.0.0/8 [1/0] via 30.0.0.2

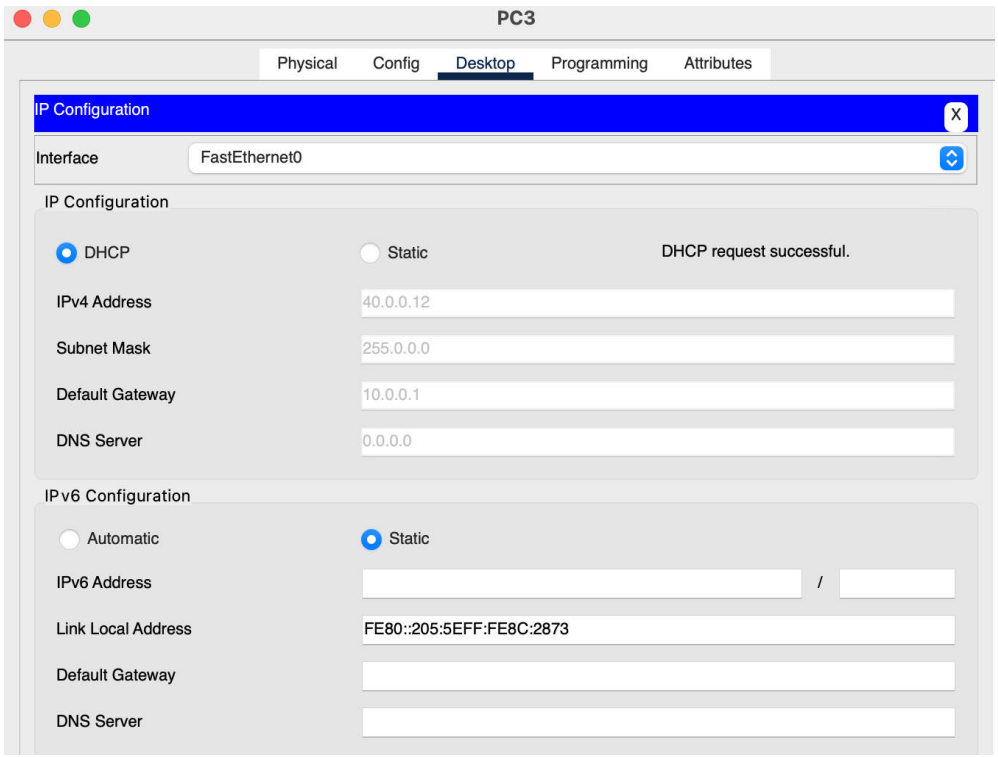

Router2
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Router(config)#ip route 10.0.0.0 255.0.0.0 30.0.0.1
Router(config)#ip route 20.0.0.0 255.0.0.0 30.0.0.1
Router(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.1
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
exit
Router con0 is now available
Press RETURN to get started.
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
S 10.0.0.0/8 [1/0] via 30.0.0.1
S 20.0.0.0/8 [1/0] via 30.0.0.1
C 30.0.0.0/8 is directly connected, Serial2/0
S 40.0.0.0/8 [1/0] via 30.0.0.1
C 50.0.0.0/8 is directly connected, FastEthernet0/0

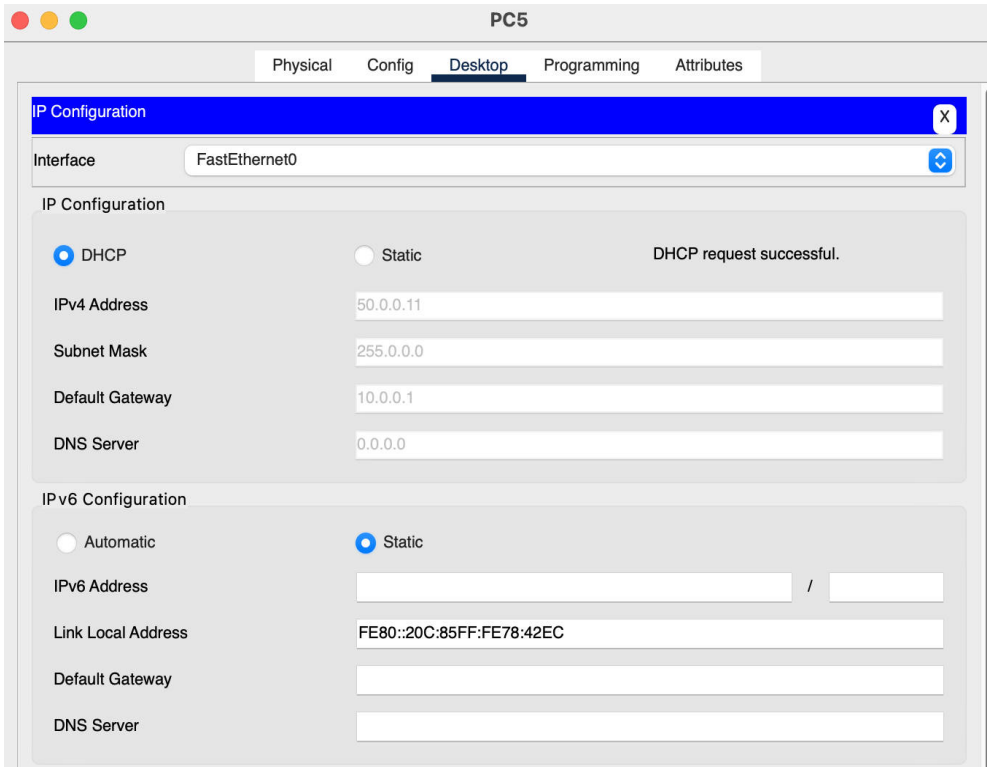
Setting IP helper address-



Obtaining IP:




PC3
Physical
Config
Desktop
Programming
Attributes
IP Configuration
Interface
FastEthernet0
IP Configuration
DHCP
Static
DHCP request successful.
IPv4 Address
40.0.0.12
Subnet Mask
255.0.0.0
Default Gateway
10.0.0.1
DNS Server
0.0.0.0
IPv6 Configuration
Automatic
Static
IPv6 Address
Link Local Address
FE80::205:5EFF:FE8C:2873
Default Gateway
DNS Server


PC5
Physical
Config
Desktop
Programming
Attributes
IP Configuration
Interface
FastEthernet0
IP Configuration
DHCP
Static
DHCP request successful.
IPv4 Address
50.0.0.11
Subnet Mask
255.0.0.0
Default Gateway
10.0.0.1
DNS Server
0.0.0.0
IPv6 Configuration
Automatic
Static
IPv6 Address
Link Local Address
FE80::20C:85FF:FE78:42EC
Default Gateway
DNS Server

Friday 14th July, 2023.

④ Title:

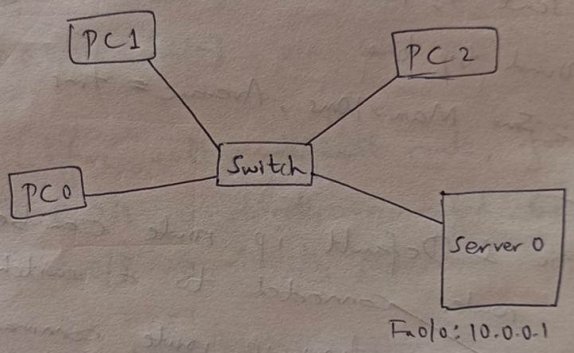
Configure DHCP within a LAN and outside LAN.

a) Within a LAN

Aim:-

To understand how to configure DHCP within a LAN.

Topology:



Procedure:

1. Connect the end devices to a switch as shown in the figure. Connect a server to the same switch.
2. set IP address for server 0 as 10.0.0.1
3. In server 0, under services click on DHCP.
4. Create a serverpool with starting IP address 10.0.0.10 & subnet mask 255.0.0.0. Click on save.

5. Click on any of the PCs, navigate to Desktop → IP Configuration.

6. Choose DHCP under IP configuration

Result:

→ PC0

IP configuration:

IP address - 10.0.0.10

Subnet Mask - 255.0.0.0

→ PC1

IP configuration:

IP address - 10.0.0.11

Subnet mask - 255.0.0.0

→ PC2

IP configuration:

IP address - 10.0.0.12

Subnet mask - 255.0.0.0

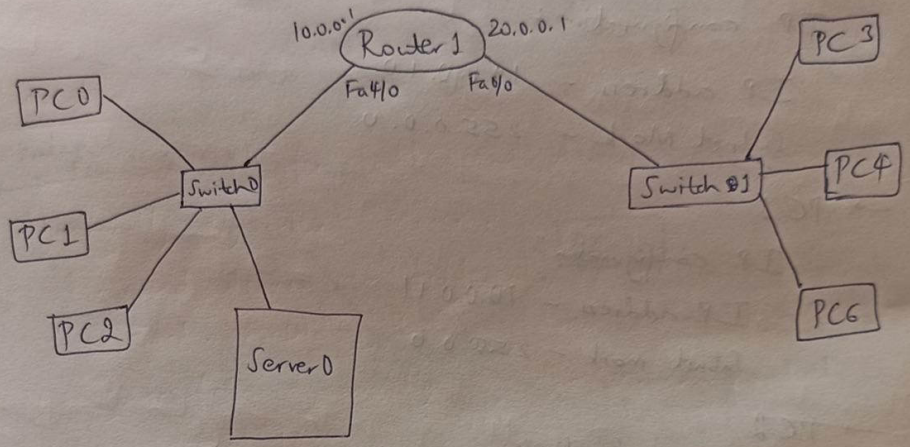
Observation:

IP addresses for all the end devices in the network i.e. PC0, PC1, PC2 is generated dynamically by the Server.

b) Outside a LAN

i) Aim: To understand how to configure DHCP for outside a LAN (with single Router)

Topology:



Procedure:

1. Connect end devices & server to one switch & only end devices to other switch.
2. Connect these two switches to a router.
3. Set IP address & gateway to server 0.

   IP: 10.0.0.2

   Gateway: 10.0.0.1

4. Configure the router, Router 1

Router(config)# Interface Fa4/0

Router(config)# ip address 10.0.0.1 255.0.0.0

Router(config)# no shutdown

Similarly set Fa0/0 to 20.0.0.1

5. Under Fa4/0 add ip helper-address as server address.

6. Create serverPool with starting address 10.0.0.10 & serverPool2 with starting address 20.0.0.10, gateway 10.0.0.1.

7. Click on any of the PCs, navigate to Desktop → IP configuration & choose DHCP.

~~Result~~:

Adding IP helper address:-

Router(config)# interface Fa0/0

Router(config-if)# ip helper-address 10.0.0.2

Router(config-if)# exit

Result:

End devices in network 10.0.0.0

→ PC0

IP address : 10.0.0.14

Subnet mask : 255.0.0.0

→ PC1

IP address : 10.0.0.16

Subnet mask : 255.0.0.0

End devices in network 20.0.0.0

→ PC3

IP address: 20.0.0.16

Subnetmask: 255.0.0.0

→ PC5

IP address: 20.0.0.18

Subnetmask: 255.0.0.0

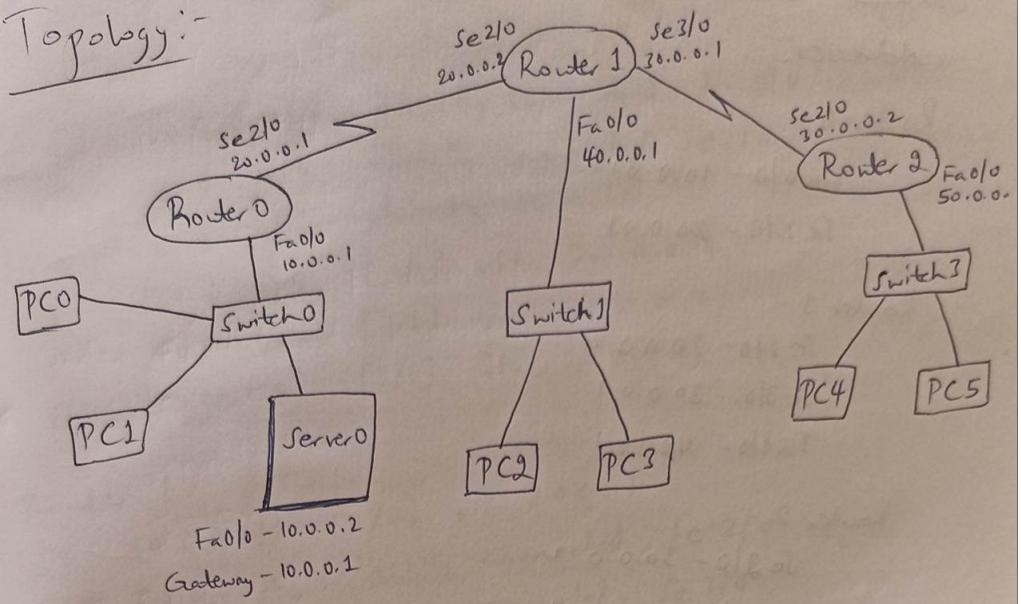
Observation:-

IP address for all end devices are generated dynamically by the server.

For end devices in different network, ip helper address is added. With that address, the ip address is generated dynamically by the server pointed by ip helper-address.

ii) Aim:- To understand how to configure DHCP outside LAN (with multiple routers)

Topology:-



Procedure:-

1. Connect the switch, Router, enddevices & server as shown in the topology.
2. Set IP address & gateway to server 0.
   IP address - Fa0/0 => 10.0.0.2
   Gateway - 10.0.0.1
3. Create 3 serverPool. Navigate to server0 → Services → DHCP.
   → ServerPool
   Starting IP address - 10.0.0.10
   → ServerPool2
   Starting IP address - 40.0.0.10
   Gateway - 10.0.0.1
   → ServerPool3
   Starting IP address - 50.0.0.10
   Gateway - 10.0.0.1

4. Configure each router with respective IP addresses.

Router 0 -

Fa0/0 - 10.0.0.1

Se2/0 - 20.0.0.1

Router 1 -

Se2/0 - 20.0.0.2

Se3/0 - 30.0.0.1

Fa0/0 - 40.0.0.1

Router 2 -

Se2/0 - 30.0.0.2

Fa0/0 - 50.0.0.1

5. Set up IP route for each network wherever necessary, using IP route command.

6. Add IP helper address for networks that doesn't have a server.

→ For Network 40.0.0.0

Under Fa0/0

Router(config)# Interface Fa0/0

Router(config-if)# ip helper-address 10.0.0.2

Router(config-if)# exit

→ For Network 50.0.0.0

Router(config)# Interface Fa0/0

Router(config-if)# ip helper-address 10.0.0.2

Router(config-if)# exit

7. Click on any of PCs, navigate to Desktop → IP configuration & choose DHCP.

Result:

Router 0 -

C	10.0.0.0/8	directly connected Fa0/0
C	20.0.0.0/8	directly connected Se2/0
~~S~~	~~40.0.0.0/8~~	~~directly connected~~
S	30.0.0.0/8	[1/0] via 20.0.0.2
S	40.0.0.0/8	[1/0] via 20.0.0.2
S	50.0.0.0/8	[1/0] via 20.0.0.2

Router 1 -

S	10.0.0.0/8	[1/0] via 20.0.0.1
C	20.0.0.0/8	directly connected, Se2/0
C	30.0.0.0/8	directly connected, Se3/0
C	40.0.0.0/8	directly connected, Fa0/0
S	50.0.0.0/8	[1/0] via 30.0.0.2

Router 2 -

S	10.0.0.0/8	[1/0] via 30.0.0.1
S	20.0.0.0/8	[1/0] via 30.0.0.1
C	30.0.0.0/8	directly connected, Se2/0
S	40.0.0.0/8	[1/0] via 30.0.0.1
C	50.0.0.0/8	directly connected, Fa0/0

PC1 -

IP Address: 10.0.0.11

PC3 -

IP address: 40.0.0.12

PC5 -

IP address: 50.0.0.11

PC1 -

IP address - 10.0.0.11

PC3 -

IP address - 40.0.0.12

Gateway - 10.0.0.1

PC5

IP address - 50.0.0.11

Gateway - 10.0.0.1

Observation:

IP addresses for end devices in all the networks are generated dynamically.

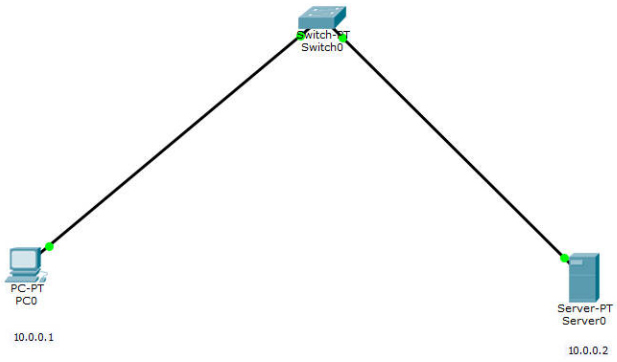
For networks that doesn't have server, IP helper address is added, which helps in generating the IP address

Experiment No. 5

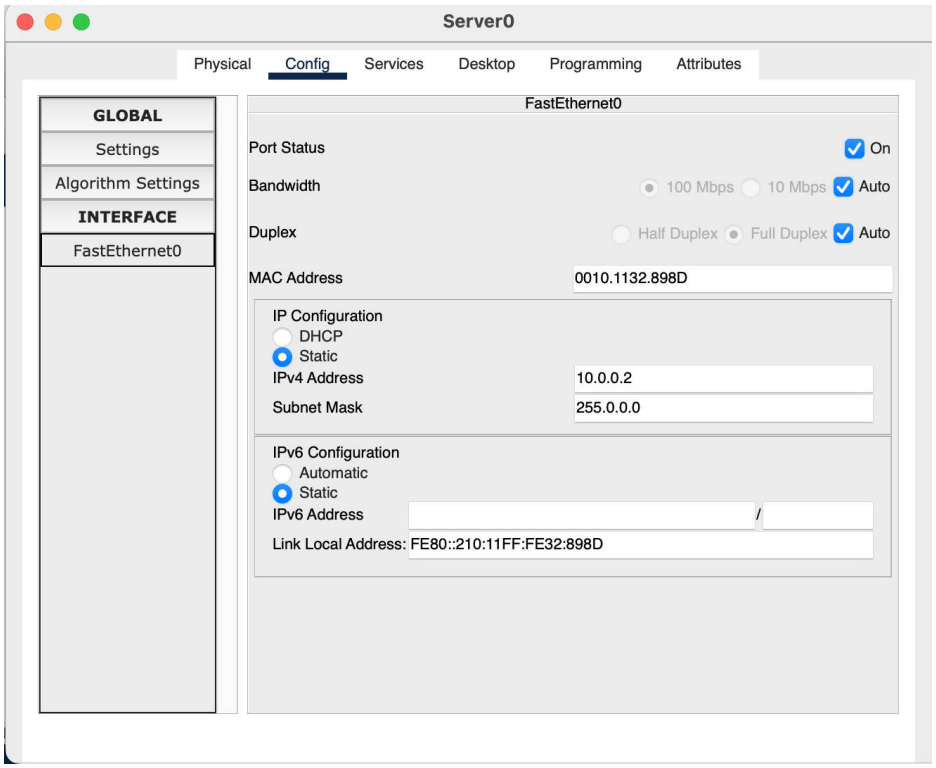
Title:

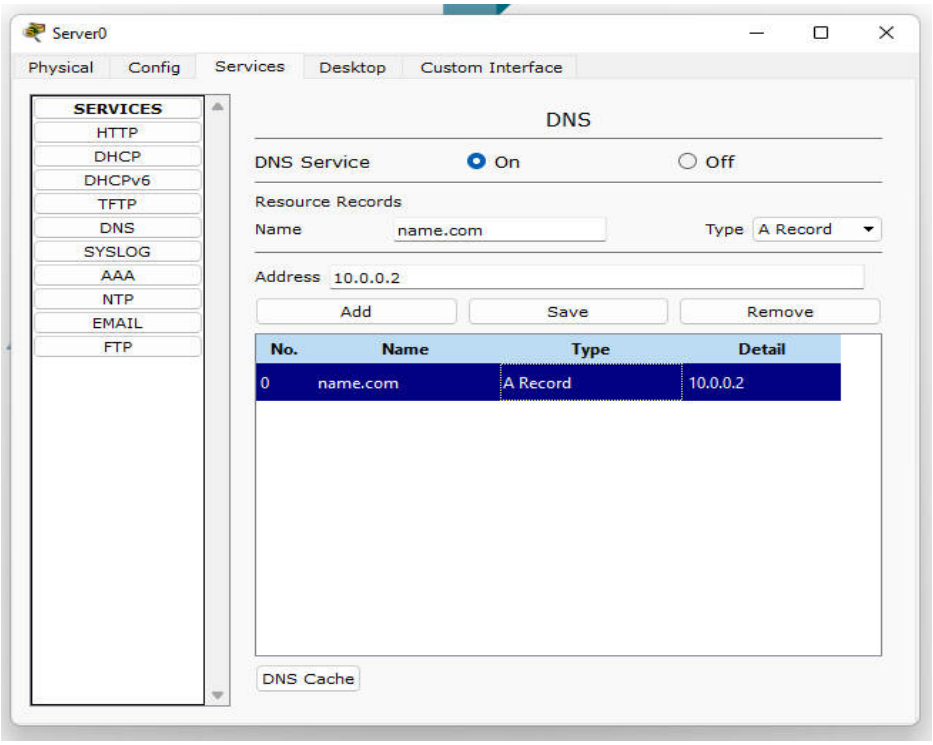
Configure Web Server, DNS within a LAN.

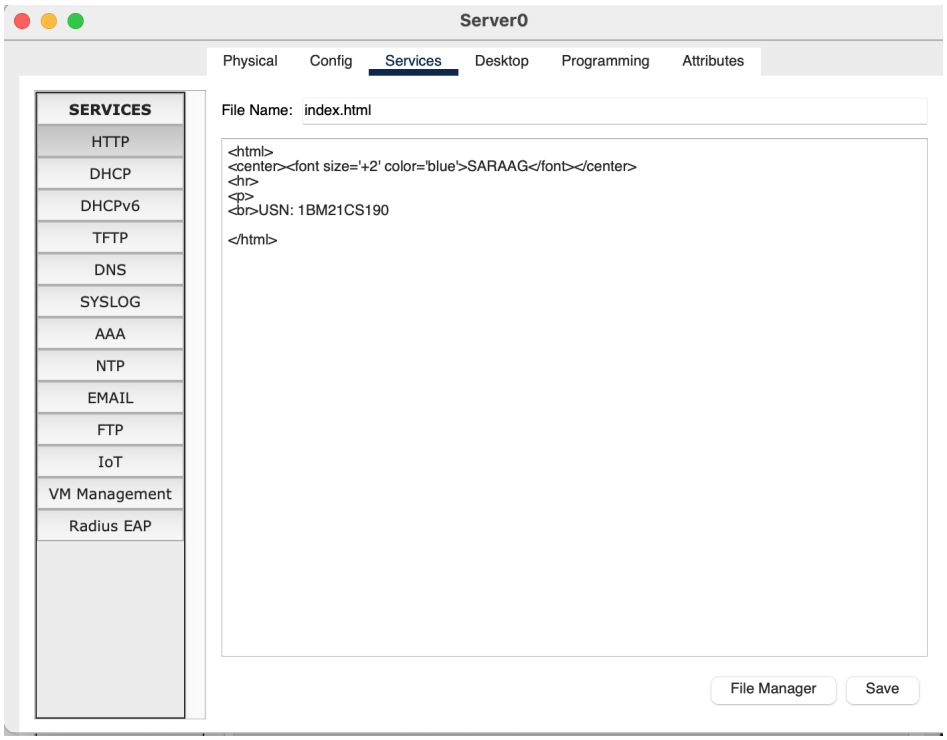
Topology:



Server:

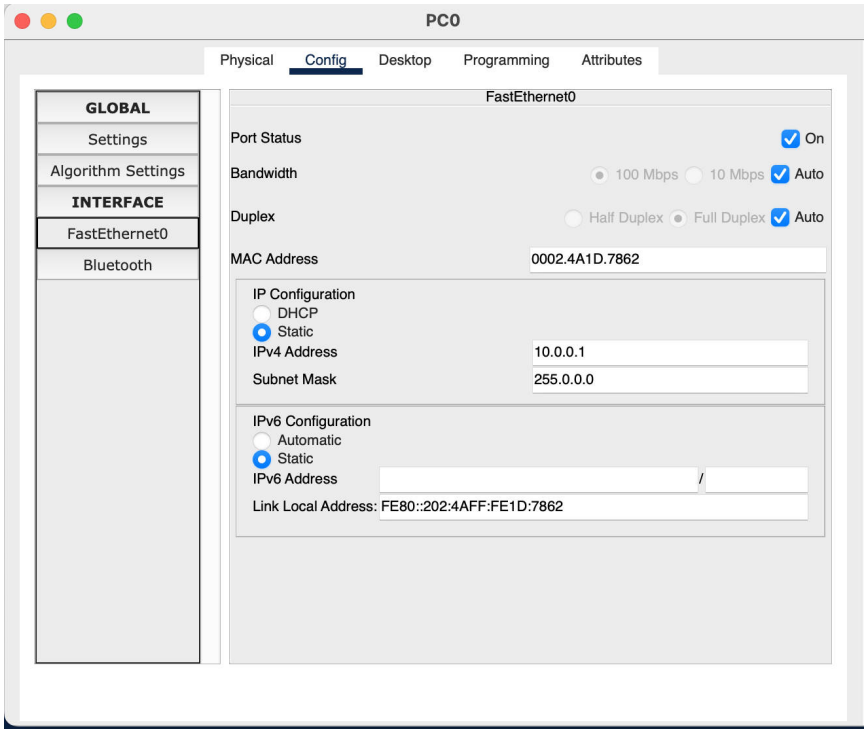



Server0
Physical
Config
Services
Desktop
Custom Interface
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
DNS
DNS Service
On
Off
Resource Records
Name
name.com
Type
A Record
Address
10.0.0.2
Add
Save
Remove
No.
Name
Type
Detail
0
name.com
A Record
10.0.0.2
DNS Cache

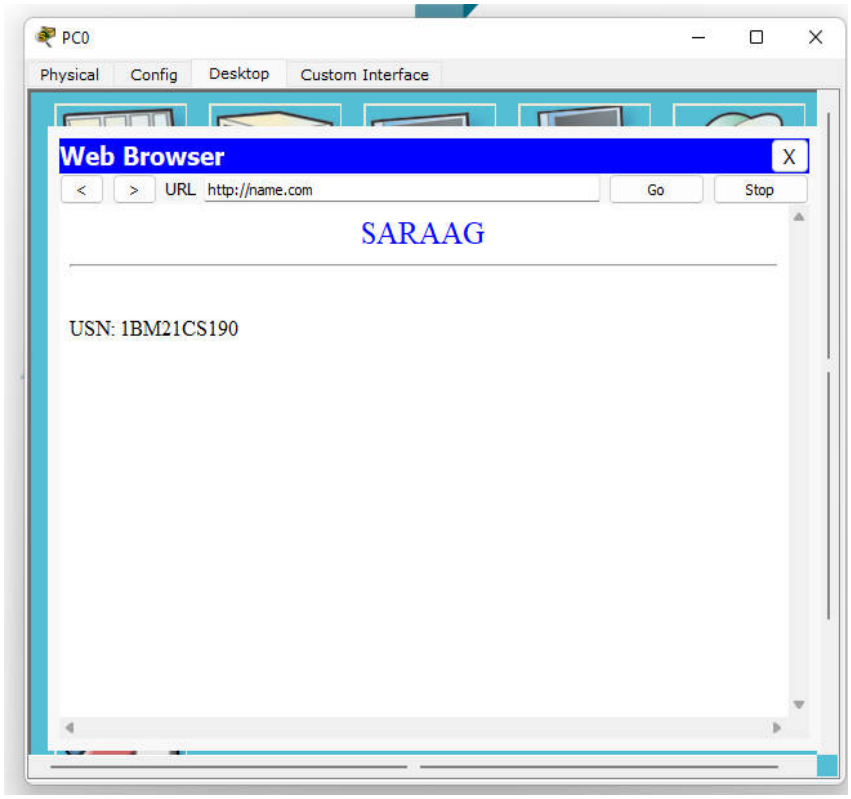

Server0
Physical
Config
Services
Desktop
Programming
Attributes
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
IoT
VM Management
Radius EAP
File Name:
index.html
<html>
SARAAG
<hr>
<p>

USN: 1BM21CS190
</html>
File Manager
Save

PC:


PC0
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
FastEthernet0
Bluetooth
FastEthernet0
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
0002.4A1D.7862
IP Configuration
DHCP
Static
IPv4 Address
10.0.0.1
Subnet Mask
255.0.0.0
IPv6 Configuration
Automatic
Static
IPv6 Address
Link Local Address:
FE80::202:4AFF:FE1D:7862

Output:


PC0
Physical
Config
Desktop
Custom Interface
Web Browser
X
URL
http://name.com
Go
Stop
SARAAG
USN: 1BM21CS190

21st July, 2023.

(5)

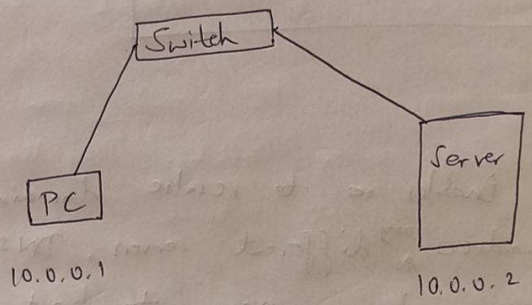
Title:

Configure Web server, DNS within LAN.

Aim:

To understand how to configure Web server, DNS within a LAN.

Topology:

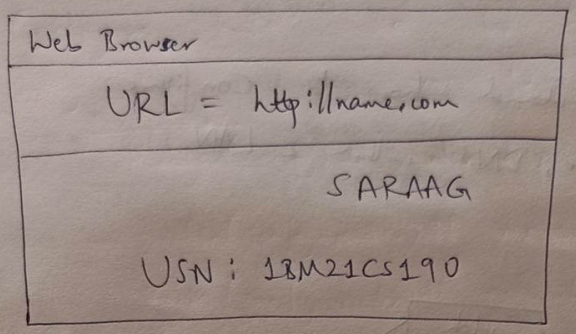


Procedure:

1. Connect the PC, Switch & Server as shown in the above topology.
2. Set IP address for PC (10.0.0.1) & Server (10.0.0.2)
3. In Server, under services click on DNS.
4. Turn on DNS service, Give name for the website and give the IP address (same as server IP) Then Click add.
5. Under services, click on HTTP. Type the HTML code in the index.html file, for the website to display.

6. Open PC, under Desktop click on Web Browser. Enter the website name with ".com" and hit 'Go'.

Result:



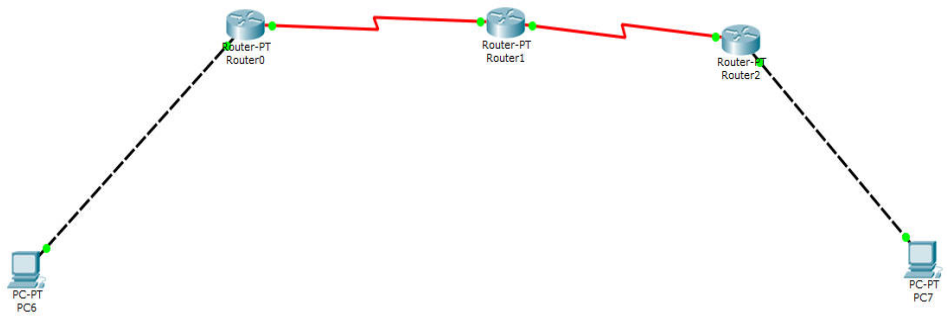
Observation:

DNS Enables to resolve domain names, IP address using different servers. DNS allows to map domain name to host address.

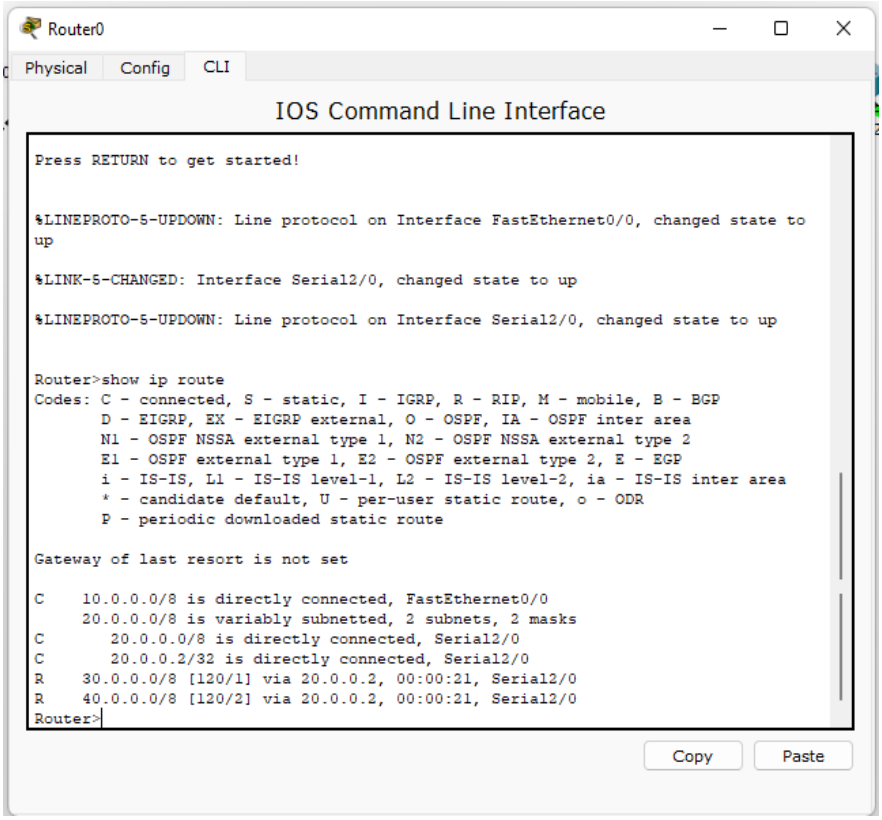
Experiment No. 6

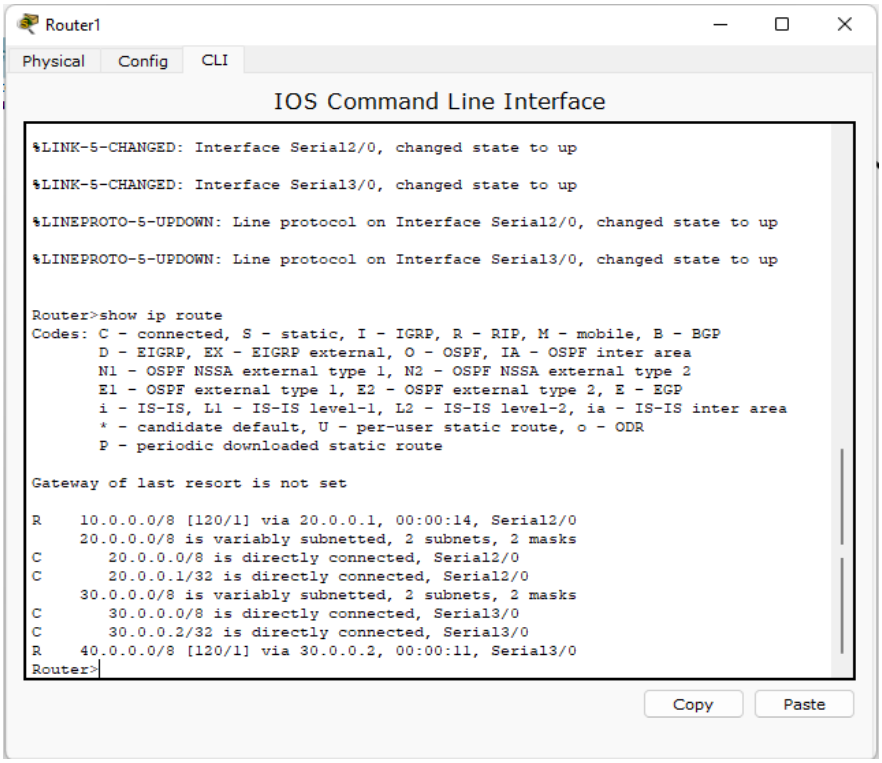
Title:
Configure RIP routing Protocol in Routers

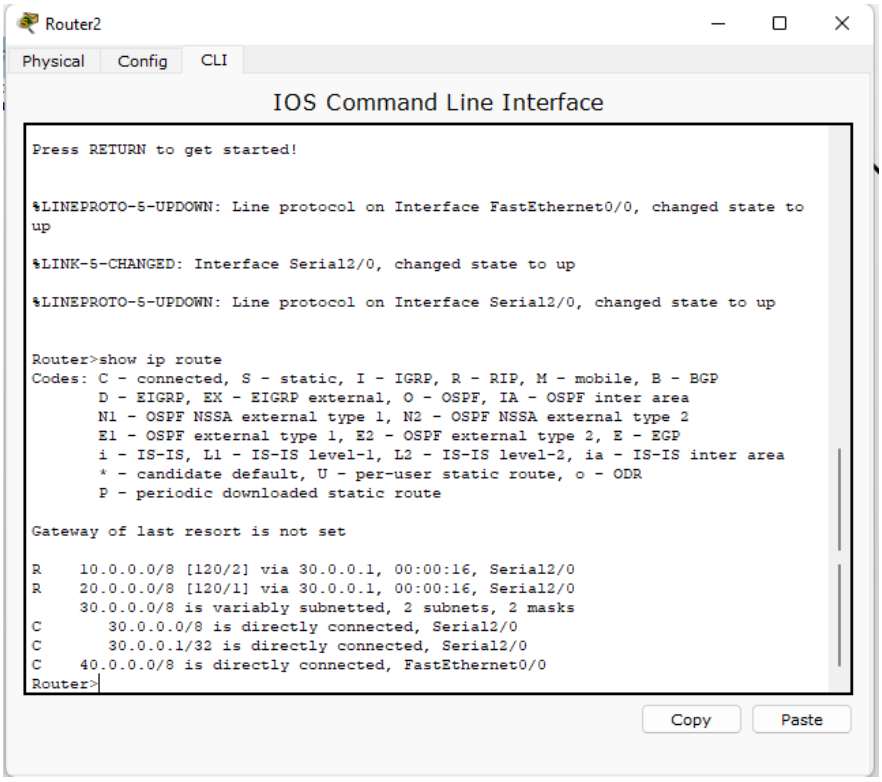
Topology:



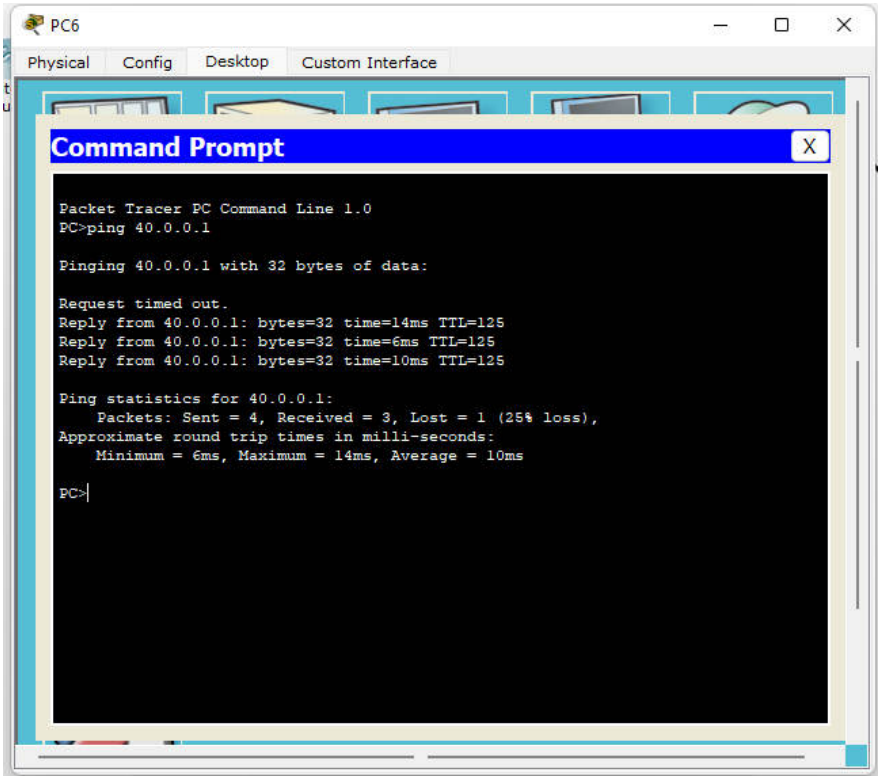
IP Route:



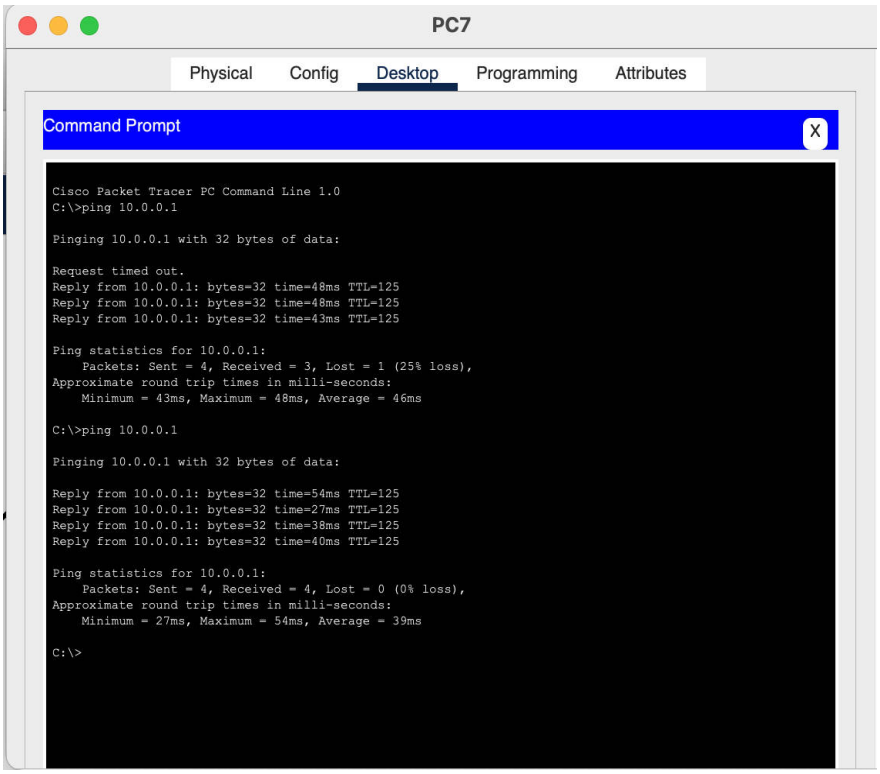

Router1
Physical
Config
CLI
IOS Command Line Interface
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
R 10.0.0.0/8 [120/1] via 20.0.0.1, 00:00:14, Serial2/0
20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 20.0.0.0/8 is directly connected, Serial2/0
C 20.0.0.1/32 is directly connected, Serial2/0
30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 30.0.0.0/8 is directly connected, Serial3/0
C 30.0.0.2/32 is directly connected, Serial3/0
R 40.0.0.0/8 [120/1] via 30.0.0.2, 00:00:11, Serial3/0
Router>
Copy
Paste


Router2
Physical
Config
CLI
IOS Command Line Interface
Press RETURN to get started!
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
R 10.0.0.0/8 [120/2] via 30.0.0.1, 00:00:16, Serial2/0
R 20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:16, Serial2/0
30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 30.0.0.0/8 is directly connected, Serial2/0
C 30.0.0.1/32 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
Router>
Copy
Paste

Pinging PC7 from PC6:



Pinging PC6 from PC7:

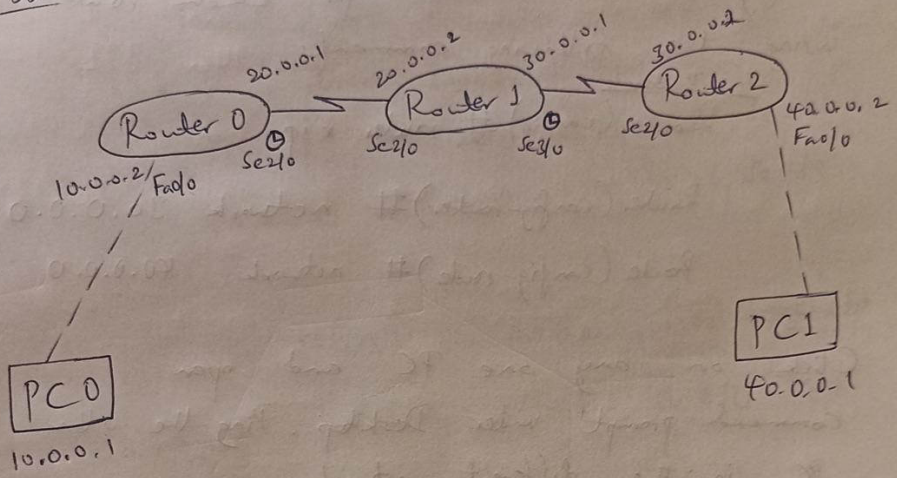


(2)

Title: Configure RIP routing protocol in Routers.

Aim: Understand how to configure RIP routing.

Topology:



Procedure:

1. Connect the routers & PCs as shown in topology.
2. Configure the end devices with their IP addresses & subnet mask.
   PC0 - 10.0.0.1 & PC1 - 40.0.0.1
3. Configure the routers with static IP addresses

Router 0 -
Fa 0/0 - 10.0.0.1
Se 2/0 - 20.0.0.1

Router 1 -
Se 2/0 - 20.0.0.2
Se 3/0 - 30.0.0.1

Router 2 -
Se 2/0 - 30.0.0.2
Fa 0/0 - 40.0.0.2

4. Set clock rate for the necessary interface using the following command

```
Router(config-if)# clock rate 64000
```

5. Configure RIP route for each router using the following commands

For Router2

```
Router(config)# router rip
Router(config-router)# network 30.0.0.0
Router(config-router)# network 40.0.0.0
```

6. Click on any one PC and open command prompt under Desktop. Ping the other PC in the different network.

Configuring Router Example

Router 1:

```
Router > enable
Router# configure terminal
Router(config)# interface se2/0
Router(config-if)# ip addres 20.0.0.2 255.0.0.0
Router(config-if)# ip address 30.0.0 encapsulation ppp
Router(config-if)# exit no shutdown
Router(config-if)# exit
Router(config)# interface se3/0
Router(config-if)# ip addres 30.0.0.1 255.0.0.0
Router(config-if)# encapsulation ppp
Router(config-if)# clock rate 64000
Router(config-if)# no shutdown
Router(config-if)# exit
```

Result:-

Router > show ip route

Router 0

C 10.0.0.0/8 is directly connected Fa0/0

20.0.0.0/8 is variably connected 2 subnets, 2 masks

C 20.0.0.0/8 is directly connected Se 2/0

20.0.0.2/32 is directly connected, Se2/0

R 30.0.0.0/8 [120/1] via 20.0.0.2, 00:00:21, Se 2/0

R 40.0.0.0/8 [120/2] via 20.0.0.2, 00:00:21, Se 2/0

Router 1

R 10.0.0.0/8 [120/1] via 20.0.0.1, 00:00:14, Se2/0

C 20.0.0.0/8 is directly connected, Se 2/0

C 30.0.0.0/8 is directly connected, Se 3/0

R 40.0.0.0/8 [120/1] via 30.0.0.2, 00:00:11, Se 3/0

Router 2

R 10.0.0.0/8 [120/1] via 30.0.0.1, 00:00:16, Se 2/0

R 20.0.0.0/8 [120/1] via 30.0.0.1, 00:00:16, Se 2/0

C 30.0.0.0/8 directly connected, Se2/0

C 40.0.0.0/8 directly connected, Fa0/0

Pinging PC1 from PC0

PC0

PC> ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data

Reply from 40.0.0.1 bytes=32 time=13ms TTL=125
Reply from 40.0.0.1 bytes=32 time=14ms TTL=125
Reply from 40.0.0.1 bytes=32 time=6ms TTL=125
Reply from 40.0.0.1 bytes=32 time=10ms TTL=125

Packets: Sent=4, Recieved=4, Loss=0% Ps

Min=6ms, Max=14ms, Avg=10ms

Observation:

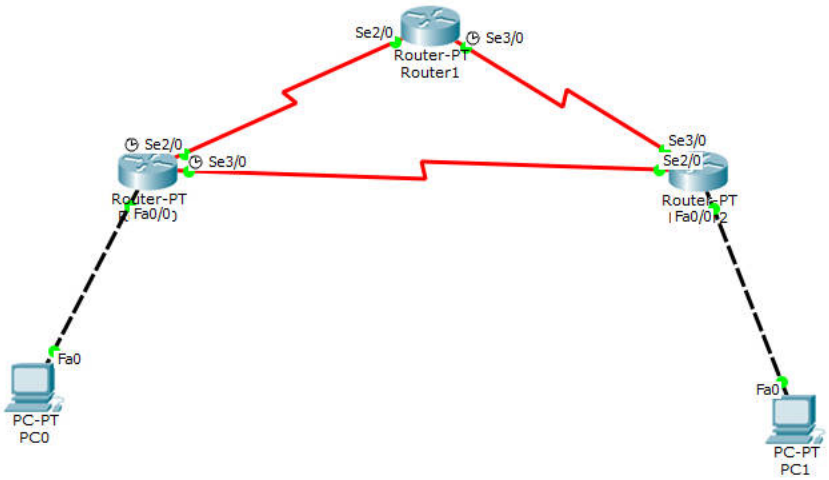
Static & RIP routing can be configured to each router using CLI. RIP routing is added using 'router $rip' command. When 'show ip route' is executed in the CLI C denotes connected and R denotes the RIP route.

Experiment No. 7

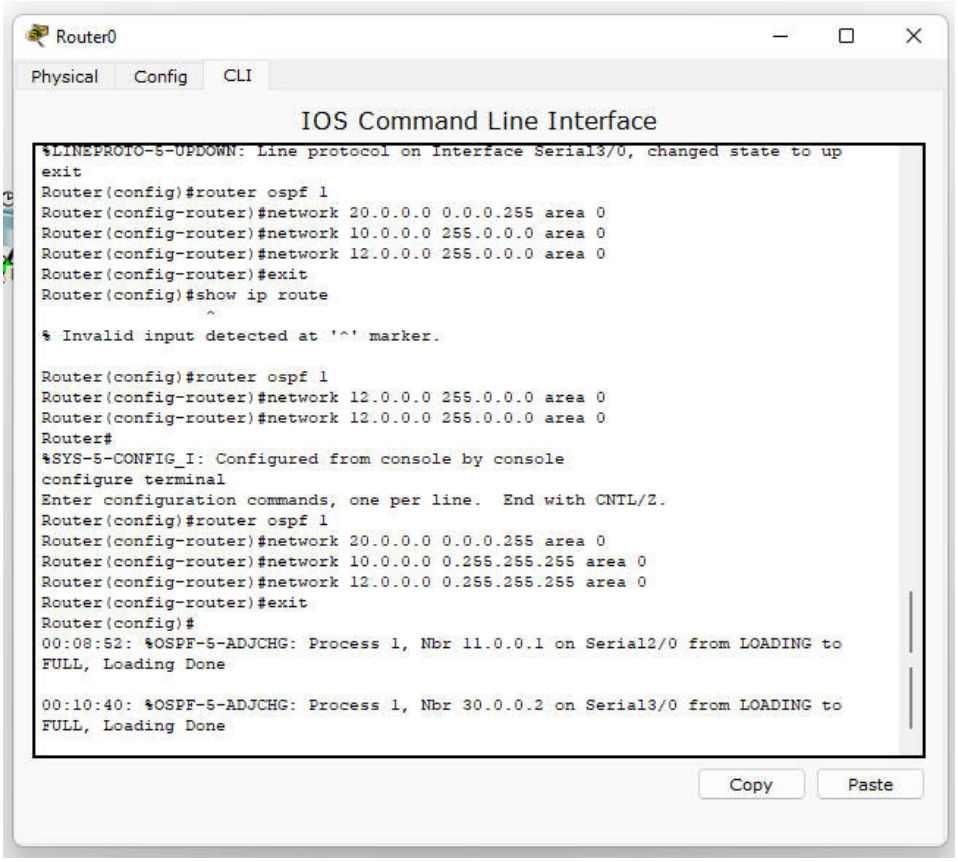
Title:

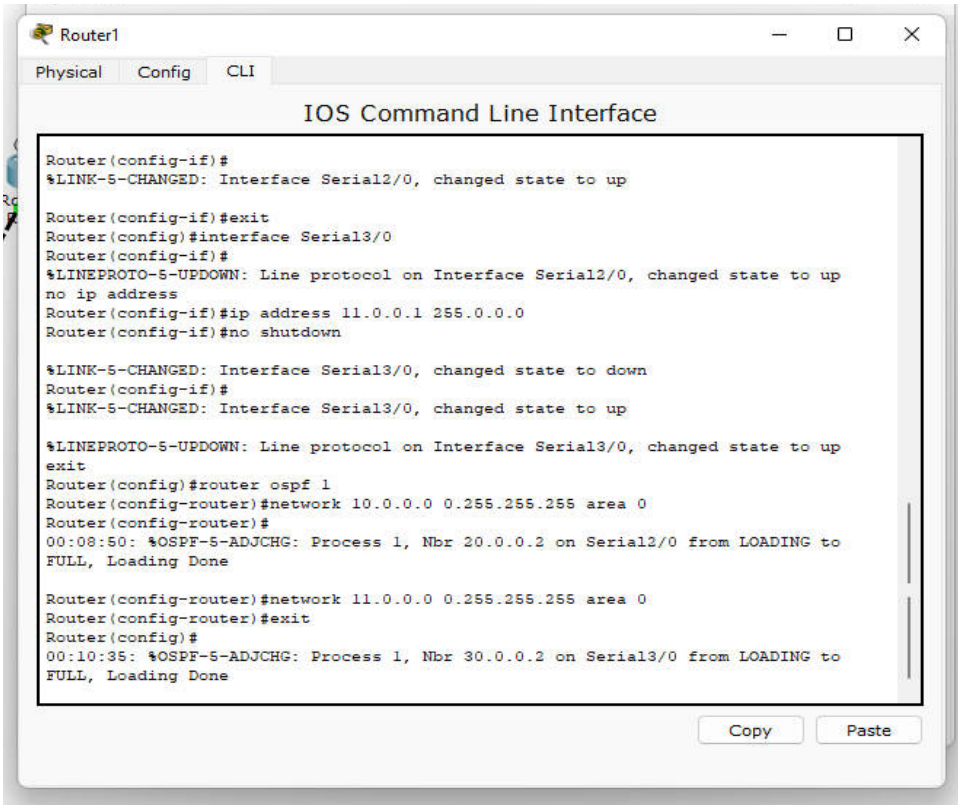
Configure OSPF routing protocol

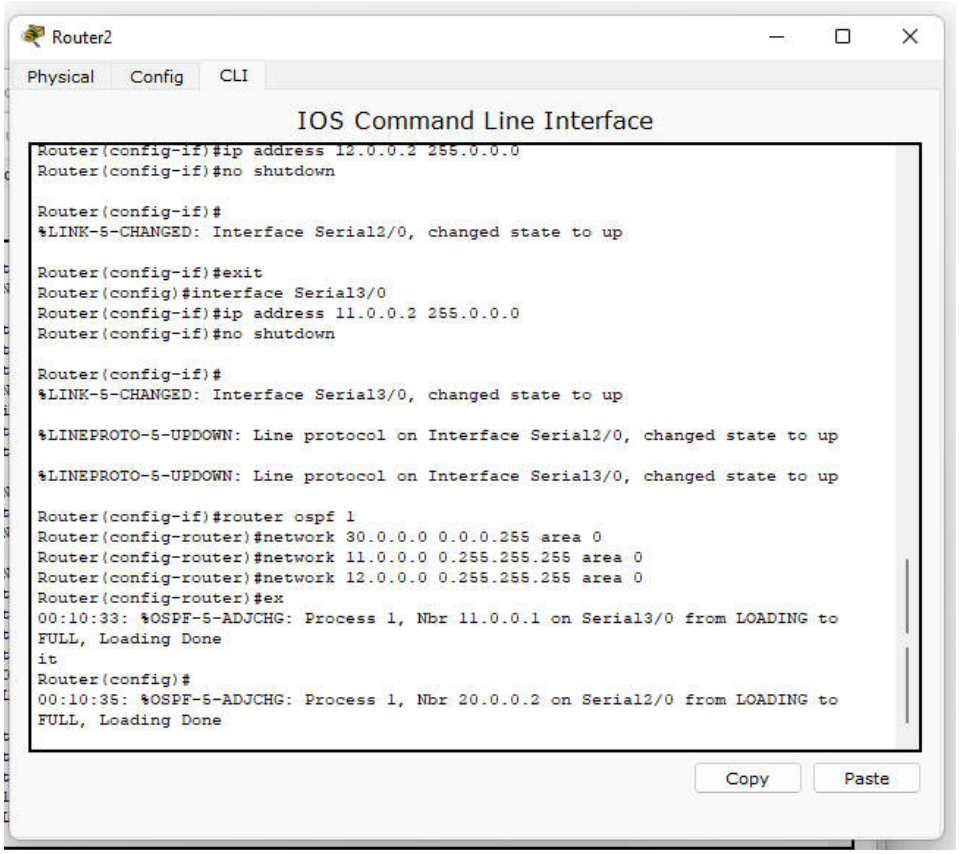
Topology:

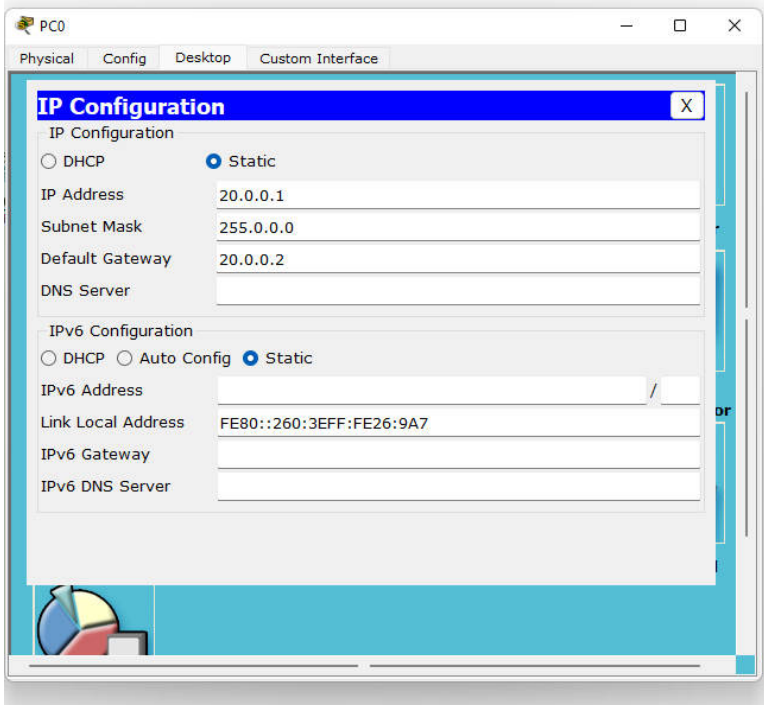


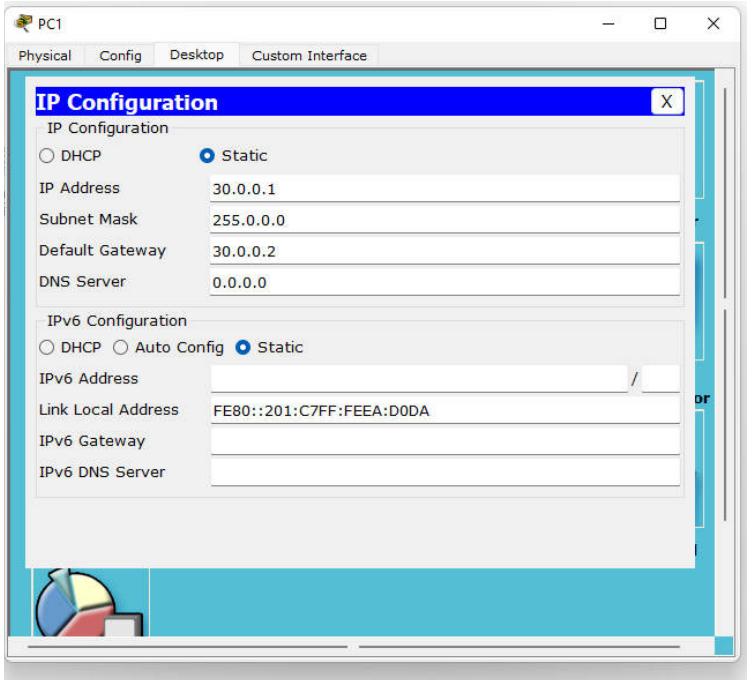
Configure Routers:

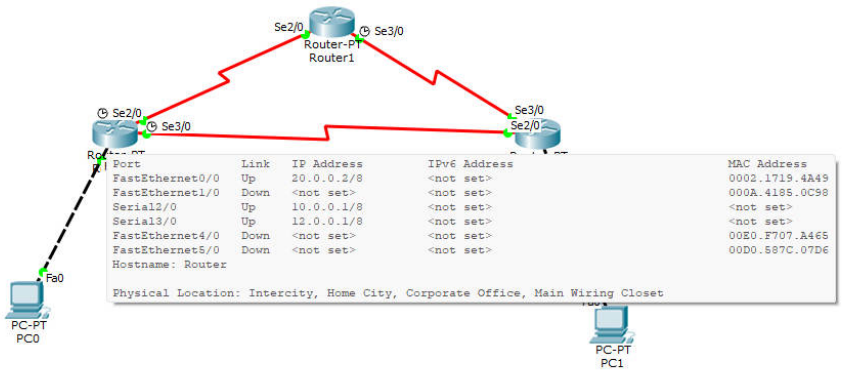


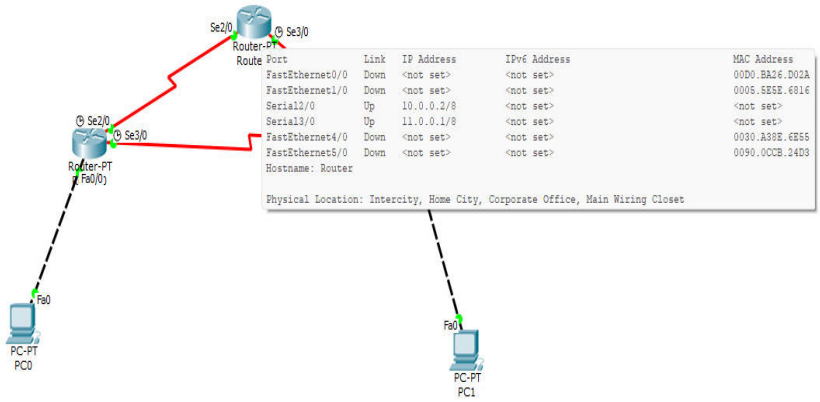


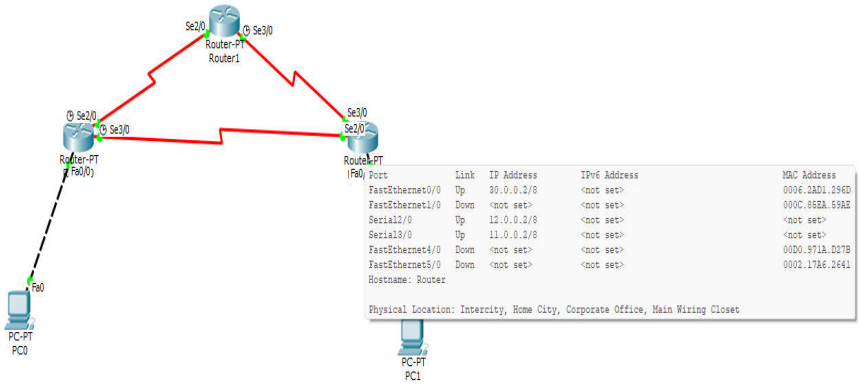

Router2
Physical
Config
CLI
IOS Command Line Interface
Router(config-if)#ip address 12.0.0.2 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
Router(config-if)#exit
Router(config)#interface Serial3/0
Router(config-if)#ip address 11.0.0.2 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
Router(config-if)#router ospf 1
Router(config-router)#network 30.0.0.0 0.0.0.255 area 0
Router(config-router)#network 11.0.0.0 0.255.255.255 area 0
Router(config-router)#network 12.0.0.0 0.255.255.255 area 0
Router(config-router)#ex
00:10:33: %OSPF-5-ADJCHG: Process 1, Nbr 11.0.0.1 on Serial3/0 from LOADING to FULL, Loading Done
it
Router(config)#
00:10:35: %OSPF-5-ADJCHG: Process 1, Nbr 20.0.0.2 on Serial2/0 from LOADING to FULL, Loading Done
Copy
Paste


PC0
Physical
Config
Desktop
Custom Interface
IP Configuration
X
IP Configuration
DHCP
Static
IP Address
20.0.0.1
Subnet Mask
255.0.0.0
Default Gateway
20.0.0.2
DNS Server
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::260:3EFF:FE26:9A7
IPv6 Gateway
IPv6 DNS Server

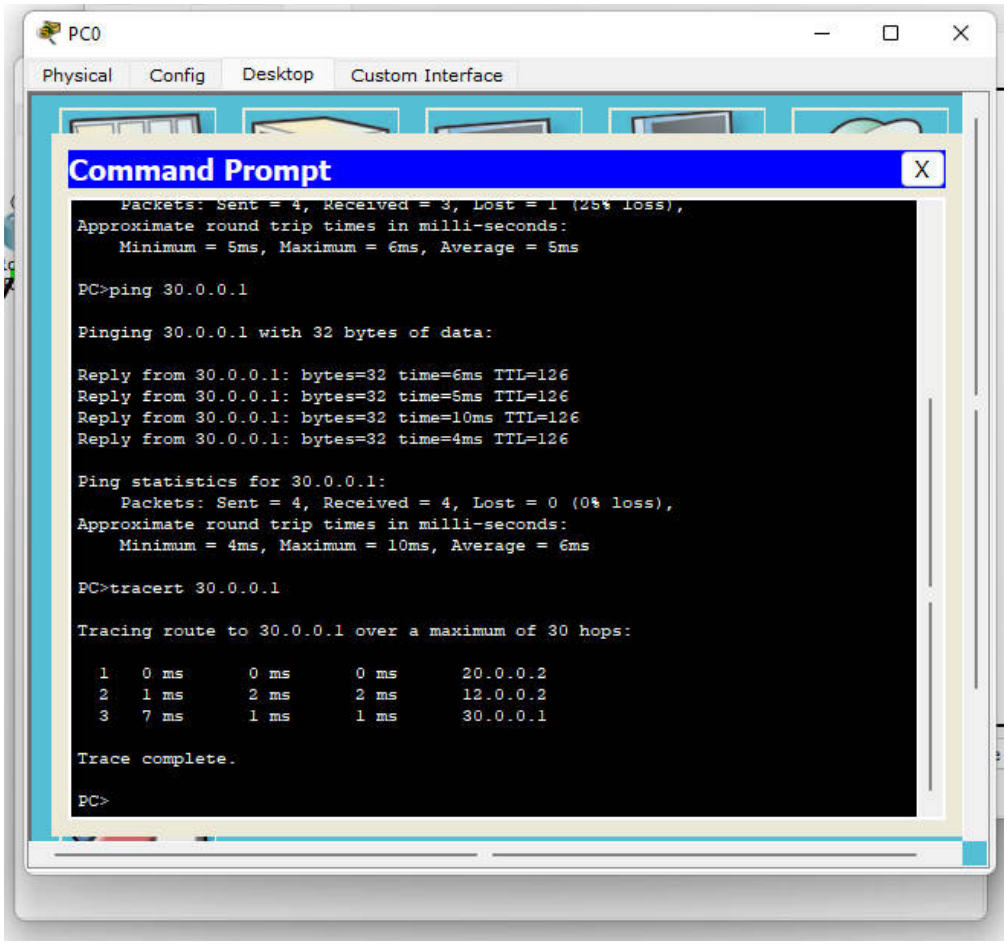

PC1
Physical
Config
Desktop
Custom Interface
IP Configuration
X
IP Configuration
DHCP
Static
IP Address
30.0.0.1
Subnet Mask
255.0.0.0
Default Gateway
30.0.0.2
DNS Server
0.0.0.0
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::201:C7FF:FEEA:D0DA
IPv6 Gateway
IPv6 DNS Server


Se2/0
Se3/0
Router-PT
Router1
Se2/0
Se3/0
Se3/0
Se2/0
Port Link IP Address IPv6 Address MAC Address
FastEthernet0/0 Up 20.0.0.2/8 <not set> 0002.1719.4A49
FastEthernet1/0 Down <not set> <not set> 000A.4185.0C98
Serial2/0 Up 10.0.0.1/8 <not set> <not set>
Serial3/0 Up 12.0.0.1/8 <not set> <not set>
FastEthernet4/0 Down <not set> <not set> 00E0.F707.A465
FastEthernet5/0 Down <not set> <not set> 00D0.587C.07D6
Hostname: Router
Physical Location: Intercity, Home City, Corporate Office, Main Wiring Closet
Fa0
PC-PT
PC0
PC-PT
PC1


Se2/0
Se3/0
Router-PT
Route
Se2/0
Se3/0
Router-PT
Fa0/0
Fa0
PC-PT
PC0
Fa0
PC-PT
PC1
Port Link IP Address IPv6 Address MAC Address
FastEthernet0/0 Down <not set> <not set> 00D0.BA26.D02A
FastEthernet1/0 Down <not set> <not set> 0005.5E5E.6816
Serial2/0 Up 10.0.0.2/8 <not set> <not set>
Serial3/0 Up 11.0.0.1/8 <not set> <not set>
FastEthernet4/0 Down <not set> <not set> 0030.A38E.6E55
FastEthernet5/0 Down <not set> <not set> 0090.0CCB.24D3
Hostname: Router
Physical Location: Intercity, Home City, Corporate Office, Main Wiring Closet


Se2/0
Se3/0
Router-PT
Router1
Se2/0
Se3/0
Router-PT
Fa0/0
Se3/0
Se2/0
Router-PT
Fa0
Fa0
PC-PT
PC0
PC-PT
PC1
Port Link IP Address IPv6 Address MAC Address
FastEthernet0/0 Up 30.0.0.2/8 <not set> 0006.2AD1.296D
FastEthernet1/0 Down <not set> <not set> 000C.85EA.59AE
Serial2/0 Up 12.0.0.2/8 <not set> <not set>
Serial3/0 Up 11.0.0.2/8 <not set> <not set>
FastEthernet4/0 Down <not set> <not set> 00D0.971A.D27B
FastEthernet5/0 Down <not set> <not set> 0002.17A6.2641
Hostname: Router
Physical Location: Intercity, Home City, Corporate Office, Main Wiring Closet

Output:


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 5ms, Maximum = 6ms, Average = 5ms
PC>ping 30.0.0.1
Pinging 30.0.0.1 with 32 bytes of data:
Reply from 30.0.0.1: bytes=32 time=6ms TTL=126
Reply from 30.0.0.1: bytes=32 time=5ms TTL=126
Reply from 30.0.0.1: bytes=32 time=10ms TTL=126
Reply from 30.0.0.1: bytes=32 time=4ms TTL=126
Ping statistics for 30.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 4ms, Maximum = 10ms, Average = 6ms
PC>tracert 30.0.0.1
Tracing route to 30.0.0.1 over a maximum of 30 hops:
1 0 ms 0 ms 0 ms 20.0.0.2
2 1 ms 2 ms 2 ms 12.0.0.2
3 7 ms 1 ms 1 ms 30.0.0.1
Trace complete.
PC>

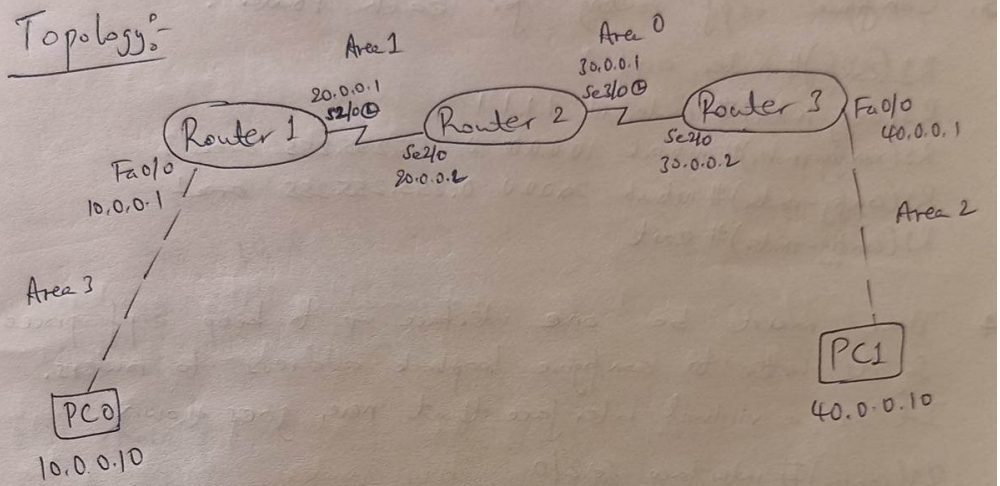
Friday

4th August, 2023.

⑦ Title:
Configure OSPF routing protocol.

Aim:
To understand ~~not~~ how to configure OSPF routing protocol and to find

Topology:-



Procedure:-

1. Create a topology as shown above
2. Configure each router with the IP addresses for required interfaces. Set up clock rate for interfaces having ⊖ symbol.

Router 2

```
R2(config)# interface Serial 3/0
R2(config-if)# ip address 30.0.0.1 255.0.0.0
R2(config-if)# encapsulation PPP
R2(config-if)# clock rate 64000
R2(config-if)# no shutdown
R2(config-if)# exit
```

R2(config)#interface se2/0
R2(config-if)# ip address 20.0.0.2 255.0.0.0
R2(config-if)#encapsulation ppp
R2(config-if)# no shutdown
R2(config-if)# exit

3. Configure ospf routing for each router.

R1(config)# router ospf 1
R1(config-router)# router-id 1.1.1.1
R1(config-router)# network 10.0.0.0 0.255.255.255 area 3
R1(config-router)# network 20.0.0.0 0.255.255.255 area 1
R1(config-router)# exit

4. There must be one interface up to keep ospf process up. So its better to configure loopback address to routers. It is a virtual interface that never goes down.

R1(config)# interface se2/0
R1(config-if)# interface loopback 0
R1(config-if)# ip add 172.16.1.252 255.255.0.0
R1(config-if)# no shutdown

This is done for each router.

5. Still R3 doesn't know about Area 3. This can be verified using "show ip route" command.

R3# show ip route
O IA 20.0.0.0/8 via 30.0.0.1, se1/0
C 40.0.0.0/8 directly connected, Fa2/0
C 30.0.0.0/8 directly connected, se1/0

So a virtual link b/w R1 & R2 must be created, to connect area 3 to area 0.

R1#

R1(config)# router ospf 1

R1(config-router)# area 1 virtual-link 2.2.2.2

R1(config-router)# exit.

Do the same for R2.

6. Ping PC0 from PC1 to check connectivity.

Result:

R3# show ip route

O IA 20.0.0.0/8 via 30.0.0.1, Se1/0

O IA 10.0.0.0/8 via 30.0.0.1, Se1/0

C 40.0.0.0/8 directly connected, Fa2/0

C 30.0.0.0/8 directly connected, Se1/0

R2# show ip route

O IA

C 20.0.0.0/8 directly connected, Se2/0

C 30.0.0.0/8 directly connected, Se3/0

C 172.16.0.0/16 directly connected, Loopback0

R1# show ip route

O IA 40.0.0.0/8 via 20.0.0.2, Se2/0

O 30.0.0.0/8 via 20.0.0.2, Se2/0

C 10.0.0.0/8 directly connected, Fa0/0

C 20.0.0.0/8 directly connected, Se2/0

Ping PC0 from PC1

PC1 > ping 10.0.0.10

Reply from 10.0.0.10, bytes=32 time=13ms TTL=125
Reply from 10.0.0.10, bytes=32 time=11ms TTL=125
Reply from 10.0.0.10, bytes=32 time=6ms TTL=125
Reply from 10.0.0.10, bytes=32 time=7ms TTL=125

Ping statistics:
Packets: Sent=4, Recieved=4, Lost=0(0% Loss)
Min=6ms, MAX=13ms, Avg=9ms

Observations

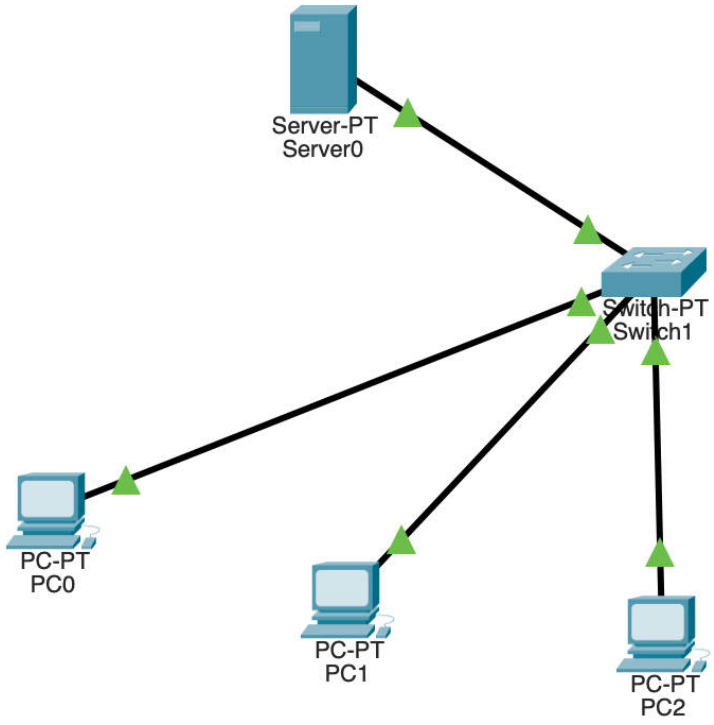
OSPF protocol enables the routers to distribute IP routing information throughout the network using the shortest path, A loopback address is needed for the routers to keep the ospf process up all the time. A virtual link is needed for some routers that doesn't know the other areas in the topology.

Experiment No. 8

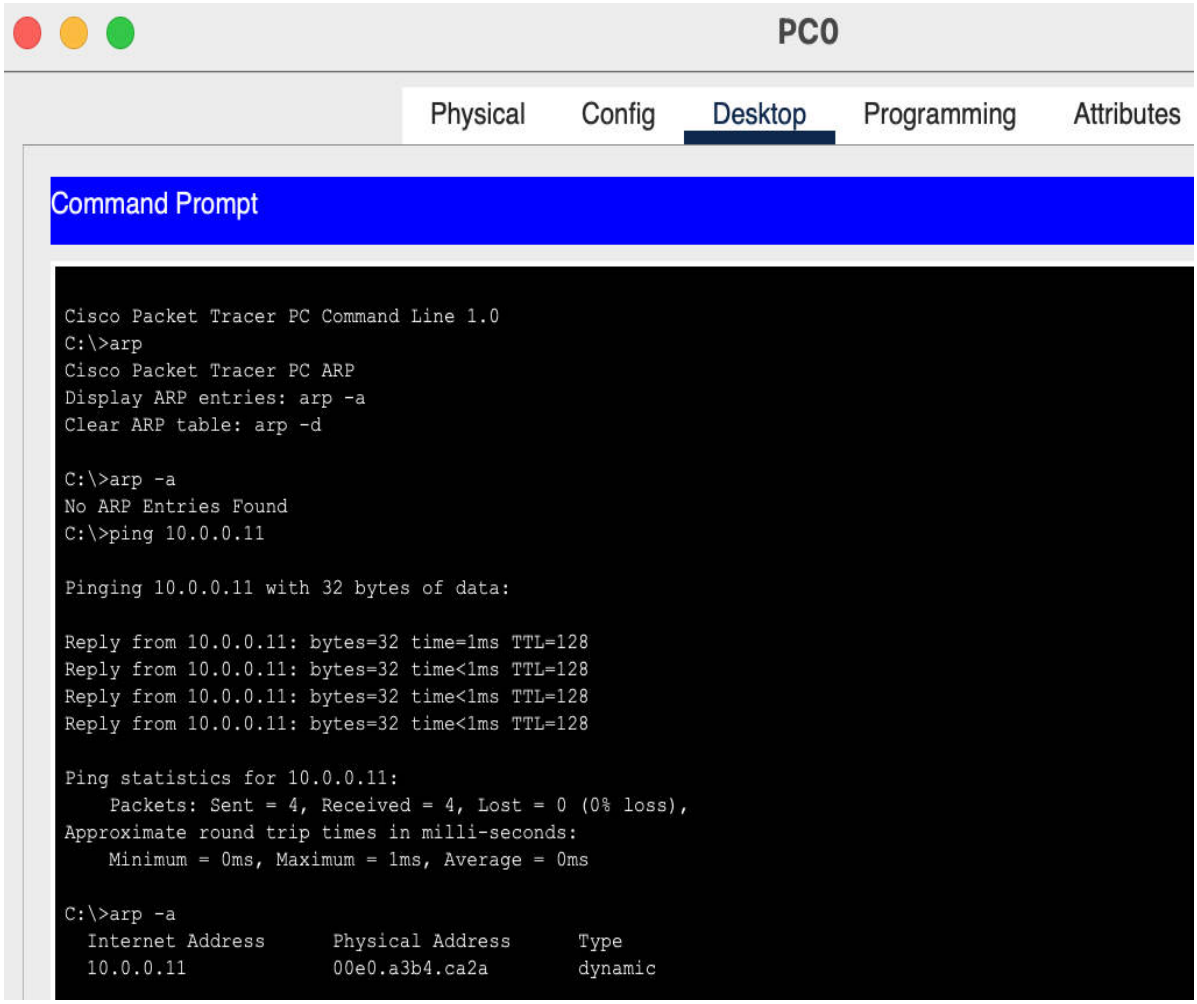
Title:

To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)

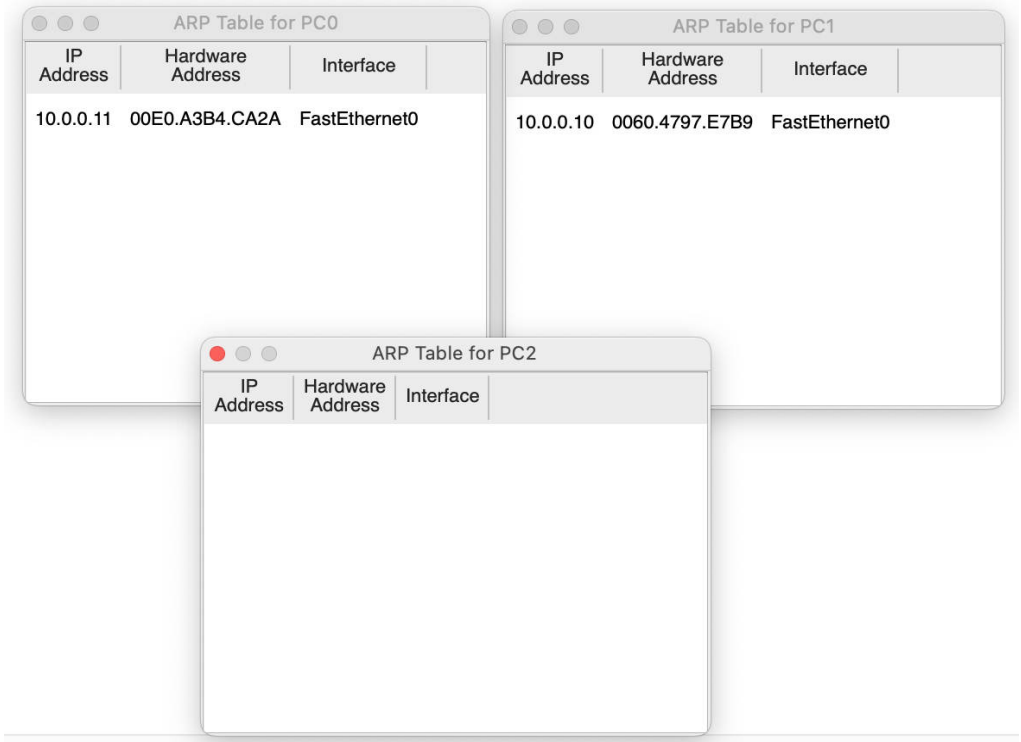
Topology:



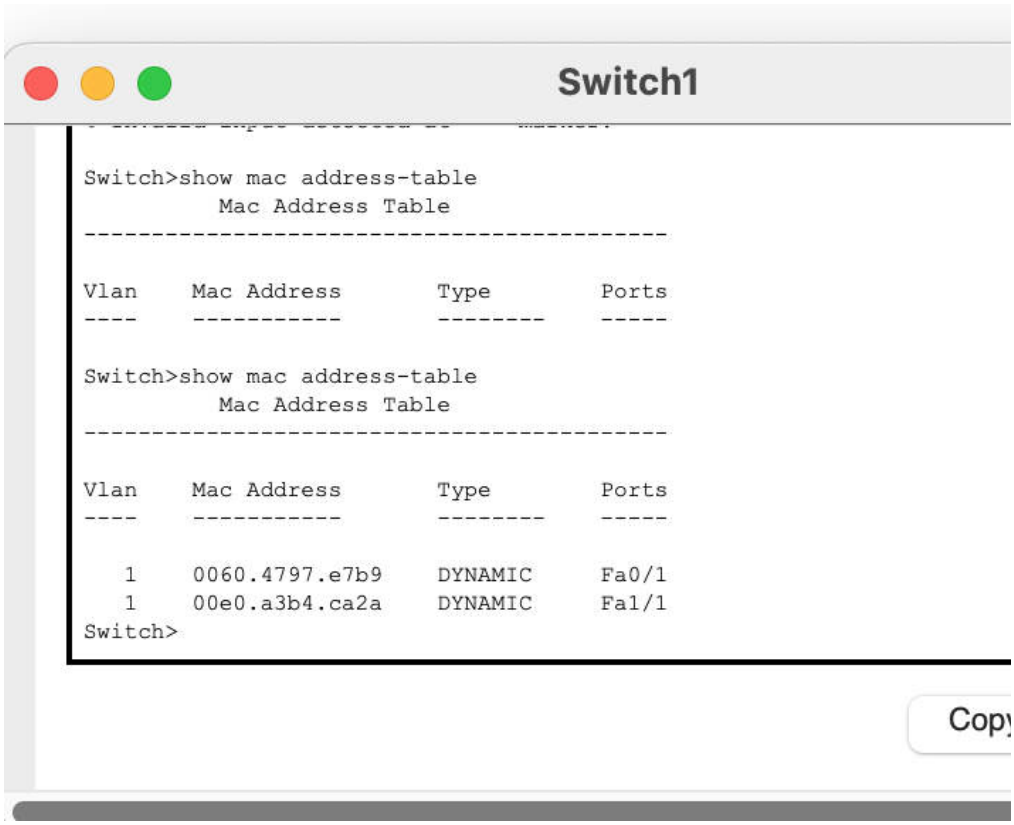
Ping PC1 from PC0:



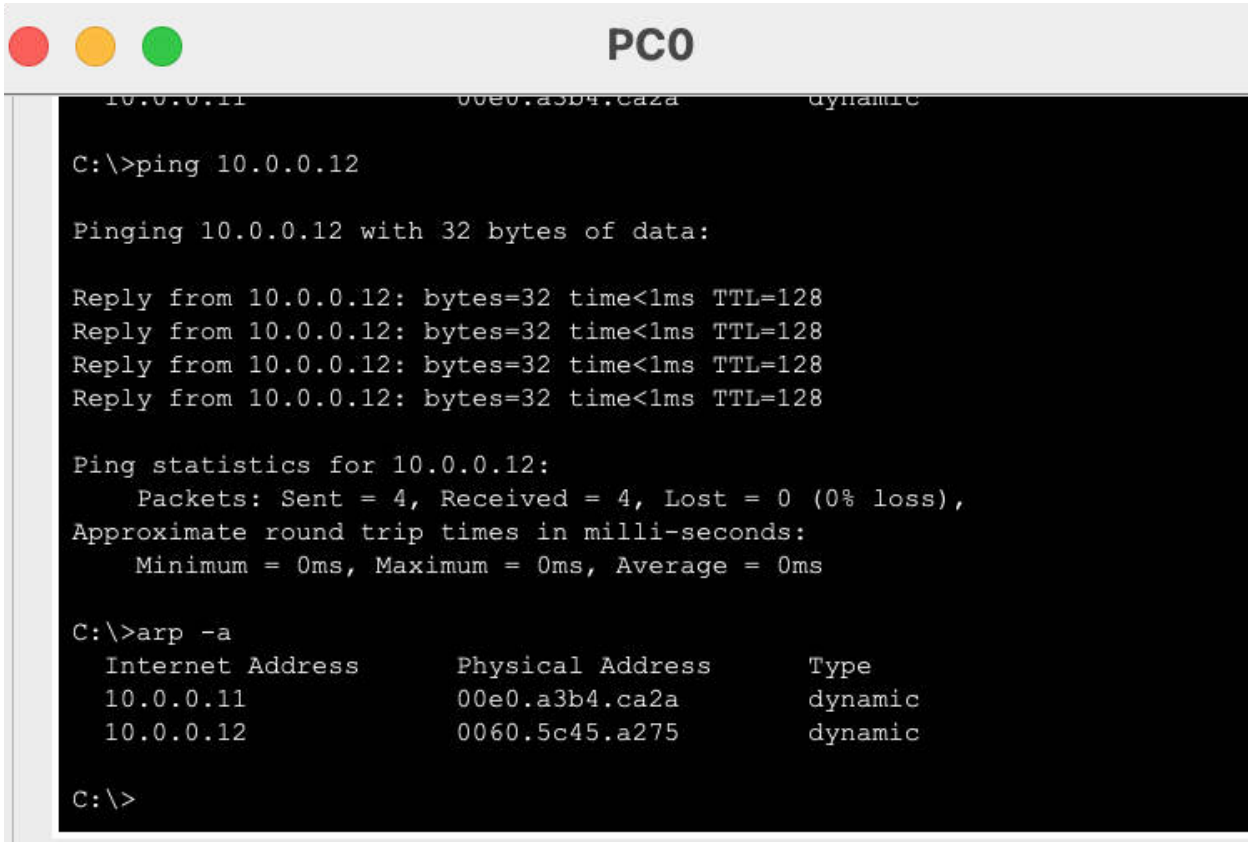
ARP table for all PCs:

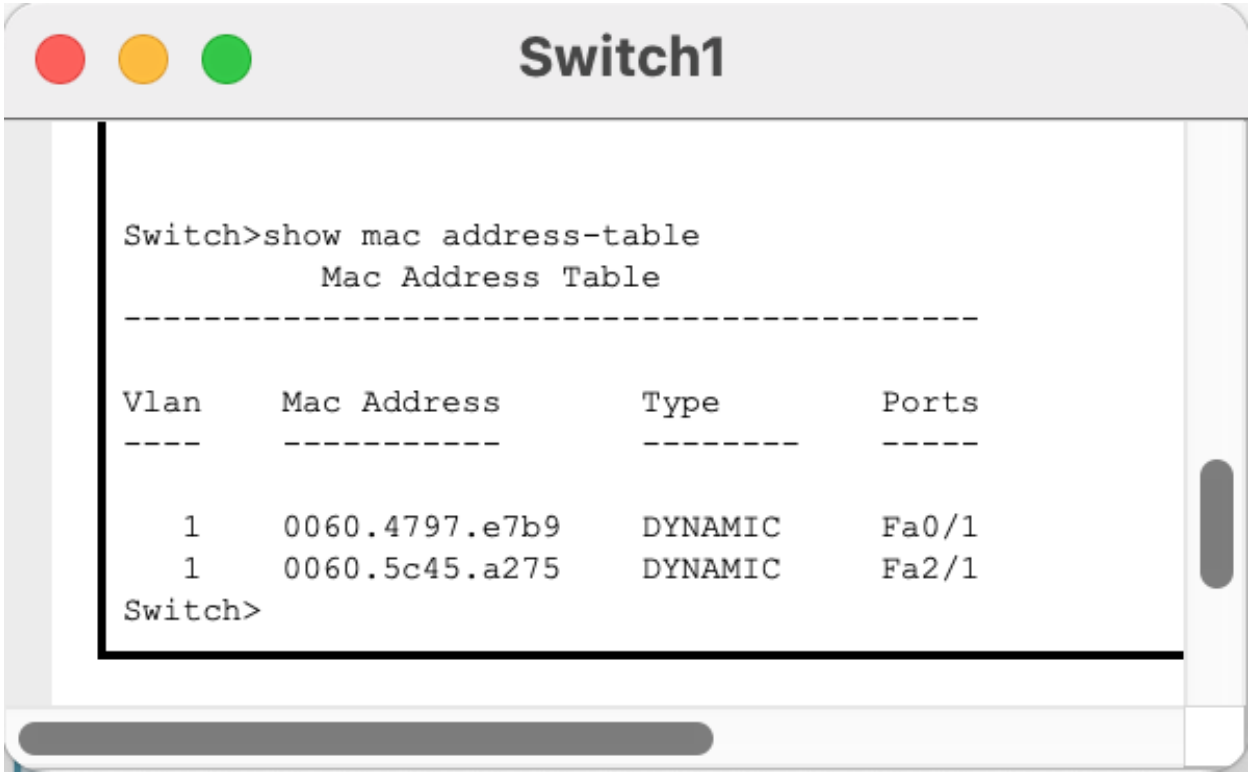


Mac Address Table:



Ping PC2 form PC0:





11th August, 2023.

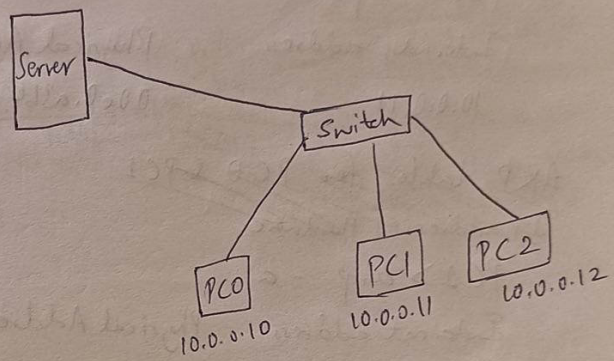
(8) Title:

To construct simple LAN and understand the concept & operation of Address Resolution Protocol (ARP).

Aim:

Understand the concept & operation of ARP

Topology:



Procedure:

1. Create topology as shown above.
2. Configure IP address for each end devices.
3. Use the "inspect tool" to see ARP table and mac-address table (or) use the following command
   PC> arp -a [ARP table]
   PC> show mac address-table [mac address table]
4. Go to simulation mode to send packets between end devices. Use "capture" button to go step by step to observe changes in ARP table, as & when new communication starts.

Results:-

Ping send packet from PC0 to PC1

Before sending packet-

PC0 > arp -a

No ARP entries found

After sending packet-

PC0 > arp -a

Internet address	Physical Address	Type
10.0.0.11	00e0.a3b4.ca2a	dynamic

~~ARP table for PC0 to PC1~~

~~IP address Hardware~~

PC1 > arp -a

Internet address	Physical Address	Type
10.0.0.10	0060.4797.e7b9	dynamic

MacAddress Table in Switch

Switch > show mac address-table

Vlan	Mac Address	Type	Ports
1	0060.4797.e7b9	DYNAMIC	Fa0/1
1	00e0.e3b4.ca2a	DYNAMIC	Fa0/1

Observation:-

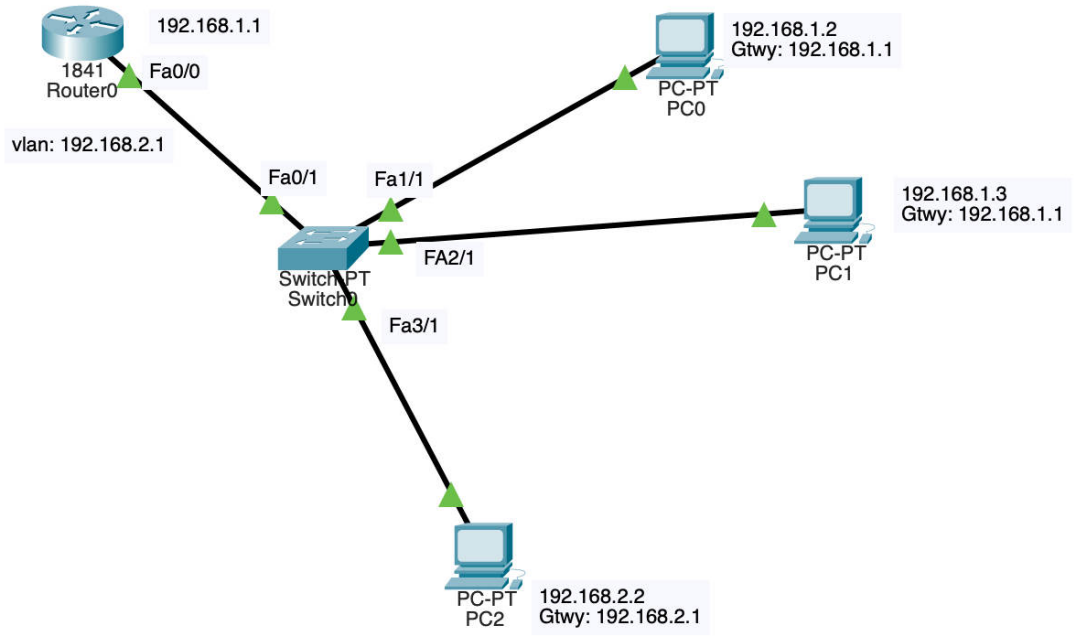
Initially the ARP table is empty, after the host encapsulates a packet into a frame, it is reflected in the MAC address table to determine the mapping of IP address to MAC address.

Experiment No. 9

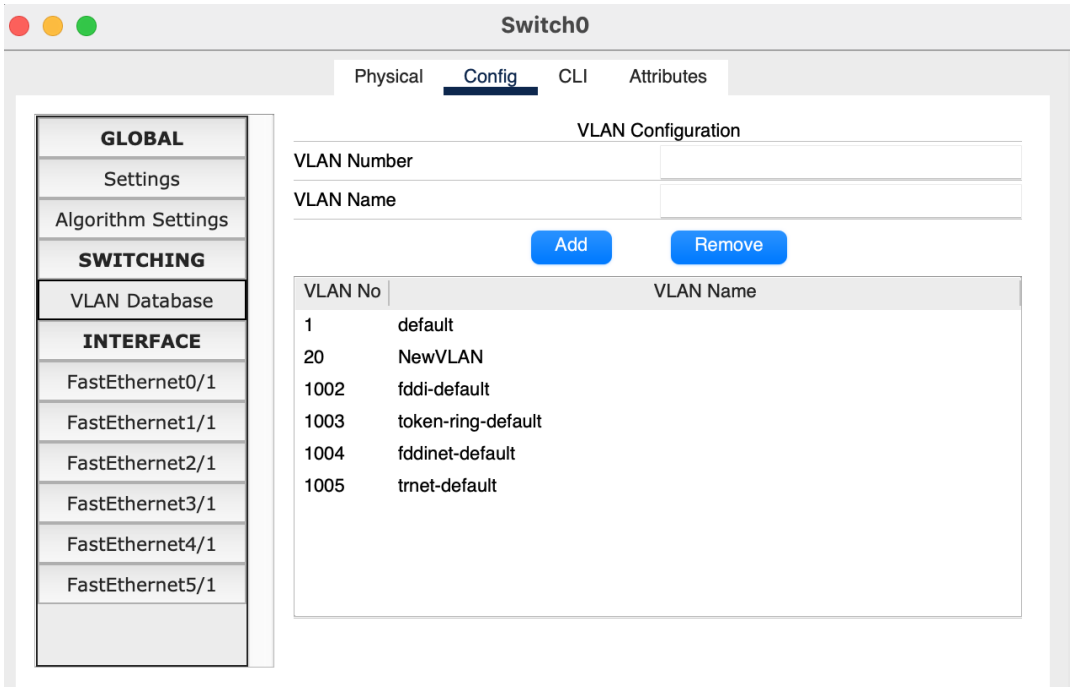
Title:

To construct a VLAN and make the PC's communicate among a VLAN

Topology:



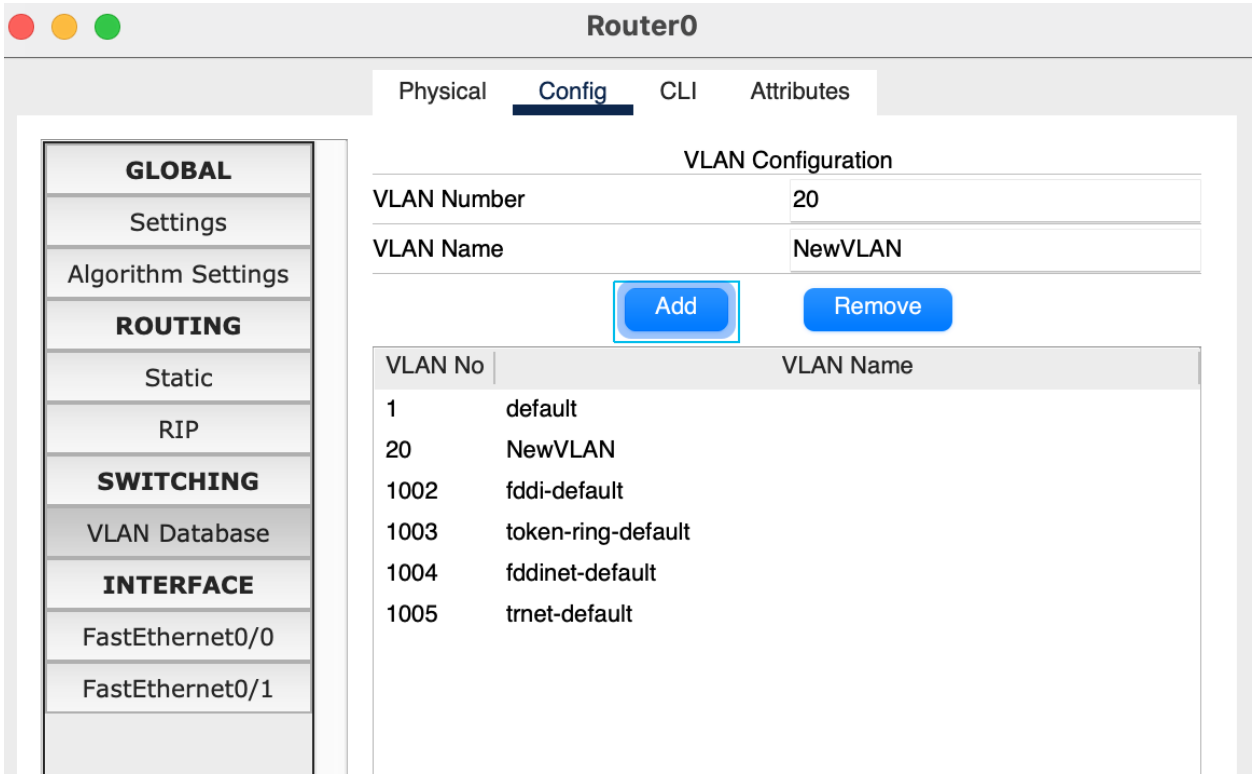
Create VLAN:

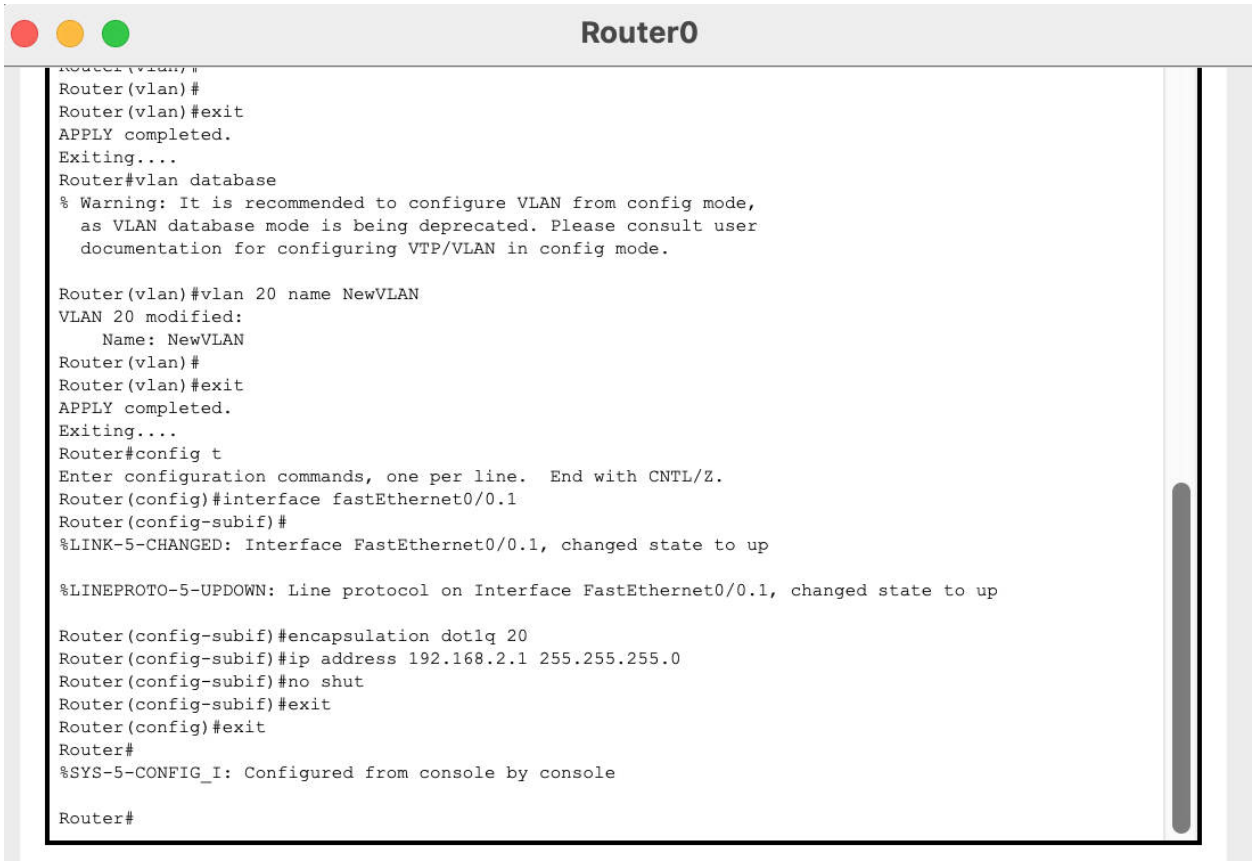


Trunking:

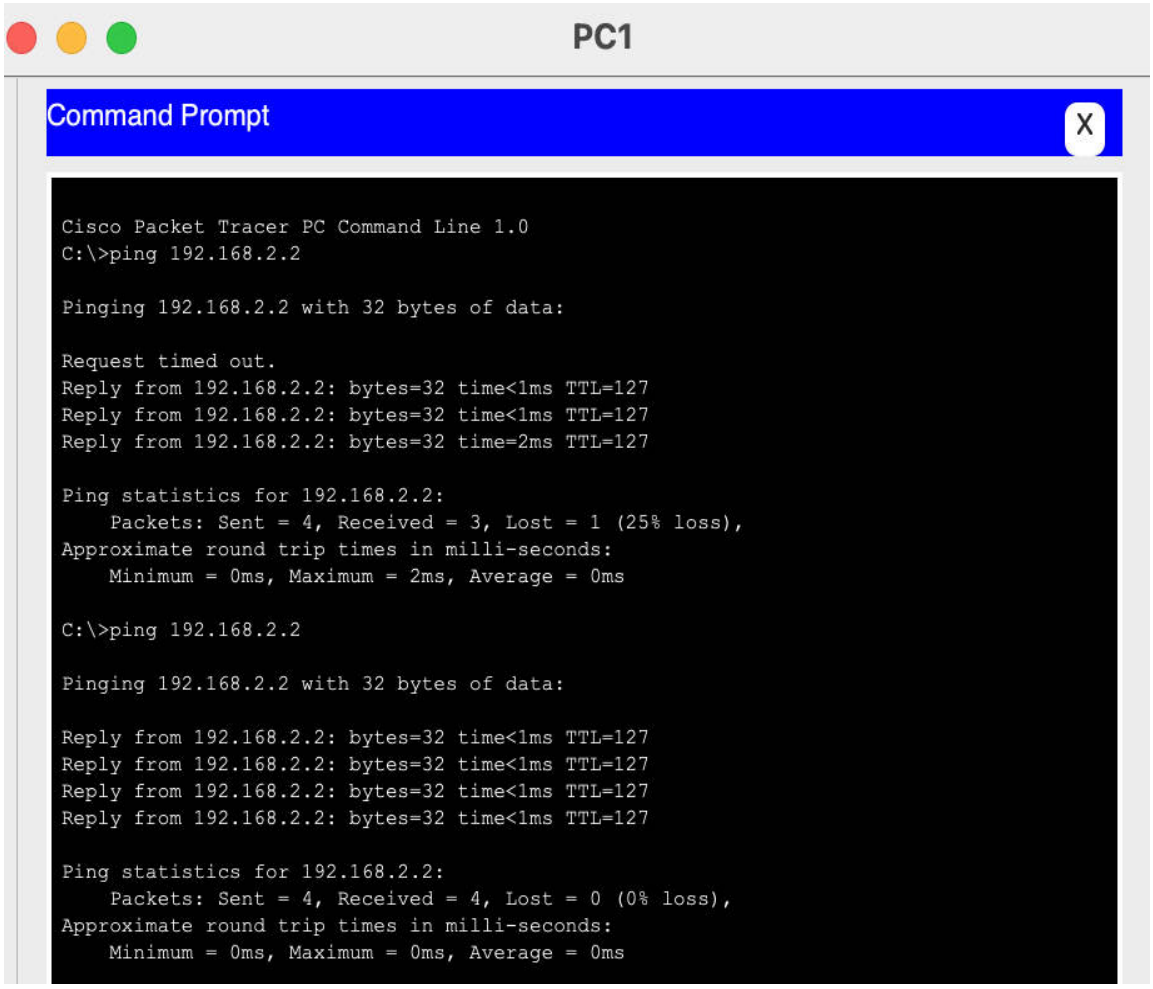
Add end devices to VLAN:

Router config:

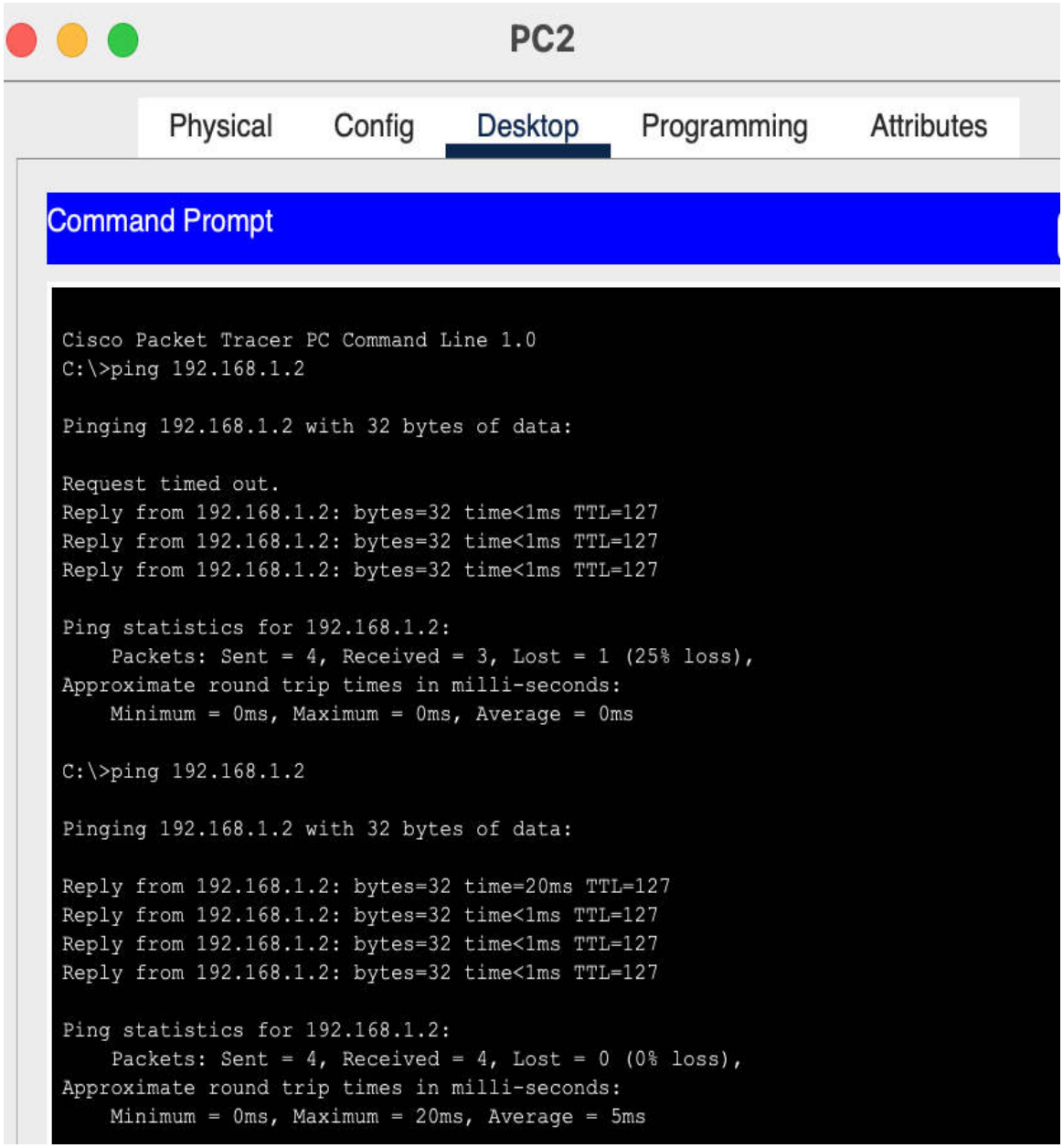




Pinging PC2(in VLAN) from PC1:



Pinging PC0 from PC2(in VLAN):



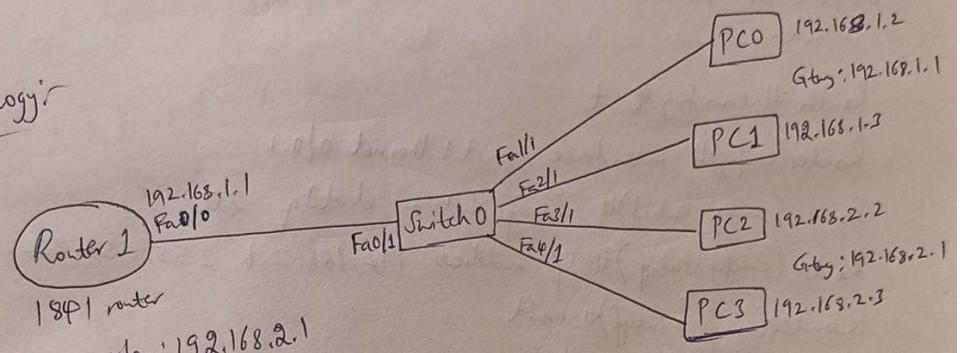
⑨ Title:-

To construct a VLAN & make the PCs communicate among VLAN.

Aim:-

Understand how to construct a VLAN.

Topology:-



Procedure:-

1. Create a topology as shown above. Use Router 1841
2. Click on switch 0. Under Config tab choose VLAN database
3. Create a VLAN by typing VLAN no. & VLANname. Then click on add.
   VLAN number: 20
   VLAN name: NewVLAN
4. Select the interface of the switch which connects it to router, here Fa 0/1 & make it Trunk
5. ~~For PC2 & PC3~~
   In the interfaces of the switch connecting to PC2 & PC3 [end devices of NewVLAN] select 20: NewVLAN under VLAN. This makes the switch understand PC2 & PC3 are in NewVLAN.

6. Click on Router. Under Config tab select VLAN Database. Add the newly created VLAN by entering its number & name.

7. Open CLI in the router. Enter the following commands.

```
Router(vlan)#exit
APPLY completed
Exiting....
Router#config t
Router(config)# interface fastEthernet 0/0.1
Router(config-subif)# encapsulation dot1q 20
Router(config-subif)# ip address 192.168.2.1 255.255.255.0
Router(config-subif)# exit
Router(config)# exit
```

8. Ping the end devices using end devices in ~~differ~~ VLAN & physical LAN to check connection.

Result:

Ping PC2 (VLAN) from PC1

```
PC1>Ping 192.168.2.2

Reply from 192.168.2.2, bytes=32, time<1ms TTL=127
Reply from 192.168.2.2, bytes=32 time<1ms TTL=127
Reply from 192.168.2.2, bytes=32, time<1ms TTL=127
Reply from 192.168.2.2 bytes=32, time<1ms TTL=127

Packets: Sent=4, Recieved=4, Lost=0 (0% Loss)
Round trip time in ms
    Min=0 Max=0 Avg=0
```

Ping PC0 from PC2(VLAN)

PC2 > ping 192.168.1.2

Reply from 192.168.1.2 bytes=32 time=20ms TTL=127
Reply from 192.168.1.2 bytes=32 time<1ms TTL=127
Reply from 192.168.1.2 bytes=32 time<1ms TTL=127
Reply from 192.168.1.2 bytes=32 time<1ms TTL=127

Packets: Sent = 4, Recieved = 4, Loss = 0 (0% loss)

Approx. Round Trip Time

Min = 0 Max = 20 Avg = 5

Observation:-

A virtual lan can be created by specifying the VLAN no. & VLAN name in the switch and the same should be added to the router to make the router identify the newly created VLAN. The physical & VLAN & virtual LAN can communicate between each other even though both have different gateways.

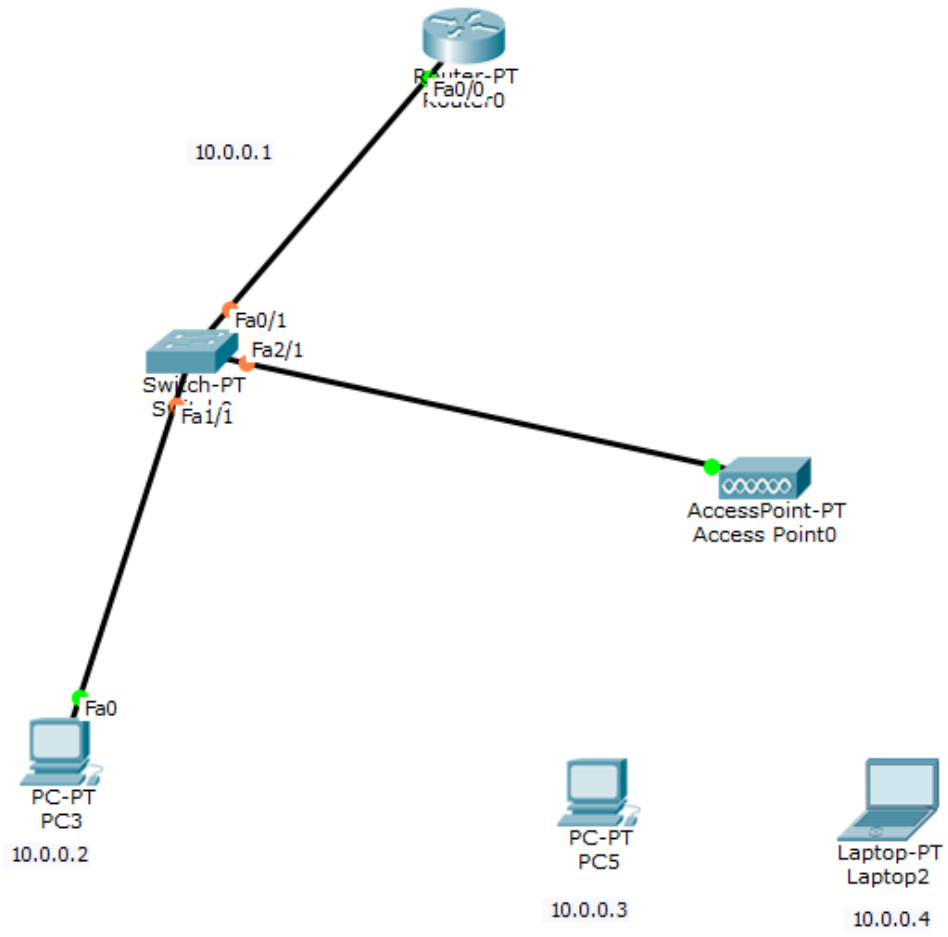
9/8/23

Experiment No. 10

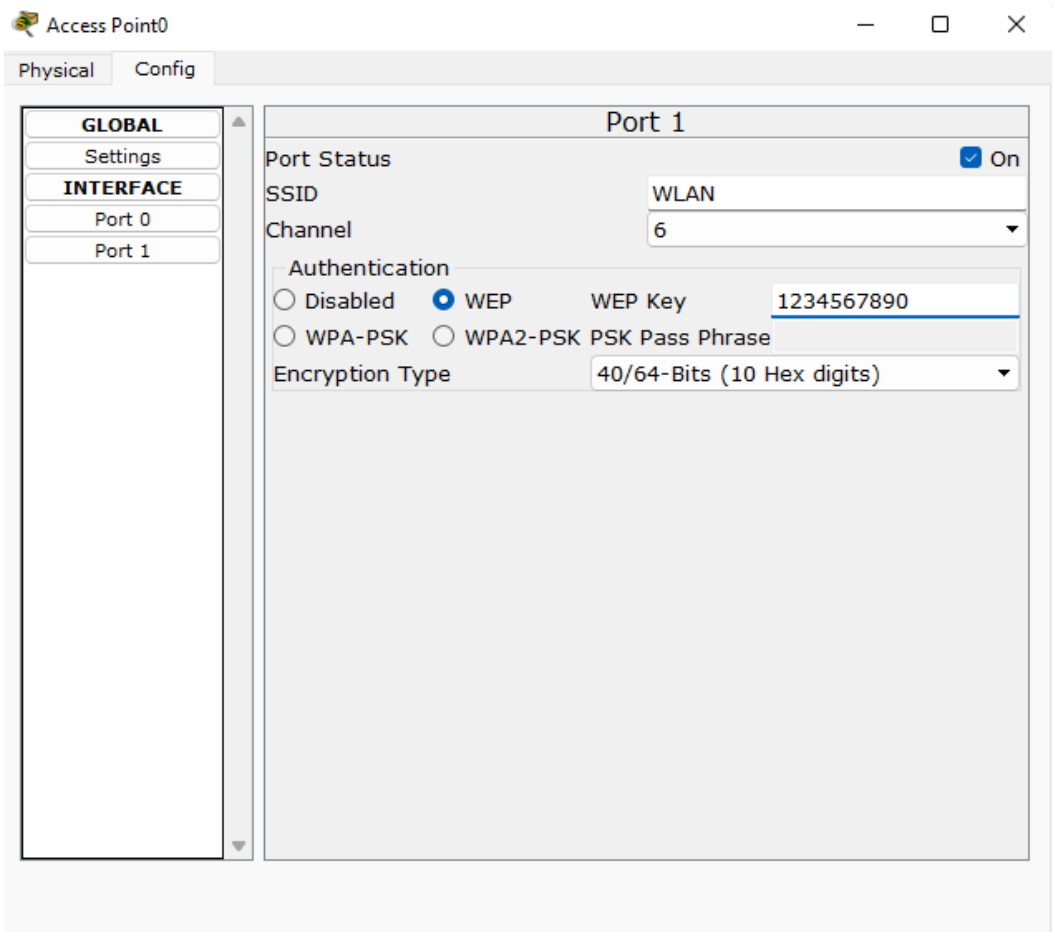
Title:

To construct a WLAN and make the nodes communicate wirelessly

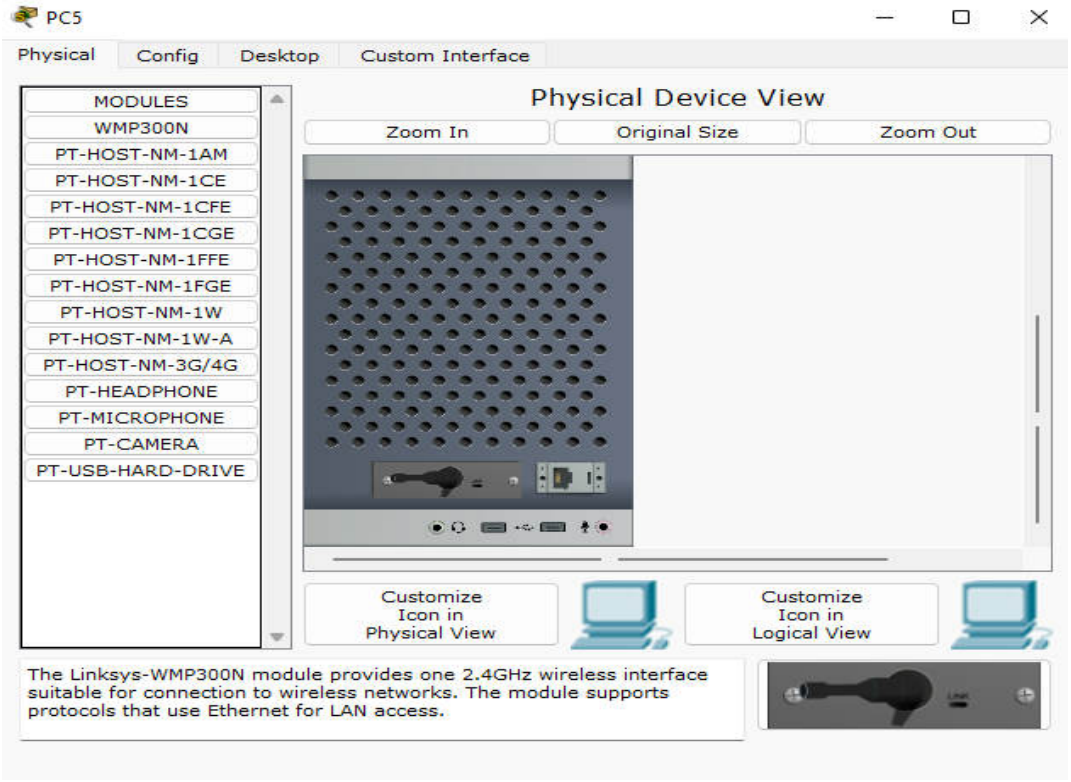
Topology:

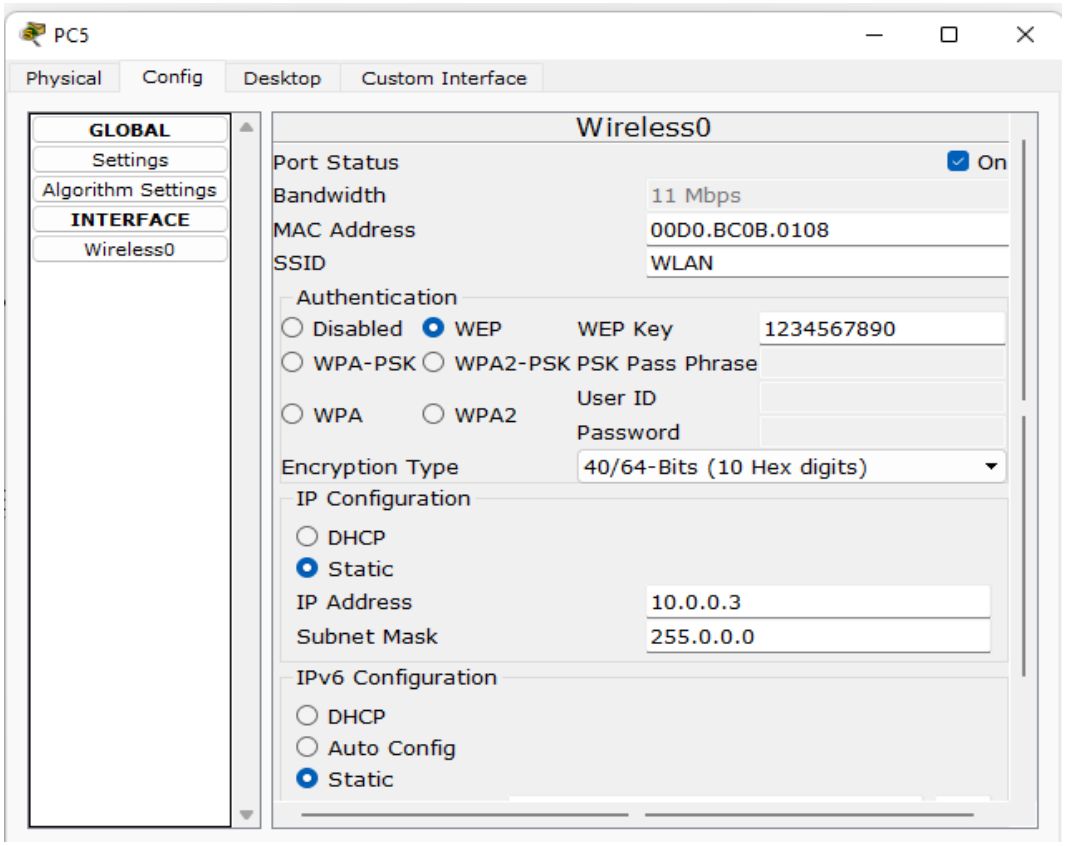


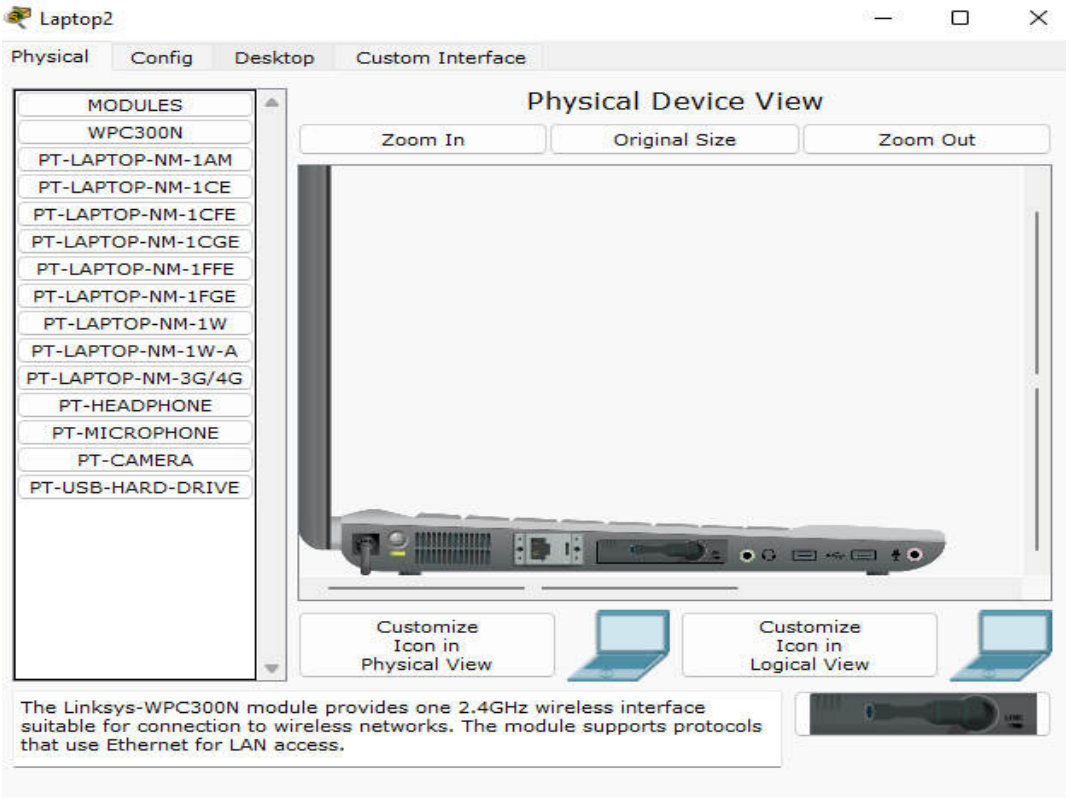
AccessPoint config:

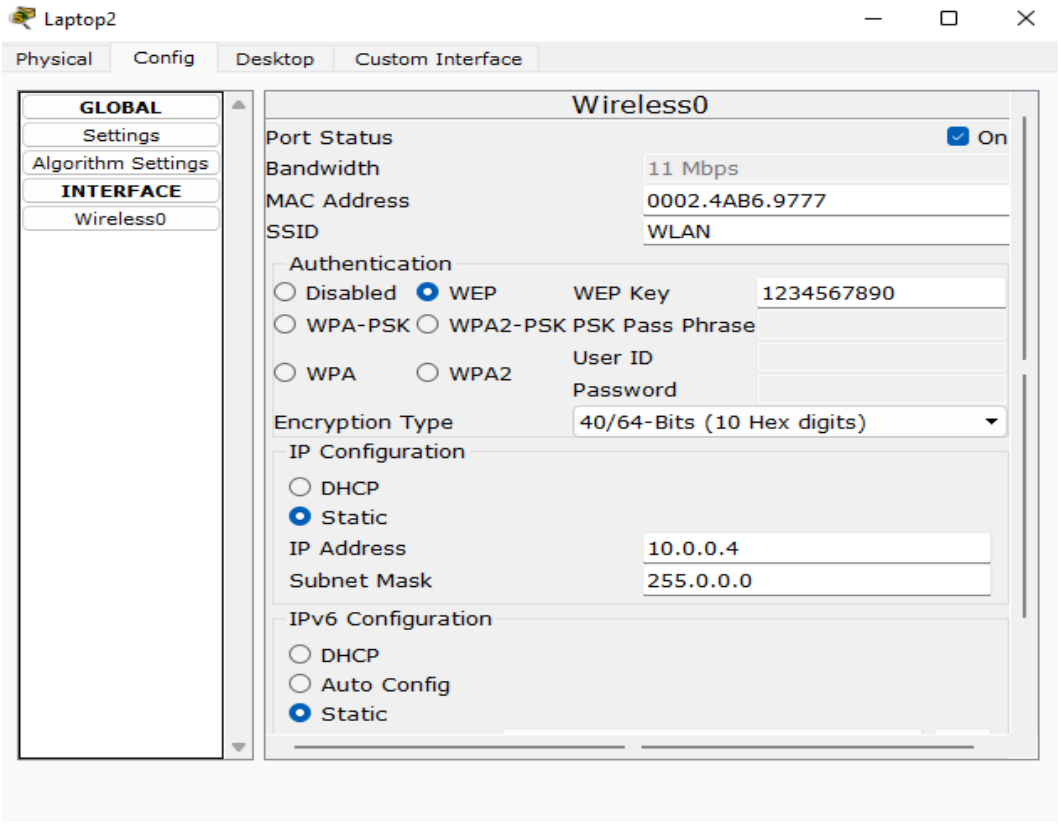


Configure wireless nodes:

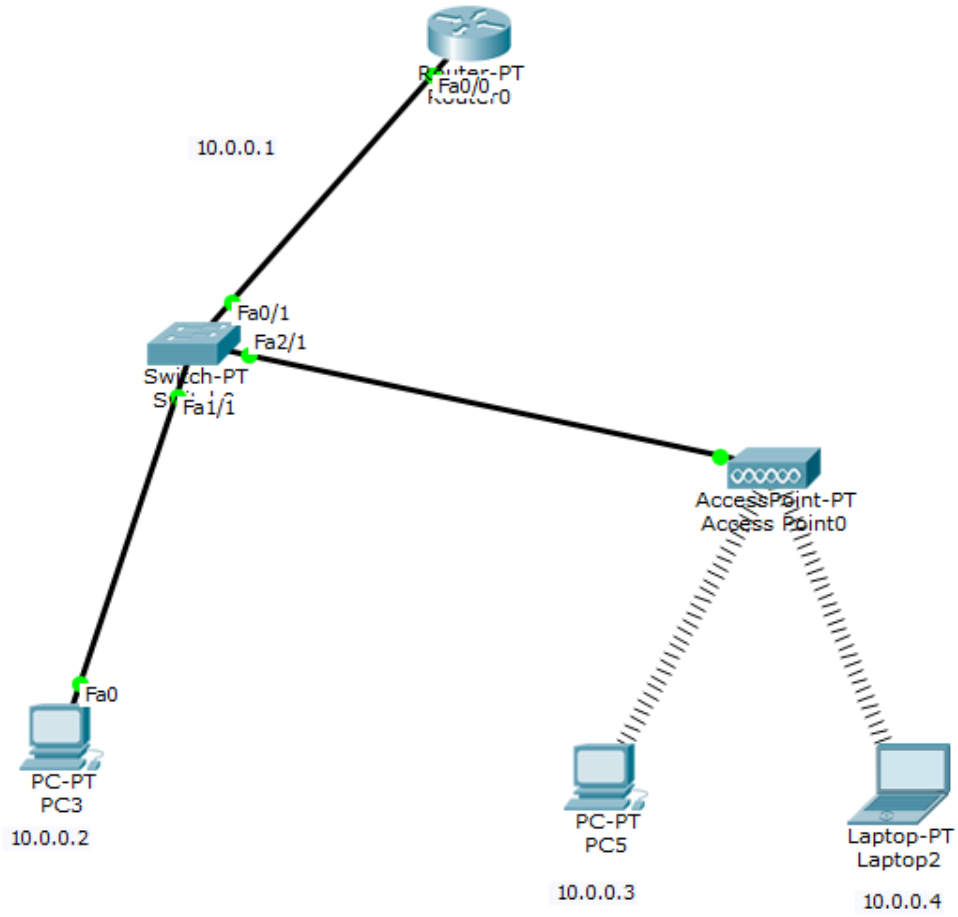



PC5
Physical
Config
Desktop
Custom Interface
GLOBAL
Settings
Algorithm Settings
INTERFACE
Wireless0
Wireless0
Port Status
On
Bandwidth
11 Mbps
MAC Address
00D0.BC0B.0108
SSID
WLAN
Authentication
Disabled
WEP
WEP Key
1234567890
WPA-PSK
WPA2-PSK
PSK Pass Phrase
WPA
WPA2
User ID
Password
Encryption Type
40/64-Bits (10 Hex digits)
IP Configuration
DHCP
Static
IP Address
10.0.0.3
Subnet Mask
255.0.0.0
IPv6 Configuration
DHCP
Auto Config
Static

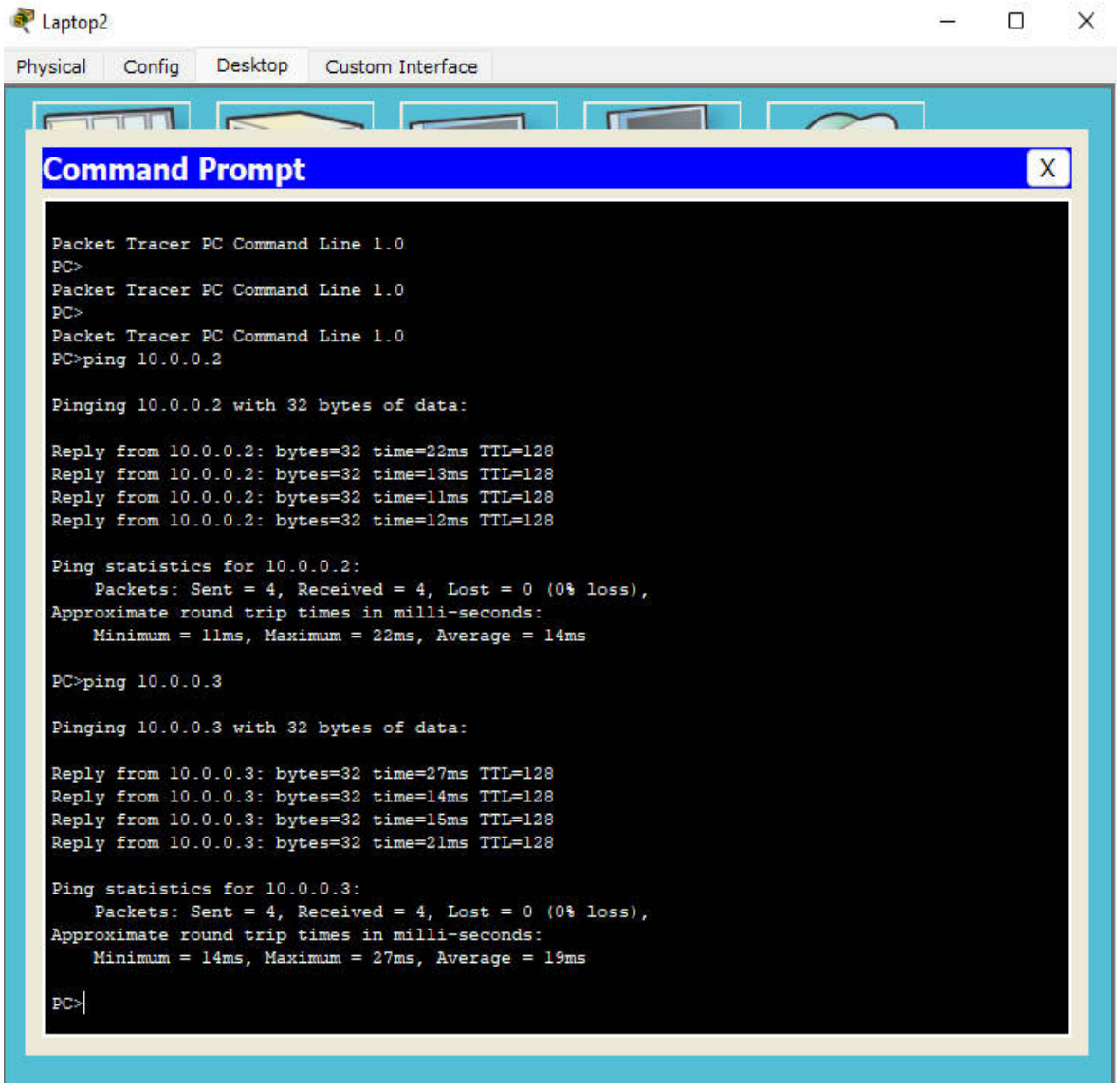

Laptop2
Physical
Config
Desktop
Custom Interface
MODULES
WPC300N
PT-LAPTOP-NM-1AM
PT-LAPTOP-NM-1CE
PT-LAPTOP-NM-1CFE
PT-LAPTOP-NM-1CGE
PT-LAPTOP-NM-1FFE
PT-LAPTOP-NM-1FGE
PT-LAPTOP-NM-1W
PT-LAPTOP-NM-1W-A
PT-LAPTOP-NM-3G/4G
PT-HEADPHONE
PT-MICROPHONE
PT-CAMERA
PT-USB-HARD-DRIVE
Physical Device View
Zoom In
Original Size
Zoom Out
Customize Icon in Physical View
Customize Icon in Logical View
The Linksys-WPC300N module provides one 2.4GHz wireless interface suitable for connection to wireless networks. The module supports protocols that use Ethernet for LAN access.

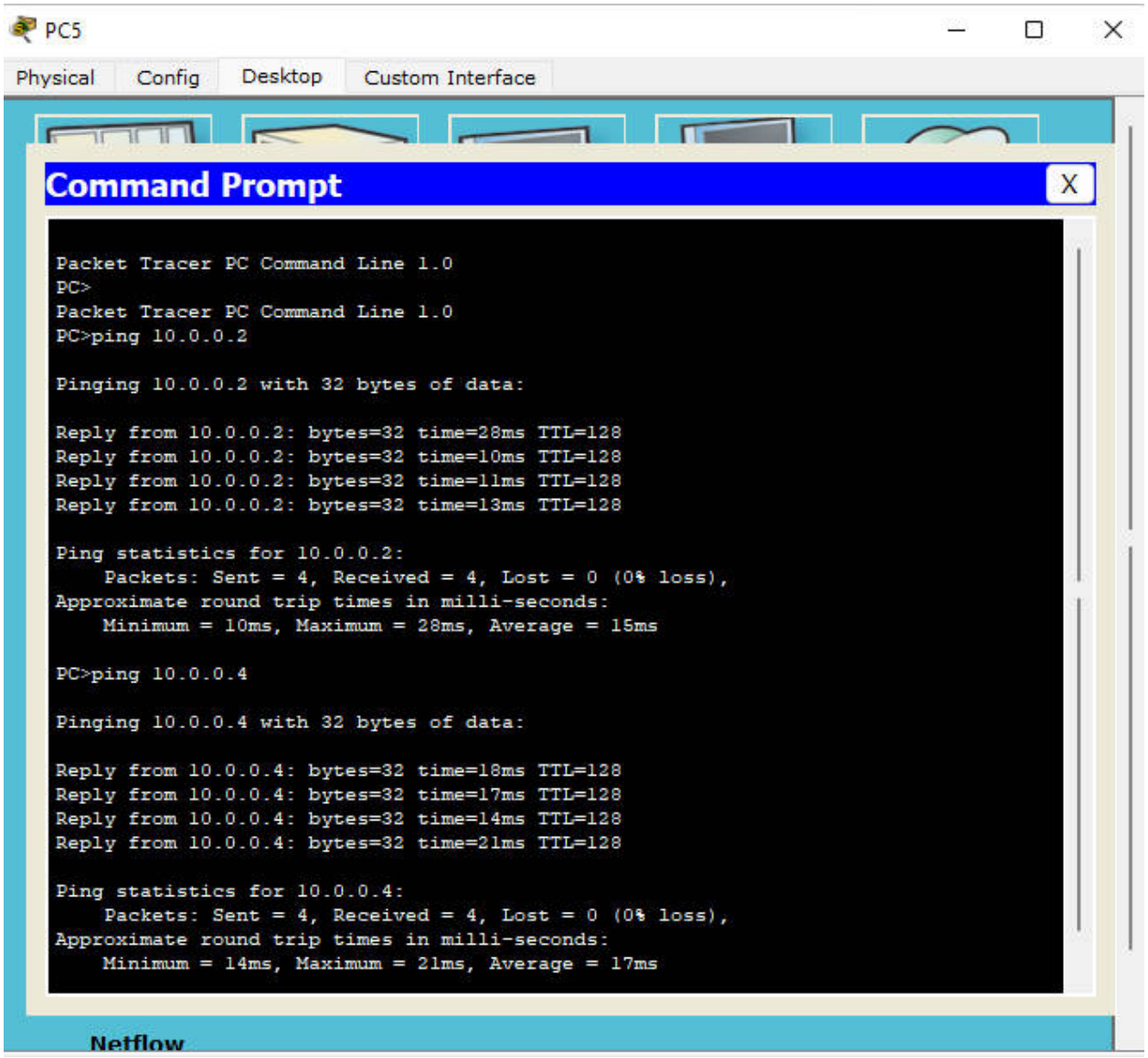

Laptop2
Physical
Config
Desktop
Custom Interface
GLOBAL
Settings
Algorithm Settings
INTERFACE
Wireless0
Wireless0
Port Status
On
Bandwidth
11 Mbps
MAC Address
0002.4AB6.9777
SSID
WLAN
Authentication
Disabled
WEP
WEP Key
1234567890
WPA-PSK
WPA2-PSK
PSK Pass Phrase
WPA
WPA2
User ID
Password
Encryption Type
40/64-Bits (10 Hex digits)
IP Configuration
DHCP
Static
IP Address
10.0.0.4
Subnet Mask
255.0.0.0
IPv6 Configuration
DHCP
Auto Config
Static

Final Topology:



Pinging end devices:




PC5
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.2
Pinging 10.0.0.2 with 32 bytes of data:
Reply from 10.0.0.2: bytes=32 time=28ms TTL=128
Reply from 10.0.0.2: bytes=32 time=10ms TTL=128
Reply from 10.0.0.2: bytes=32 time=11ms TTL=128
Reply from 10.0.0.2: bytes=32 time=13ms TTL=128
Ping statistics for 10.0.0.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 10ms, Maximum = 28ms, Average = 15ms
PC>ping 10.0.0.4
Pinging 10.0.0.4 with 32 bytes of data:
Reply from 10.0.0.4: bytes=32 time=18ms TTL=128
Reply from 10.0.0.4: bytes=32 time=17ms TTL=128
Reply from 10.0.0.4: bytes=32 time=14ms TTL=128
Reply from 10.0.0.4: bytes=32 time=21ms TTL=128
Ping statistics for 10.0.0.4:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 14ms, Maximum = 21ms, Average = 17ms
Netflow

18th August, 2022.

(10)

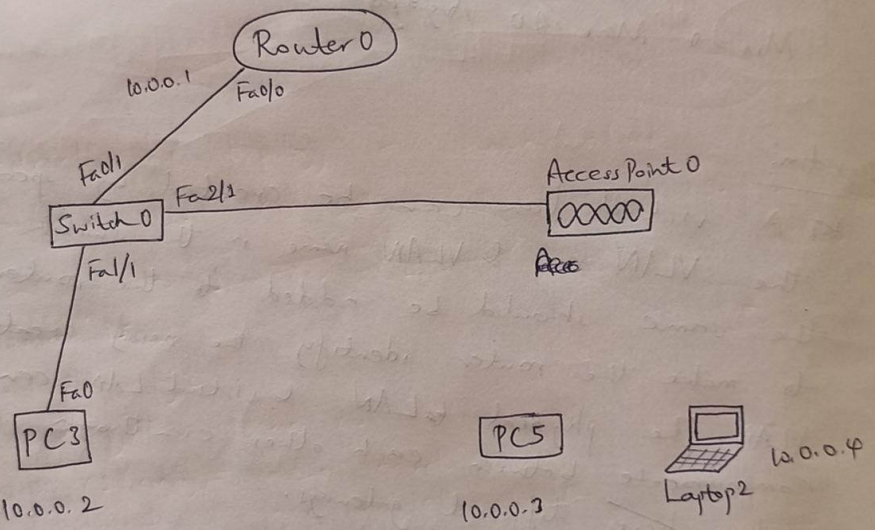
Title:

To construct a WLAN & make the nodes communicate wirelessly.

Aim:

Understand how to construct a WLAN

Topology:



Procedure:-

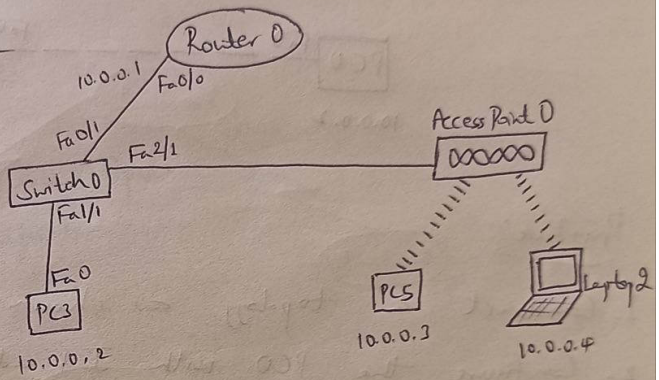
1. Create the topology as shown above
2. Configure PC3 & Router 0 as normally done
3. Configure Access Point 0, goto port 1 and give SSID name (any name), here WLAN
4. Select WEP and give any 10 digit hex key (1234567890, here). Configure PC5 & Laptop 2 with wireless standards.

5. Switch off the device. Drag existing PT-HOST-NM-1AM to Component list. Drag WMP300N wireless interface into the empty port & switch on the device.

6. In the config tab, a new wireless interface will be added. Now configure, SSID, WEP, WEP key, IP address & gateway to the devices (PC5, Laptop 2).

Result:-

Final Topology -



Pinging End Devices -

Laptop 2 > ping 10.0.0.2

Reply from 10.0.0.2 : bytes = 32 time = 22ms TTL = 120
Reply from 10.0.0.2 : bytes = 32 time = 13 ms TTL = 120
Reply from 10.0.0.2 : bytes = 32 time = 11ms TTL = 120
Reply from 10.0.0.2 : bytes = 32 time = 12ms TTL = 120

Packets: Sent = 4, Recieved = 4, Loss = 0 (0% Loss)

Min = 11ms Max = 22ms Avg = 14ms

Observation:-

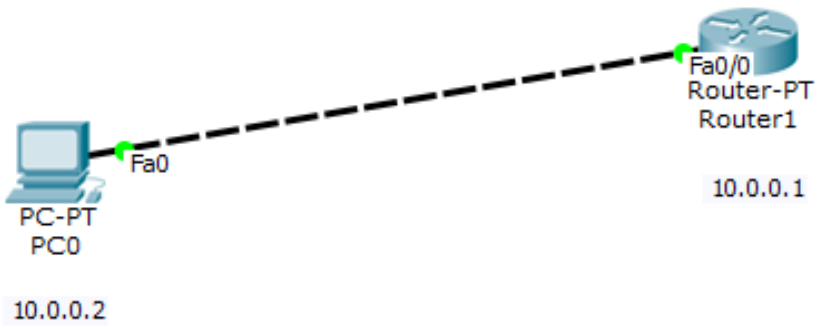
To connect devices wirelessly, the end devices are configured with wireless interface (WMP300N component) and the corresponding SSID, WEP & WEP-key of WLAN is added. The Access Point is used to create a WLAN by specifying the SSID & the WEP-key.

Experiment No. 11

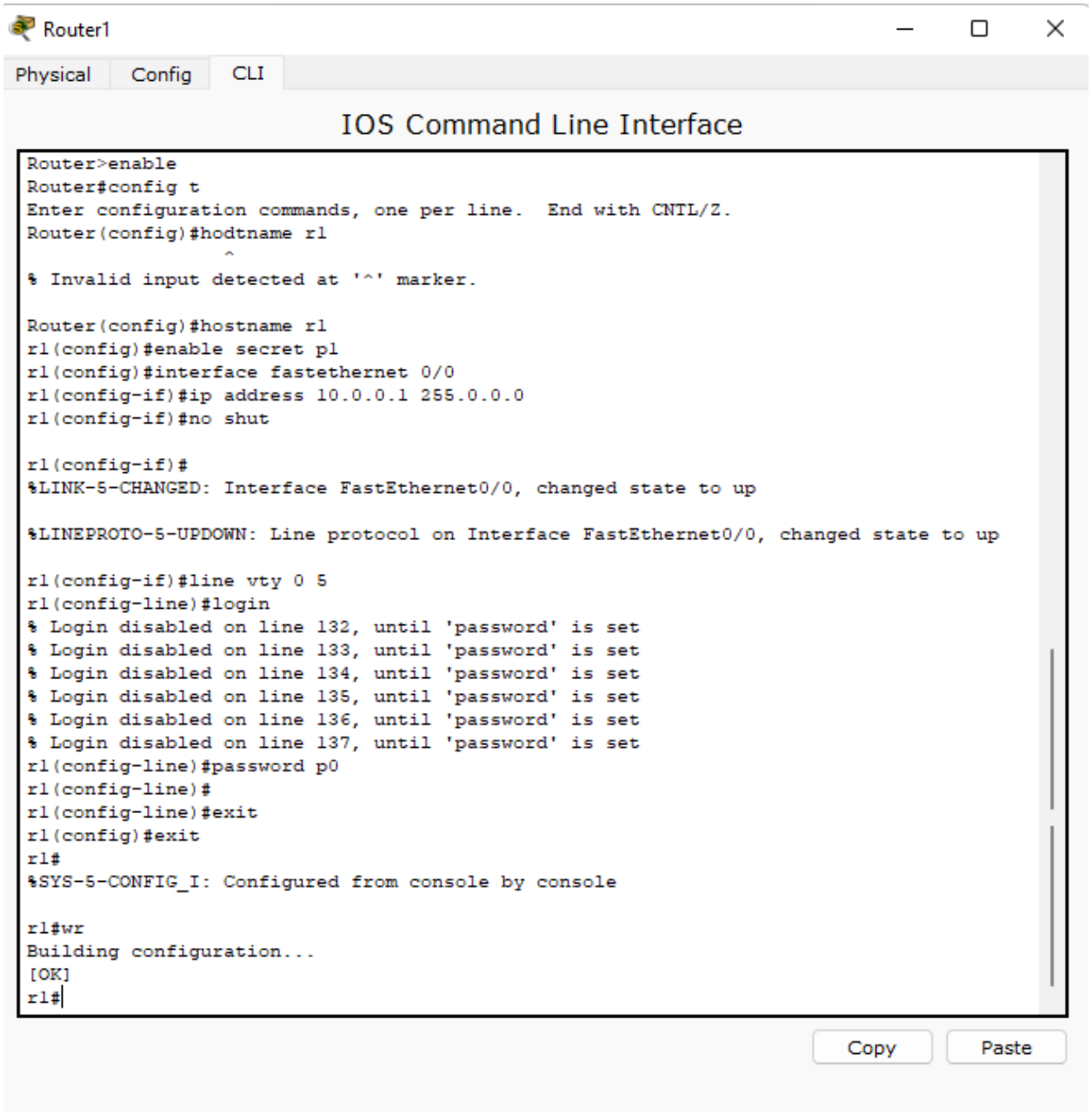
Title:

To understand the operation of TELNET by accessing the router in the server room from a PC in the IT office.

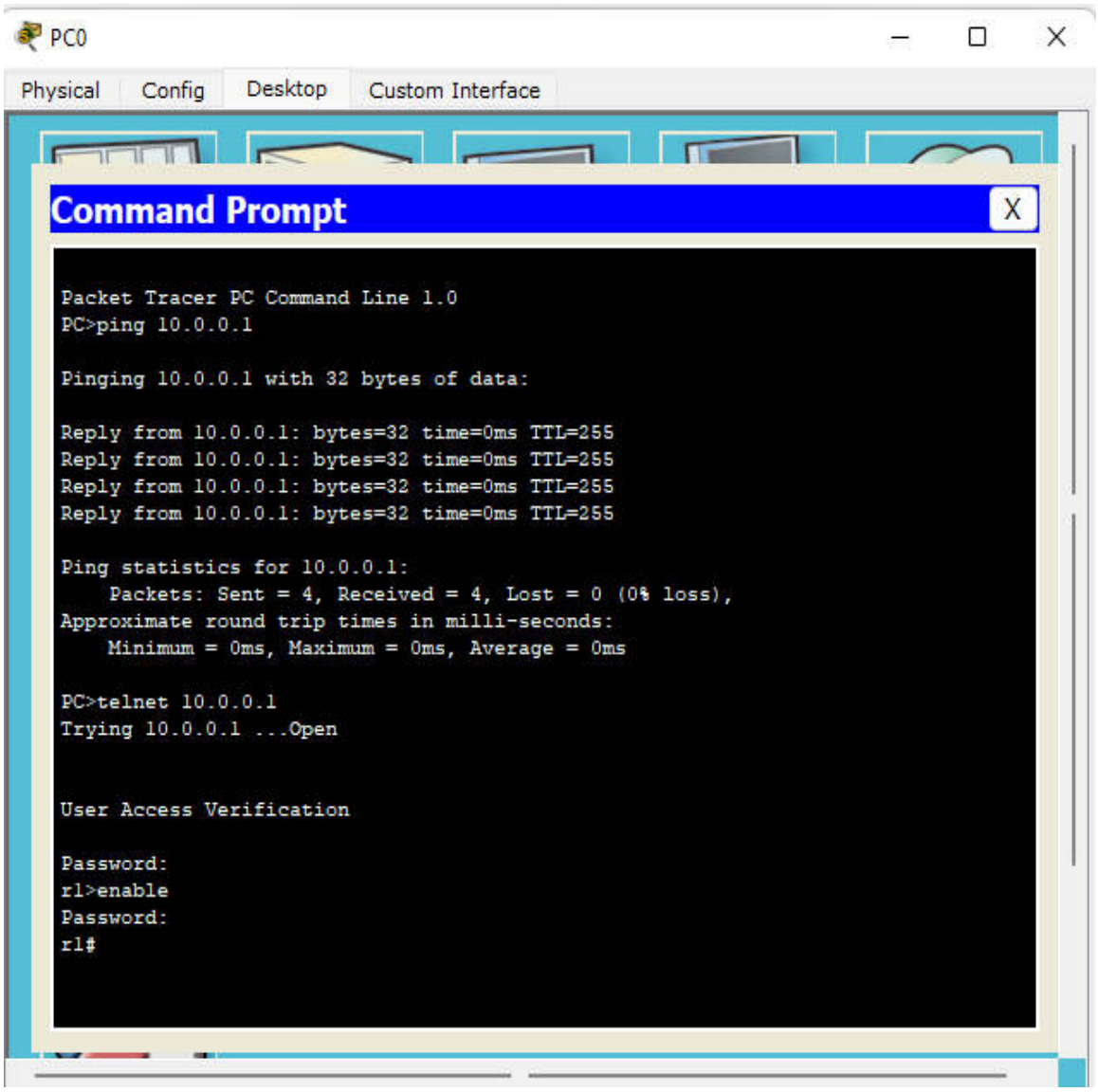
Topology:

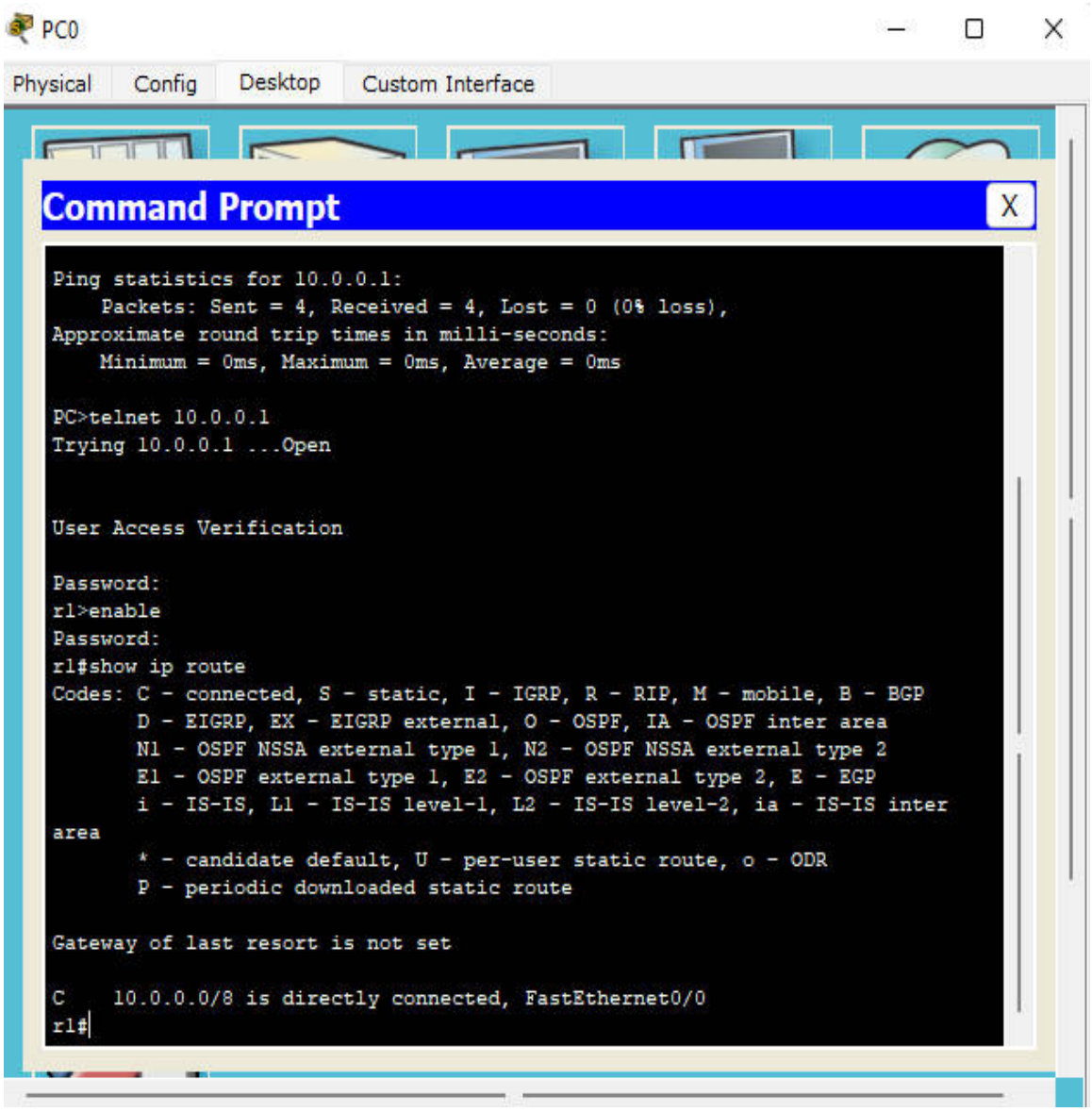


Router config:



Pinging & accession Router CLI from PC:



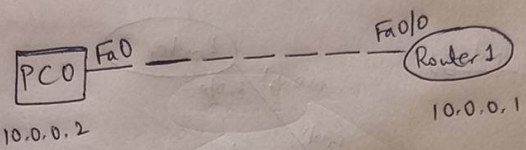

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>telnet 10.0.0.1
Trying 10.0.0.1 ...Open
User Access Verification
Password:
rl>enable
Password:
rl#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
C 10.0.0.0/8 is directly connected, FastEthernet0/0
rl#

(11)

Title/Aim:

To understand the operation of TELNET by accessing the router in server room from a PC in IT office.

Topology:



Procedure:

1. Construct a topology as shown above.
2. Configure the PC0 with IP address 10.0.0.2.
3. In router 1 oper CLI and enter the following commands.

```
Router# config t
Router(config)# hostname r1
r1(config)# enable secret p1
r1(config)# interface fastethernet 0/0
r1(config-if)# ip address 10.0.0.1 255.0.0.0
r1(config-if)# no shutdown
r1(config-if)# line vty 0 5 [to allow access for 6 users]
r1(config-line)# login
r1(config-line)# password p0
r1(config-line)# exit
r1(config)# exit
r1# wr [to save changes in router]
```

4. Router 0's ~~config~~ CLI can be opened from PC0's command prompt using following commands.

```
PC> telnet 10.0.0.1
Password:
r1 > enable
Password:
r1#
```

Result:

```
PC0>Telnet 10.0.0.1
    Trying 10.0.0.1 .... Open
User Access Verification
Password: p0
r1>enable
Password: p1
r1# show ip route
C   10.0.0.0/8 directly connected Fa0/0
```

Observation:

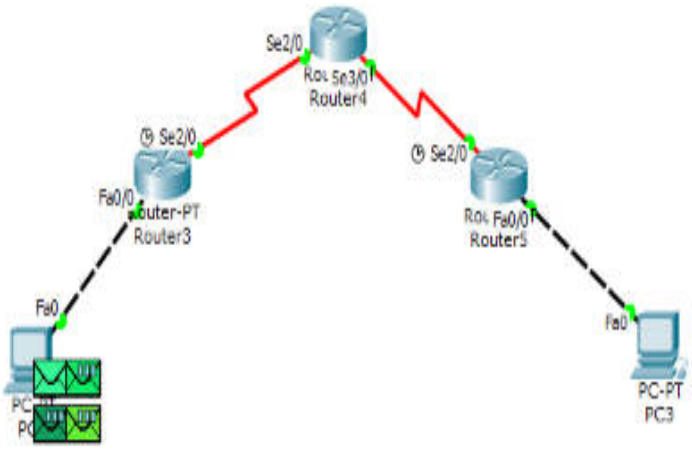
By using the "line vty 0 5" command ~~the~~ ~~no.~~ virtual access to the router and the no. of users having this access can be set. The "enable secret" command is used to set the password for enabling the router & under r1(config-line)# "password" command is used to set login password.

Experiment No. 12

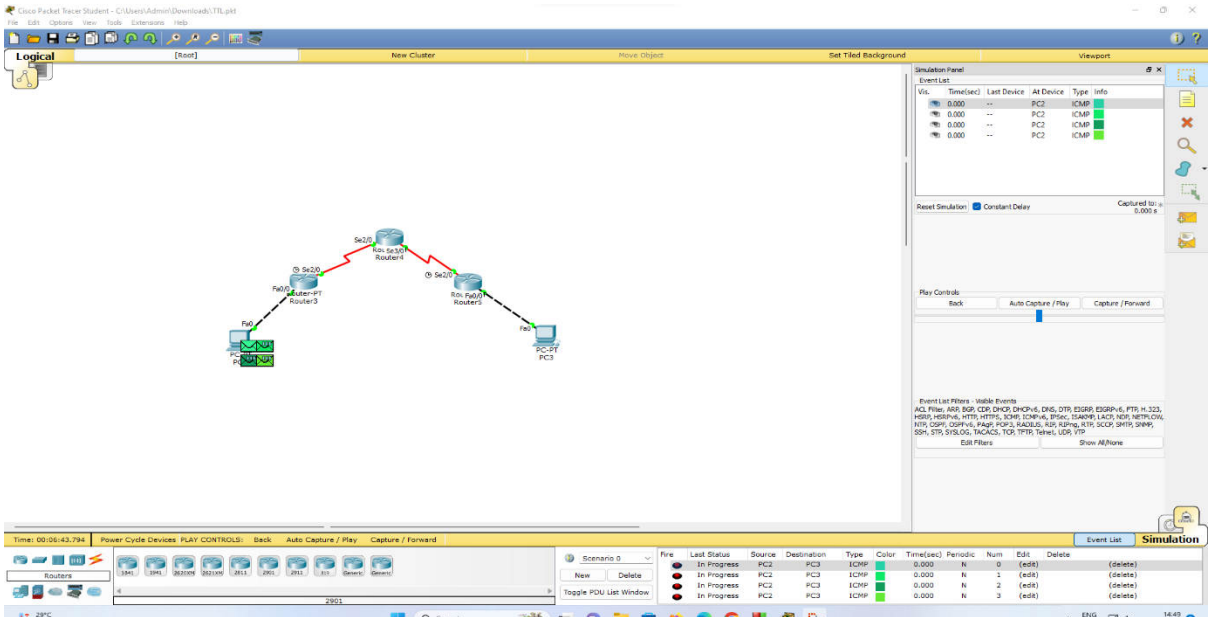
Title:

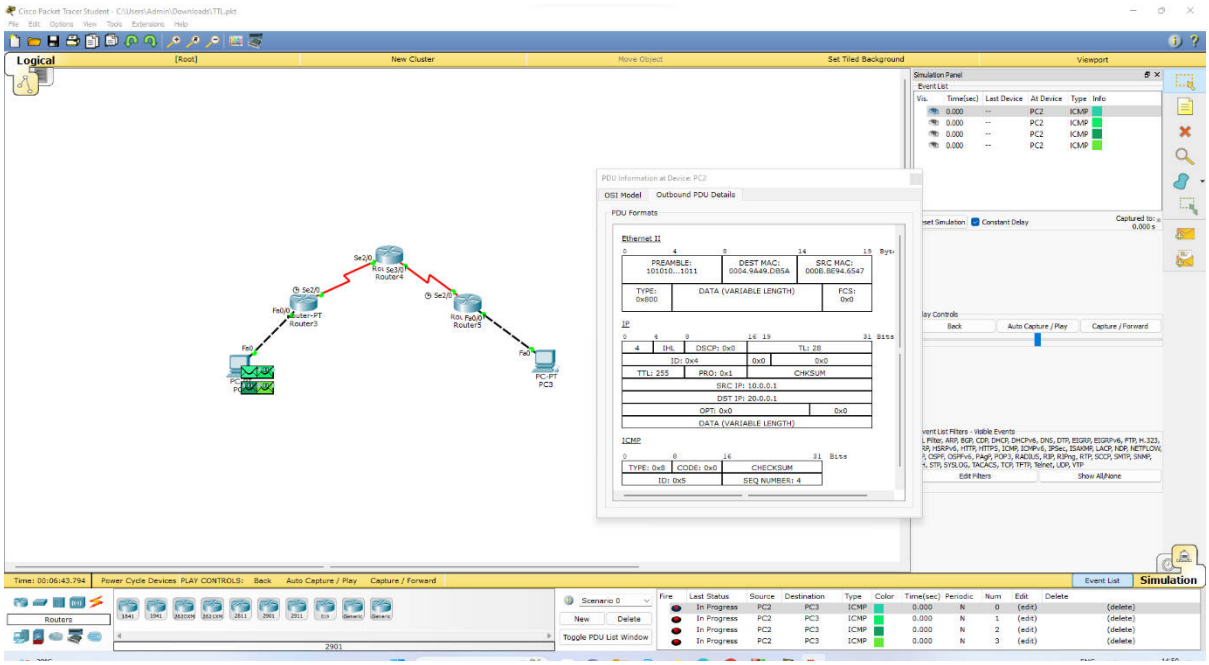
Demonstrate the TTL/ Life of a Packet

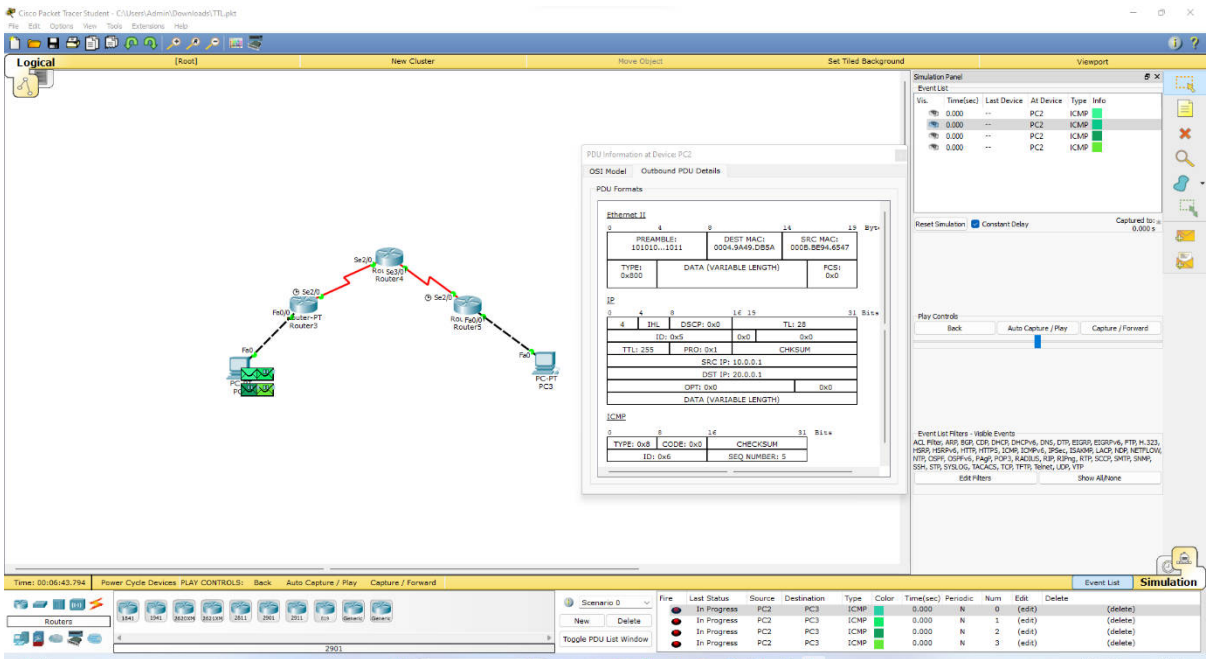
Topology:

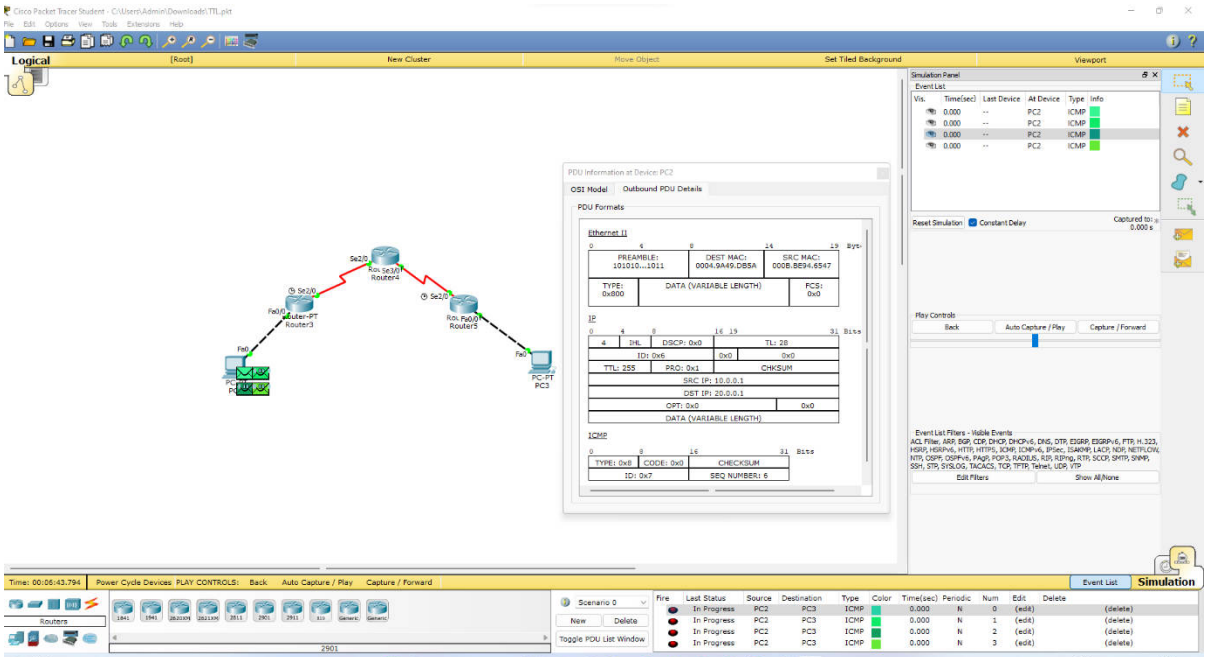


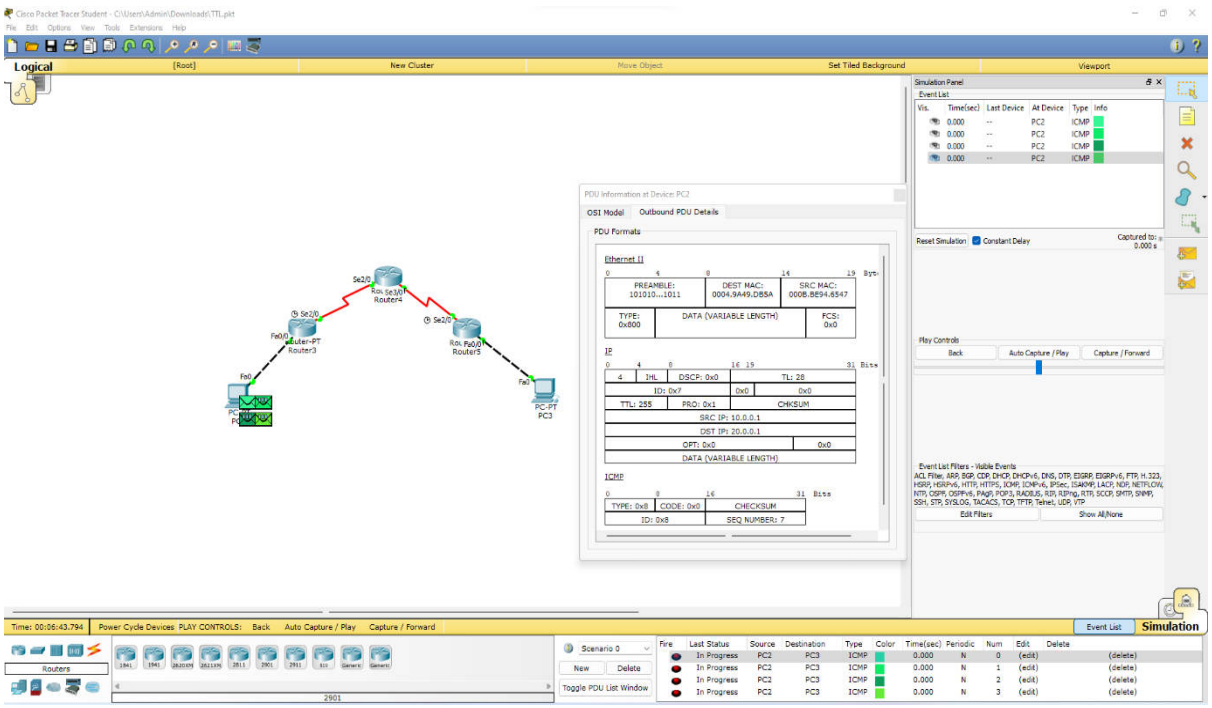
Sending PDU from one PC to another:

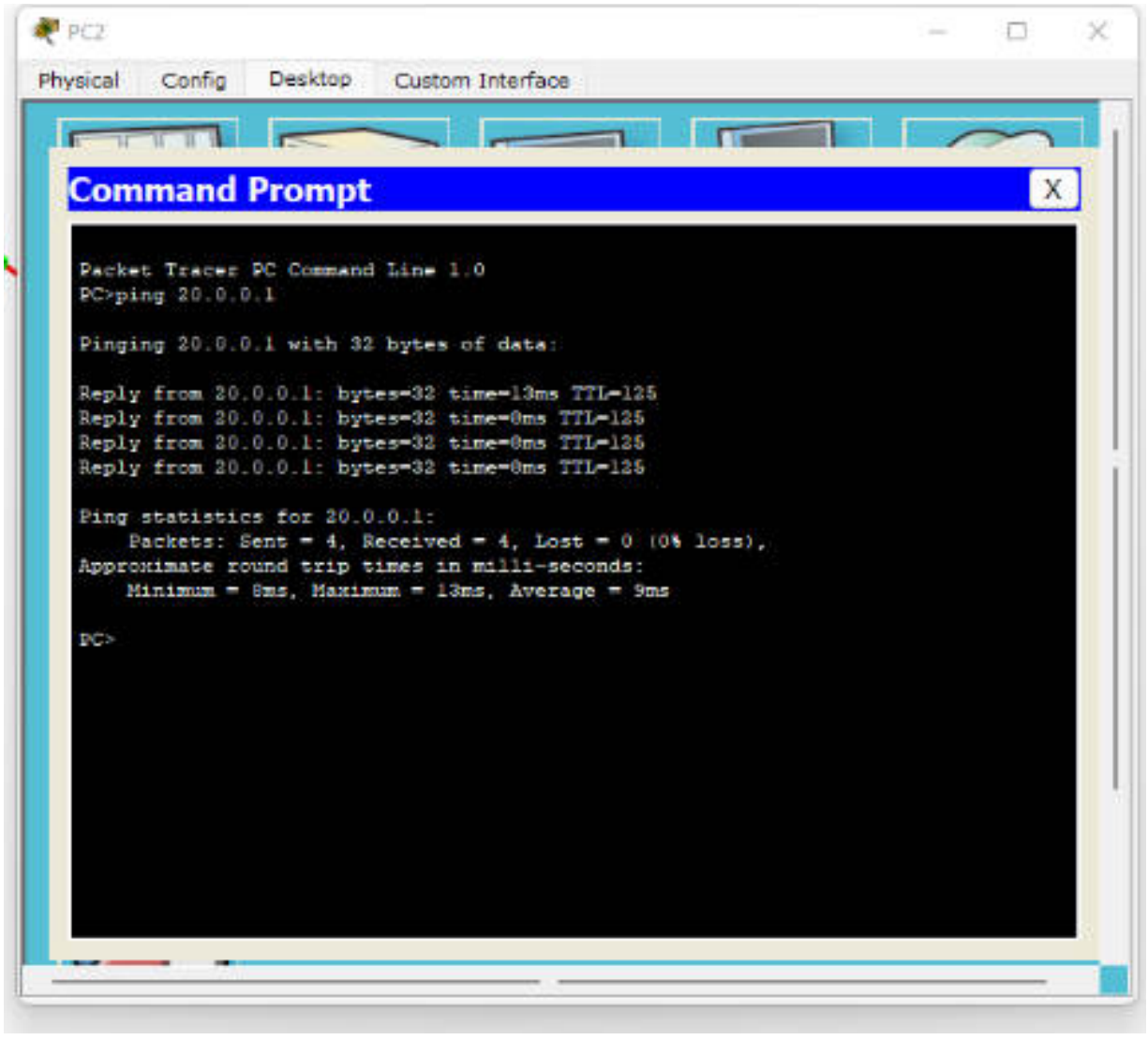











PC2
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.1
Pinging 20.0.0.1 with 32 bytes of data:
Reply from 20.0.0.1: bytes=32 time=13ms TTL=125
Reply from 20.0.0.1: bytes=32 time=0ms TTL=125
Reply from 20.0.0.1: bytes=32 time=0ms TTL=125
Reply from 20.0.0.1: bytes=32 time=0ms TTL=125
Ping statistics for 20.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 13ms, Average = 9ms
PC>

(12)

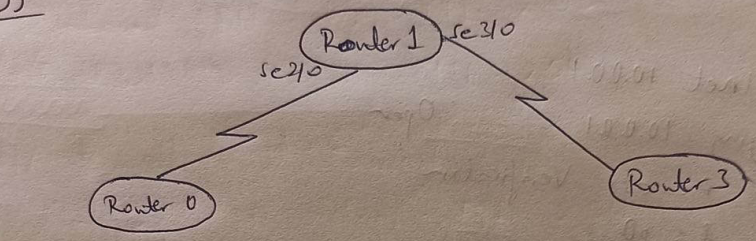
Title:-

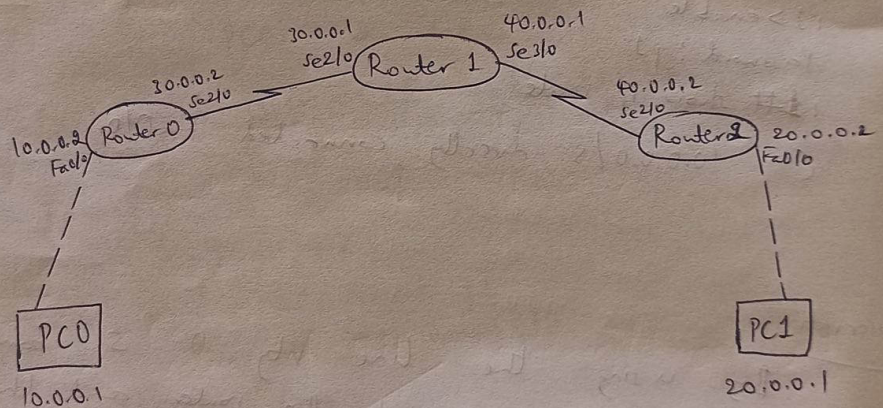
Demostrate the TTL/Life of a Packet.

Aim:-

To understand TTL/life of a packet

Topology:-





Procedure:-

1. Create a topology with 2 pcs & 3 routers as shown above.
2. Configure IP address as 10.0.0.1 & 20.0.0.1 for PC0 & PC1 respectively.

3. Configure IP addresses for routers and set default gateway.

Router 0

```
Router# config t
Router(config)# interface fa0/0
Router(config-if)# ip address 10.0.0.2 255.0.0.0
Router(config-if)# no shut
Router(config-if)# exit
Router(config)# interface se2/0
Router(config-if)# ip address 30.0.0.2 255.0.0.0
Router(config-if)# no shut
Router(config-if)# exit
Router(config)# ip router 0.0.0.0 0.0.0.0 30.0.0.1
Router(config)# exit
```

Similarly configure Router1 & Router2

4. In simulation mode, send a simple PDU from one PC to another

5. Click on PDU during every transfer to see the Inbound & Outbound PDU details, use capture button to capture every transfer

6. Observe the TTL value for the PDU when it crosses each router.

Result :-

Sending PDU from PC0 to PC1

PDU information at PC0:

Outbound PDU details - TTL = 255

PDU info at Router 0:

Inbound PDU details - TTL = 255

Outbound PDU details - TTL = 254

PDU info at Router 1:

Inbound PDU details - TTL = 254

Outbound PDU details - TTL = 253

PDU info at Router 2:

Inbound PDU details - TTL = 253

Outbound PDU details - TTL = 252

Observation :-

The value of TTL decreases/changes by 1 every time it crosses a router.

CYCLE 2

Experiment No. 1

Title:

Write a program for error detecting code using CRC-CCITT (16-bits).

Code:

```
#include<stdio.h>
char m[50],g[50],r[50],q[50],temp[50];
void caltrans(int);
void crc(int);
void calram();
void shiftl();
int main()
{
int n,i=0;
char ch,flag=0;
printf("Enter the frame bits:");
while((ch=getc(stdin))!='\n')
m[i++]=ch;
n=i;
for(i=0;i<16;i++)
m[n++]='0';
m[n]='\0';
printf("Message after appending 16 zeros:%s",m);
for(i=0;i<=16;i++)
g[i]='0';
g[0]=g[4]=g[11]=g[16]='1';g[17]='\0';
printf("\ngenerator:%s\n",g);
crc(n);
printf("\n\nquotient:%s",q);
caltrans(n);
printf("\ntransmitted frame:%s",m);
```

```
printf("\nEnter transmitted freme:");
scanf("\n%s",m);
printf("CRC checking\n");
crc(n);
printf("\n\nlast remainder:%s",r);
for(i=0;i<16;i++)
if(r[i]!='0')
flag=1;
else
continue;
if(flag==1)
printf("Error during transmission");
else
printf("\n\nReceived freme is correct");
}
void crc(int n)
{
int i,j;
for(i=0;i<n;i++)
temp[i]=m[i];
for(i=0;i<16;i++)
r[i]=m[i];
for(i=0;i<n-16;i++)
{
if(r[0]=='1')
{
q[i]='1';
calram();
}
else
{
q[i]='0';
shiftl();
}
r[16]=m[17+i];
r[17]='\0';
for(j=0;j<=17;j++)
```

```
temp[j]=r[j];
}
q[n-16]='\0';
}
void calram()
{
int i,j;
for(i=1;i<=16;i++)
r[i-1]=((int)temp[i]-48)^((int)g[i]-48)+48;
}
void shiftl()
{
int i;
for(i=1;i<=16;i++)
r[i-1]=r[i];
}
void caltrans(int n)
{
int i,k=0;
for(i=n-16;i<n;i++)
m[i]=((int)m[i]-48)^((int)r[k++]-48)+48;
m[i]='\0';
}
```

Output:

```
Enter the frame bits:1011
Message after appending 16 zeros:1011000000000000000
generator:10001000000100001


quotient:1011
transmitted frame:10111011000101101011
Enter transmitted freme:10111011000101101011
CRC checking


last remainder:0000000000000000

Received freme is correct
```

```
Enter the frame bits:1011
Message after appending 16 zeros:1011000000000000000
generator:10001000000100001


quotient:1011
transmitted frame:10111011000101101011
Enter transmitted freme:101
CRC checking


last remainder:0001000000100001 Error during transmission
```

Cycle - 2 1st September, 2023.

Title:-

(11) Write a program for error detecting code using CRC - CCITT

Code:-

```
#include<stdio.h>
char m[50], g[50], r[50], q[50], temp[50];
void cal_trans(int);
void crc(int);
void caltrans();
void shift();
int main(){
  int n, i=0, flag=0;
  char ch;
printf("Enter the frame bits: ");
while ( ch=getc(stdin)!='\n')
    m[i++]=ch;
    n=i;
for (i=0; i<16; i++)
    m[i++]='0';
    r[i]='\0';
printf("Message after appending 16 zeros: %s", m);
for (i=0; i<=16; i++)
    g[i]='0';
    g[0]=g[ ]=g[ ]=g[16]='1'; g[17]='\0';
printf("\n Generator %s\n", g);
crc(n);
printf("\n Quotient: %s\n", g);
cal_trans(n);
printf("\n Transmitted frames: %s", m);
printf("Enter transmitted frames: ");
scanf("%s", m);
```

```
printf("CRC checking ");
crc(n);
printf("Last remainder = %s",r);
for(i=0; i<16 ;i++)
    if(r[i]!='0')
      flag=1;
    else
      continue;
  if(flag==1)
    printf("Error during transmission");
  else
    printf("Recieved frame is correct");
}
void crc(int n){
  int i,j;
  for(i=0; i<n;i++)
    temp[i]=m[i];
  for(i=0; i<n-16 ;i++){
    if(r[0]=='1'){
      q[i]='1';
      calram(n);}
    else
      q[i]='0';
    shift();}
  r[16]=m[17+i];
  r[17]='\0';
  for(j=0; j<=17;j++)
    temp[j]=r[j];}
  q[n-16]='\0';
}
```

```
void calram(){
  int i,j;
  for(i=1;i<=16;i++)
    r[i-1]=((int)temp[i]-48)^((int)g[i]-48)+48;}
void shift(){
  int i;
  for(i=1;i<=16;i++)
    r[i-1]=r[i];
}
void calbar(int n){
    int i,k=0;
    for(i=n-16;i<n;i++)
      m[i]=((int)m[i]-48)^((int)r[k++]-48)+48;
      m[i]='\0';}
```

Output:-

Enter frame bits: 1011

Message after appending 16 zeroes: 1011 0000 0000 0000 0000

Generator: 1 0001 0000 0001 0000 1

Quotient: 1011

Transmitted frame: 1011 1011 0001 0110 1011

Enter Transmitted frame: 1011 1011 0001 0110 1011

Last remainder: 0000 0000 0000 0000

Recieved frame is correct.

Experiment No. 2

Title:

Write a program for congestion control using Leaky bucket algorithm.

Code:

```
#include<stdio.h>

int main(){
   int incoming, outgoing, buck_size, n, store = 0;
   printf("Enter bucket size, outgoing rate and no of inputs: ");
   scanf("%d %d %d", &buck_size, &outgoing, &n);

   while (n != 0) {
      printf("Enter the incoming packet size : ");
      scanf("%d", &incoming);
      printf("Incoming packet size %d\n", incoming);
      if (incoming <= (buck_size - store)){
         store += incoming;
         printf("Bucket buffer size %d out of %d\n", store, buck_size);
      } else {
         printf("Dropped %d no of packets\n", incoming - (buck_size - store));
         printf("Bucket buffer size %d out of %d\n", store, buck_size);
         store = buck_size;
      }
      store = store - outgoing;
      printf("After outgoing %d packets left out of %d in buffer\n", store,
buck_size);
      n--;
   }
}
```

Output:

```
Enter bucket size, outgoing rate and no of inputs: 8 6 4
Enter the incoming packet size : 3
Incoming packet size 3
Bucket buffer size 3 out of 8
After outgoing -3 packets left out of 8 in buffer
Enter the incoming packet size : 3
Incoming packet size 3
Bucket buffer size 0 out of 8
After outgoing -6 packets left out of 8 in buffer
Enter the incoming packet size : 4
Incoming packet size 4
Bucket buffer size -2 out of 8
After outgoing -8 packets left out of 8 in buffer
Enter the incoming packet size : 3
Incoming packet size 3
Bucket buffer size -5 out of 8
After outgoing -11 packets left out of 8 in buffer
```

(2) Title: Write a program for congestion control using Leaky bucket algorithm.

Code:-

```
#include<stdio.h>
int main(){
  int in,out,bucket_size,n,store=0;
printf("Enter bucket size, outgoing rate & no. of inputs: ");
scanf("%d %d %d", &bucket_size,&out,&n);
while(n!=0){
   printf("Enter the incoming packet size: ");
   scanf("%d",&in);
   printf("Incoming packet size %d\n",in);
    if(in<=(bucket_size-store)){
        store+=in;
   printf("Bucket buffer size %d out of %d\n",store,bucket_size);
   else{
     printf("Dropped %d no. of packets\n",in,(bucket_size-store));
     printf("Bucket buffer size %d out of %d\n",store,bucket_size);
     store=bucket_size;
   }
   store-=out;
   printf("After outgoing %d packets left out of %d in buffer",store,bucket);
    n--;
  }
}
```

Output:

Enter bucket size, outgoing rate & no. of inputs: 8 6 4 1
Enter the incoming packet size: 3
Bucket buffer size 3 out of 8
After outgoing -3 packets left out of 8 in buffer

Experiment No. 3

Title:

Using TCP/IP sockets, write a client-server program to make the client send the file name and the server to send back the contents of the requested file if present.

Code:

ClientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence = input("\nEnter file name: ")
clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print ('\nFrom Server:\n')
print(filecontents)
clientSocket.close()
```

ServerTCP.py

```
from socket import *
serverName="127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
    print ("The server is ready to receive")
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()
    file=open(sentence,"r")
    l=file.read(1024)
    connectionSocket.send(l.encode())
    print ('\nSent contents of ' + sentence)
    file.close()
    connectionSocket.close()
```

Output:

```
/Users/mac/PycharmProjects/cn-lab/venv/bin/python /Users/mac/PycharmProjects/cn-lab/ServerTCP.py
The server is ready to receive
```

```
/Users/mac/PycharmProjects/cn-lab/venv/bin/python /Users/mac/PycharmProjects/cn-lab/ClientTCP.py

Enter file name: ServerTCP.py

From Server:

from socket import *

serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
```

```
while 1:
    print("The server is ready to receive")
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()

    file = open(sentence, "r")
    l = file.read(1024)

    connectionSocket.send(l.encode())
    print('\nSent contents of ' + sentence)
    file.close()
    connectionSocket.close()


Process finished with exit code 0
```

```
/Users/mac/PycharmProjects/cn-lab/venv/bin/python /Users/mac/PycharmProjects/cn-lab/ServerTCP.py
The server is ready to receive

Sent contents of ServerTCP.py
The server is ready to receive
```

(3)

Title:-

Use TCP/IP sockets, write a client-server program to make client sending the file name & the server to send back the contents of requested file if present.

Code:-

ClientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("\nEnter file name: ")
clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print(filecontents)
clientSocket.close()
```

ServerTCP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
while(1):
    print("The server is ready to receive")
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()
    file = open(sentence, "r")
    l = file.read(1024)
    connectionSocket.send(l.encode())
```

```
print ("\n Sent contents of "+ sentence)
file.close()
connectionSocket.close()
```

Output:-

ServerTCP.py

The server is ready to recieve

Sent contents of ServerTCP.py

~~The server is ready to recieve~~

ClientTCP.py

Enter file name: ServerTCP.py

From Server:

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
while 1:
    print("The server is ready to recieve")
    connectionSocket, addr = serverSocket.accept()
    sentence = connectionSocket.recv(1024).decode()
    file = open(sentence, "r")
    l = file.read(1024)
    connectionSocket.send(l.encode())
    print('\n Sent contents of ' + sentence)
    file.close()
    connectionSocket.close()
```

Experiment No. 4

Title:

Using UDP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

Code:

ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\nEnter file name:  ")
clientSocket.sendto(bytes(sentence,"utf-8"),(serverName, serverPort))
filecontents,serverAddress = clientSocket.recvfrom(2048)
print ('\nReply from Server:\n')
print (filecontents.decode("utf-8"))
# for i in filecontents:
   # print(str(i), end = '')
clientSocket.close()
clientSocket.close()
```

ServerUDP.py

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print ("The server is ready to receive")
while 1:
        sentence, clientAddress = serverSocket.recvfrom(2048)
        sentence = sentence.decode("utf-8")
        file=open(sentence,"r")
        con=file.read(2048)
          serverSocket.sendto(bytes(con,"utf-8"),clientAddress)
        print ('\nSent contents of ', end = ' ')
        print (sentence)
       #  for i in sentence:
          #  print (str(i), end = '')
        file.close()
```

Output:

```
/Users/mac/PycharmProjects/cn-lab/venv/bin/python /Users/mac/PycharmProjects/cn-lab/ServerUDP.py
The server is ready to receive
```

```
/Users/mac/PycharmProjects/cn-lab/venv/bin/python /Users/mac/PycharmProjects/cn-lab/ClientUDP.py

Enter file name:  ServerUDP.py

Reply from Server:

from socket import *

serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print("The server is ready to receive")
while 1:
    sentence, clientAddress = serverSocket.recvfrom(2048)
    sentence = sentence.decode("utf-8")
    file = open(sentence, "r")
    con = file.read(2048)
```

```
    serverSocket.sendto(bytes(con, "utf-8"), clientAddress)

    print('\nSent contents of ', end=' ')
    print(sentence)
    #  for i in sentence:
    #  print (str(i), end = '')
    file.close()


Process finished with exit code 0
```

```
/Users/mac/PycharmProjects/cn-lab/venv/bin/python /Users/mac/PycharmProjects/cn-lab/ServerUDP.py
The server is ready to receive

Sent contents of  ServerUDP.py
```

④ Title:
Using UDP sockets, write a client-server program to make client sending the file name & the server to send back the contents of the requested file if present.

Code:

ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
sentence = input("\nEnter file name: ")
clientSocket.sendto(bytes(sentence,"utf-8"),(serverName,serverPort))
fileContents, serverAddress = clientSocket.recvfrom(2048)
print('\nReply from Server: \n')
print(fileContents.decode("utf-8"))
clientSocket.close()
clientSocket.close()
```

ServerUDP.py

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind("127.0.0.1", serverPort))
print("The server is ready to recieve")
While 1:
    sentence, clientAddress = serverSocket.recvfrom(2048)
    sentence = sentence.decode("utf-8")
    file.open(sentence,"r")
    con = file.read(2048)
    serverSocket.sendto(bytes(con,"utf-8"),clientAddress)
    print('\nSent contents of', end=' ')
    print(sentence)
    file.close()
```

O/P:

ServerUDP.py

The server is ready to recieve

Sent contents of ServerUDP.y

ClientUDP.py

Enter file name: ServerUDP.py

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
while 1:
    print("The server is ready to recieve")
    sentence, clientAddress = serverSocket.recvfrom(2048)
    sentence = sentence.decode("utf-8")
    file.open(sentence, "r")
    l = file.read(2048)
    serverSocket.sendto(bytes(l,"utf-8"), clientAddress)
    print('\n Sent contents of ', end = " ")
    print(sentence)
    file.close()
```

8/9/23