VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

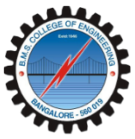
LAB REPORT
on

COMPUTER NETWORKS LAB

Submitted by

DEEKSHA S (1BM21CS048)

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
JUN-2023 to SEP-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 "COMPUTER NETWORKS LAB" carried out by **DEEKSHA S (1BM21CS048),** who is 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 a **COMPUTER NETWORKS - (22CS4PCCON)** work prescribed for the said degree.

M Lakshmi Neelima
Assistant Professor
Department of CSE
BMSCE, Bengaluru

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

Index

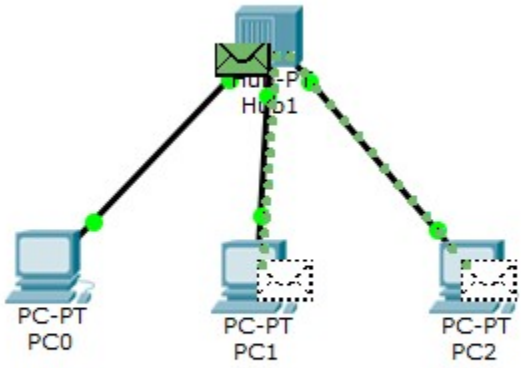
Cycle - 1

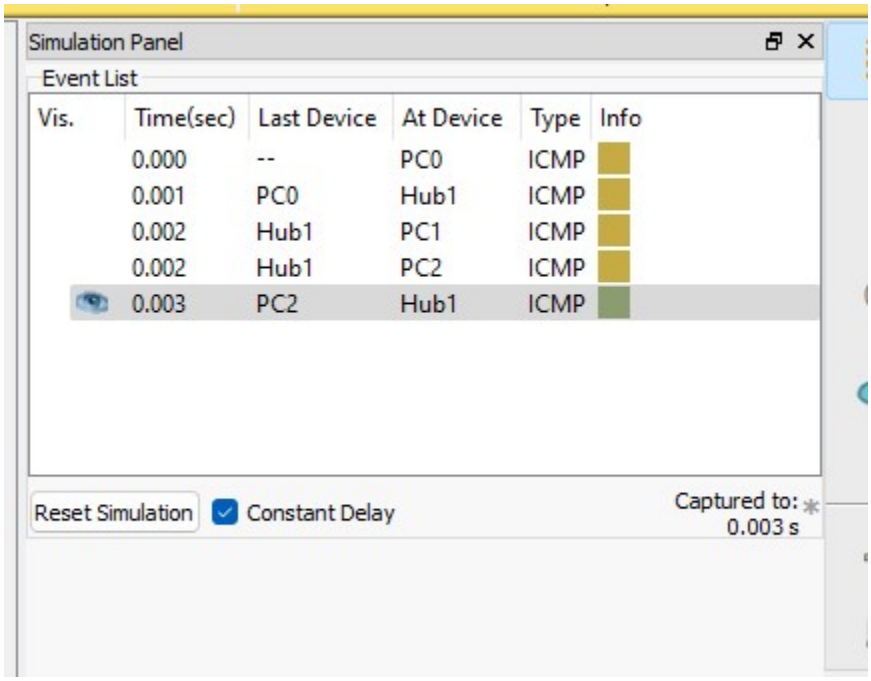
Experiment 1

Aim: Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate the ping message.

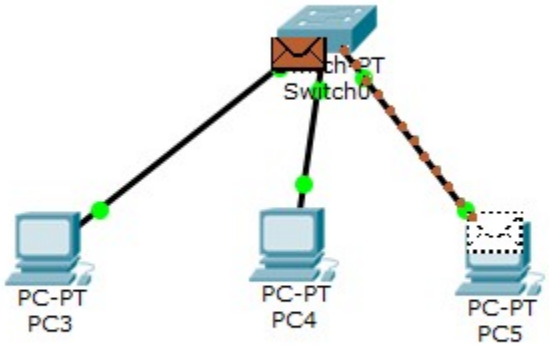
Topology:

Topology with Hub as connecting Device:

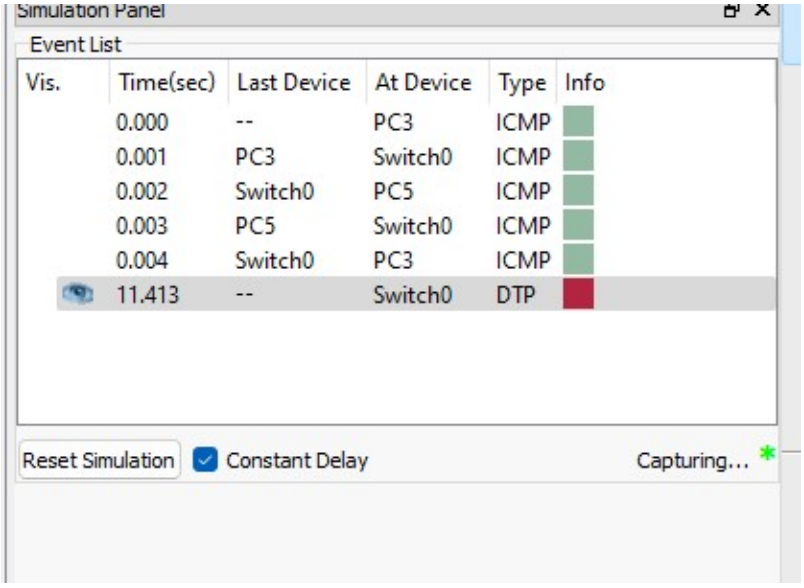




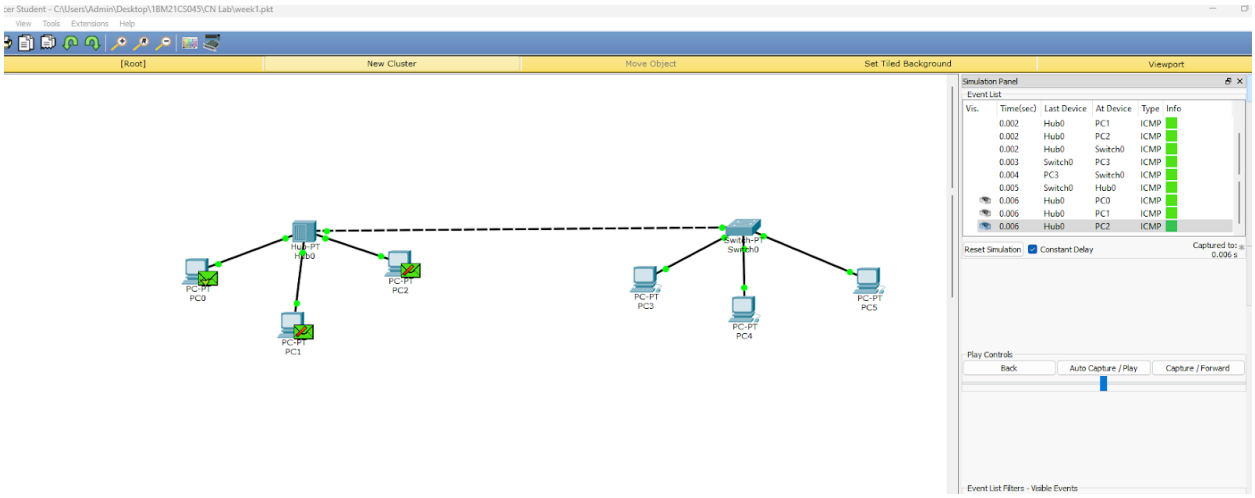
Topology with Switch as connecting Device:



Simulation Panel:



Topology with Switch and Hub together:



Observation:

Q1 Create a ~~tog~~ topology & stimulate sending a simple PDU from source to destination using a simple hub & switch as connecting domains.

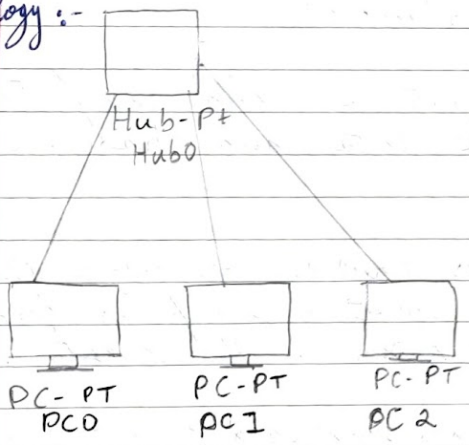
Hub : Physical layer
Switch : Data link layer.

Hub : unicast - half duplex

Switch : Broadcast - Full duplex

Aim: To create topology & stimulate sending packets (simple PDU) from source to destination using Hub.

Topology :-



Hub - 8 connected with 3 PCs.

Procedure:

Step 1: Select The network is started by selecting end devices as 3 PC-PTs, i.e generic PC.

Step 2: Select Generic Hub, Hub-PT and place all of them in the workspace.

Step 3: Now, connect the 3 PCs to the Hub using Copper-Straight Through cable. PC0 - Port 0, PC1 - Port 1, PC2 - Port 2.

Step 4: Open the config window of all PCs and open FastEthernet0. Set the IP Address of all the 3 PCs. For ex: 10.0.0.1 (PC0), 10.0.0.2 (PC1), 10.0.0.3 (PC2) respectively.

Step 5: Click on DNS server.

Step 6: Add a simple PDU from PC0 to PC2.

Step 7: In Simulation mode click on Auto Capture / Play.

Observation:

→ PC 0 sends a packet to Hub 0.
→ Hub 1 sends the packet to both PC1 and PC2
→ PC2 accepts the packet and sends acknowledgement to Hub 0. PC1 rejects it since it is not addressed to it.
→ The acknowledgement packet from Hub 0 is

sent to PC0 and PC1. PC0 accepts it and PC1 rejects it

Ping Output :- 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 = 0ms TTL = 128
Reply from 10.0.0.2 : bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.2 : bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.2 : bytes = 32 time = 0ms 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 = 0ms, Maximum = 0ms, Average = 0ms

Switch - 3 PCS

Procedure:

Step 1: The network is started by selecting end devices as 3 PC-PTS, ie generic PC.

Step 2: Select Generic switch - PT and place these devices in the workspace.

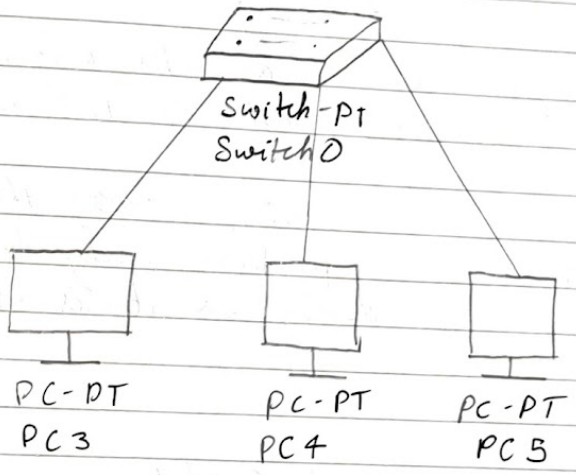
Step 3: Now, connect the 3 PCs to the switch using copper straight through cable.
~~wire PC3 - Port0, PC - 2n Switch:~~
PC3 - FastEthernet 0/1, PC4 - FastEthernet 1/1
PC 5 - FastEthernet 2/1.
In PC - choose FastEthernet 0 connection

Step 4 - Open config window of all PCs and in FastEthernet 0, set IP addresses.
PC 3 - 10.0.0.4, PC4 - 10.0.0.5
PC 5 - 10.0.0.6
Then click on DNS server.

Step 5 - Add a simple PDU from PC 3 to PC5.

Step 6: In Simulation mode Click on Auto Capture /Play.

Topology.



Ping Output in real time mode: In PC 3

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 for 10.0.0.6:

Ro packets : Set Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0 ms, Maximum = 0 ms, Average = 0 ms

Observation

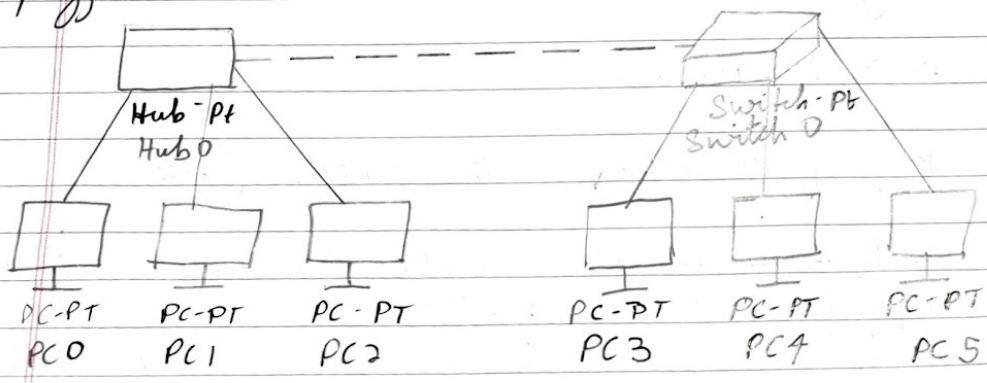
PC 3 to PC 5

→ PC 3 sends a packet to Switch 0.
→ switch 0 sends the packets to all devices (PC 4, PC 5) in first time.
→ PC 5 accepts and sends acknowledgement.
→ Switch sends it to all and PC 3 accepts.

→ From next time Switch learns the devices & it sends packets only to correct source devices as it is smart device.

Aim: Connecting Hub and Switch.

Topology:



Procedure

Step 1: Prevously obtained Hub topology and switch topology are connected using Copper-Cross over cable
Port 3-hub to FastEthernet 3/1.

Step 2: Add simple PDU from PC0 to PC3.

Ping output

ping 10.0.0.4

Pinging 10.0.0.4 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=0ms TTL = 128
Reply from 10.0.0.4: bytes = 32 time = 3ms TTL = 128
Reply from 10.0.0.4: bytes = 32 time = 0ms TTL = 128

Ping Statistics for 10.0.0.4:
Packets: Sent = 4, Received = 4, Lost = 0 (0% lost),
Approximate round trip time in milli-seconds:
Minimum = 0ms, Maximum = 3 ms, Average = 1 ms

Observation: PC0 to PC3

→ PC0 sends ~~PC0 to~~ packet to Hub 0. ~~#~~
→ Hub0 sends the packets to all PC1, PC2 and Switch.
→ Switch sends the packets to all PC3, PC4, PC5 at first instance.
→ PC3 accepts the packet and it is sent

PC0 via Hub 0. But Hub 0 also sends it to PC1 and PC2 as well where it is rejected.

→ On the second instance, PC0 sends packet again to Hub 1. It sends to all P1, PC2, & ~~PC3, PC4 and~~ Switch. Switch sends to only PC_3 as it has already learnt about the device and acknolegement is sent via Hub 1 Hub 0 sends it to all, where PC0 accepts it and so on.

15/6/2023

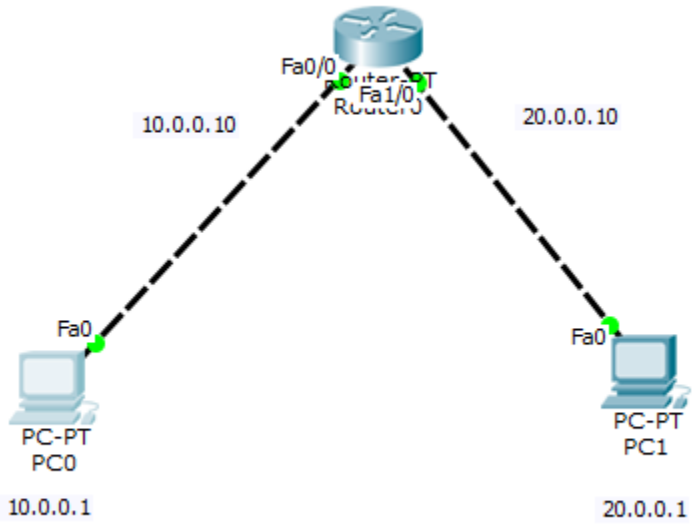
Experiment 2

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

2A:

Topology:

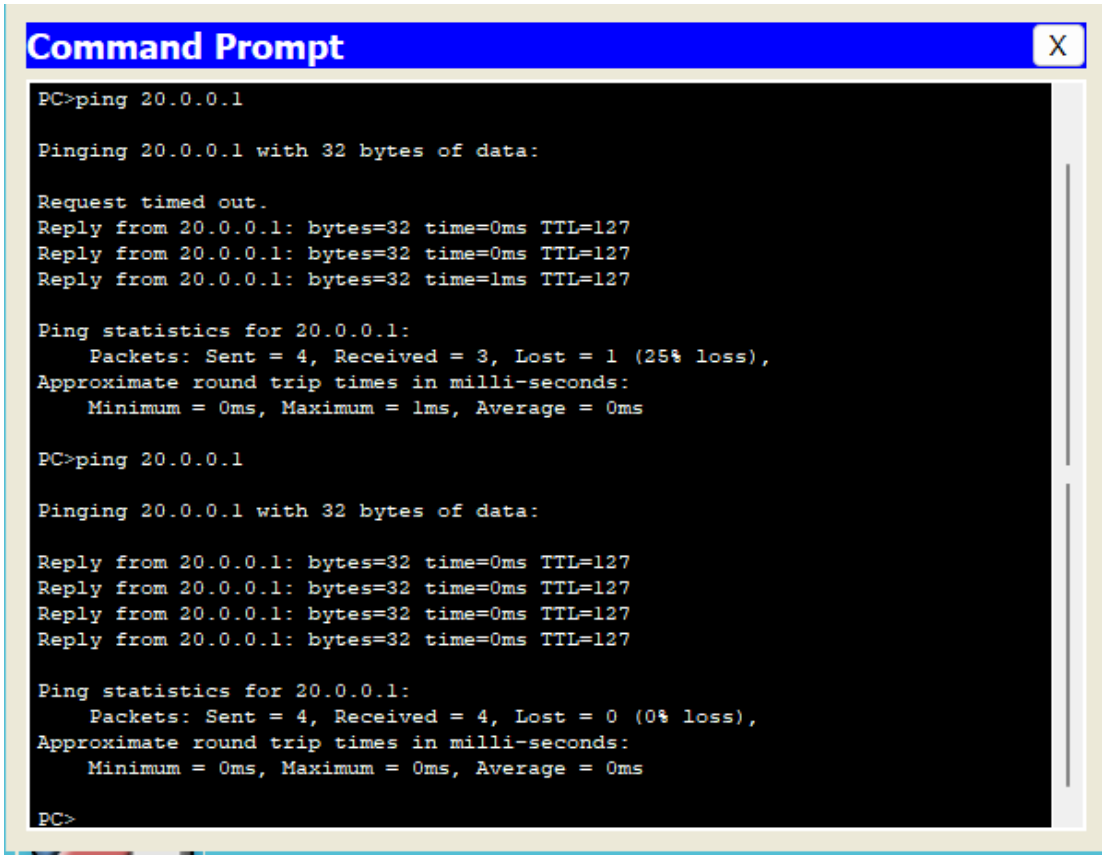
Topology with 1 router and 2 PCs:



Configuration of Router:

```
Continue with configuration dialog? [yes/no]: n


Press RETURN to get started!



Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet0/0
Router(config-if)#ip address 10.0.0.10 255.0.0.0
Router(config-if)#no shut

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

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#interface fastethernet1/0
Router(config-if)#ip address 20.0.0.10 255.0.0.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet1/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet1/0, changed state to up
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    10.0.0.0/8 is directly connected, FastEthernet0/0
C    20.0.0.0/8 is directly connected, FastEthernet1/0
Router#
```

Ping Output:


Command Prompt
X
PC>ping 20.0.0.1
Pinging 20.0.0.1 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=1ms TTL=127
Ping statistics for 20.0.0.1:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
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=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
Reply from 20.0.0.1: bytes=32 time=0ms TTL=127
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
PC>

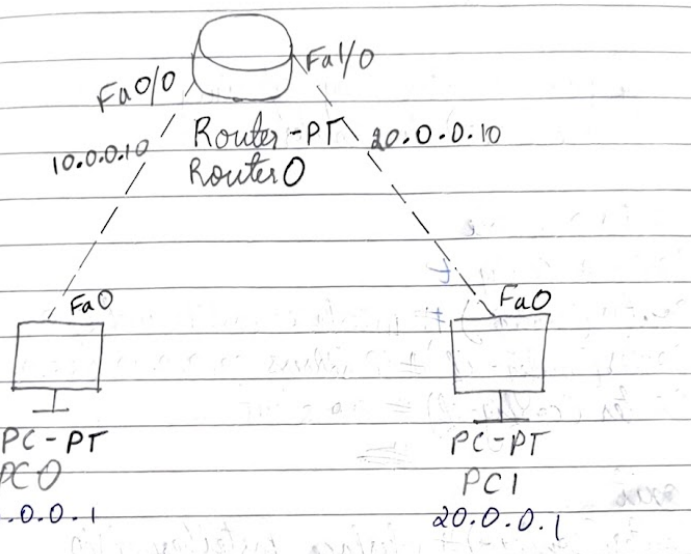
Observation:

22/6/23 Experiment 2.

AIM: Q2. Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply.

2A) Aim :- Configure using 1 router and 2 PCs.

Topology :-



Procedure :-

→ The network is started by selecting end devices PC0 and PC1 as generic PCs.

→ Select Generic Router, Router-PT and place all of them in the workspace.

→ Now, connect the PCs to the router using

Copper cross-over cable.

PC0 → FastEthernet 0 to FastEthernet 0/0
PC1 → FastEthernet 0 to FastEthernet 1/0

→ Setup IP address of the end ~~PC0~~ devices PC0 and PC1, by clicking config → FastEthernet0 → IP address

PC0 → 10.0.0.1
PC1 → 20.0.0.1

→ Configure router by opening Command Line Interface (CLI) on router 0.

```
Router > enable
Router # config t
Router(config) # interface fastethernet 0/0
Router(config-if) # ip address 10.0.0.10 255.0.0.0
Router(config-if) # no shut
exit
Router(config) # interface fastethernet 1/0
Router(config-if) # ip address 20.0.0.10 255.0.0.0
Router(config-if) no shut
exit
Router(config) # exit
show ip route
```

C 10.0.0.0/8 is directly connected, FastEthernet 0/0
C 20.0.0.0/8 is directly connected, FastEthernet 1/0

→ Setup the gateway for PC0 and PC1
PC0 : 10.0.0.10
PC1 : 20.0.0.10

Observation :-

→ Green lights appear on the wires when no shut commands are written which indicate that they are ready for data transmission.

~~Ping Output in PC0 :~~

→ When PC1 is pinged for the first time, there is 25% loss.
→ From next ping, there are no losses.

Ping output in PC0 :-

PC> ping 20.0.0.1

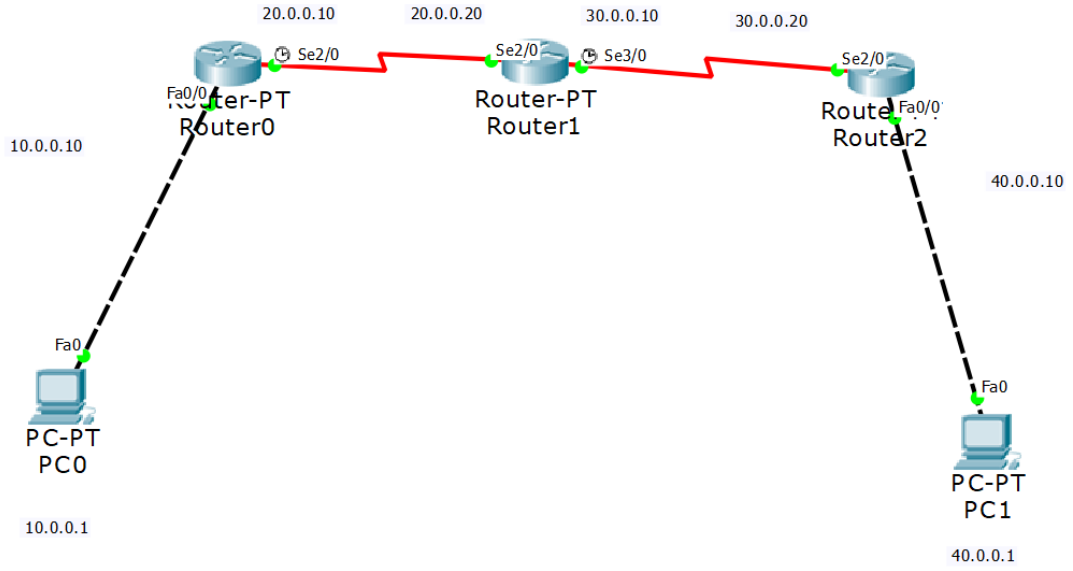
Pinging 20.0.0.1 with 32 bytes of data :

Request timed out.
Reply from 20.0.0.1 : bytes = 32 time = 0 ms TTL = 127
Reply from 20.0.0.1 : bytes = 32 time = 0 ms TTL = 127
Reply from 20.0.0.1 : bytes = 32 time = 0 ms TTL = 127

Ping statistics for 20.0.0.1 :
Packets : Sent = 4, Received = 3, Loss = 1 (25% loss)
Approximate round trip times in milli-seconds :
Minimum = 0ms, Maximum = 1ms, Average = 0ms

2B:

Topology:



Configuration of Routers:

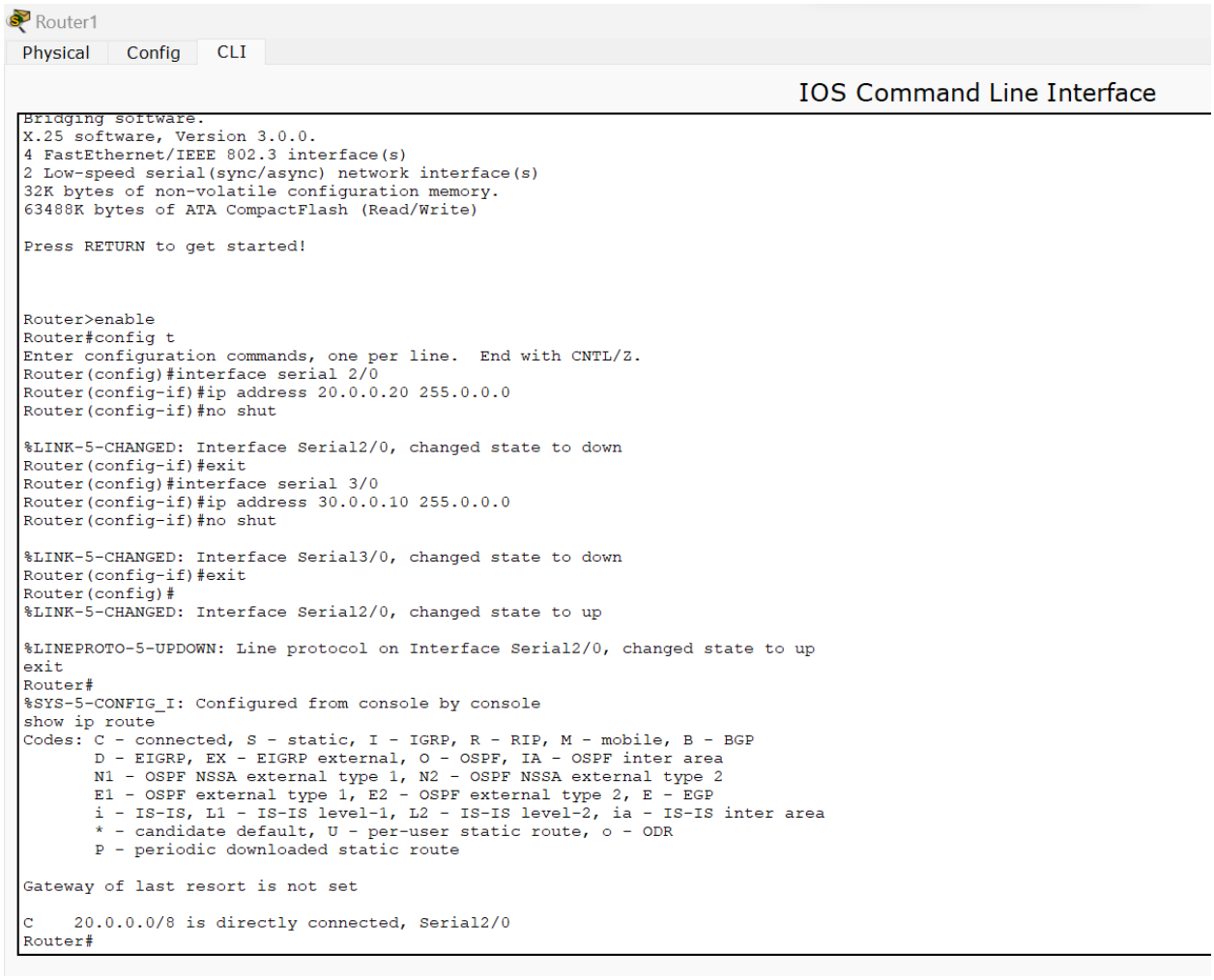
Router0

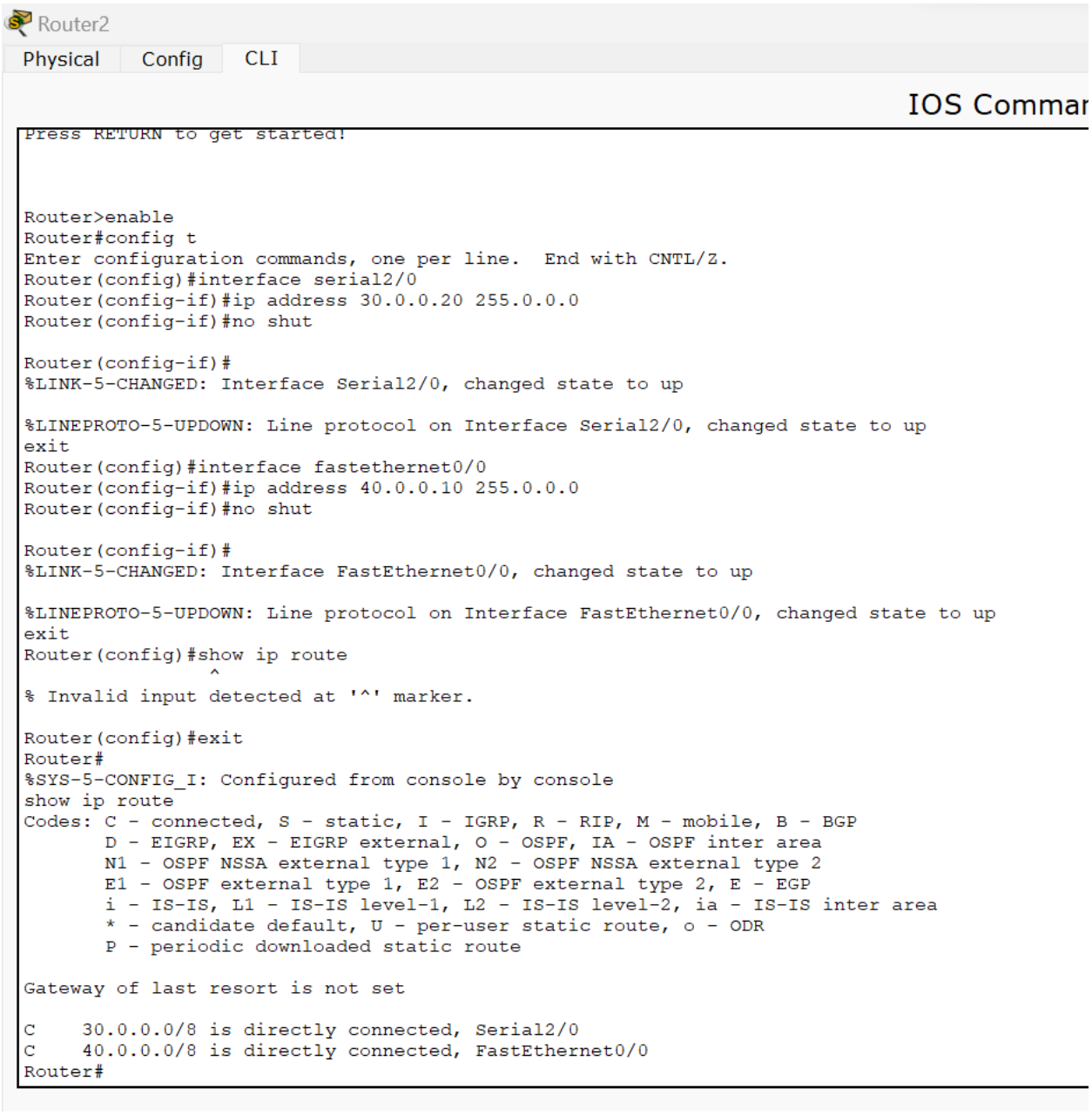
Physical | Config | CLI

IOS Command Line Interface

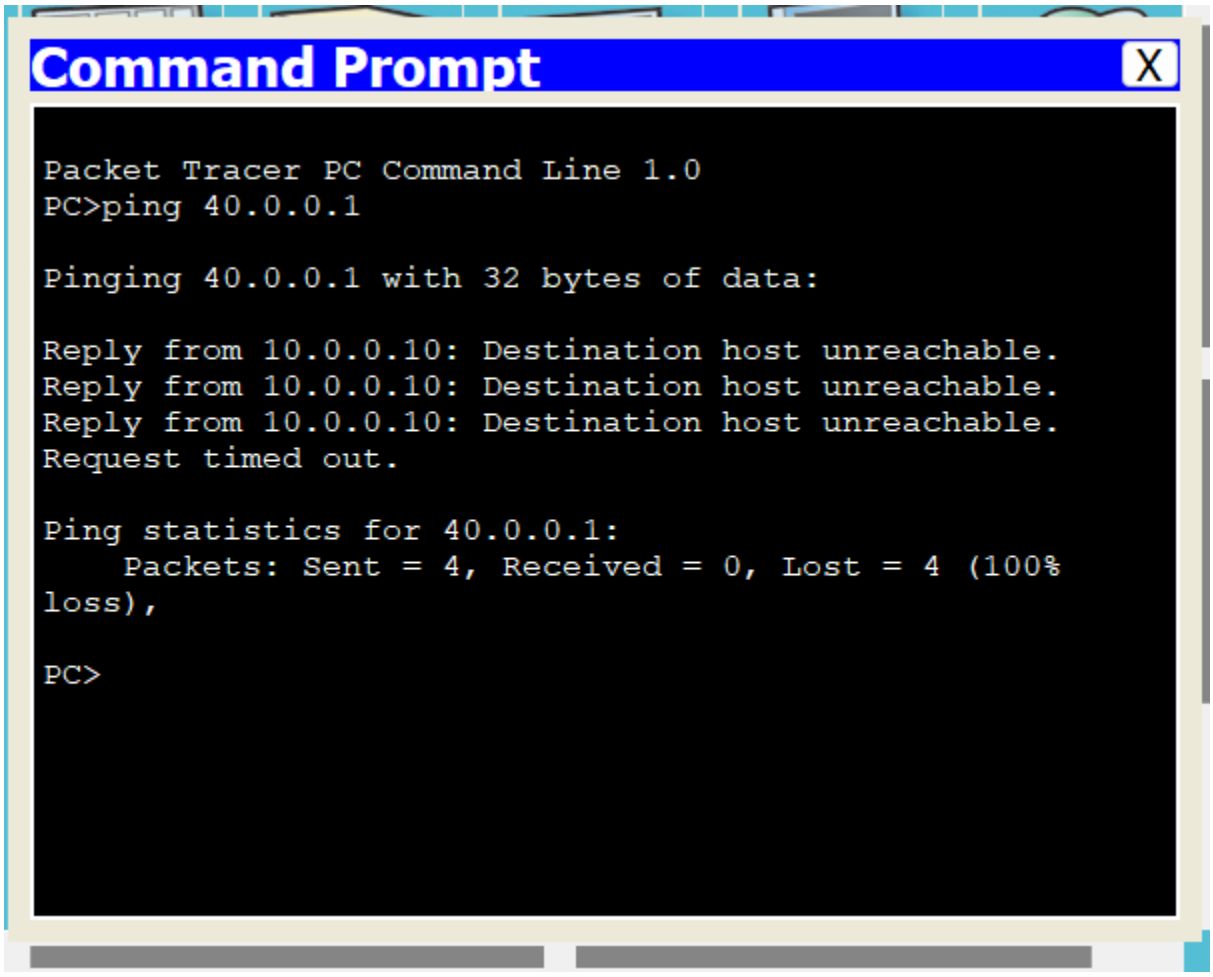
```
.
Processor board ID PT0123 (0123)
PT2005 processor: part number 0, mask 01
Bridging software.
X.25 software, Version 3.0.0.
4 FastEthernet/IEEE 802.3 interface(s)
2 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)

Press RETURN to get started!



Router>n
Translating "n"...domain server (255.255.255.255)
% Unknown command or computer name, or unable to find computer address

Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet0/0
Router(config-if)#ip address 10.0.0.10 255.0.0.0
Router(config-if)#no shut

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

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#exit
Router#
```

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface serial2/0
Router(config-if)#ip address 20.0.0.10 255.0.0.0
Router(config-if)#no shut

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
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    10.0.0.0/8 is directly connected, FastEthernet0/0
C    20.0.0.0/8 is directly connected, Serial2/0
Router#
```


Router1
Physical
Config
CLI
IOS Command Line Interface
Bridging software.
X.25 software, Version 3.0.0.
4 FastEthernet/IEEE 802.3 interface(s)
2 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)
Press RETURN to get started!
Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial2/0, changed state to down
Router(config-if)#exit
Router(config)#interface serial 3/0
Router(config-if)#ip address 30.0.0.10 255.0.0.0
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
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, Serial2/0
Router#


Router2
Physical
Config
CLI
IOS Commar
Press RETURN to get started!
Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial2/0
Router(config-if)#ip address 30.0.0.20 255.0.0.0
Router(config-if)#no shut
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
exit
Router(config)#interface fastethernet0/0
Router(config-if)#ip address 40.0.0.10 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#show ip route
^
% Invalid input detected at '^' marker.
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#

Ping output before static routing:



Router 0 :

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 30.0.0.0 255.0.0.0 20.0.0.20
```

```
Router(config)#ip route 40.0.0.0 255.0.0.0 20.0.0.20
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    10.0.0.0/8 is directly connected, FastEthernet0/0
C    20.0.0.0/8 is directly connected, Serial2/0
S    30.0.0.0/8 [1/0] via 20.0.0.20
S    40.0.0.0/8 [1/0] via 20.0.0.20
Router#
```

Router 1:

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.10
Router(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.20
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 20.0.0.10
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 30.0.0.20
Router#
```

Router 2:

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 10.0.0.0 255.0.0.0 30.0.0.10
Router(config)#ip route 20.0.0.0 255.0.0.0 30.0.0.10
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 30.0.0.10
S    20.0.0.0/8 [1/0] via 30.0.0.10
C    30.0.0.0/8 is directly connected, Serial2/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
Router#
```

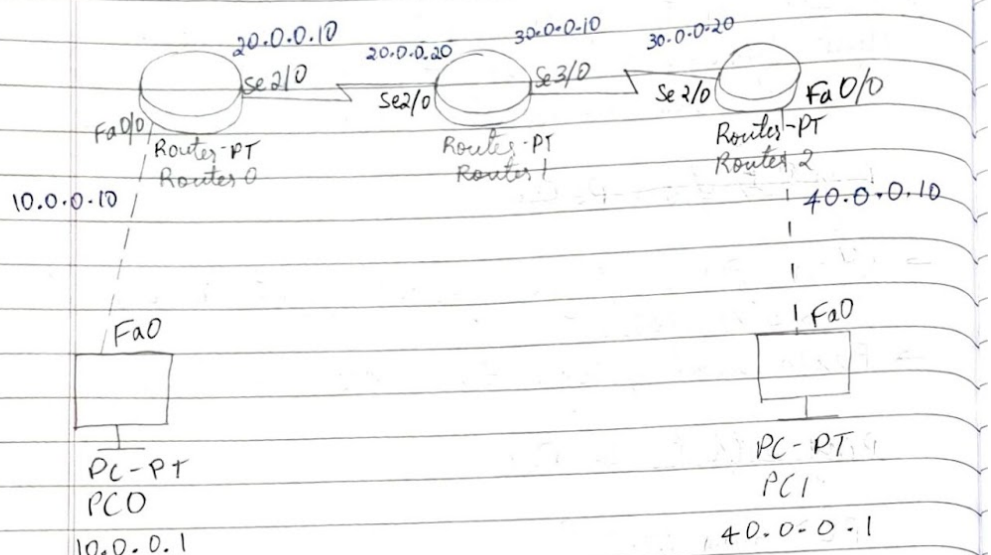
Ping output after static routing:

```
PC>ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data:

Request timed out.
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125

Ping statistics for 40.0.0.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 2ms, Average = 2ms

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=35ms TTL=125
Reply from 40.0.0.1: bytes=32 time=4ms TTL=125
Reply from 40.0.0.1: bytes=32 time=27ms TTL=125
Reply from 40.0.0.1: bytes=32 time=13ms TTL=125

Ping statistics for 40.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 35ms, Average = 19ms

PC>
```

Observation :

2B) Aim :- Configure using 3 routers and 2 PCs.

Topology :-



Procedure :

→ The Network is Started by selecting end devices PC0 and PC1 as generic PCs.

→ Select 3 generic routers, Router PTs and place all of them on the workspace.

→ Connect PC0 to Router 0 and PC1 to Router 2 using copper cross over cable.

→ Setup the IP address of the end devices PC0 & PC1,

by clicking config → FastEthernet 0 → IP address.
PC0 → 10.0.0.1
PC1 → 40.0.0.1

→ Configure the routers by opening CLI.

In Router 0 :-

```
Router >enable
Router # config t
Router (config) # interface fastethernet 0/0
Router (config-if) # ip address 10.0.0.10 255.0.0.0
Router (config-if) # no shut
exit
Router (config) # interface serial 2/0
Router (config-if) # ip address 20.0.0.10 255.0.0.0
Router (config-if) # no shut
exit
exit
```

In Router 1 :-

```
Router > enable
Router # config t
Router (config-if) # interface serial 2/0
Router (config-if) # ip address 20.0.0.20 255.0.0.0
Router (config-if) # no shut
Router (config-if) # exit
Router (config) # interface serial 3/0
Router (config-if) # ip address 30.0.0.10 255.0.0.0
Router (config-if) # no shut
exit
exit
```

In Router 2:-

```
Router>enable
Router# config t
Router(config)# interface serial 2/0
Router(config-if)# ip address 30.00.20 255.0.0.0
Router (config-if)# no shut.
exit
Router (config)# interface fastethernet 0/0
Router (config-if)# ip address 40.0.0.10 255.0.0.0
Router (config-if)# no shut
exit
exit
```

→ Set up Gateway for PC0 & PC1. PC0 : 10.0.0.10
PC1 : 40.0.0.10

IP Route table :-

Router 0:-

```
Router# Show ip route
C  10.0.0.0/8 is directly connected, Fast Ethernet 0/0
C  20.0.0.0/8 is directly connected, Serial 2/0
```

Router 1:-

```
Router# Show ip route
C  20.0.0.0/8 is directly connected, Serial 2/0
C  30.0.0.0/8 is directly connected, Serial 3/0
```

Router 2:

```
Router# Show ip route
C  30.0.0.0/8 is directly connected, Serial 2/0
C  40.0.0.0/8 is directly connected, FastEthernet 0/0
```

Ping Output : In PC0

PC>ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data:

Reply from 10.0.0.10 : Destination host unreachable
Reply from 10.0.0.10 : Destination host unreachable
Reply from 10.0.0.10 : Destination host unreachable

Ping statiestics for 40.0.0.1:
Packets : Sent = 4, Received = 0, Lost = 4 (100% loss)

~~Ob~~ Observation :-

→ Green lights appear on the wires when no shut commands are ~~ee~~ written indicating that they are ready for data transmission

→ Here, destination host was unreachable as Router 0 has no knowledge about the network 30.0.0.0 and 40.0.0.0 and the Packets got stuck. We need to set ip route explicitly.

→ Now configure the Routers which does not have data of the other network. ~~ADD~~ the network in CLI.

In Router 0:

ip route 30.0.0.0 255.0.0.0 20.0.0.20
ip route 40.0.0.0 255.0.0.0 20.0.0.20

Routes 1:

ip route 10.0.0.0 255.0.0.0 20.0.0.10
ip route 40.0.0.0 255.0.0.0 30.0.0.20

Routes 2:

ip route 10.0.0.0 255.0.0.0 30.0.0.10
ip route 20.0.0.0 255.0.0.0 30.0.0.10

IP route table now :-

Routes 0:

C 10.0.0.0/8 is directly connected, FastEthernet 0/0
C 20.0.0.0/8 is directly connected, Serial 2/0
S 30.0.0.0/8 [1/0] via 20.0.0.20
S 40.0.0.0/8 [1/0] via 20.0.0.20

Routes 1:

S 10.0.0.0/8 [1/0] via 20.0.0.10
C 20.0.0.0/8 is directly connected, Serial 2/0
C 30.0.0.0/8 is directly connected, Serial 3/0
S 40.0.0.0/8 [1/0] via 30.0.0.20

Routes 2:

S 10.0.0.0/8 [1/0] via 30.0.0.10
S 20.0.0.0/8 [1/0] via 30.0.0.10
C 30.0.0.0/8 is directly connected, Serial 2/0
C 40.0.0.0/8 is directly connected, FastEthernet 0/0

Ping Output now:

PC> Ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data:

Request timed out.
Reply from 40.0.0.1 : bytes = 32 time = 2ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 2ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 2ms TTL = 125

Ping statistics for 40.0.0.1:
Packets : Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 2ms, Average = 2ms

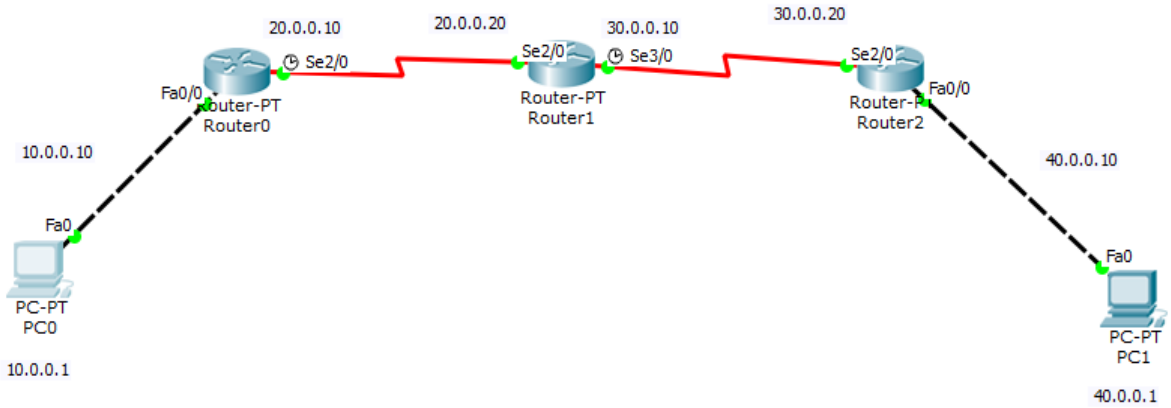
→ Now, when PC1 is pinged for the first time, there is 25% loss
→ From next ping, there are no losses.

13/7/2023

Experiment 3

Aim: Configure default route, static route to the Router

Topology:



Configurations:

Roter 0 :

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 10.0.0.10 255.0.0.0
Router(config-if)#no shut

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

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#interface serial 2/0
Router(config-if)#ip address 20.0.0.10 255.0.0.0
Router(config-if)#no shut

%LINK-5-CHANGED: Interface Serial2/0, changed state to down
Router(config-if)#exit
Router(config)#exit

Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 20.0.0.20
Router(config)#
```

```
Router>enable
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 20.0.0.20 to network 0.0.0.0

C    10.0.0.0/8 is directly connected, FastEthernet0/0
C    20.0.0.0/8 is directly connected, Serial2/0
S*   0.0.0.0/0 [1/0] via 20.0.0.20
Router#
```

Roter 1 :

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
interface serail 3/0
                         ^
% Invalid input detected at '^' marker.

Router(config)#interface serial 3/0
Router(config-if)#ip address 30.0.0.10 255.0.0.0
Router(config-if)#no shut

%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#exit
```

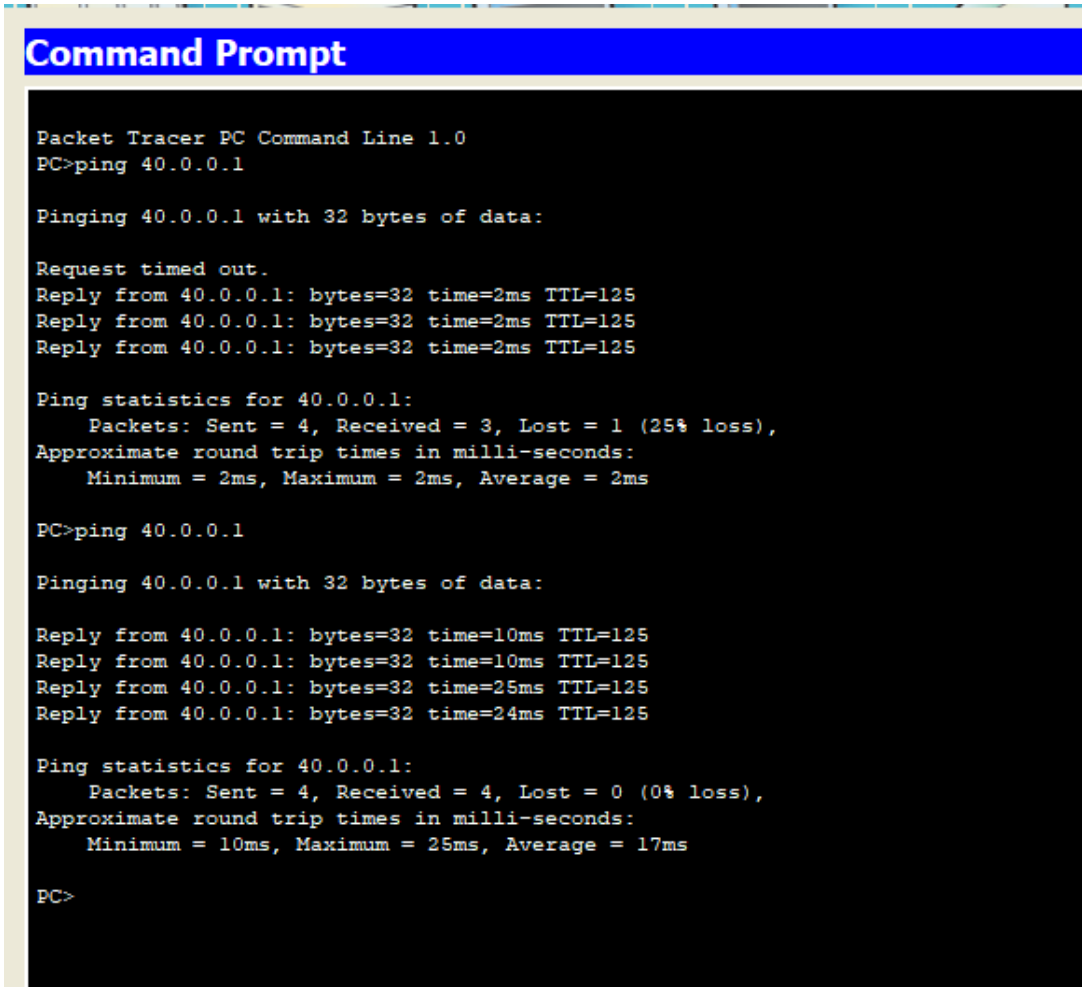
```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.10
Router(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.20
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
show ip route
Codes: C - connected, 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 20.0.0.10
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 30.0.0.20
```

Roter 2 :

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface serail 2/0
                             ^
% Invalid input detected at '^' marker.

Router(config)#interface serial 2/0
Router(config-if)#ip address 30.0.0.20 255.0.0.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exi
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
t
Router(config)#exit
```

```
Router>enable
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 30.0.0.10
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 30.0.0.10 to network 0.0.0.0

C    30.0.0.0/8 is directly connected, Serial2/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
S*   0.0.0.0/0 [1/0] via 30.0.0.10
Router#
```

Ping Output:


Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data:
Request timed out.
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 2ms, Average = 2ms
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=10ms TTL=125
Reply from 40.0.0.1: bytes=32 time=10ms TTL=125
Reply from 40.0.0.1: bytes=32 time=25ms TTL=125
Reply from 40.0.0.1: bytes=32 time=24ms TTL=125
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 10ms, Maximum = 25ms, Average = 17ms
PC>

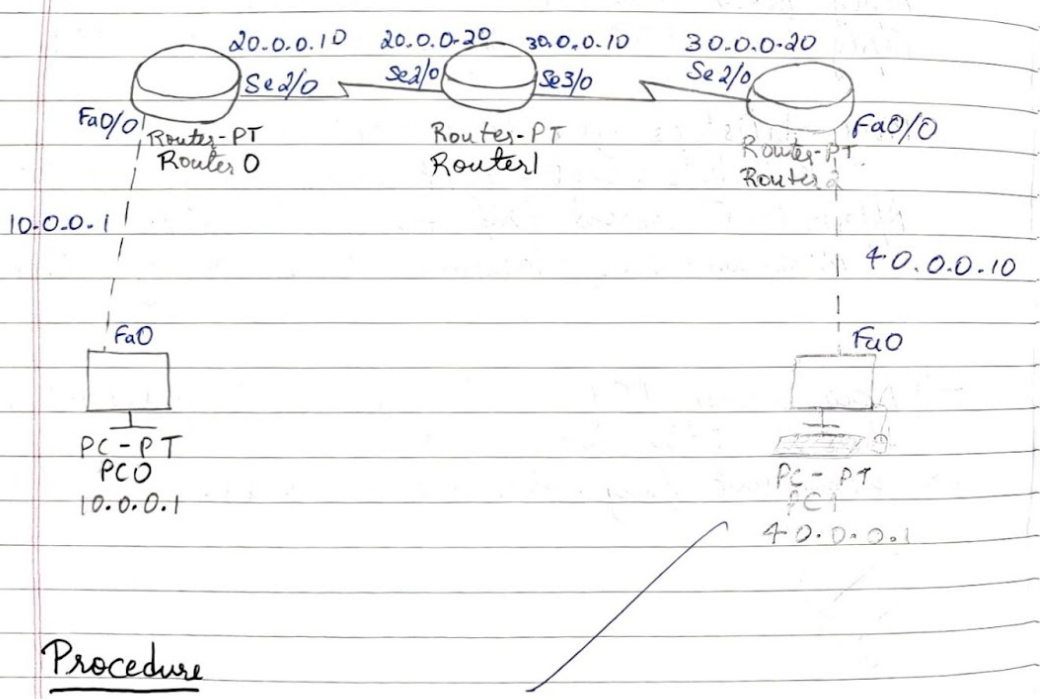
Observation:

13/7/'23

Lab 3 :

Aim: Configure default route, static route to the Router.

Topology :



Procedure

→ Create the same topology as 2B experiment.

→ Setup IP addres of PC0 & PC1
PC0 → 10.0.0.1
PC1 → 40.0.0.1
config → Fastethernet0 → IP address

→ Config IP address of Router 0, Router1 & Router 2 following steps in 2B experiment.

→ Setup Gateway for PC0 & PC1 :

PC0 : 10.0.0.10

PC1 : 40.0.0.10

→ Setup static routing for Router1 ~~and Router 2~~ following steps in 2B experiment

→ For Router 0 and Router 2, we will setup default routing.

Router 0:

Router(config) # ip route 0.0.0.0 0.0.0.0 20.0.0.20

Router # show ip route

C 10.0.0.0/8 is directly connected, FastEthernet0/0
C 20.0.0.0/8 is directly connected,
S* 0.0.0.0/0 [1/0] via 20.0.0.20

Router 2:

Router(config) # ip route 0.0.0.0 0.0.0.0 30.0.0.10

Router # Show ip route

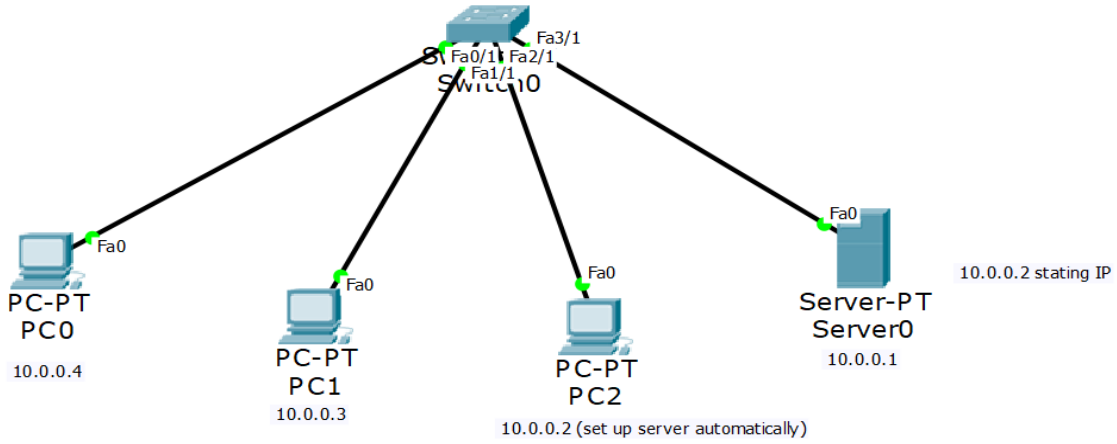
C 30.0.0.0/8 is directly connected, Serial 2/0
C 40.0.0.0/8 is directly connected, FastEthernet 0/0
S* 0.0.0.0/0 [1/0] via 30.0.0.10

Experiment 4

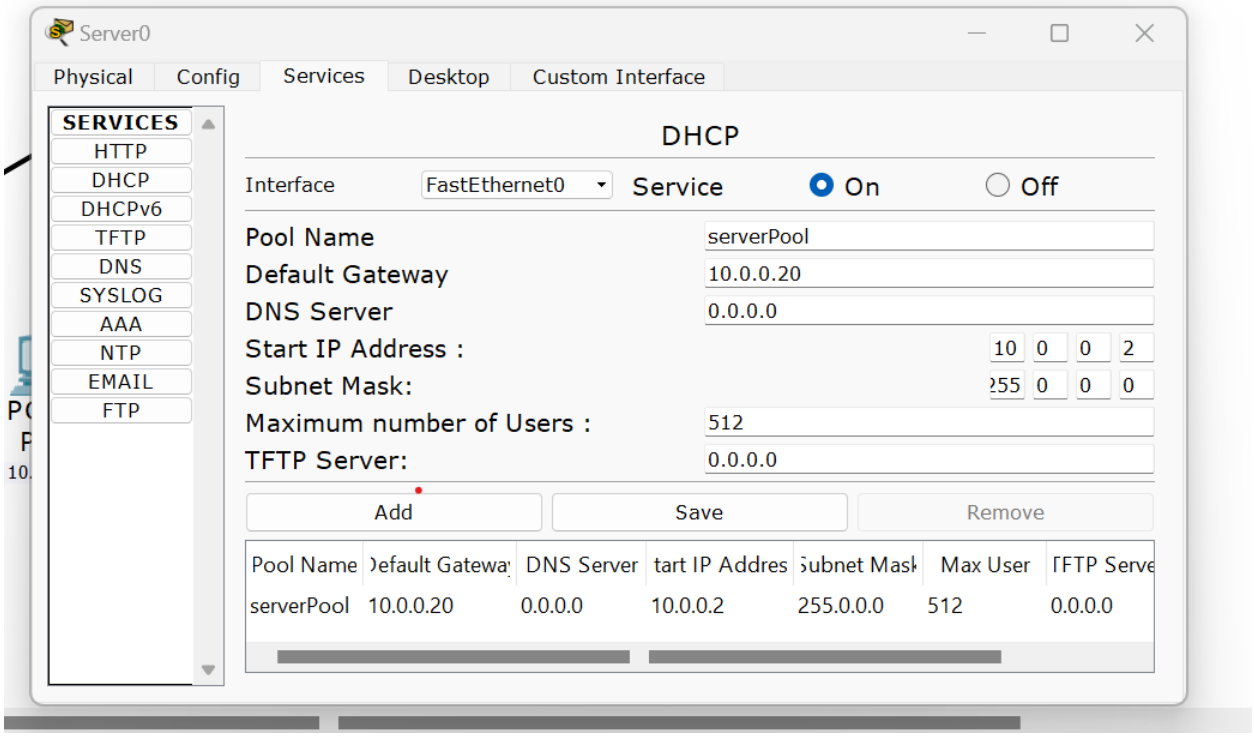
Aim: Configure DHCP within a LAN and outside LAN.

4A: Within a LAN.

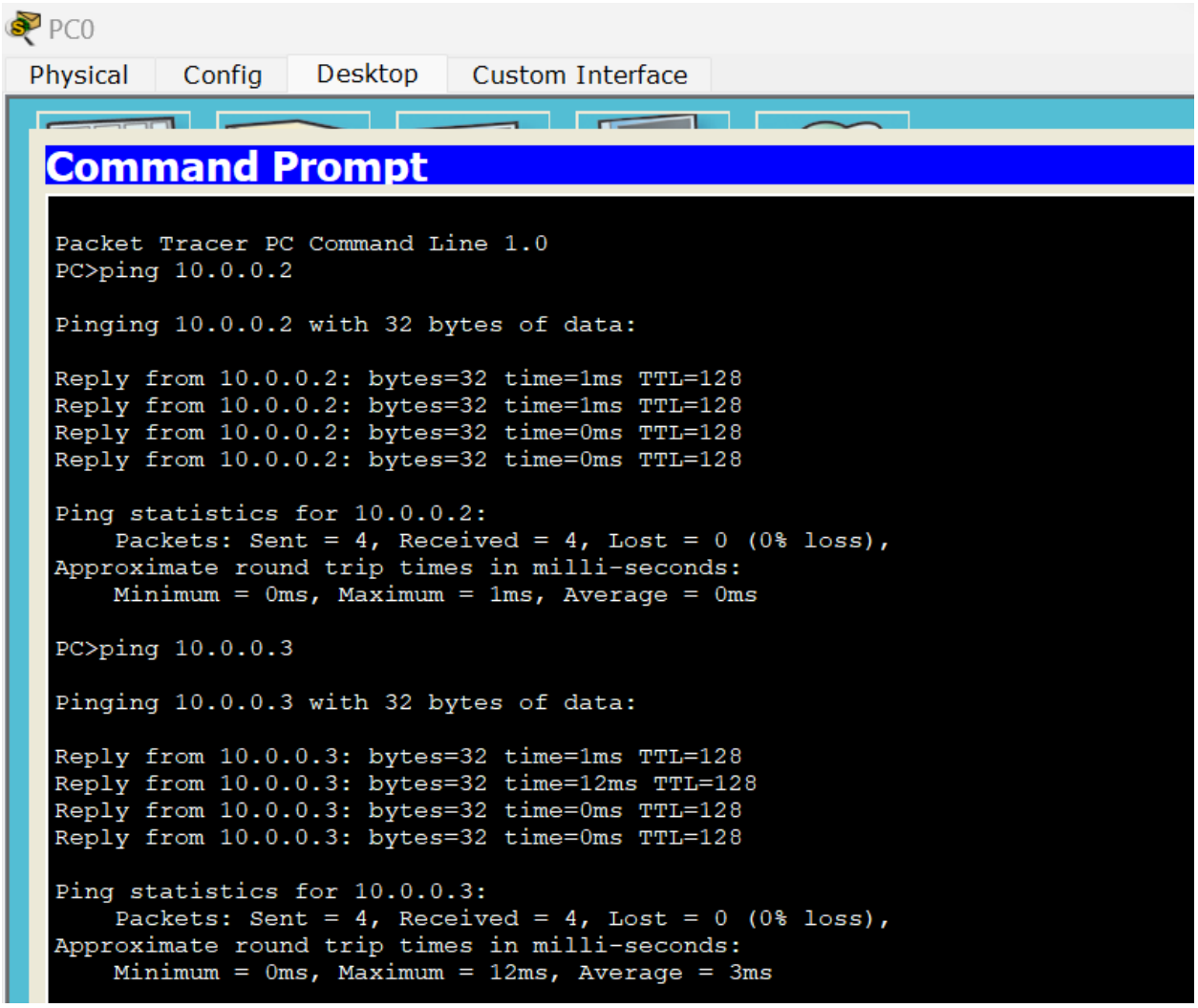
Topology:



Server 0 :



Ping Output :


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
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=1ms TTL=128
Reply from 10.0.0.2: bytes=32 time=1ms TTL=128
Reply from 10.0.0.2: bytes=32 time=0ms TTL=128
Reply from 10.0.0.2: bytes=32 time=0ms 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 = 0ms, Maximum = 1ms, Average = 0ms
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=1ms TTL=128
Reply from 10.0.0.3: bytes=32 time=12ms 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, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 12ms, Average = 3ms

Observation :

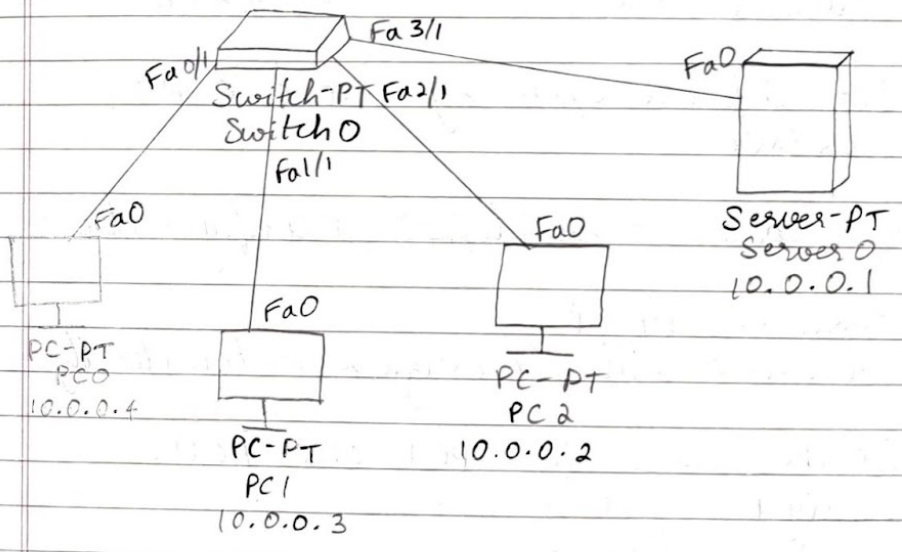
13/7/23

Lab 4:

Aim: Configure DHCP within a LAN and outside LAN.

4A :-

Topology :



Procedure

1. The network is started by selecting end devices PC-PTs, PC0, PC1 and PC2. Select Server-PT too.

2. Select a generic switch / Switch-PT and place all of them on the workspace.

3. Connect PCs and Server to switch using copper Straight-Through cable.

4. Set the IP address of server 0
Desktop → IP configuration → IP Address = 10.0.0.1
Click on subnet mask.

5. Server 0 → Services → DHCP → On

Set start IP address as 10.0.0.2
Subnet Mask: 255.0.0.0

6. Set default Gateway : 10.0.0.20.
↳ Save

7. Open PC2, → Desktop → IP configuration

Turn on DHCP
10.0.0.2 will be assigned automatically.

8. Do the same for PC1 and PC0.
PC1 : 10.0.0.3
PC0 : 10.0.0.4

Ping Output:
In PC0
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 = 1ms TTL = 128

Reply from 10.0.0.2 : bytes = 32 time = 1ms : TTL = 128
Reply from 10.0.0.2 : bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.2 : bytes = 32 time = 0ms 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 = 0ms, Maximum = 1ms, Average = 0 ms.

Observation :

→ We observe that IP addresses are set automatically to PC0, PC1 and PC2 when we enable DNCP IP configuration.

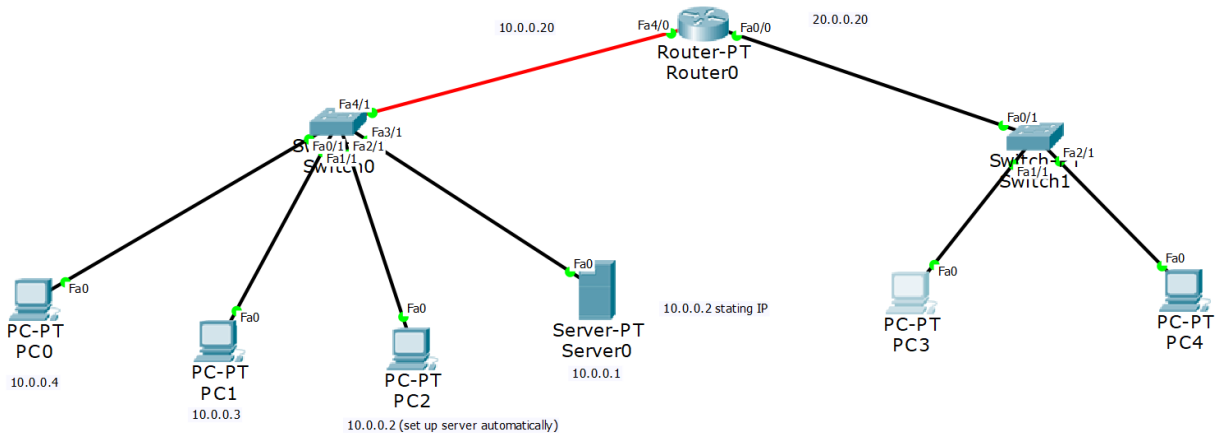
→ This is useful for large networks with 100s of PCs.

→ For all PCs Gateway is automatically set to 10.0.0.20.

17/8/2023

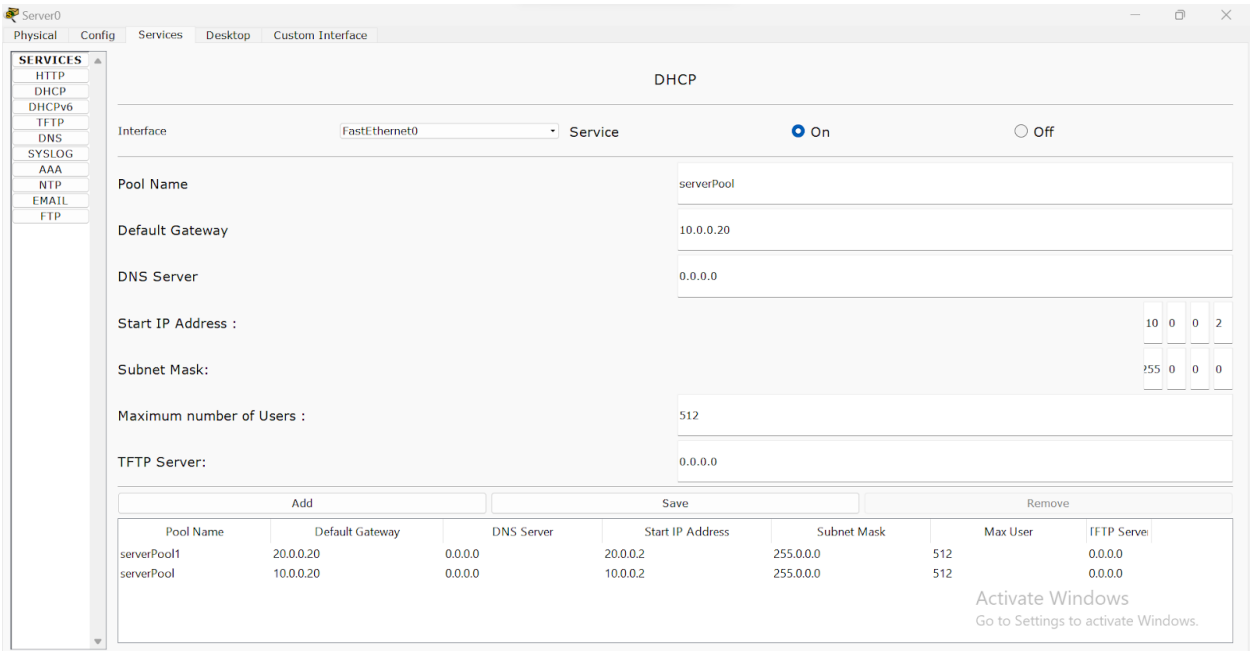
4B: Outside a LAN.

Topology:

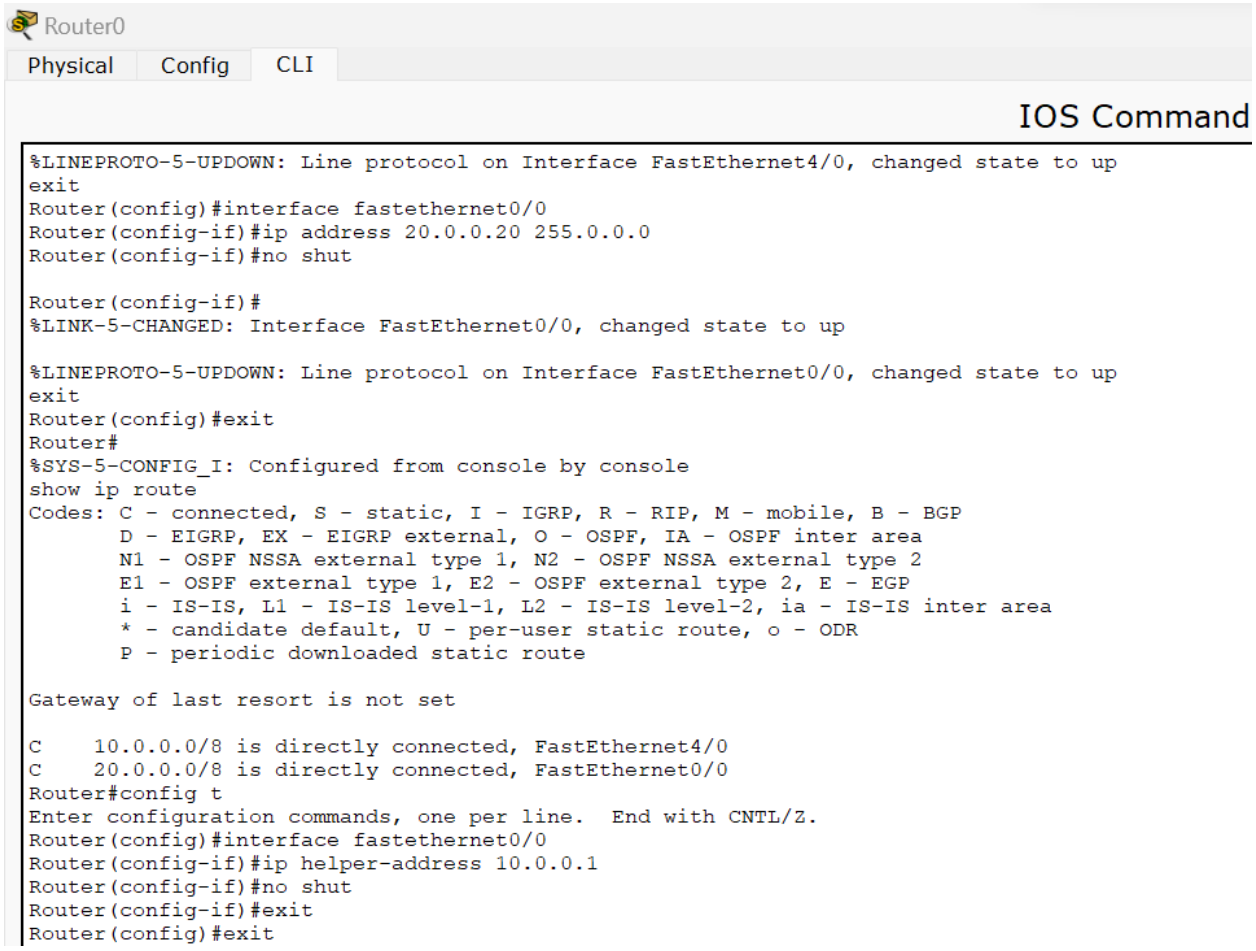


Configurations :

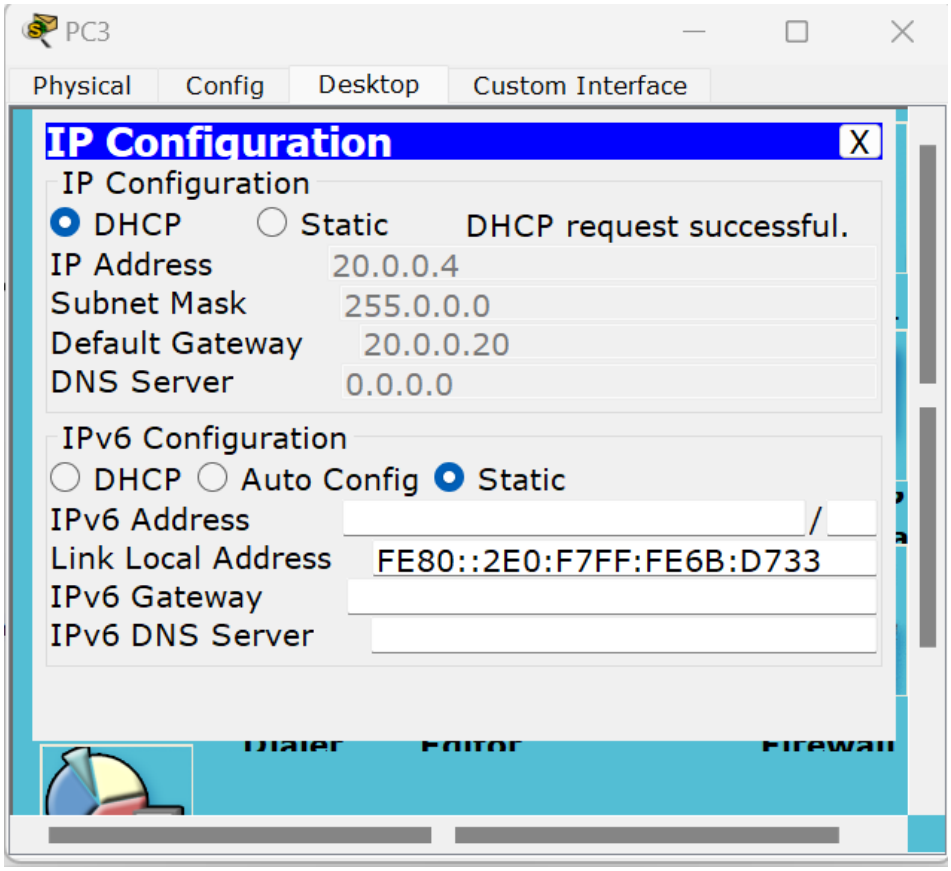
Server 0 :



Router 0 :


Router0
Physical
Config
CLI
IOS Command
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet4/0, changed state to up
exit
Router(config)#interface fastethernet0/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
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 10.0.0.0/8 is directly connected, FastEthernet4/0
C 20.0.0.0/8 is directly connected, FastEthernet0/0
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fastethernet0/0
Router(config-if)#ip helper-address 10.0.0.1
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#exit

Automation IP is assigned in the PCs by Server 0 via DHCP:



Ping Output :

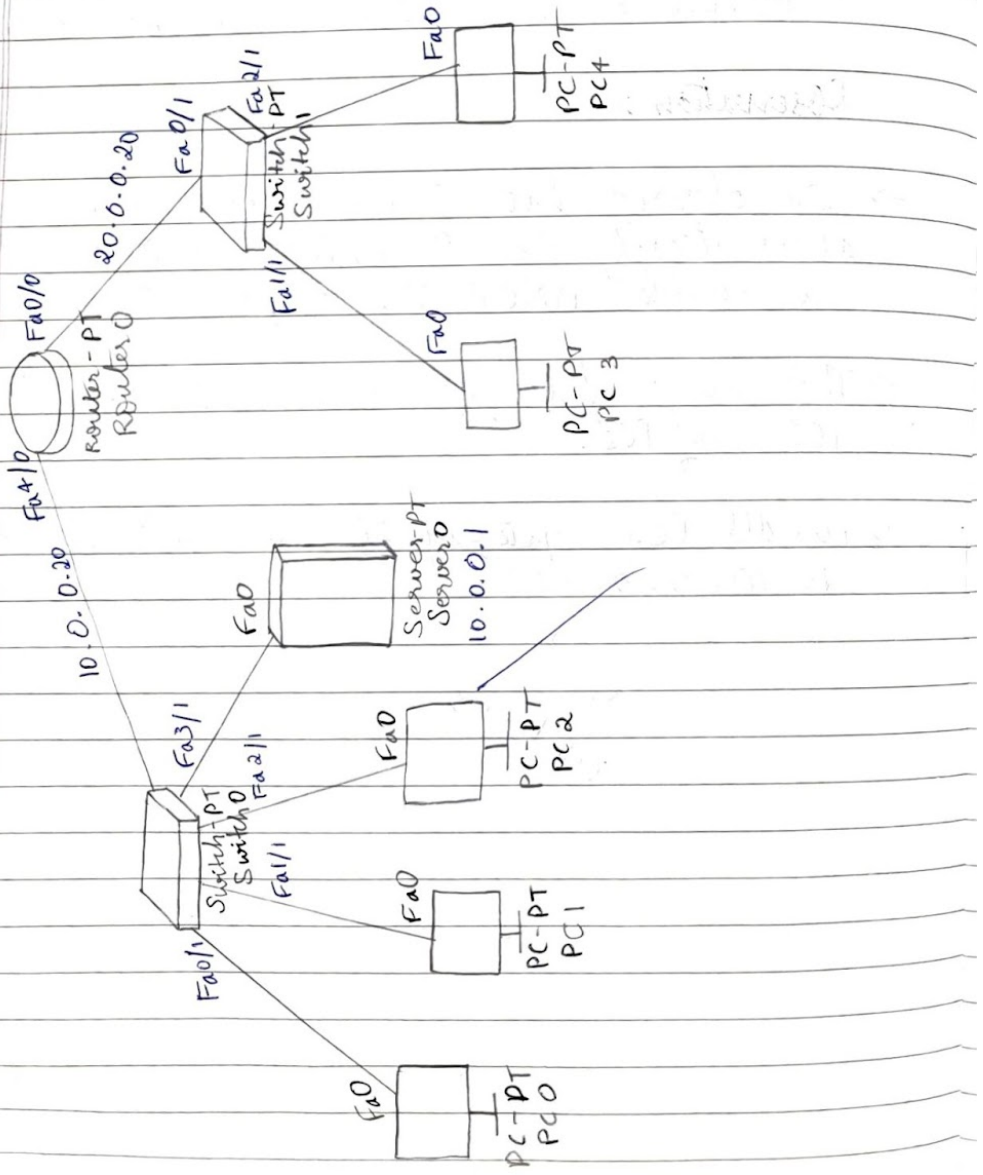
```
PC>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time=1ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127

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

Observation :

4B: (Continuation of 4A)

Aim: Configure DHCP ~~and~~ outside LAN.

Topology:



Procedure :-

1. To the topology created in 4A, connect a ~~Bo~~ Generic Router 0 using copper Straight through wire.

2. To a Switch 1, connect 2 PCs, PC0 and PC1.

3. Connect the step 2 topology to Router 0.

4. In Router 0, open CLI.
   Set IP addresses for interfaces.
   Fa 4/0 : 10.0.0.20
   Fa 0/0 : 20.0.0.20

5. In Router 0,
   interface fastethernet 0/0
   Router (config-if) # ip helper-address 10.0.0.1
   Router (config-if) # no shoshut

6. In Server 0,
   ↳ Config → Settings → Gateway: 10.0.0.20

7. In Server 0,
   ↳ Services → DHCP → Pool Name: serverPool1
   Set Default Gateway : 20.0.0.20

   Start IP Address : 20.0.0.2
   Subnet Mask : 255.0.0.0

   ↳Add.

8. PC 3
↳ Desktop ↳ IP Configuration ↳ DHCP
20.0.0.2 set automatically for PC 3

PC 4
20.0.0.3 set automatically.

Ping Output
On PC 0

PC> 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=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127

~~Observation:~~

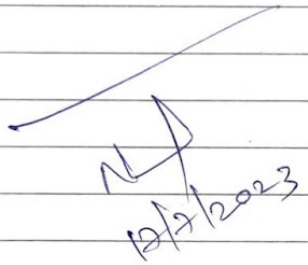
→ ~~IP address is set Automa~~
Ping statistics for 20.0.0.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms

Observation:

→ IP addresses are set automatically on PC3 & PC4 by Server 0.

→ PC3 : 20.0.0.2
PC4 : 20.0.0.3

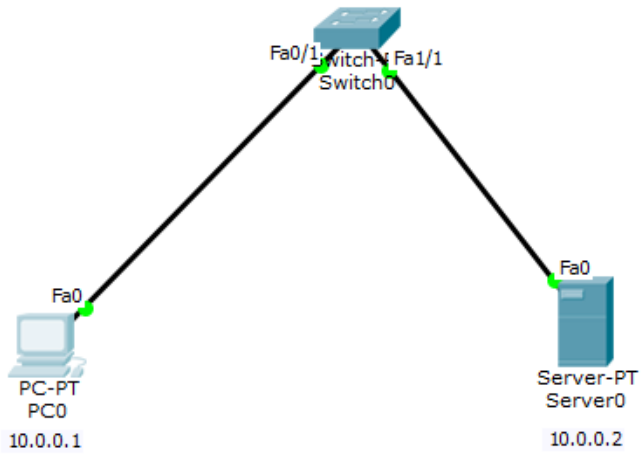
→ We can successfully ping PC3 from PC0 with no loss.



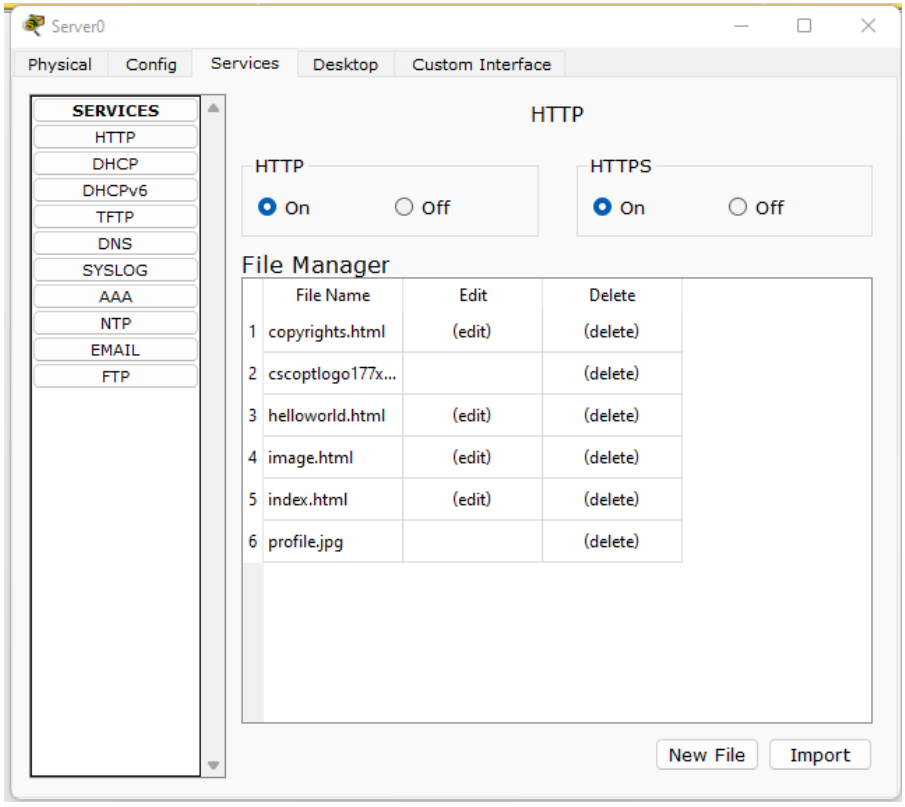
Experiment 5

Aim : Configure Web Server, DNS within a LAN.

Topology:

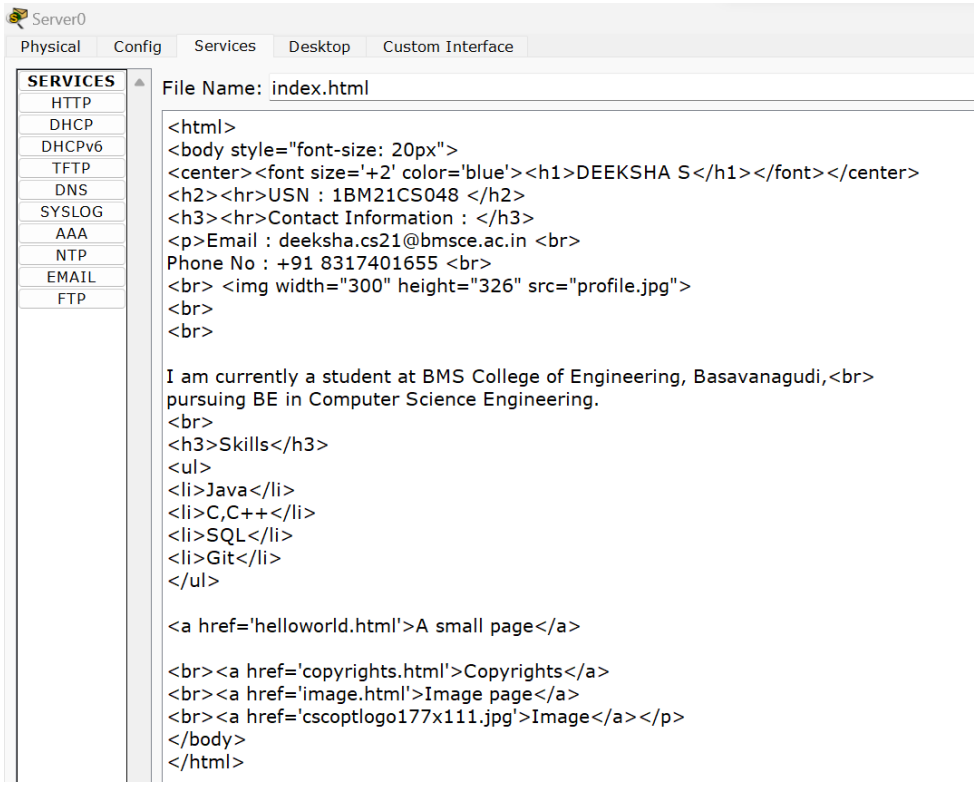


Server 0 :



Website:

Code:



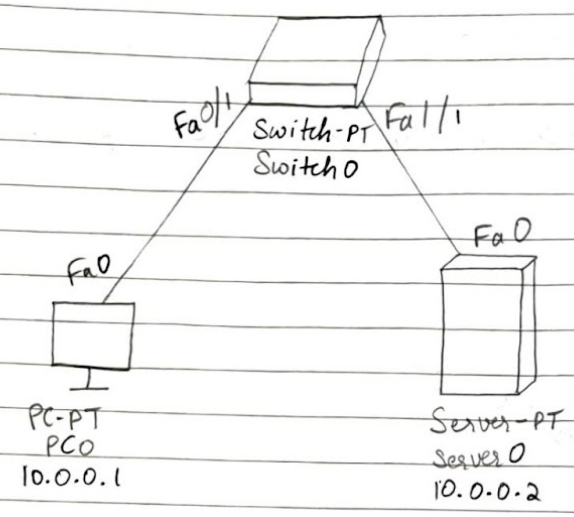
Observation :

20/7/23

Aim:

Lab 5: Configure Web Server, DNS within a LAN

Topology:



Procedure :-

→ The Network is started by selecting end devices PC-PT and Server-PT.

→ Select Switch-PT and add all of them in the workspace.

→ Connect PC0 and Server 0 to Switch-0 using copper Straight-Through cable.

→ Set IP address for PC0 and Server 0
config → Fastethernet → IP address

PC0 : 10.0.0.1
Server0 : 10.0.0.2

→ Open desktop in PC → Desktop → Web Browser
In URL, type 10.0.0.2 and click Go.
You can see Cisco Packet Tracer website.

→ In Server0, → Services → HTTP, then edit index.html.
Code the HTML & CSS and create your CV.

Output :-

→ In PC0 → Desktop → Webbrowser → URL
type 10.0.0.2.
The created CV will be visible.

→ In Server 0, → Services → DNS → On
Name : deeksha
Address : 10.0.0.2
Click on add.

Observation :-

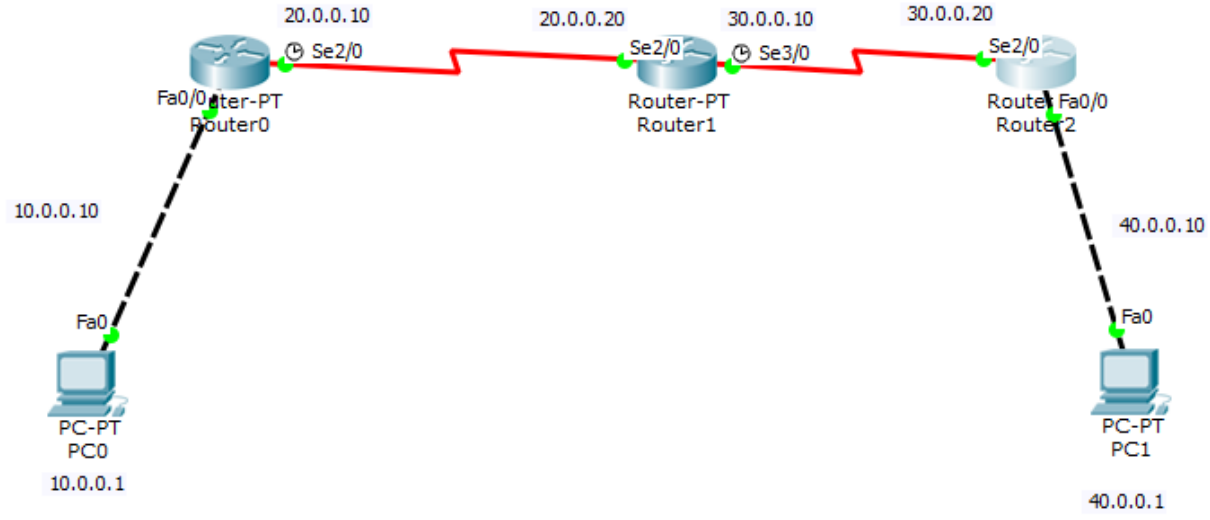
→ In PC0 → Desktop → Webbrowser → URL
When we type the domain name "deeksha" the website created in Server0 opens.

24/7/2023

Experiment 6

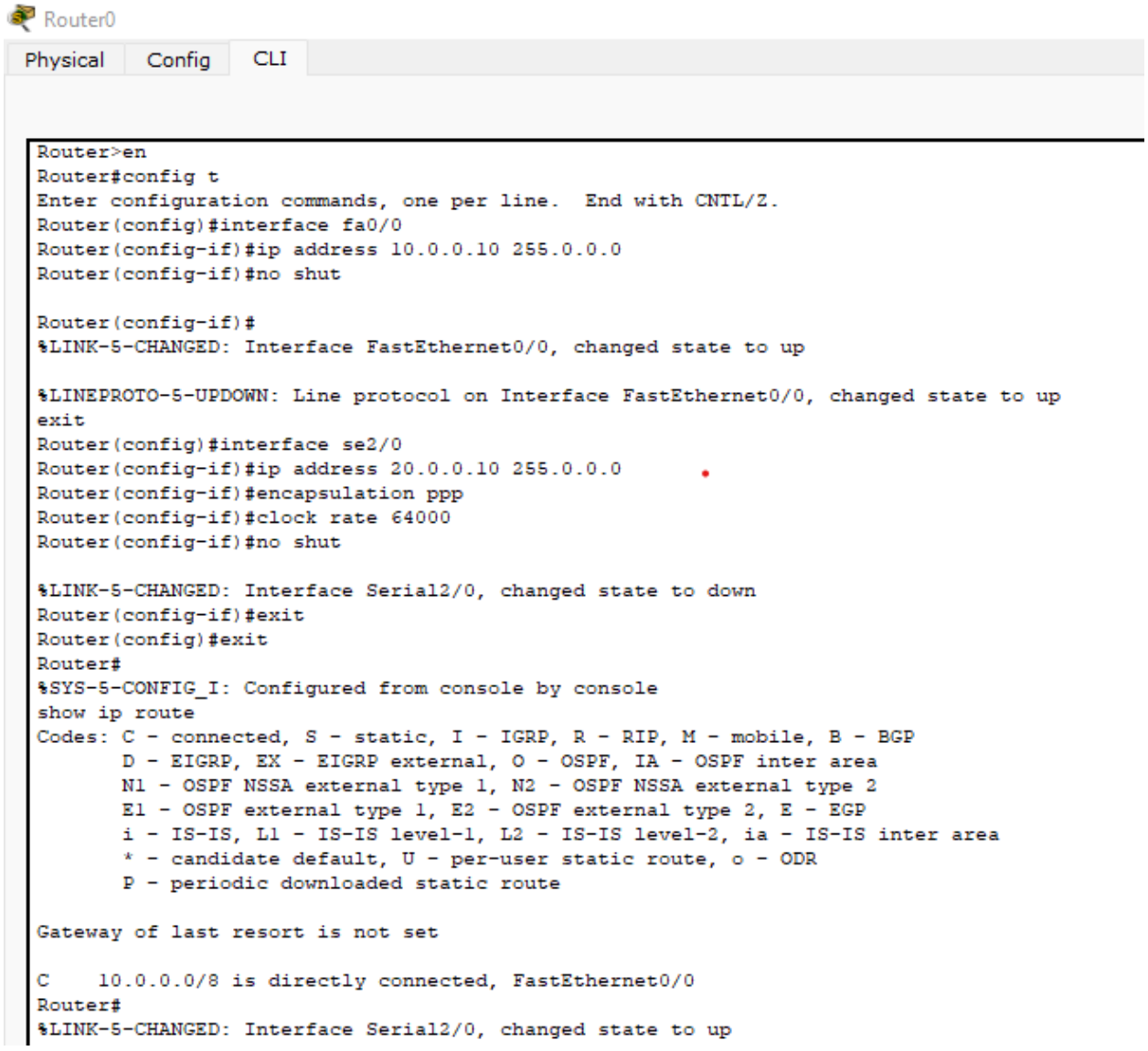
Aim: Configure RIP routing Protocol in Routers.

Topology:

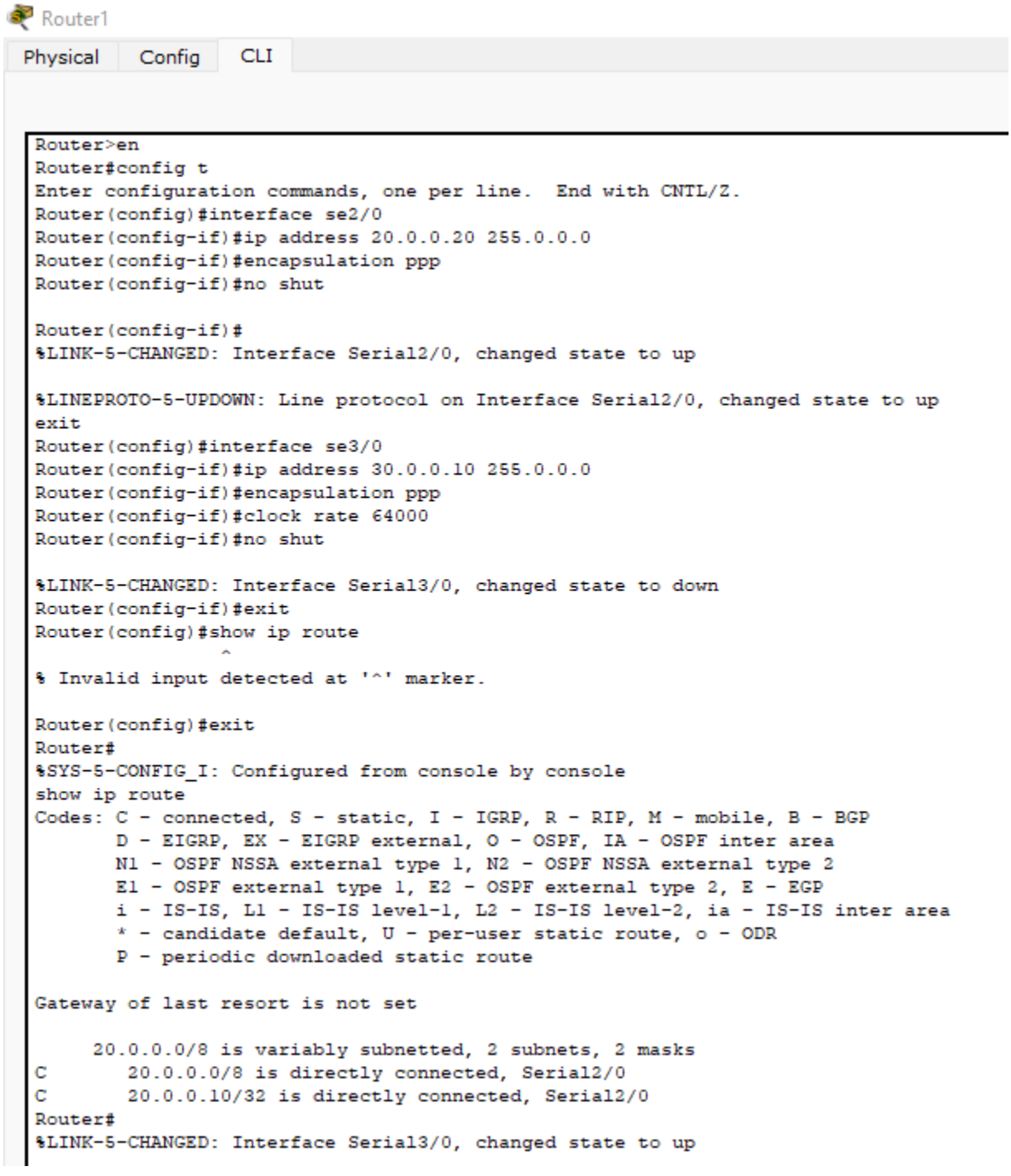


Configuration:

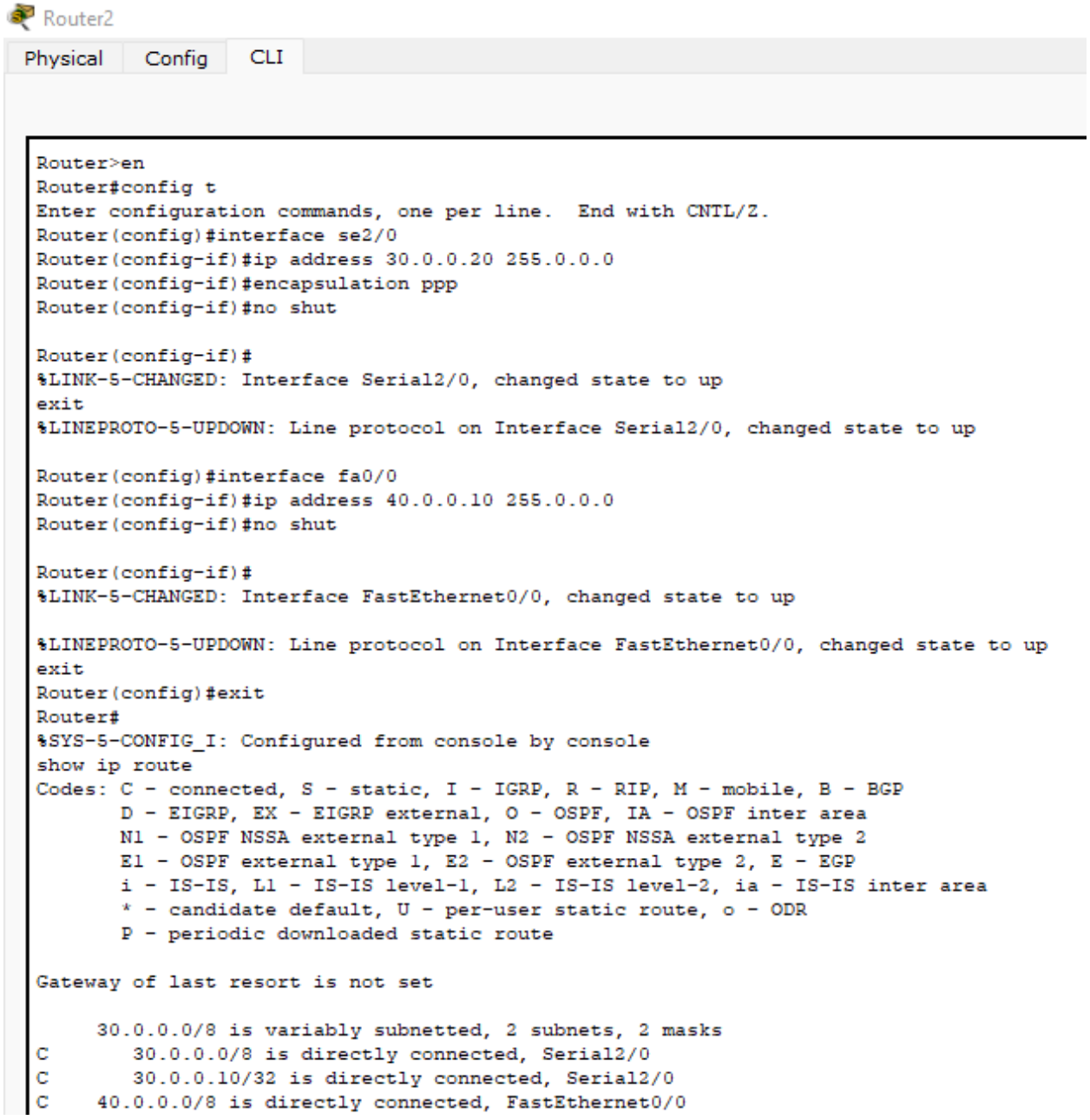
Router 0 :



Router 1 :


Router1
Physical
Config
CLI
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface se2/0
Router(config-if)#ip address 20.0.0.20 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#no shut
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
exit
Router(config)#interface se3/0
Router(config-if)#ip address 30.0.0.10 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#clock rate 64000
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#show ip route
^
% Invalid input detected at '^' marker.
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
20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 20.0.0.0/8 is directly connected, Serial2/0
C 20.0.0.10/32 is directly connected, Serial2/0
Router#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

Router 2 :


Router2
Physical
Config
CLI
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface se2/0
Router(config-if)#ip address 30.0.0.20 255.0.0.0
Router(config-if)#encapsulation ppp
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router(config)#interface fa0/0
Router(config-if)#ip address 40.0.0.10 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
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
30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C 30.0.0.0/8 is directly connected, Serial2/0
C 30.0.0.10/32 is directly connected, Serial2/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0

RIP routing:

Router 0:

```
Router#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 20.0.0.0
Router(config-router)#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    10.0.0.0/8 is directly connected, FastEthernet0/0
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial2/0
C       20.0.0.20/32 is directly connected, Serial2/0
R    30.0.0.0/8 [120/1] via 20.0.0.20, 00:00:18, Serial2/0
R    40.0.0.0/8 [120/2] via 20.0.0.20, 00:00:18, Serial2/0
Router#
```

Router 1:

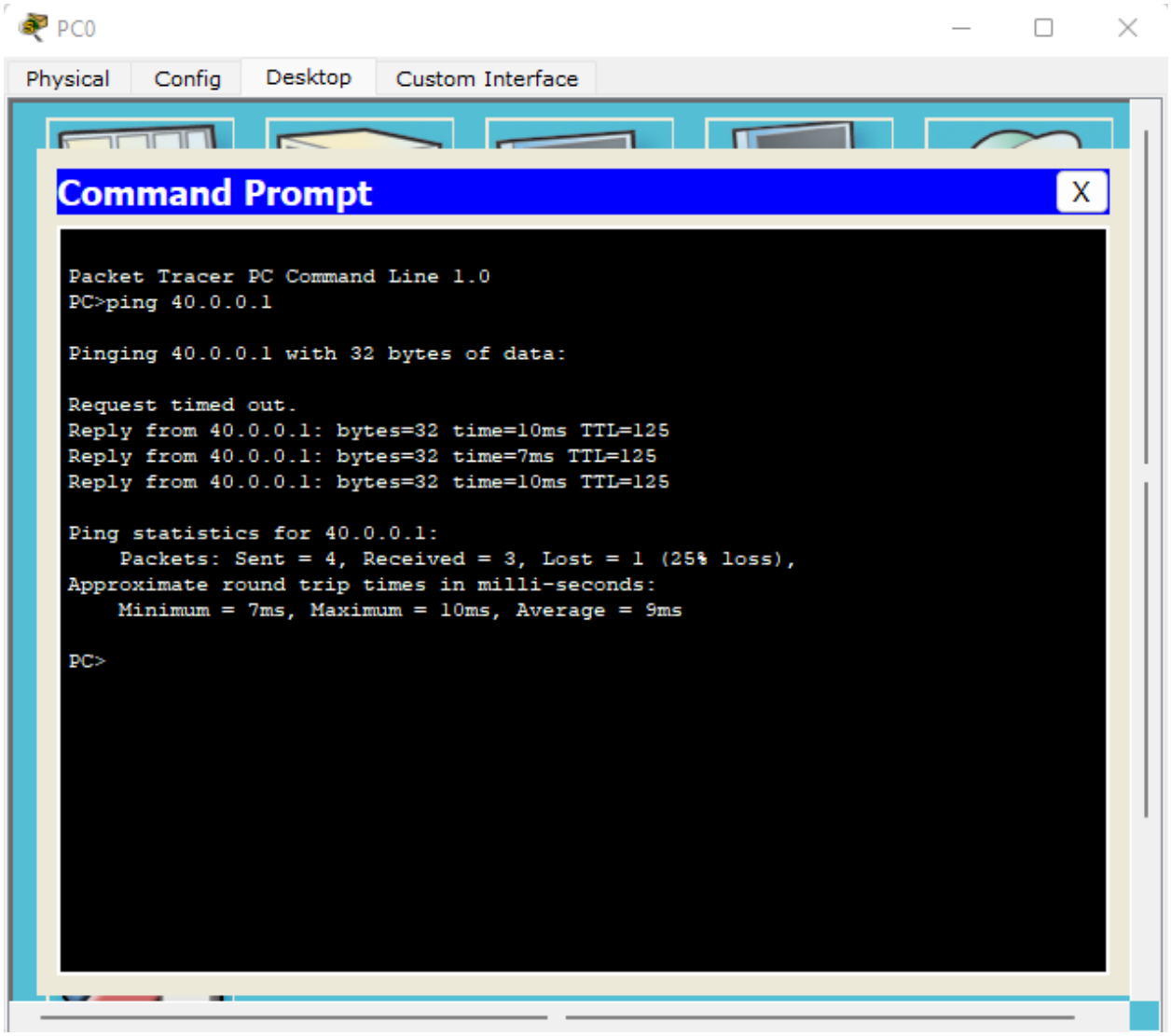
```
Router#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 20.0.0.0
Router(config-router)#network 30.0.0.0
Router(config-router)#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

R    10.0.0.0/8 [120/1] via 20.0.0.10, 00:00:20, 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       20.0.0.10/32 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       30.0.0.20/32 is directly connected, Serial3/0
R    40.0.0.0/8 [120/1] via 30.0.0.20, 00:00:19, Serial3/0
Router#
```

Router 2:

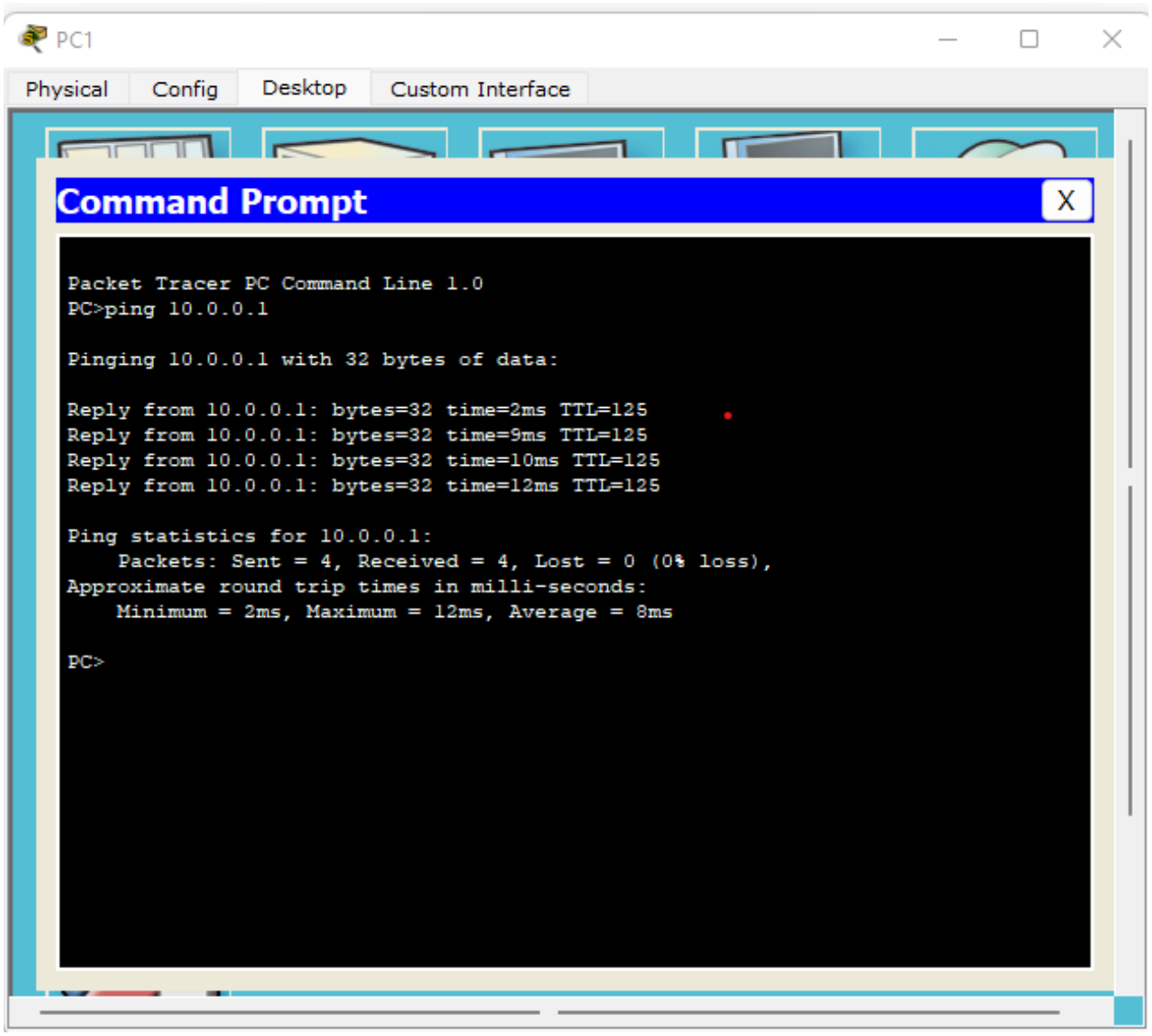
```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#network 30.0.0.0
                ^
% Invalid input detected at '^' marker.

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

R    10.0.0.0/8 [120/2] via 30.0.0.10, 00:00:14, Serial2/0
R    20.0.0.0/8 [120/1] via 30.0.0.10, 00:00:14, 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       30.0.0.10/32 is directly connected, Serial2/0
C    40.0.0.0/8 is directly connected, FastEthernet0/0
Router#
```

Ping Outputs:

P0:



P1:

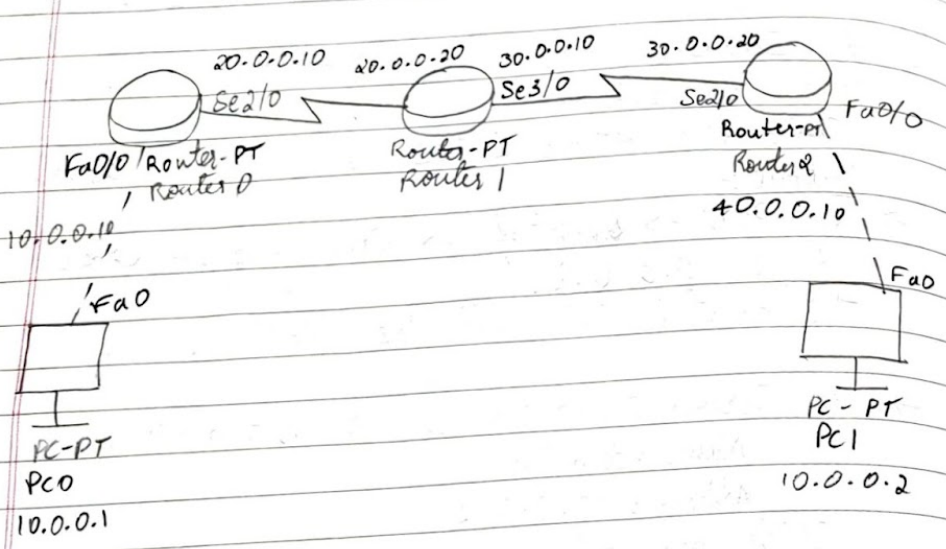

PC1
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=9ms TTL=125
Reply from 10.0.0.1: bytes=32 time=10ms TTL=125
Reply from 10.0.0.1: bytes=32 time=12ms TTL=125
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 12ms, Average = 8ms
PC>

Observation :

20/1/'23.

Aim: Lab 6 : Configure RIP routing Protocol in Routes.

Topology :-



Procedure :

→ Repeat the topology in 4B experiment.

→ Set IP address of PC0 & PC1
PC0 : 10.0.0.1
PC1 : 10.0.0.2

→ Set Gateway for PC0 & PC1
PC0 : 10.0.0.10 PC1 : ~~40.0.0.20~~ 40.0.0.10

→ Set IP addresses of Routers' interfaces using previous procedure.

Now:

Router 0:

Router(config)# interface serial 2/0
config-if# encapsulation ppp
Router clock rate 64000
no shut
exit

Router 1:

interface serial 2/0
encapsulation ppp
no shut
exit
interface se 3/0
encapsulation ppp
clock rate 64000
no shut
exit

Router 2:

interface se 2/0
encapsulation ppp
no shut
exit
interface fa 0/0
no shut
exit

Router 0:-

```
Router(config)# router rip
Router(config-router)# network 10.0.0.0
Router(config-router)# network 20.0.0.0
exit
exit
```

Router 1:

```
Router(config)# router rip
Router(config-router)# network 20.0.0.0
Router(config-router)# network 30.0.0.0
exit
exit
```

Router 2:

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

IP routing :-

We see that in each Router C indicates direct connection.

We see that RIP connection is established in each Router

Ping Output :-

PC> ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data :

Request timed out.
Reply from 40.0.0.1 : bytes = 32 time = 10ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 7ms TTL = 125
Reply from 40.0.0.1 : bytes = 32 time = 10ms TTL = 125

Ping statistics for 40.0.0.1 :
Packets : Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds :
Minimum = 7ms, Maximum = 10ms, Average = 9ms

Observation :-

→ Through RIP, routing is established in the network.

→ First packet is lost as we have seen before.

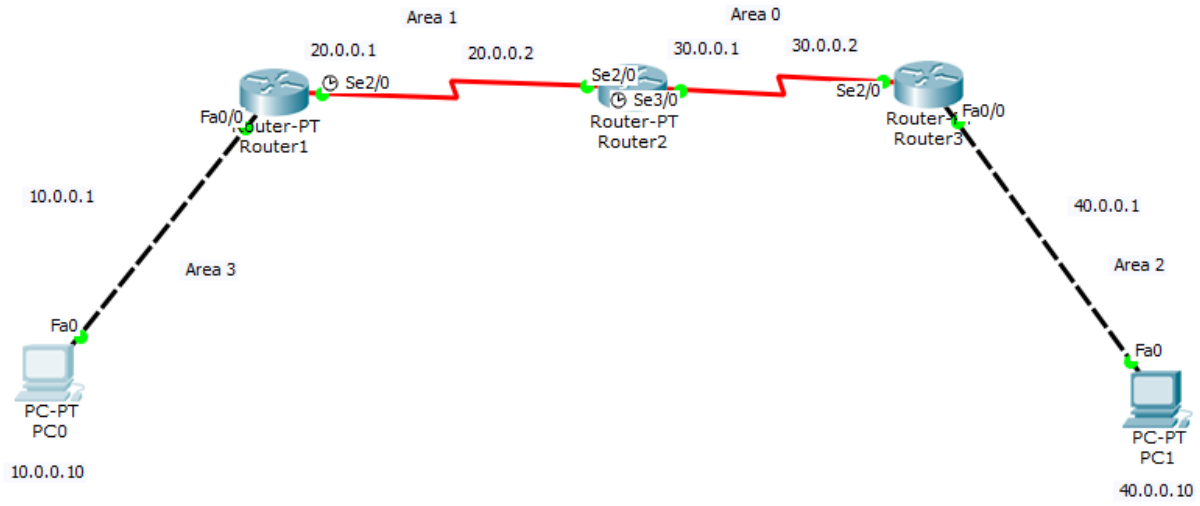
→ Then, remaining packets are transmitted successfully.

21/7/2023

Experiment 7

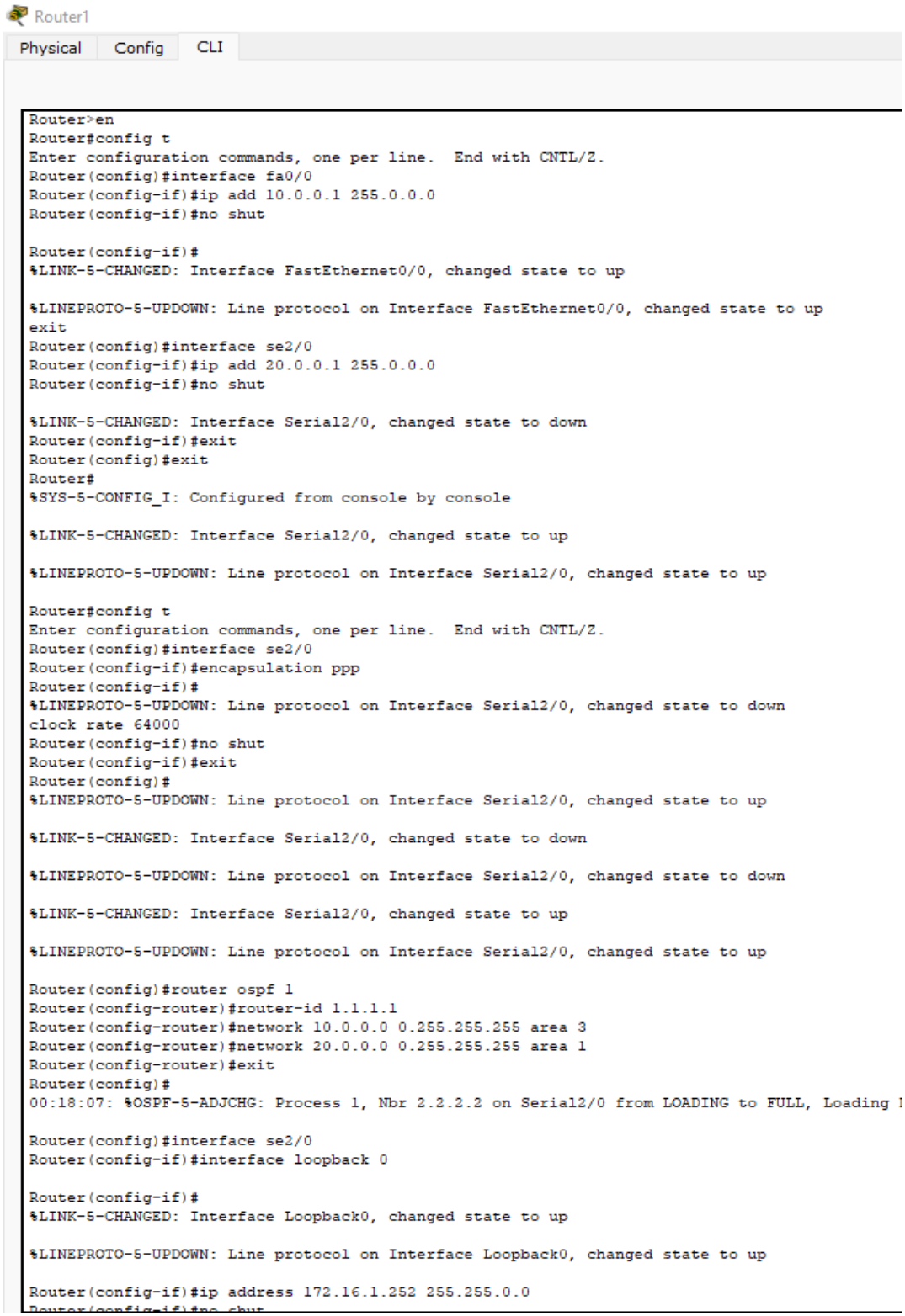
Aim : Configure OSPF routing protocol.

Topology:



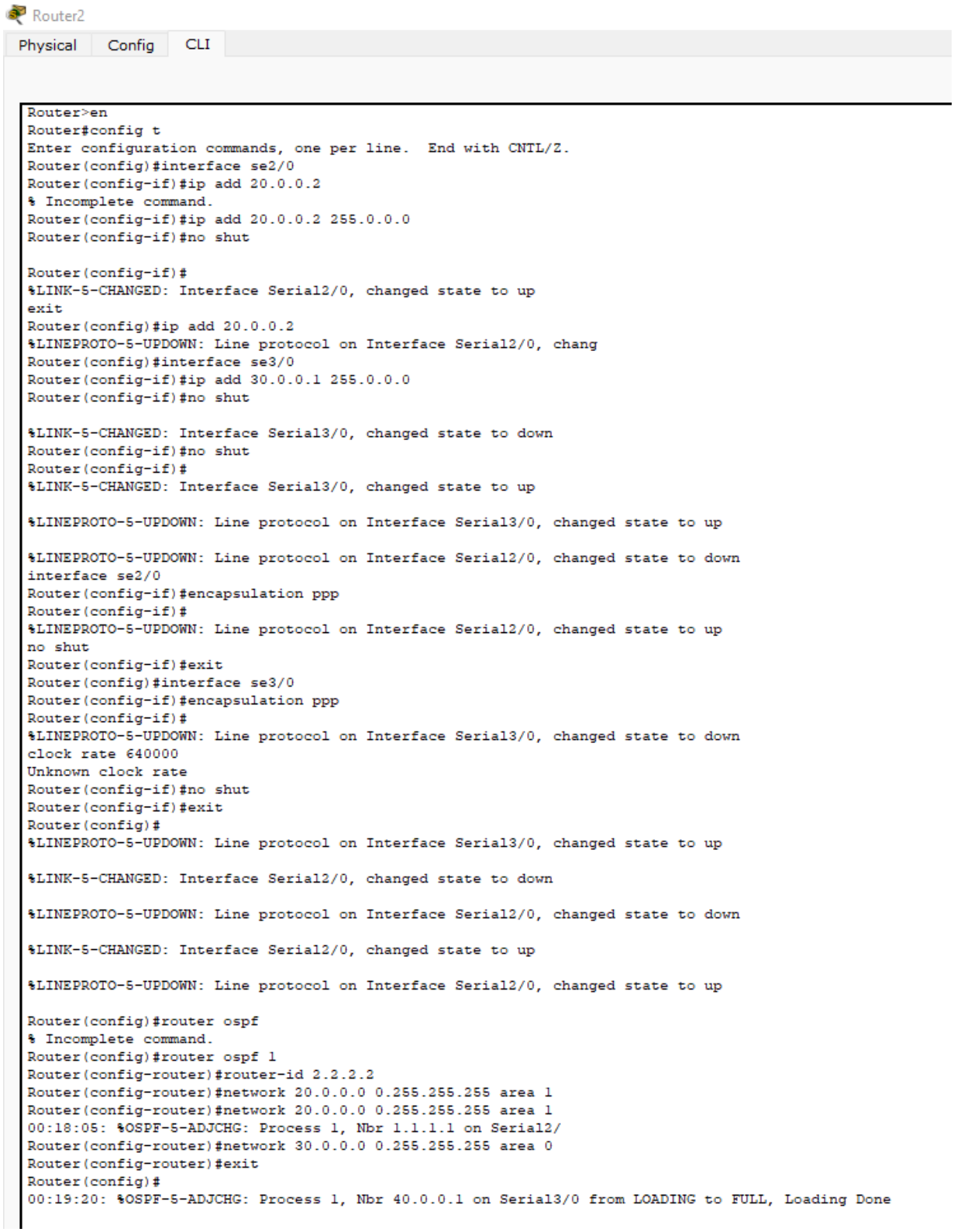
Configurations:

Router 1 :



```
Router(config)#interface se2/0
Router(config-if)#interface loopback 0

Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

Router(config-if)#ip address 172.16.1.252 255.255.0.0
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#area 1 virtual-link 2.2.2.2
Router(config-router)#exit
Router(config)#
00:24:20: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on OSPF_VL0 from LOADING to FULL, Loading Done
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    10.0.0.0/8 is directly connected, FastEthernet0/0
     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, Serial2/0
C       20.0.0.2/32 is directly connected, Serial2/0
O    30.0.0.0/8 [110/128] via 20.0.0.2, 00:00:54, Serial2/0
O IA 40.0.0.0/8 [110/129] via 20.0.0.2, 00:00:54, Serial2/0
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

Router 2 :


Router2
Physical
Config
CLI
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface se2/0
Router(config-if)#ip add 20.0.0.2
% Incomplete command.
Router(config-if)#ip add 20.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
Router(config)#ip add 20.0.0.2
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, chang
Router(config)#interface se3/0
Router(config-if)#ip add 30.0.0.1 255.0.0.0
Router(config-if)#no shut
%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#no shut
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
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
interface se2/0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
no shut
Router(config-if)#exit
Router(config)#interface se3/0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to down
clock rate 640000
Unknown clock rate
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
%LINK-5-CHANGED: Interface Serial2/0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router(config)#router ospf
% Incomplete command.
Router(config)#router ospf 1
Router(config-router)#router-id 2.2.2.2
Router(config-router)#network 20.0.0.0 0.255.255.255 area 1
Router(config-router)#network 20.0.0.0 0.255.255.255 area 1
00:18:05: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on Serial2/
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
Router(config)#
00:19:20: %OSPF-5-ADJCHG: Process 1, Nbr 40.0.0.1 on Serial3/0 from LOADING to FULL, Loading Done

Router2

Physical | Config | CLI

IOS Command Line Interface

```
Router(config)#interface se3/0
Router(config-if)#interface loopback 0

Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ip add 172.16.1.253 255.255.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to down

00:23:22: %OSPF-5-ADJCHG: Process 1, Nbr 40.0.0.1 on Serial3/0 from FULL to DOWN, Neighbor Down: Interface down or detached

%LINK-5-CHANGED: Interface Serial3/0, changed state to up

Router(config-if)#exi
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up
t
00:23:33: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0

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

00:23:41: %OSPF-5-ADJCHG: Process 1, Nbr 40.0.0.1 on Serial3/0 from LOADING to FULL, Loading Done

Router#
00:23:43: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0
config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#
00:23:53: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0

Router(config-router)#area 1 virtual-link
00:24:03: %OSPF-4-ERRRCV: Received invalid packet: mismatch area ID, from backbone area must be virtual-link but not found from 20.0.0.2, Serial2/0
1.1.1.1
Router(config-router)#
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#c
00:24:18: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on OSPF_VL0 from LOADING to FULL, Loading Done

% Ambiguous command: "c"
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

     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
```

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#area 1 virtual-link 1.1.1.1
Router(config-router)#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

O IA 10.0.0.0/8 [110/65] via 20.0.0.1, 00:00:34, 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
O IA 40.0.0.0/8 [110/65] via 30.0.0.2, 00:01:24, Serial3/0
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

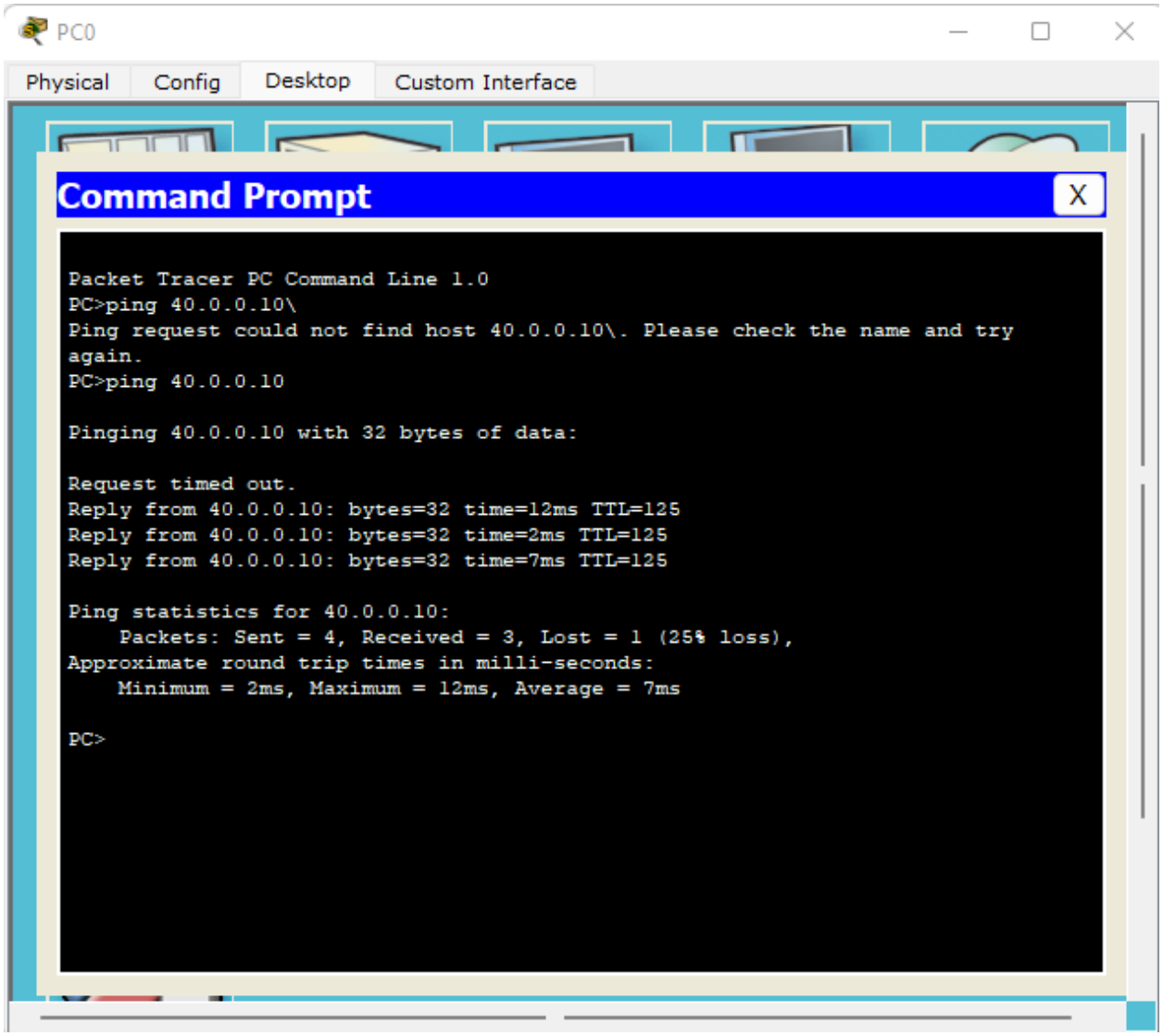
Router 3 :


Router3
Physical
Config
CLI
IOS Commar
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip add 40.0.0.1 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
exit
Router(config)#interface se2/0
Router(config-if)#ip add 30.0.0.2 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
exit
Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
Router(config)#interface se2/0
Router(config-if)#encapsulation ppp
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
no shut
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#
00:18:56: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from LOADING to FULL, Loading Done
Router(config-router)#network 40.0.0.0 0.255.255.255 area 2
Router(config-router)#exit
Router(config)#interface se2/0
Router(config-if)#interface loopback 0
Router(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
ip add 172.16.1.254 255.255.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
00:22:58: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from FULL to DOWN, Neighbor Down: Interface down or detached
%LINK-5-CHANGED: Interface Serial2/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
00:23:18: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on Serial2/0 from LOADING to FULL, Loading Done
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

```
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

O IA 10.0.0.0/8 [110/129] via 30.0.0.1, 00:05:53, Serial2/0
O IA 20.0.0.0/8 [110/128] via 30.0.0.1, 00:06:30, 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
C    172.16.0.0/16 is directly connected, Loopback0
Router#
```

Ping Output:

P0:



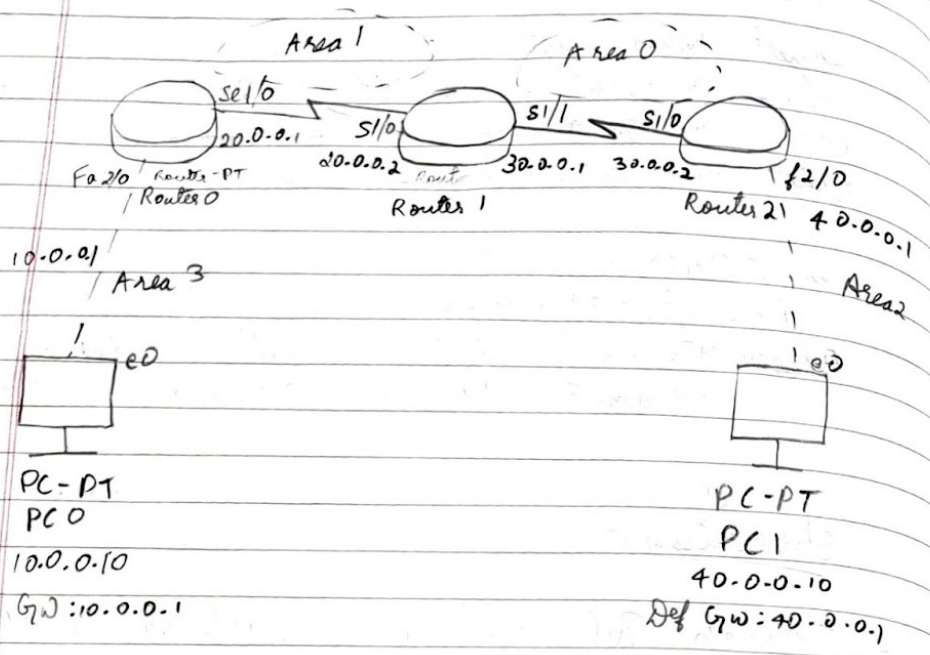
Observation :

27/7/23

Lab 7

Aim : Configure OSPF routing Protocol

Topology :-



Procedure.

1 → Configure the PCs with IP address and Gateway according to topology seen above.

2 → Configure each of the Routers according to the IP address in topology.

3 → Encapsulation ppp and clock ~~so~~ rate need to be set as done in RIP Protocol experiment

4→ Now, Enable ip routing by configuring ospf routing protocol in all routers

In Router R1,
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
ext

In Router R2,
R2(config)# router ospf 1
R2(config-router)# router-id 2.2.2.2
R2(config-router)# network 20.0.0.0 0.255.255.255 area 1
R2(config-router)# network 30.0.0.0 0.255.255.255 area 0
R2(config-router)# exit

In Router R3,
R3(config)# router ospf 1
R3(config-router)# router-id 3.3.3.3
R3(config-router)# network 30.0.0.0 0.255.255.255 area 0
R3(config-router)# network 40.0.0.0 0.255.255.255 area 2
R3(config-router)# exit

5→ Loopback

In R1, R1(config)# interface s1/0
R1(config-if)# interface loopback 0
R1(config-if)# ip address 172.16.1.252 255.255.0.0
R1(config-if)# no shut

In R2, R2(config)# interface s1/1
R2(config-if)# interface loopback 0
R2( )# ip address 172.16.1.253 255.255.0.0
R2( )# no shut

In R3,
R2(config)# interface s1/0 R2(config-if)# interface loopback 0
R2(config-if)# ip address 172.16.1.254 255.255.0.0
~~R2(config-if)#~~
R2(config-if)# no shut

6 → Virtual Links

In Router R1,
R1(config)# router ospf 1
R1(config-router)# area 1 virtual-link 2.2.2.2

In Router R2,
R2(config)# router ospf 1
R2(config-router)# area 1 virtual-link 1.1.1.1
R2(config-router)# exit

Ping Output

Ping > 40.0.0.10
Pinging 40.0.0.10 with 32 bytes of data:
Request timed out
Reply from 40.0.0.10: bytes=32 time=13ms TTL=125
Reply from 40.0.0.10: bytes=32 time=2ms TTL=125

Reply from 40.0.0.10 : bytes=32 time=10ms TTL=125

Ping statistics for 40.0.0.10
Packets: sent = 4, Received = 3, lost = 1 (25% loss)
Approximate round trip in milli-seconds:
Minimum = 2ms, Maximum = 13ms, Average = 12.5ms

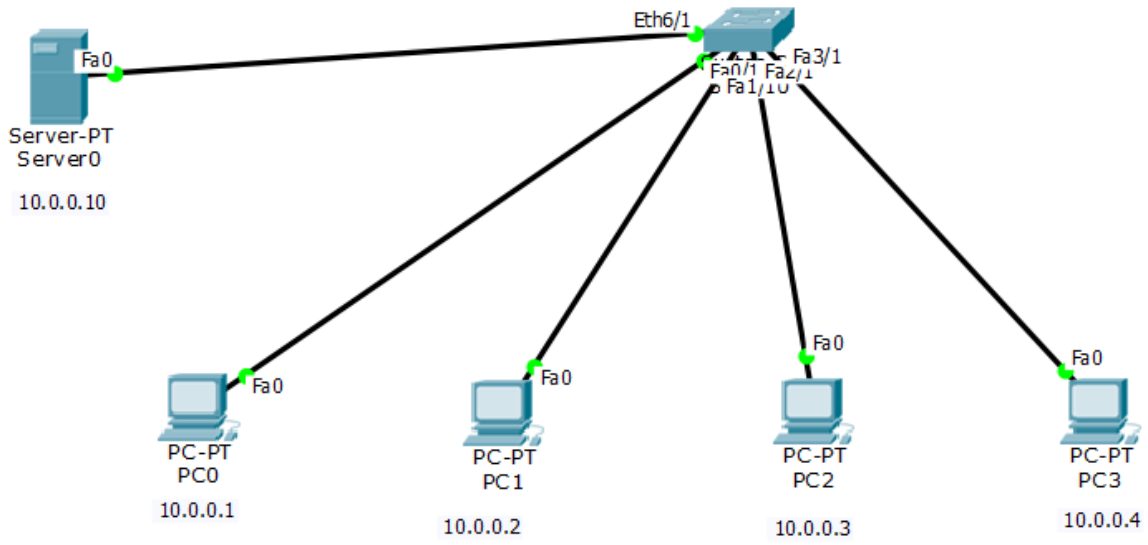
Observation
OSPF protocol is used to find the best path for packets as they pass through a set of connected networks.

8/12/2023

Experiment 8

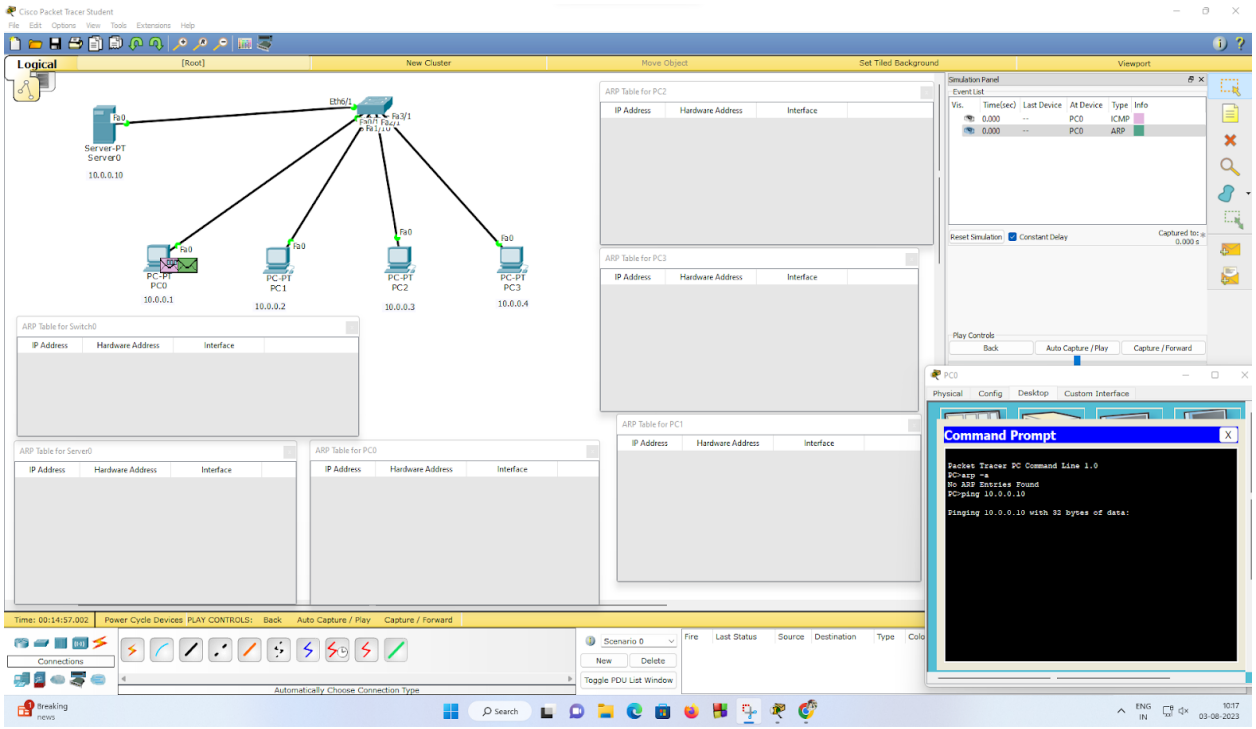
Aim : To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)

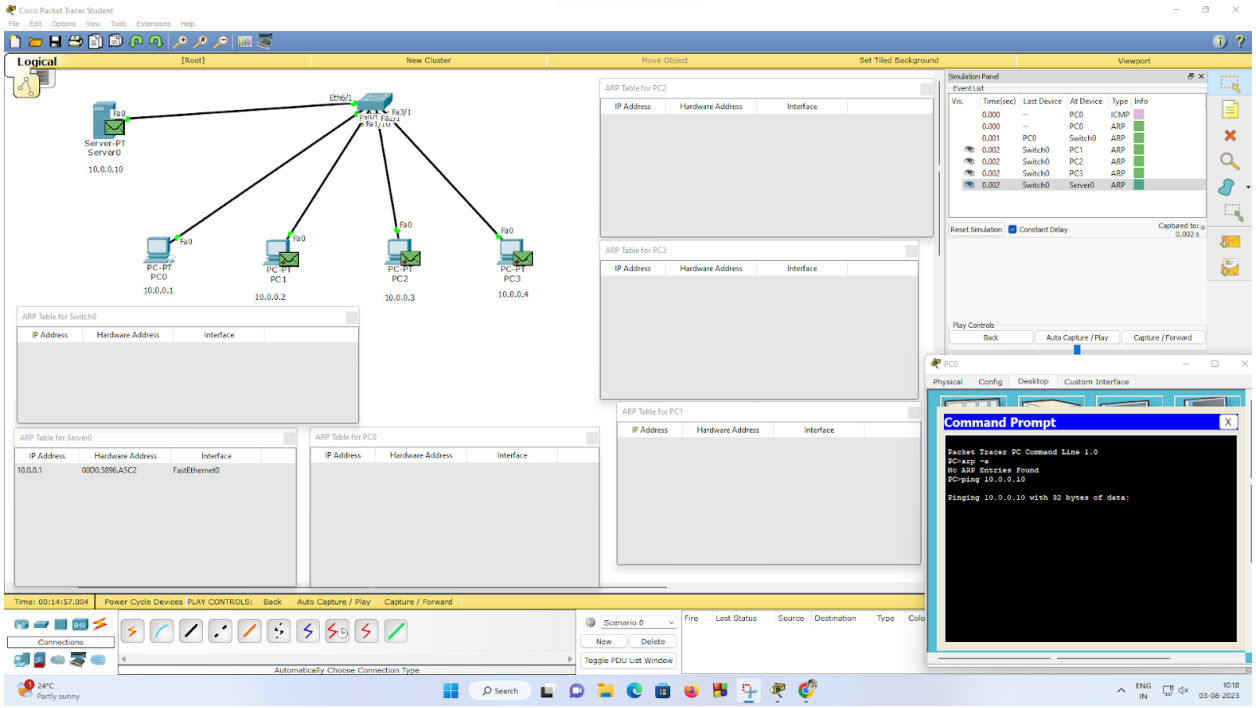
Topology:

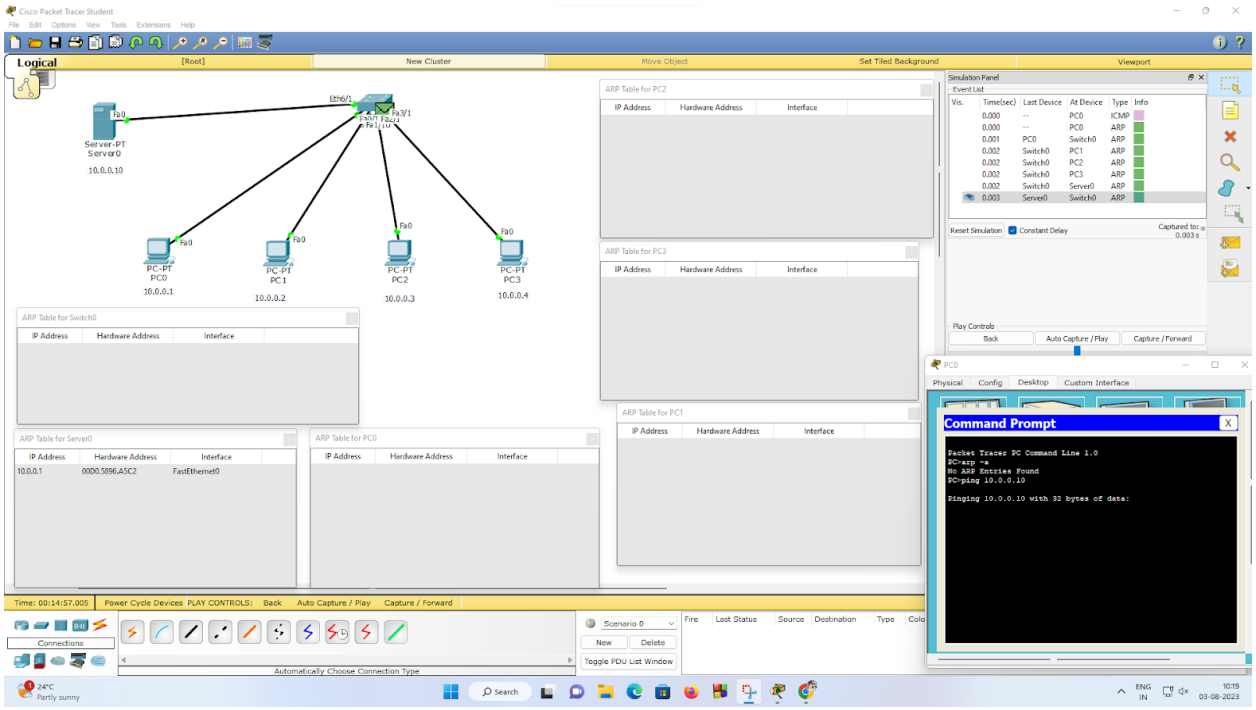


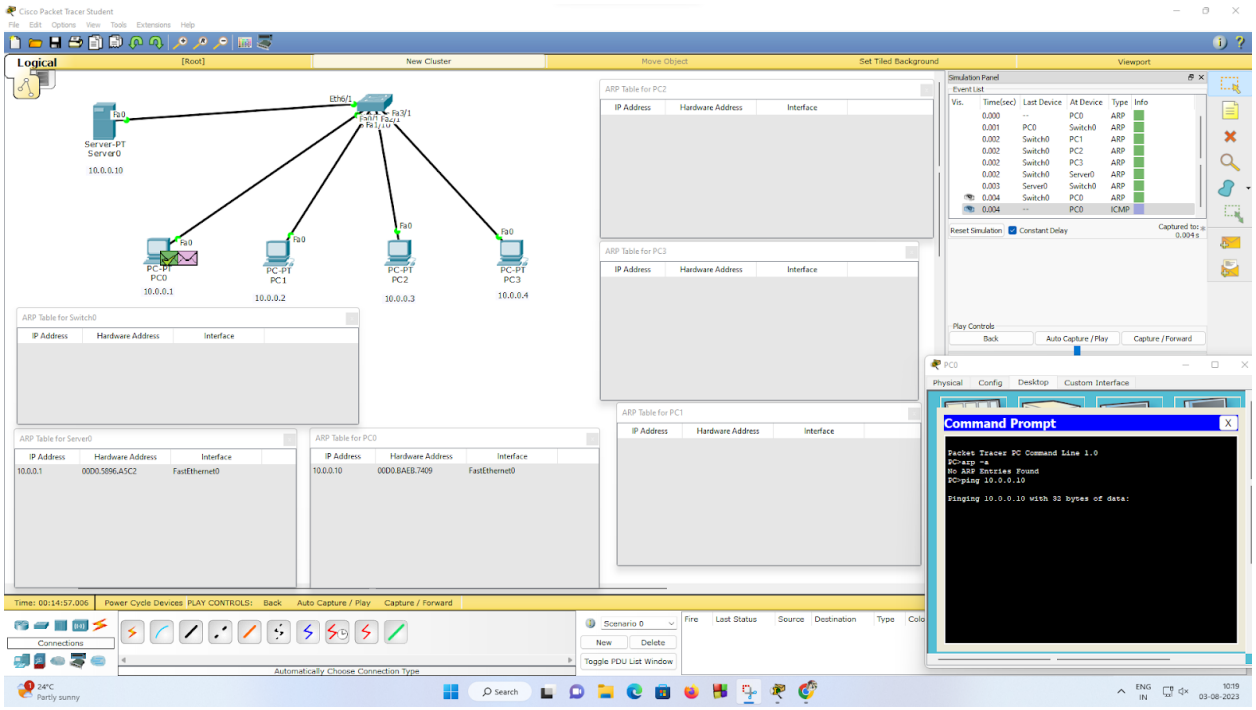
ARP Tables while pinging :

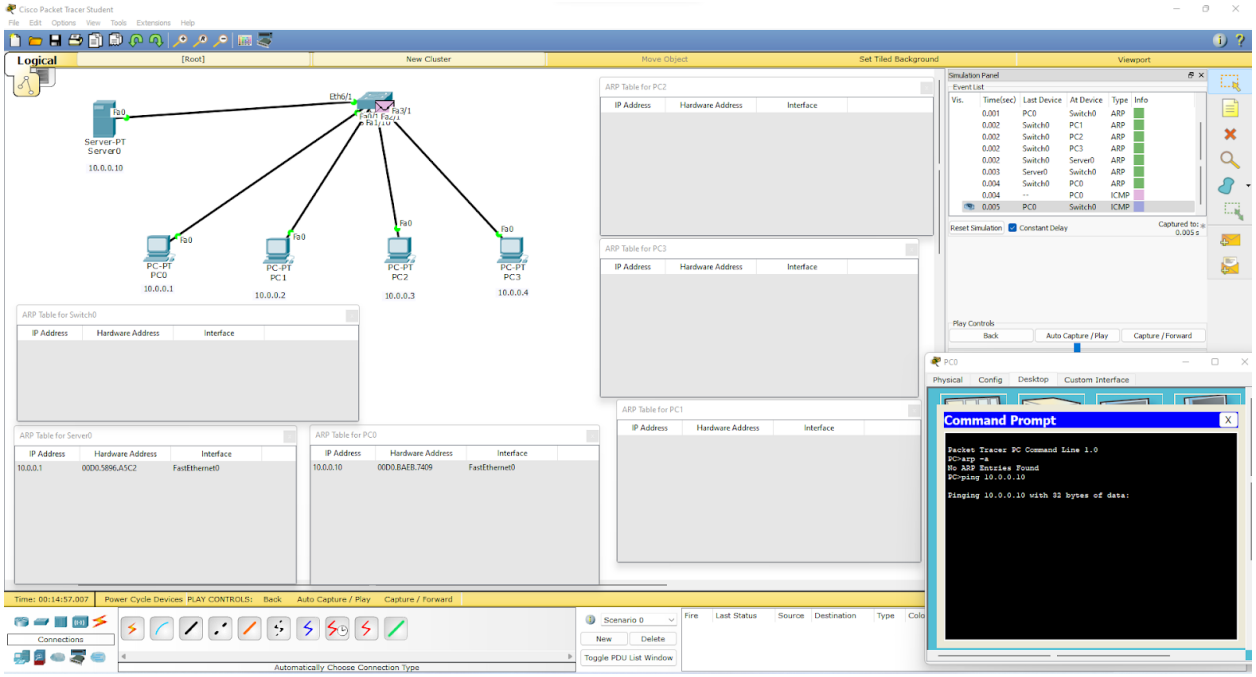
Ping from PC0 to Server0:

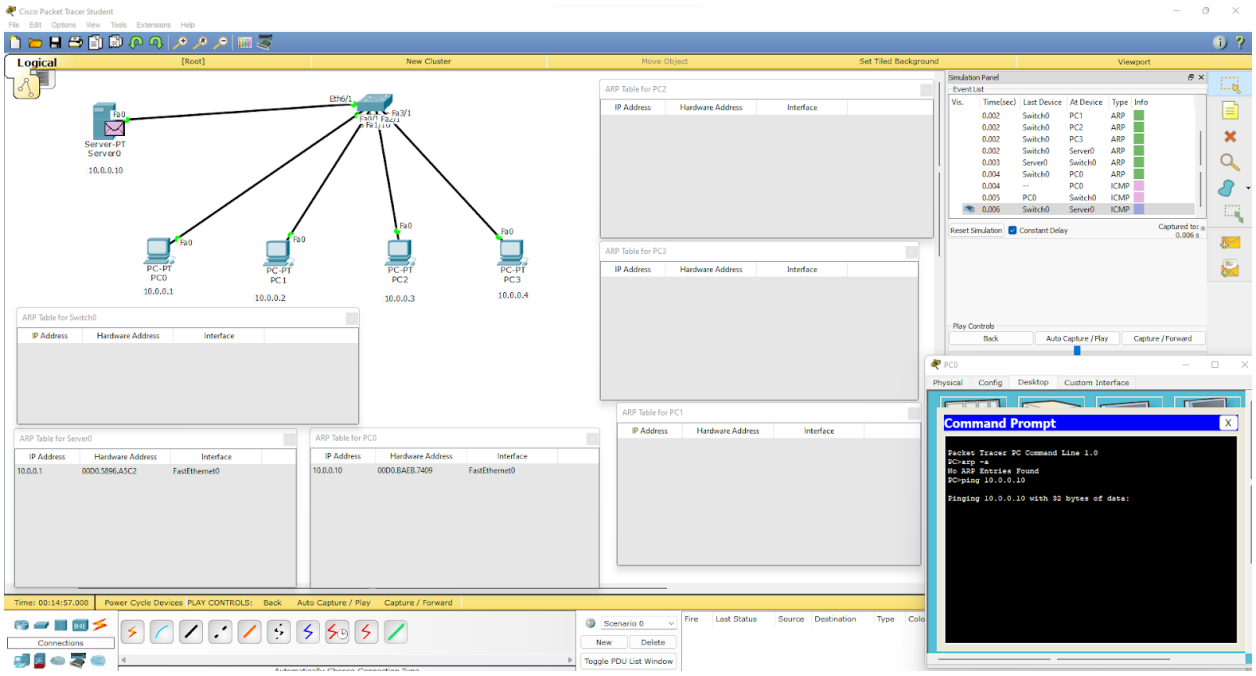


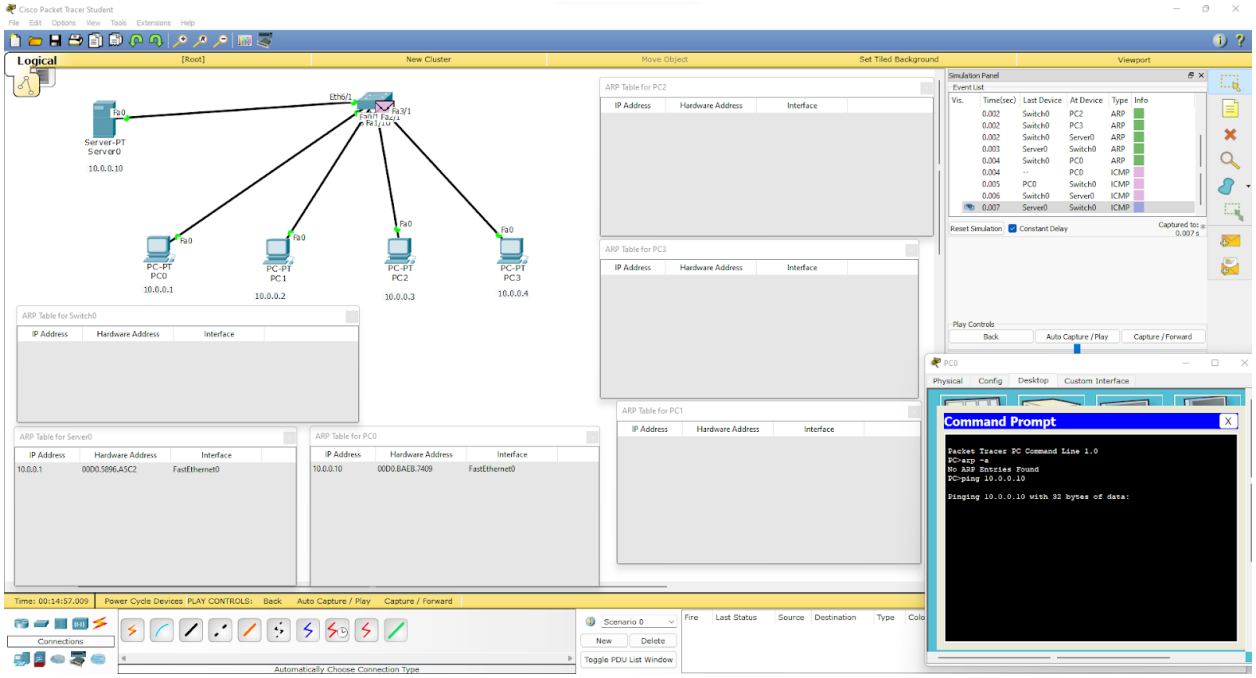


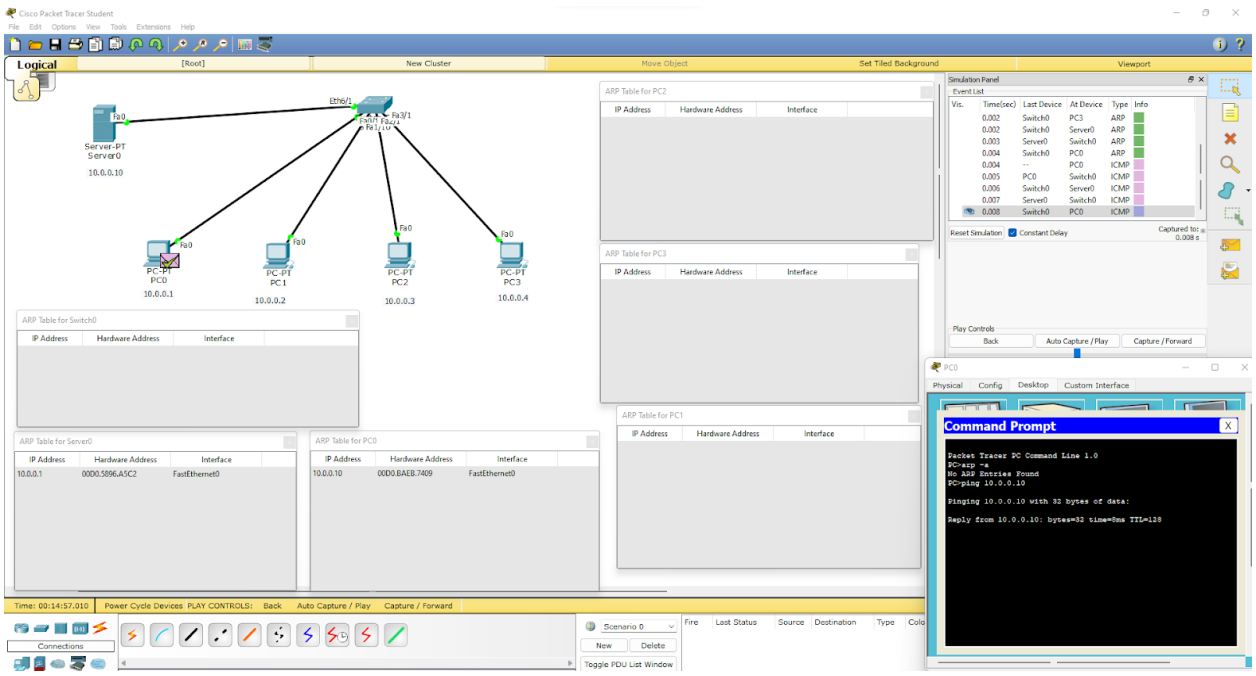

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:

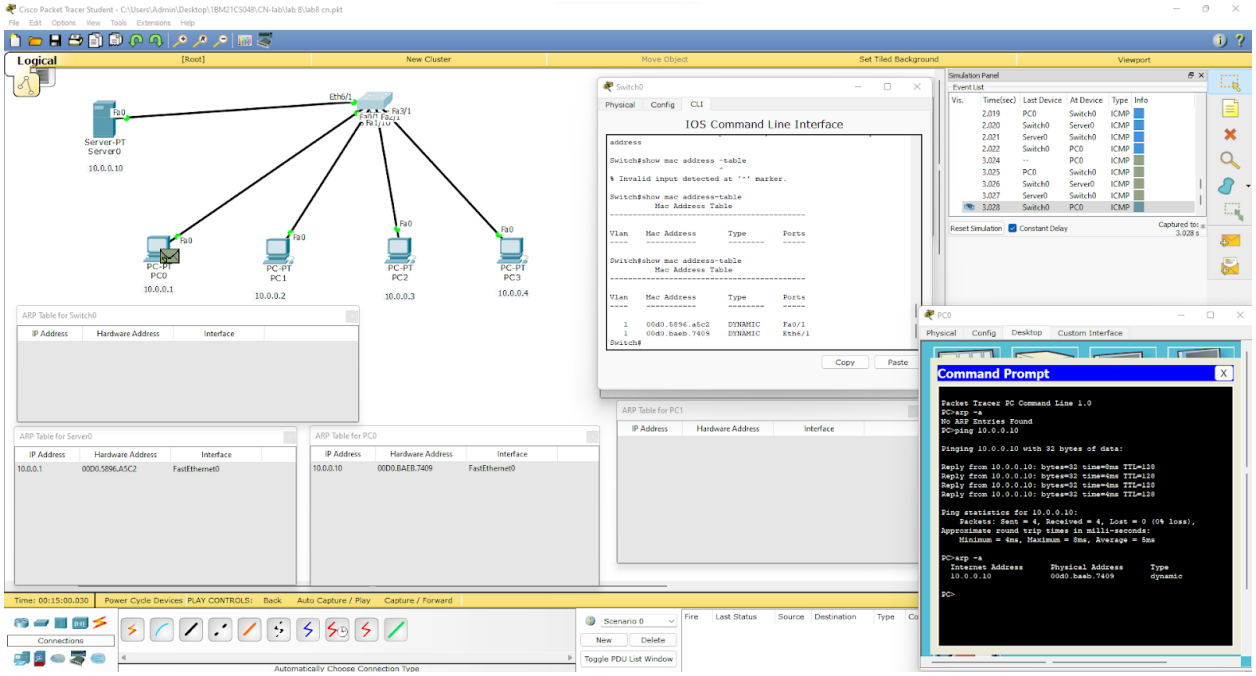

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:


Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:

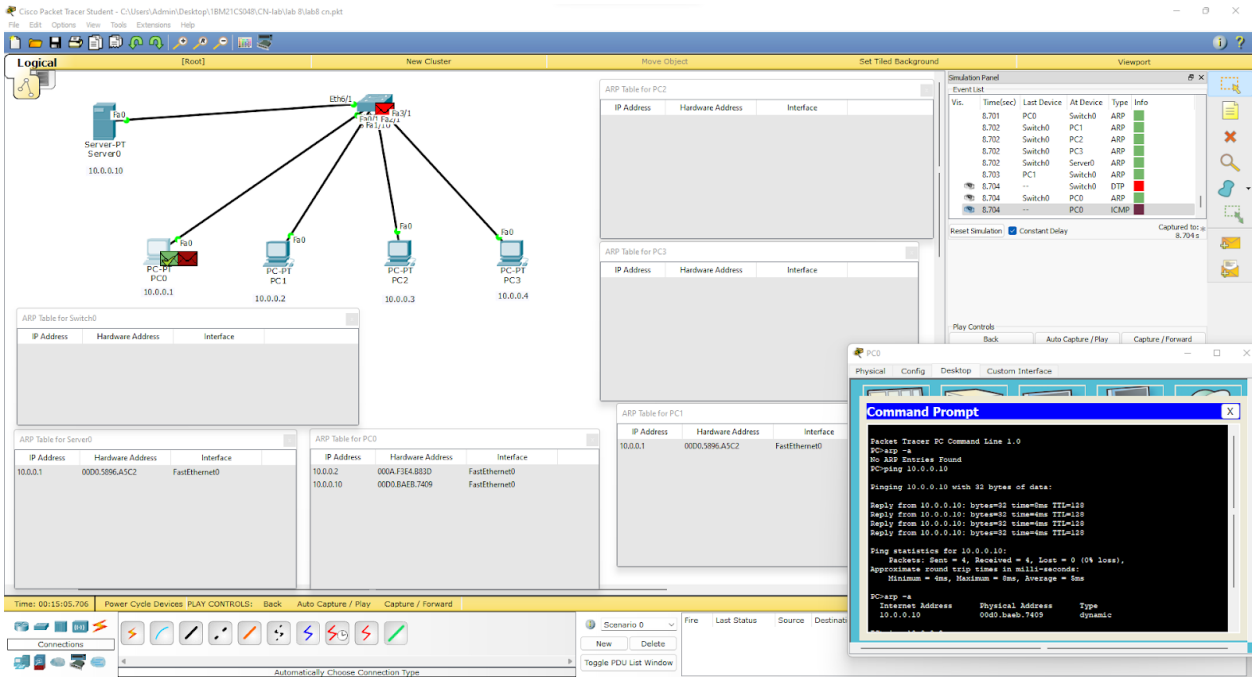

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:

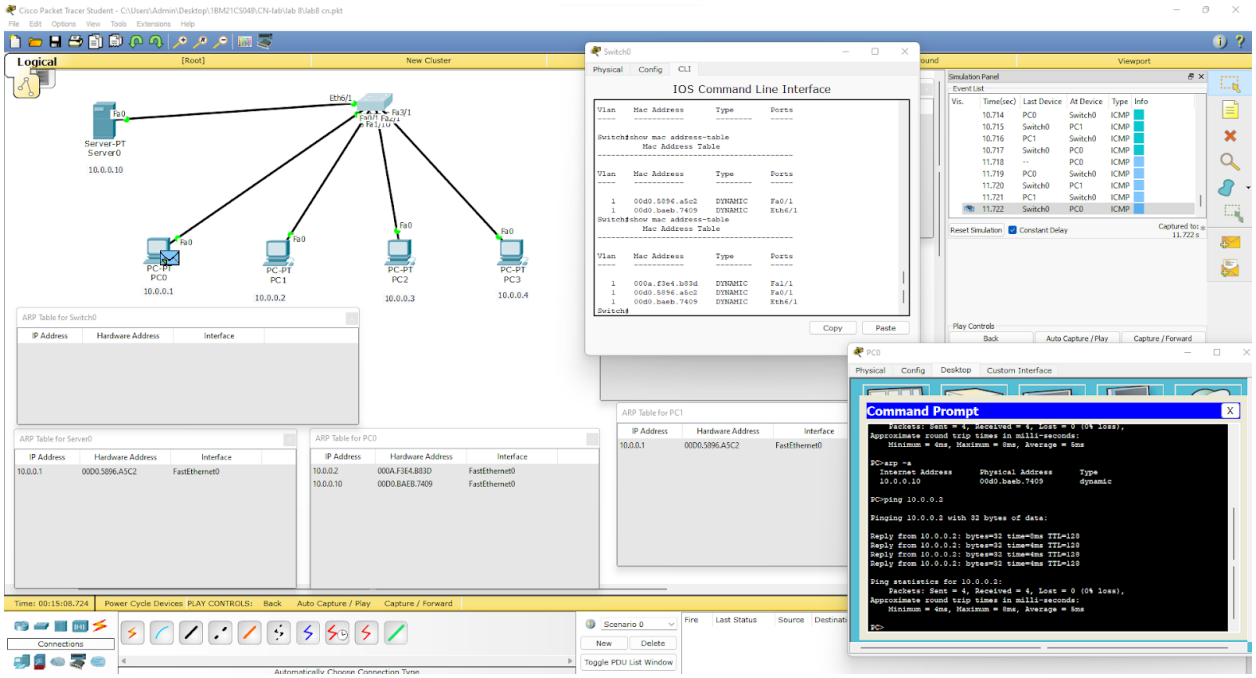

Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:


Logical
Server-PT Server0
10.0.0.10
PC-PT PC0
10.0.0.1
PC-PT PC1
10.0.0.2
PC-PT PC2
10.0.0.3
PC-PT PC3
10.0.0.4
Command Prompt
Packet Tracer PC Command Line 1.0
PC>arp -a
No ARP Entries Found
PC>ping 10.0.0.10
Pinging 10.0.0.10 with 32 bytes of data:
Reply from 10.0.0.10: bytes=32 time=0ms TTL=128

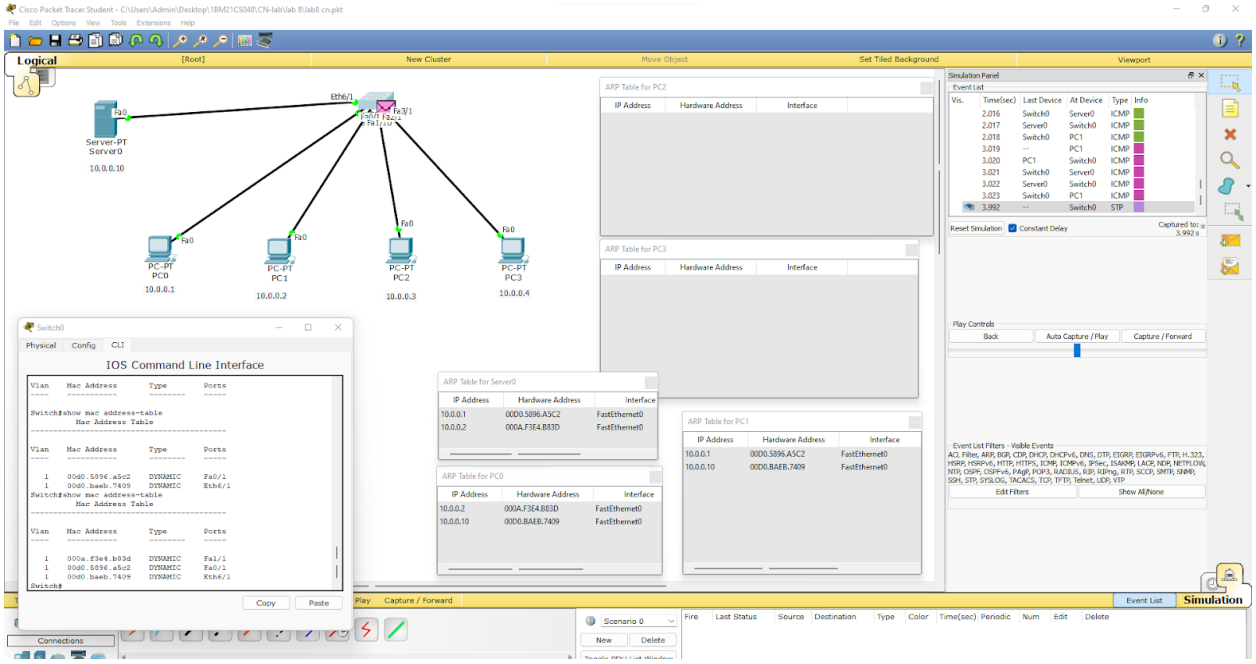


Ping from PC0 to PC1:





Final ARP Tables after pinging:



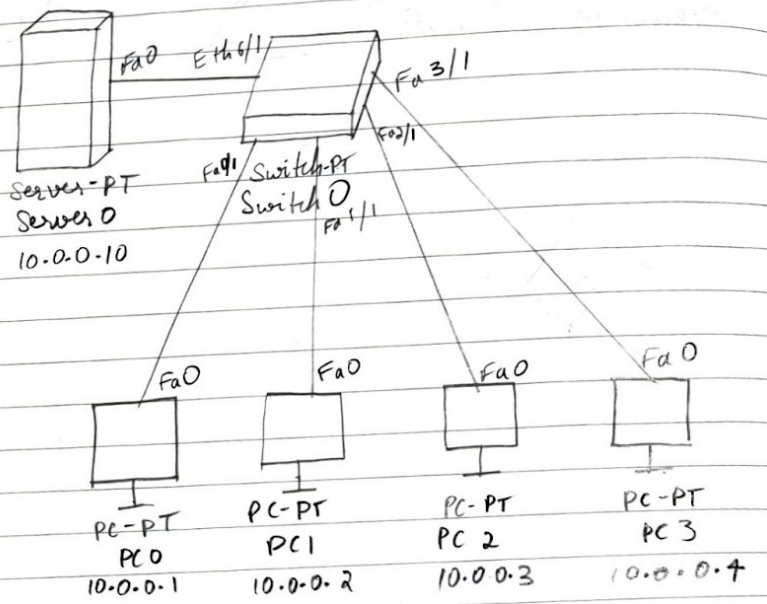
Observation:

3/8/'23

Lab8:

Aim: To construct simple LAN and understand the concept and operation of address resolution protocol (ARP)

Topology:



Procedure

1) Create a topology of 4 PCs and a server and a switch as shown in the diagram.

2) Set IP address of all devices as shown in the topology.

3) Use inspect tool to click on devices to open ARP tables of each device.

4) In command prompt of PC0, write arp -a to see arp table.
   o/p: No ARP Entries Found as no pinging done yet.

5) In CLI of switch, enter
Switch > enable
~~# show MAC~~
show mac address-table
Now, mac address table is empty.

6) Go to Simulation mode.
ping the server from PC0
Click on capture button in Simulation panel to go step by step. ~~change~~

28/12/22

∈ Ping Output :-

```
PC > arp -a
NO ARP Entries found
PC> ping 10.0.0.10

Pinging 10.0.0.10 with 32 bytes of data:

Reply from 10.0.0.10 : bytes=32 time=8ms TTL=128
Reply from 10.0.0.10 : bytes=32 time=4ms TTL=128
Reply from 10.0.0.10 : bytes=32 time=4ms TTL=128
Reply from 10.0.0.10 : bytes=32 time=4ms TTL=128

Ping statistics for 10.0.0.10:
    Packets : Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 8ms, Average = 5ms

PC>arp -a
  Internet Address      Physical Address      Type
  10.0.0.10             0060.baab.7409        dynamic
```

Observation:

→ We can see mac address-table of switch in CLI

→ We can observe that, when we ping and click on capture on simulation panel, ARP request is broadcasted and ARP reply is unicasted.

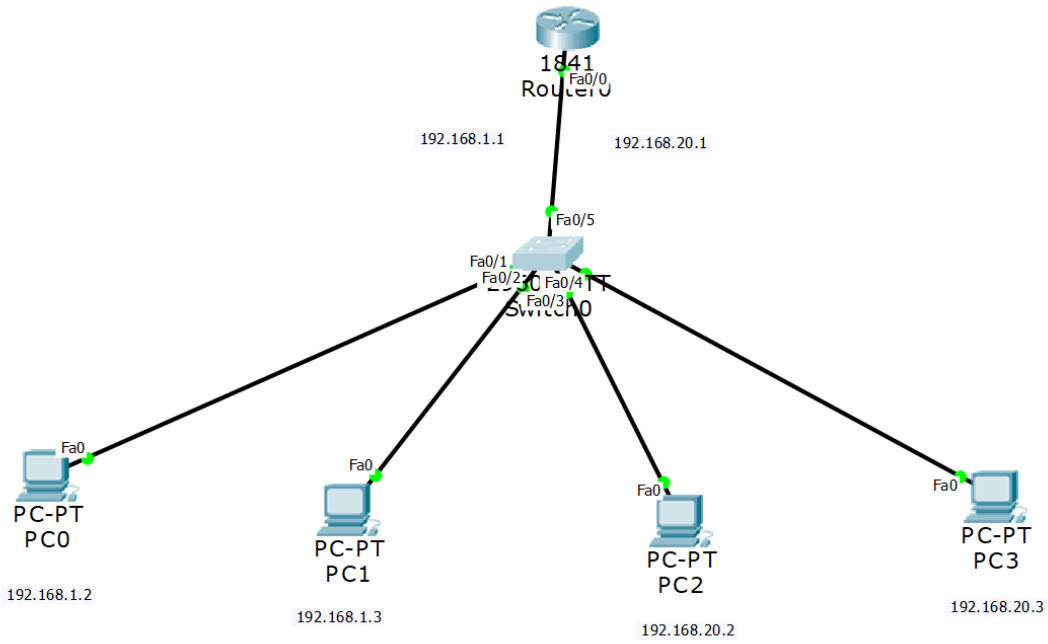
→ Then ICMP packets are sent.

→ As and when capture is clicked, ARP table is updated accordingly for PC and server.

Experiment 9

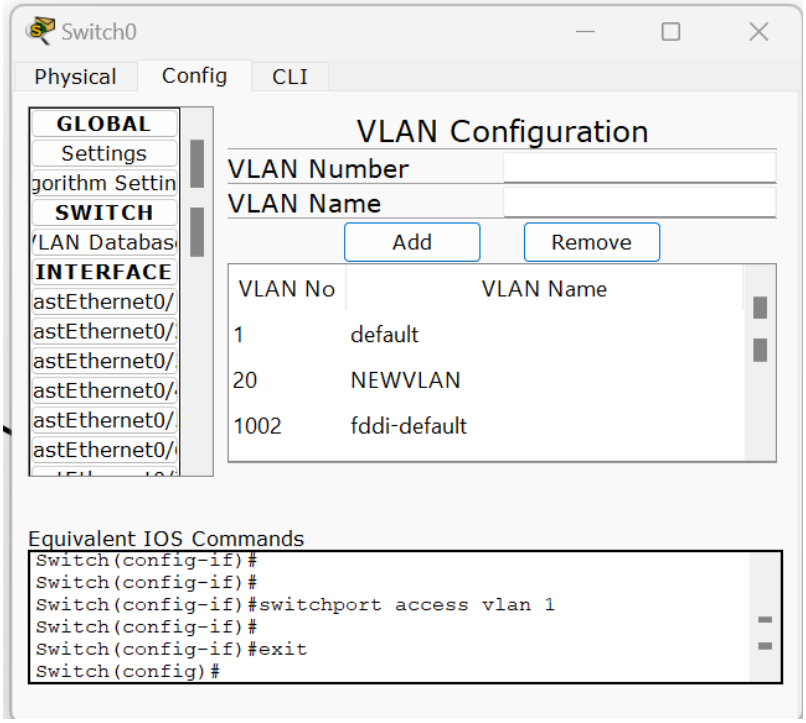
Aim : To construct a VLAN and make the PC's communicate among a VLAN

Topology:

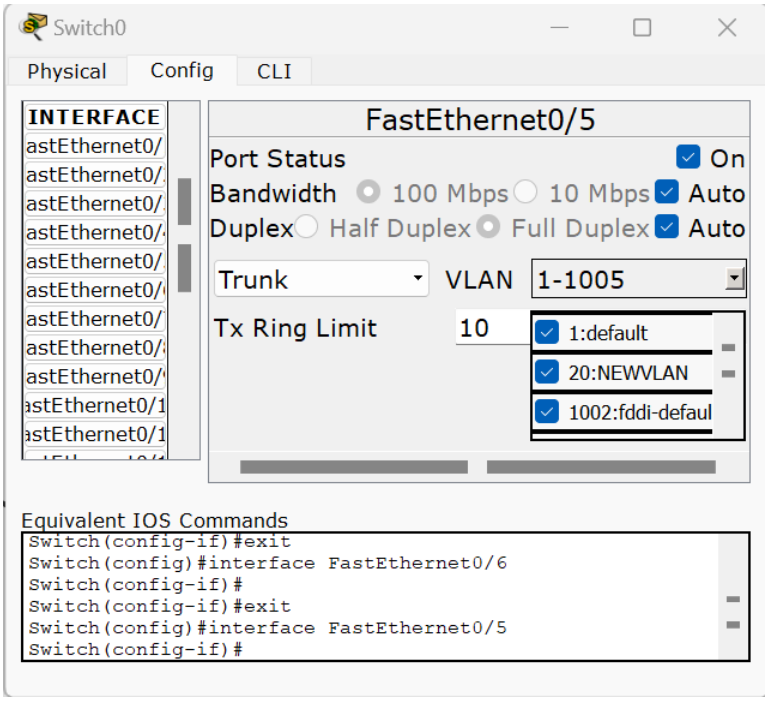


Configurations:

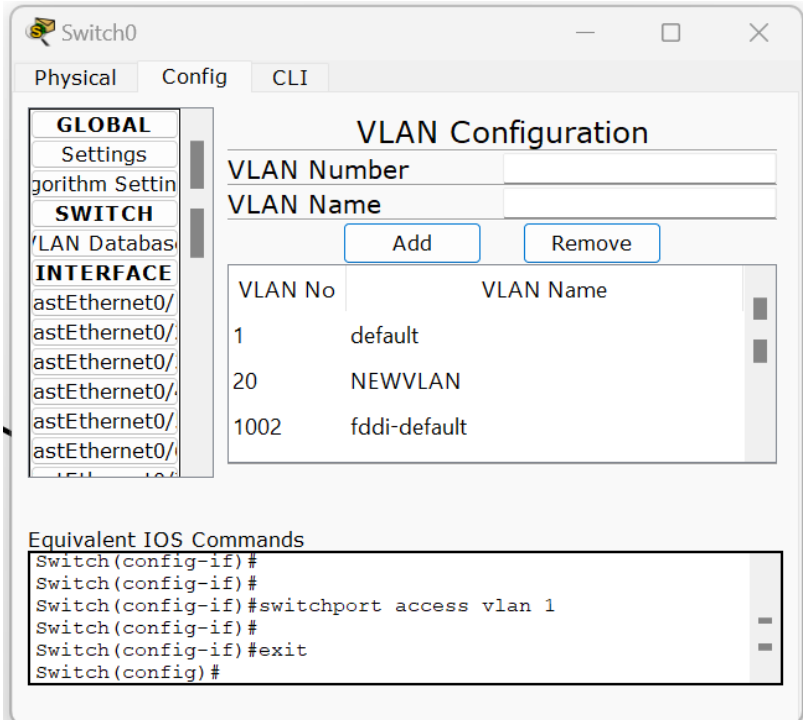
Switch VLAN Database:



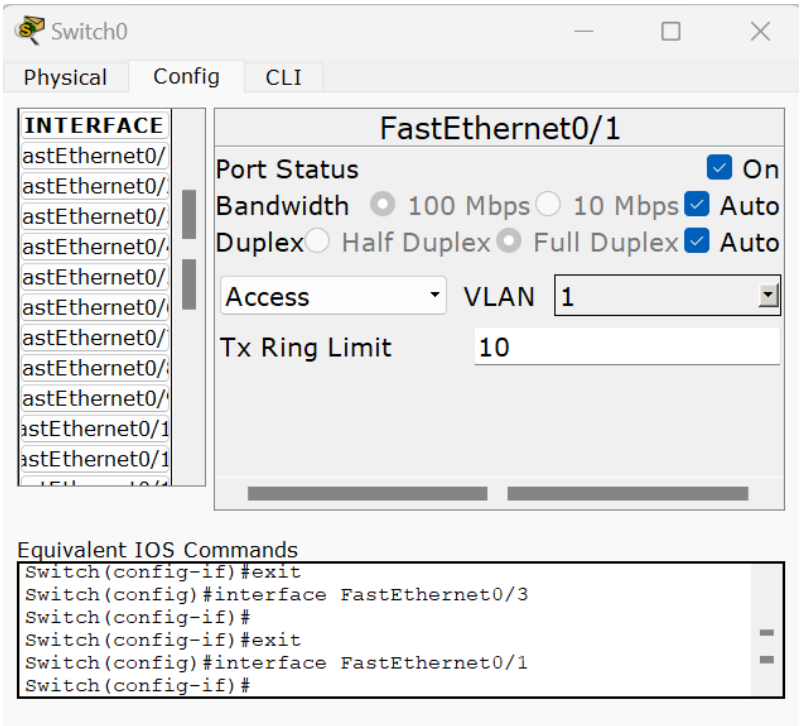
Switch FastEthernet0/5



Switch FastEthernet0/3 and FastEthernet0/4

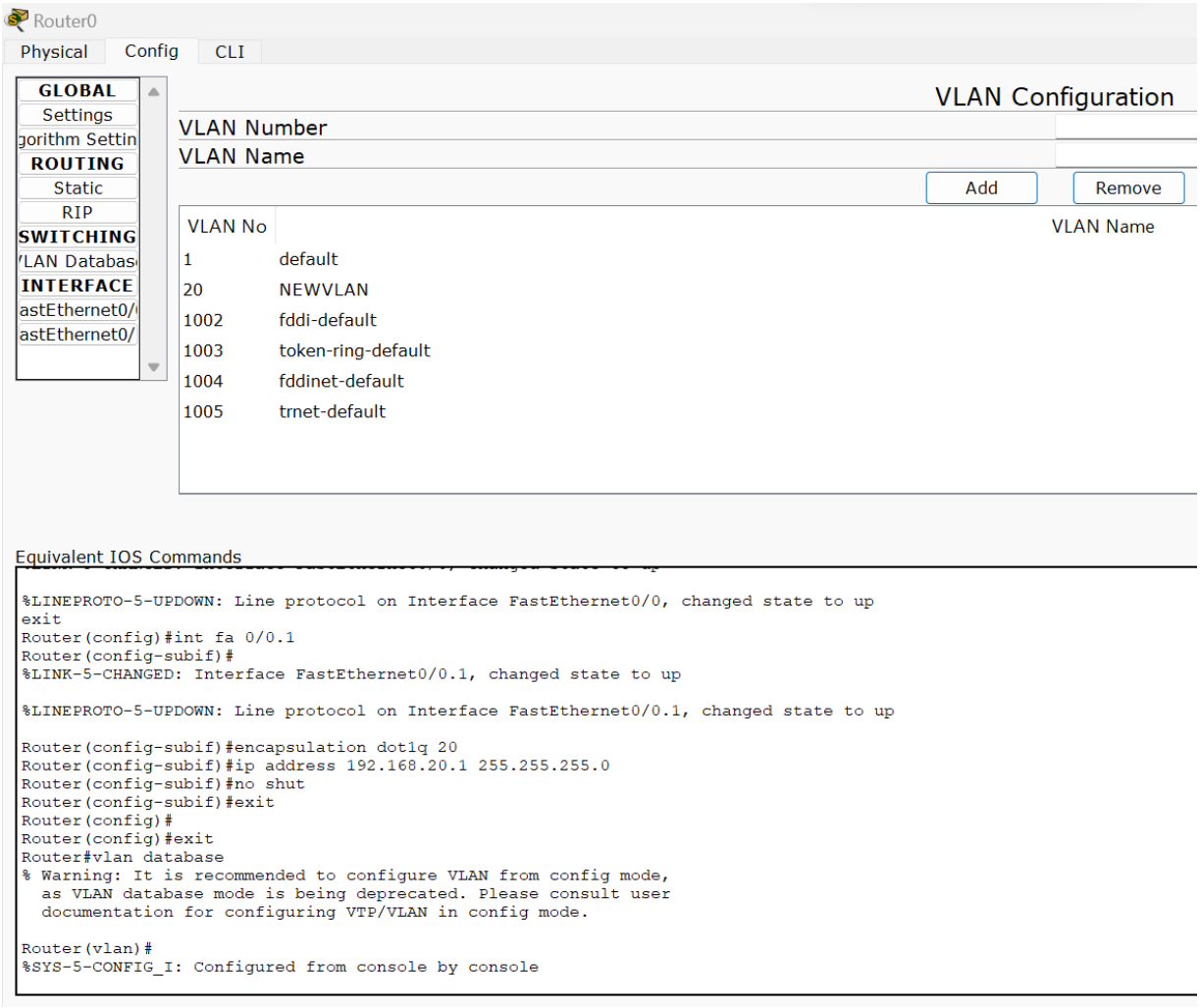


Switch FastEthernet0/1 and FastEthernet0/2



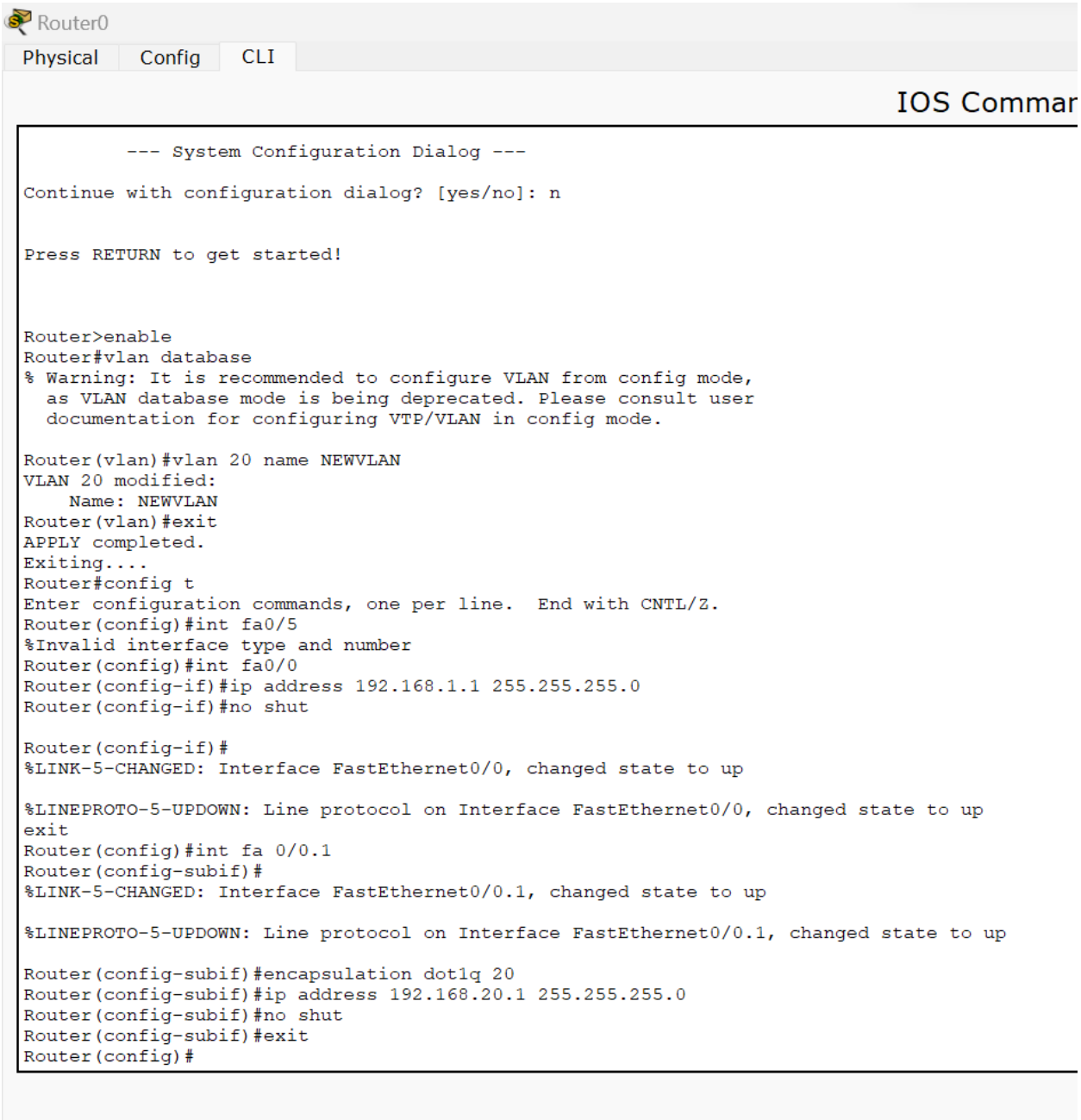
Router 0 :

VLAN DataBase:



Router 0 :

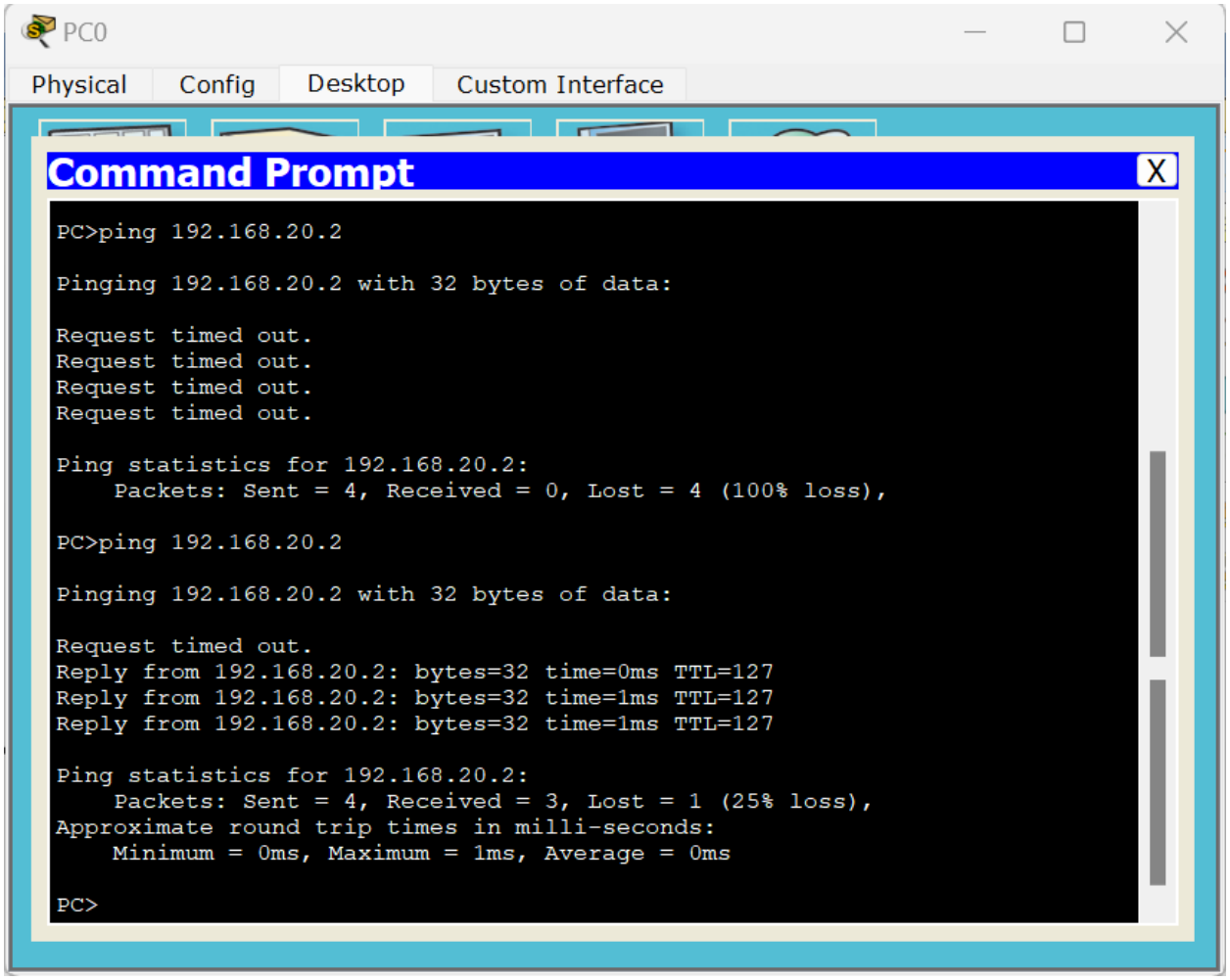
CLI:



Ping Outputs:

P0:

Before and after VLAN configuration was successful.

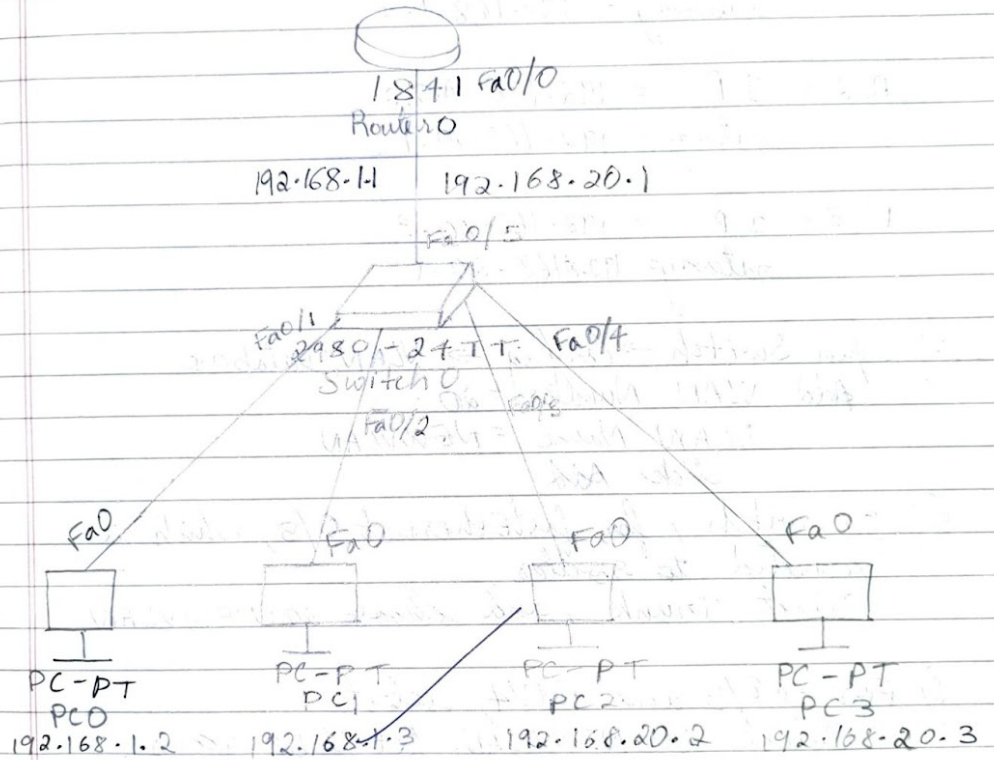


Observation:

Lab 9

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

Topology:



Procedure:

1) Create the topology as shown using 1841 Router and 2980 Switch. Connect 4 PCs to them as shown using copper straight through cable.

2) We use class C addressing here.

3) Set IP addresses and Gateways for PCs.

PC 0 : IP = 192.168.1.2
Gateway = 192.168.1.1

PC 1 : IP = 192.168.1.3
Gateway = 192.168.1.1

PC 2 : IP = 192.168.20.2
Gateway = 192.168.20.1

PC 3 : IP = 192.168.20.3
Gateway = 192.168.20.1

4) Open Switch → Config → VLAN Database
Add VLAN Number = 20
VLAN Name = NEWVLAN
Click Add

5) In Switch, for fastEthernet 0/5, which is connected to router,
Select Trunk, and choose 20:NEWVLAN

6) For Fa0/3 and Fa0/4, select
Select 20:NEWVLAN. Let it be Access.
For Fa0/1 and Fa0/2, leave it as it is.

7) Open Router → Config → VLAN Database
Add VLAN Number : 20
VLAN Name : NEWVLAN

8) In Router0 CLI :

Router>en
Router#(vlan)#exit
Router#config t

Router(config)#int fa0/0
Router(config)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shut
Router(config-if)#exit

Router(config)#int fa 0/0.1
Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip address 192.168.20.1 255.255.255.0
Router(config-subif)#no shut
Router(config-subif)#exit
Router(config)#exit

AL
10/8/2023

Ping Output :-

PC > ping 192.168.20.2

Pinging 192.168.20.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.20.2: bytes=32 time=0ms TTL=127
Reply from 192.168.20.2: bytes=32 time=1ms TTL=127
Reply from 192.168.20.2: bytes=32 time=1ms TTL=127

Ping statistics for 192.168.20.2:
    Packets: Sent=4, Received=3, Lost=1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum=0ms, Maximum=1ms, Average=0ms.

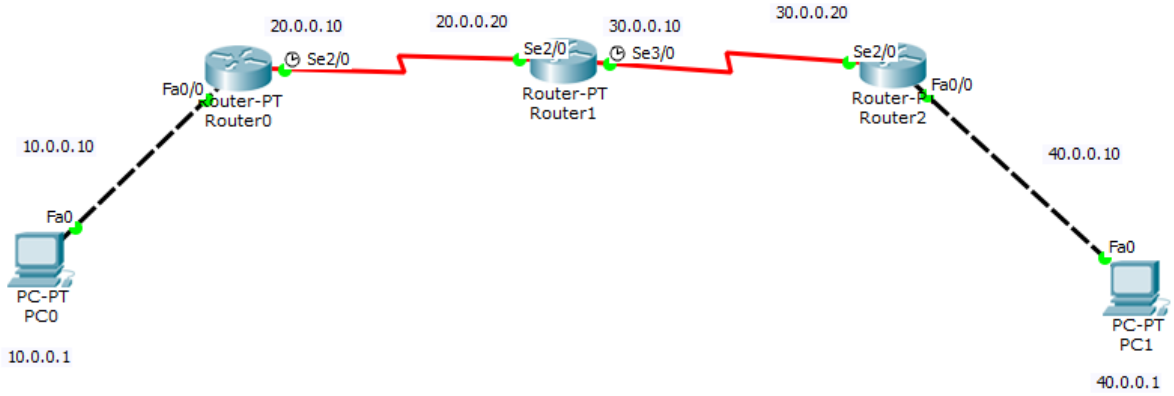
Observation :-

- We can observe that ~~before~~ after VLAN is configured we can succesfully ping PC2 (192.168.20.2) from PC0 (192.168.1.2).
- PC2 and PC3 are grouped together and communication among them is done via VLAN.
- 192.168.20.1 is a subgo interface of the Router.

Experiment 10

Aim : Demonstrate the TTL/ Life of a Packet

Topology :

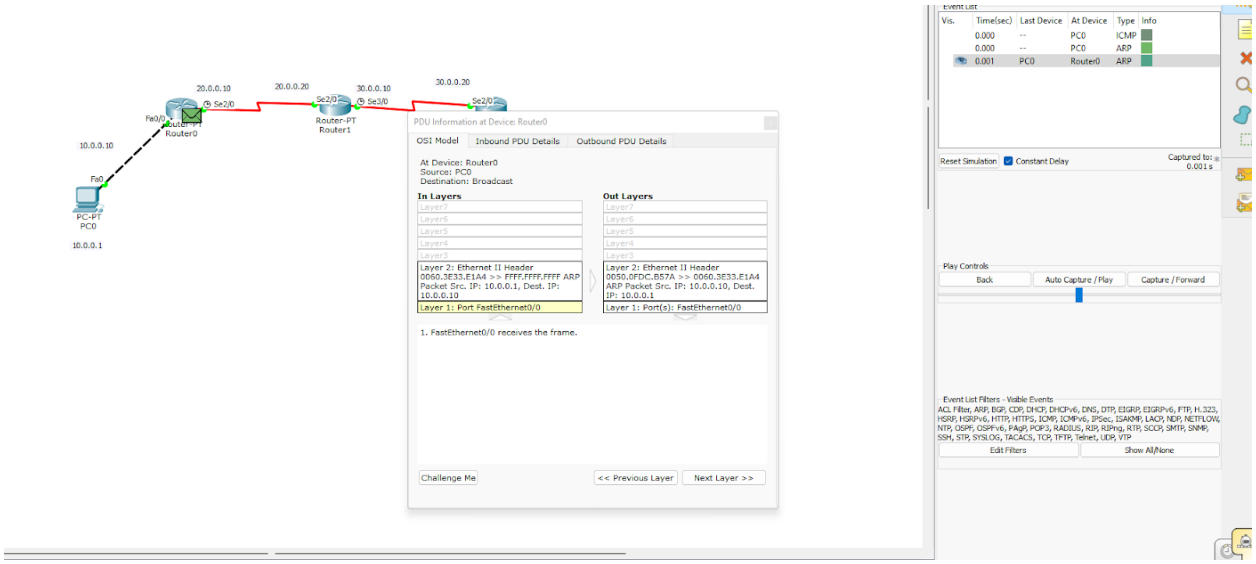


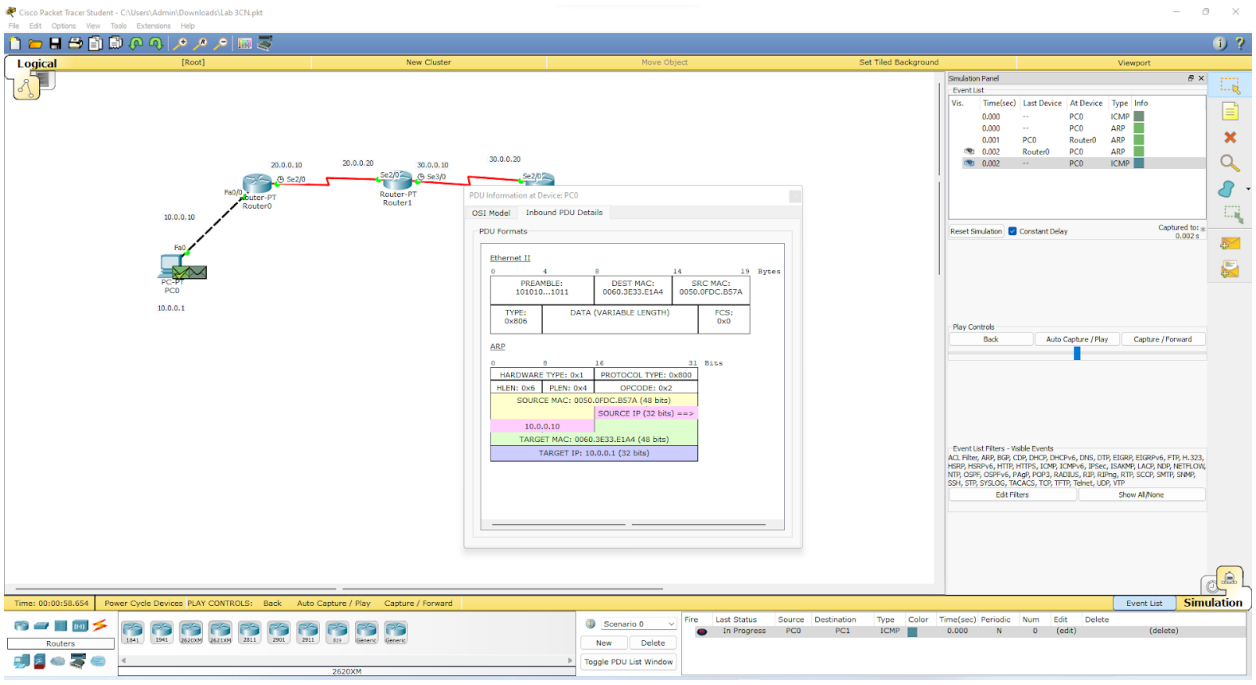
Configurations :

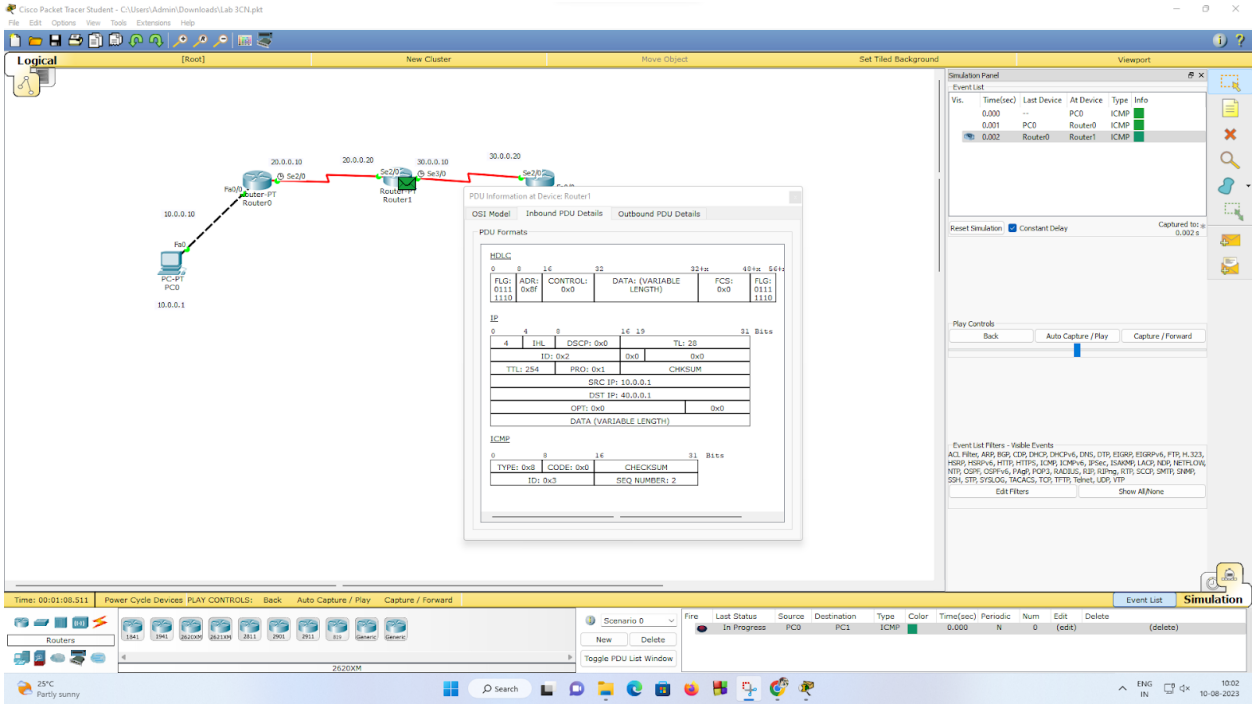
Configure the devices as per static / default / dynamic routing.
Above is done using static routing.

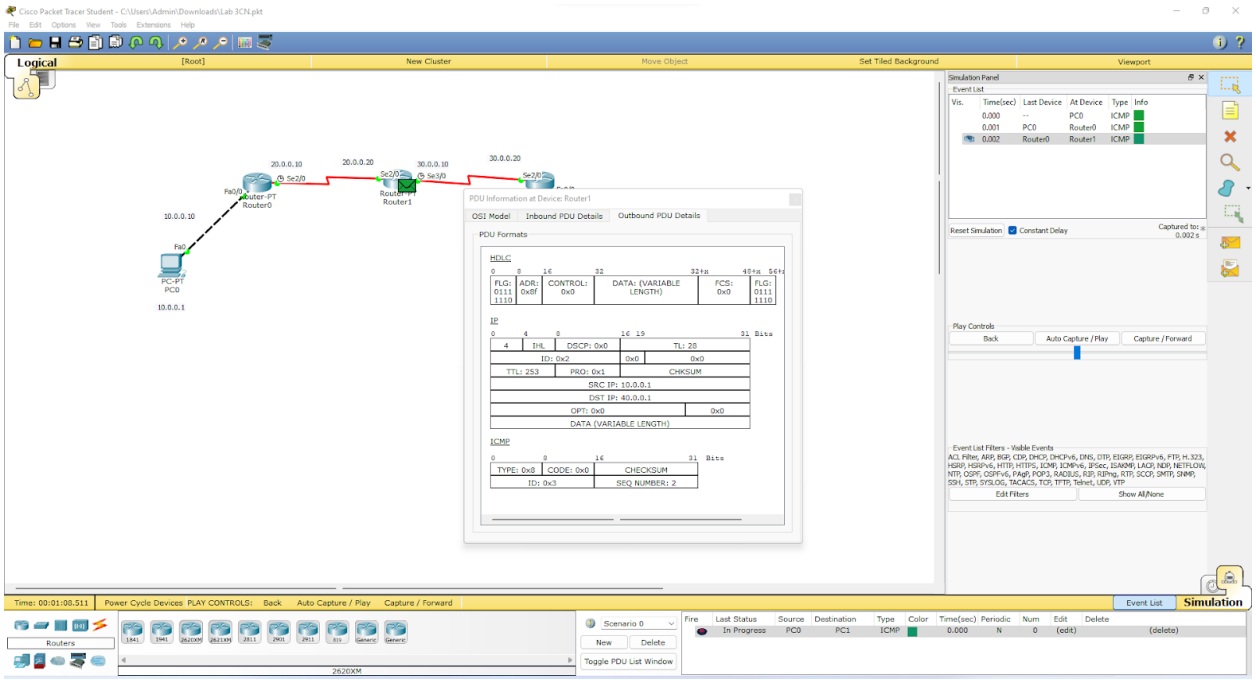
Output and PDU Details:

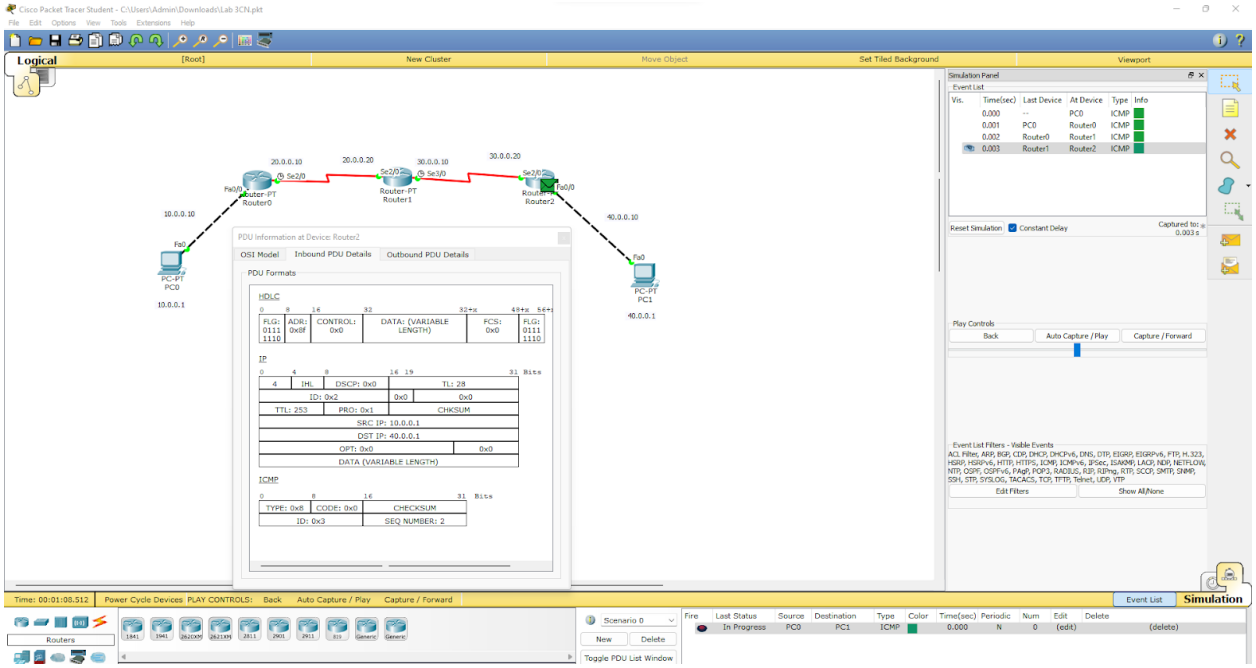
Simple PDU sent from PC0 to PC1 in simulation mode.

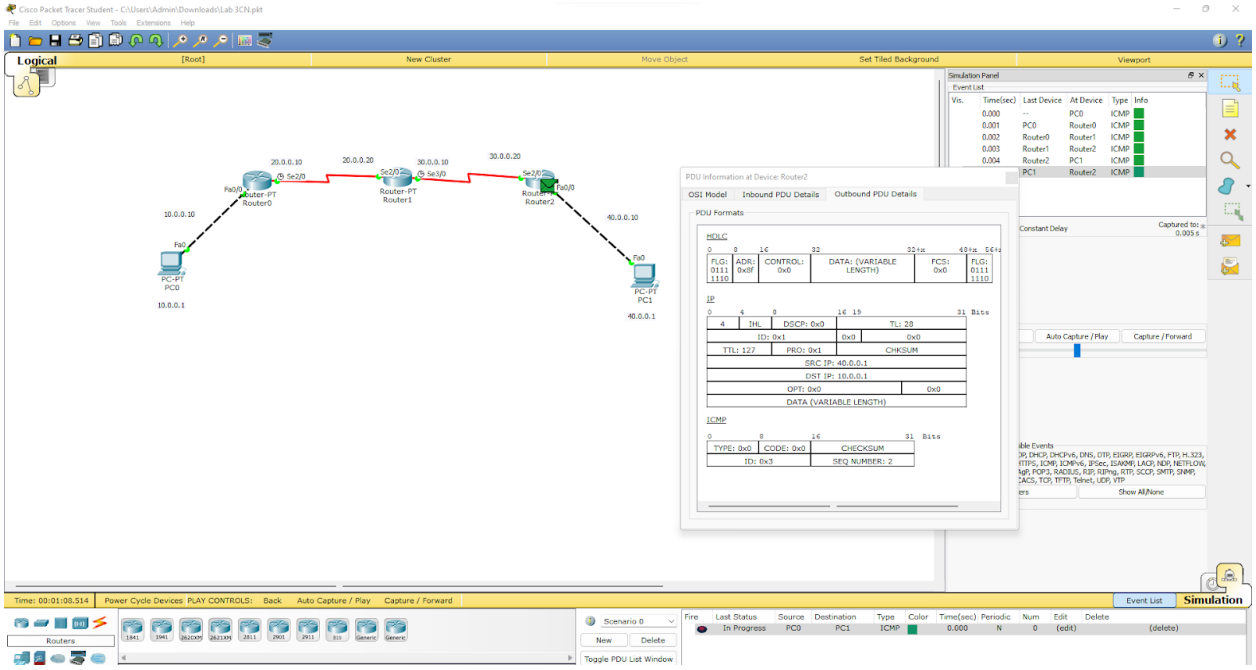


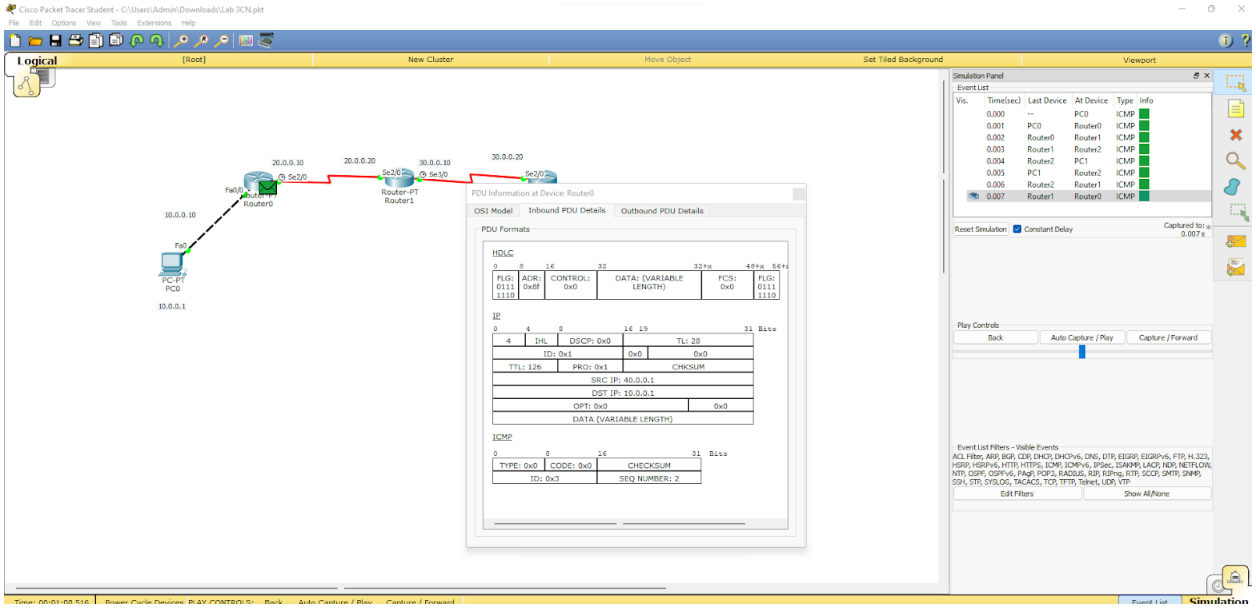


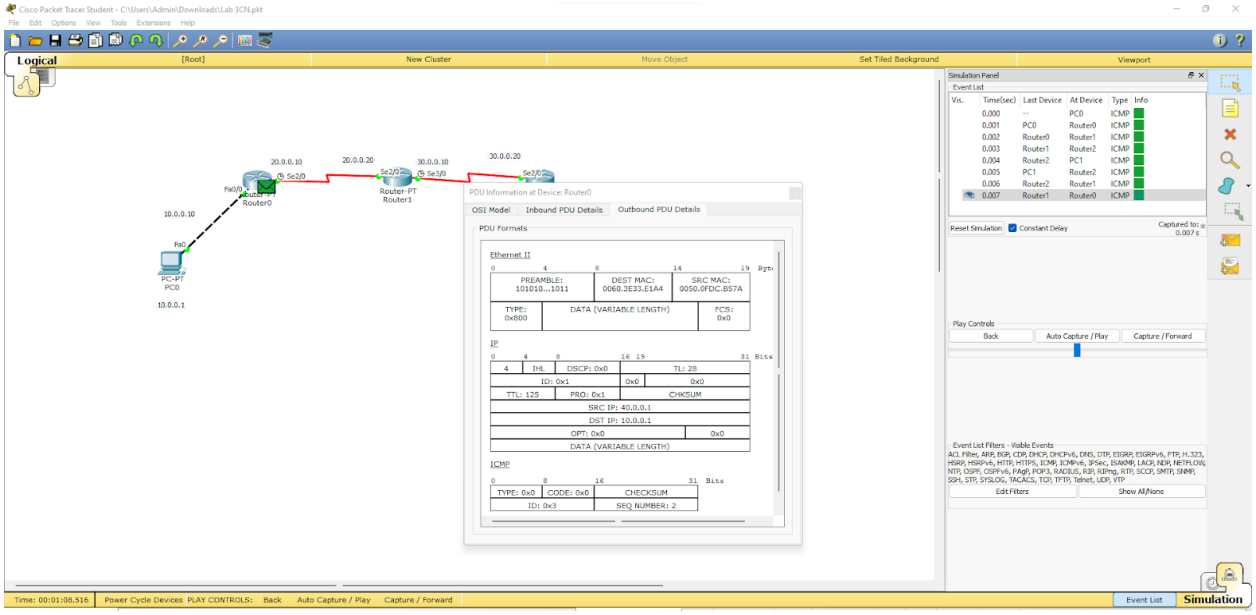

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
PC-PT PC0
10.0.0.10
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
10.0.0.1
PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Time: 00:01:08.511
Power Cycle Devices PLAY CONTROLS:
Simulation
Routers
Scenario 0
New
Delete
Toggle PDU List Window

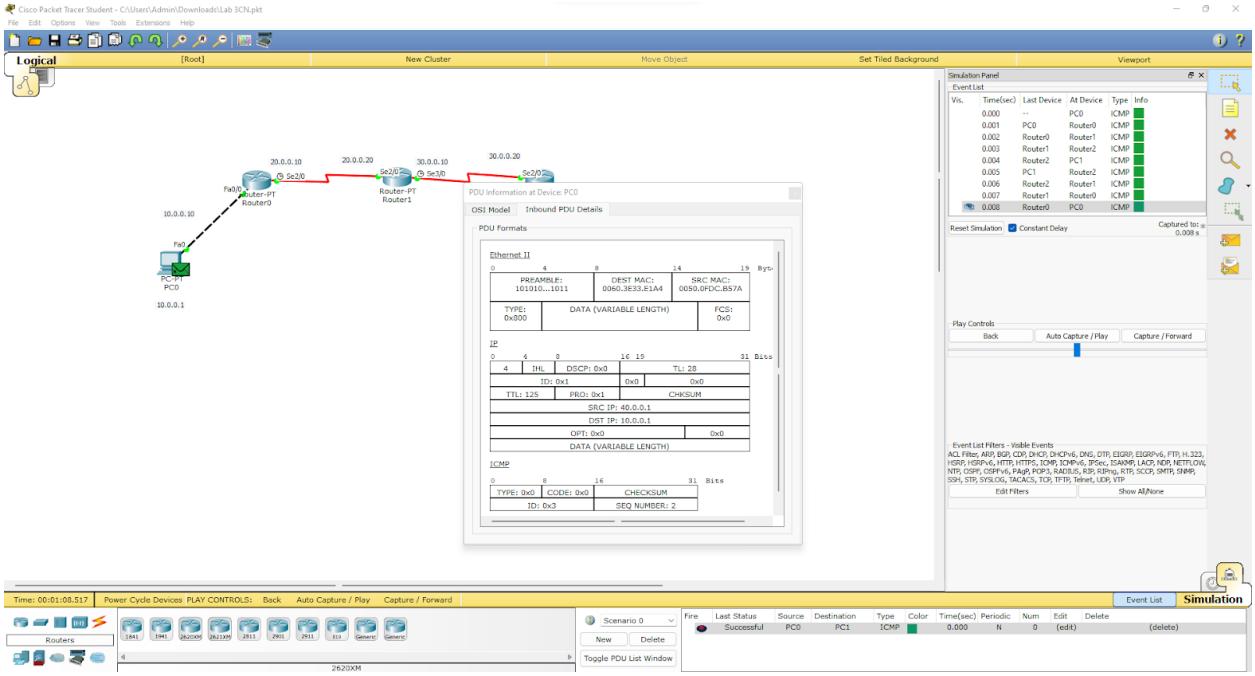

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
PC-PT PC0
10.0.0.10
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
10.0.0.1
PDU Information at Device: Router1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Time: 00:01:08.511
Power Cycle Devices PLAY CONTROLS:
Simulation
Routers
Scenario 0
New
Delete
Toggle PDU List Window


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
Router2
PC0
PC1
10.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
40.0.0.10
10.0.0.1
40.0.0.1
PDU Information at Device: Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Simulation


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
Router2
PC0
PC1
10.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
40.0.0.10
10.0.0.1
40.0.0.1
PDU Information at Device: Router2
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Simulation


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
PC-PT
PC0
10.0.0.10
10.0.0.1
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
PDU Information at Device: Router0
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
HDLC
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Simulation


Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router0
Router1
PC-PT
PC0
10.0.0.10
10.0.0.1
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
PDU Information at Device: Router0
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
Ethernet II
IP
ICMP
Simulation Panel
Event List
Play Controls
Back
Auto Capture / Play
Capture / Forward
Simulation

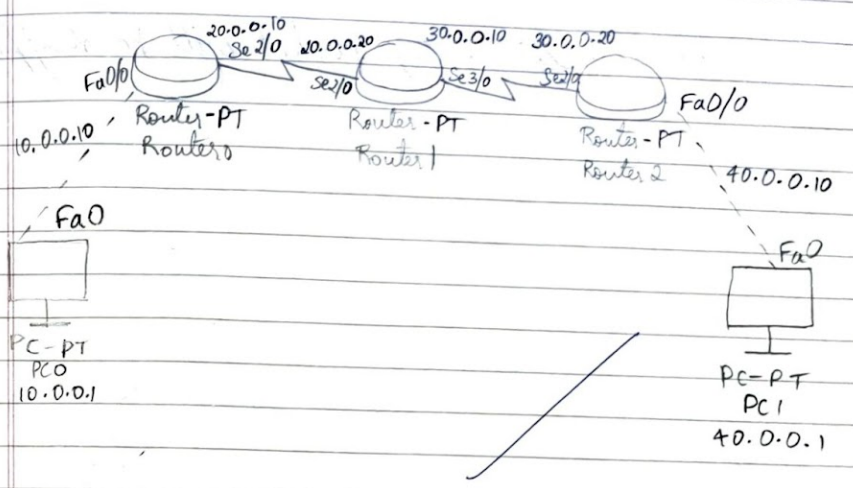

Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
20.0.0.10
20.0.0.20
30.0.0.10
30.0.0.20
Se2/0
Se3/0
Fa0/0
Router-PT
Router0
Router1
10.0.0.10
Fa0
PC-PT
PC0
10.0.0.1
PDU Information at Device: PC0
OSI Model
Inbound PDU Details
PDU Formats
Ethernet II
PREAMBLE: 10101010...1011
DEST MAC: 0060.3E33.E1A4
SRC MAC: 0050.0FDC.B57A
TYPE: 0x800
DATA (VARIABLE LENGTH)
FCS: 0x0
IP
4
IHL
DSCP: 0x0
TL: 28
ID: 0x1
0x0
TTL: 125
PRO: 0x1
CHKSUM
SRC IP: 40.0.0.1
DST IP: 10.0.0.1
OPT: 0x0
DATA (VARIABLE LENGTH)
ICMP
TYPE: 0x0
CODE: 0x0
CHECKSUM
ID: 0x3
SEQ NUMBER: 2
Simulation Panel
Event List
Reset Simulation
Constant Delay
Play Controls
Back
Auto Capture / Play
Capture / Forward
Edit Filters
Show All/None
Time: 00:01:08.517
Power Cycle Devices
PLAY CONTROLS:
Routers
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Status
Source
Destination
Type
Color
Time(sec)
Periodic
Num
Edit
Delete
Successful
PC0
PC1
ICMP
0.000
N
0
(edit)
(delete)
2620XM
Simulation

Observation :

Lab 10

Aim : Demonstrate the TTL / Life of a Packet.

Topology :



Procedure :

1) Create the topology as shown and configure the PCs and routers as shown above. static routing done. We can do any type.

2) In simulation mode, send a packet from PC0 to PC1. Simple PDU sent.

3) Click ~~of~~ on capture / Forword button on Simulation panel step by step.

4) At each step click on the packet to view the Inbound PDU Details and Outbound PDU details.

Observations:-

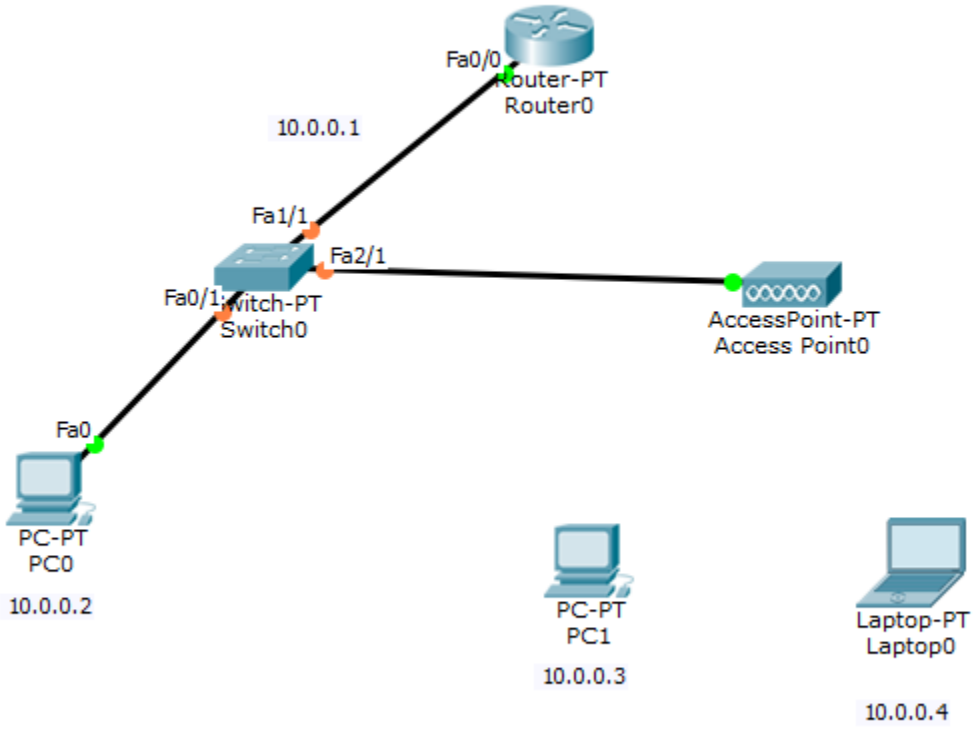
- We can observe that there is a difference of 1 in TTL when the packet crosses every router.
- If TTL becomes 0, packet will be dropped.

16/8/2023

Experiment 11

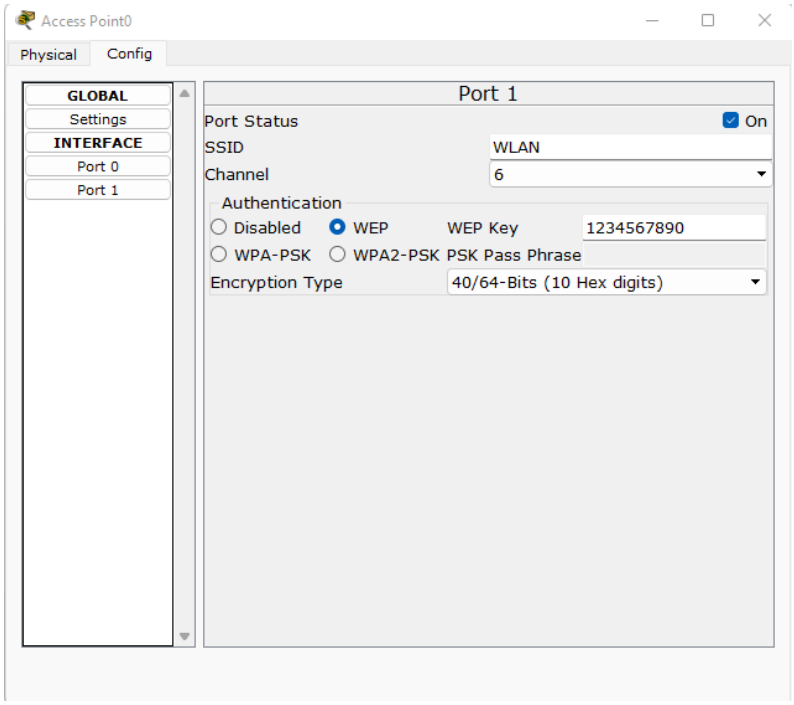
Aim : To construct a WLAN and make the nodes communicate wirelessly

Topology:

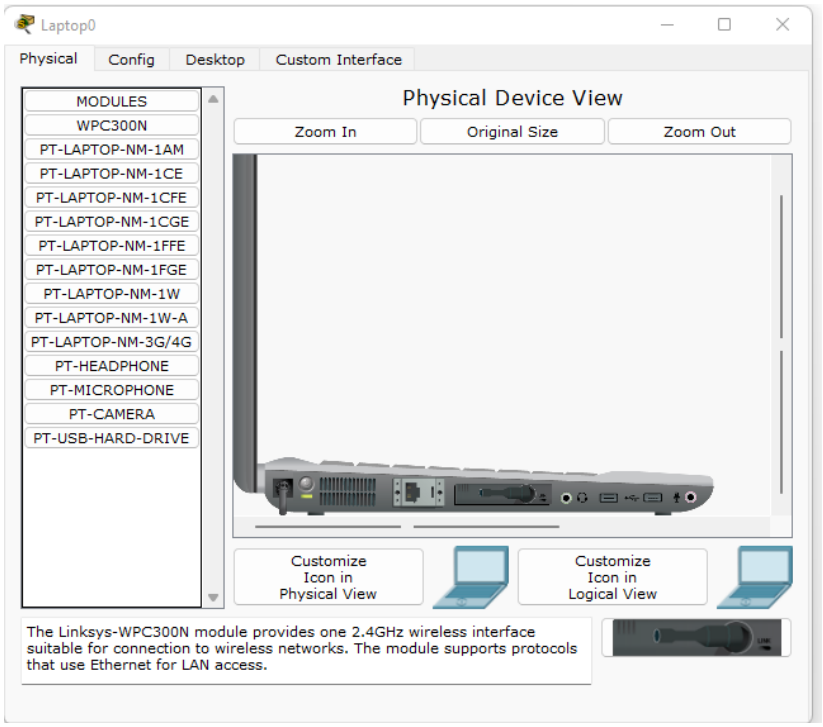


Configurations:

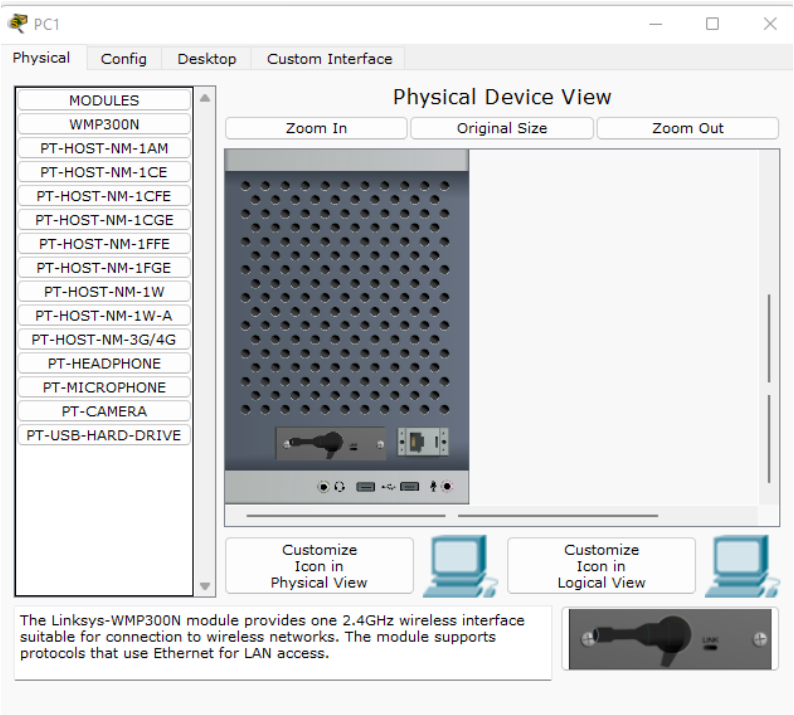
Access Point0:



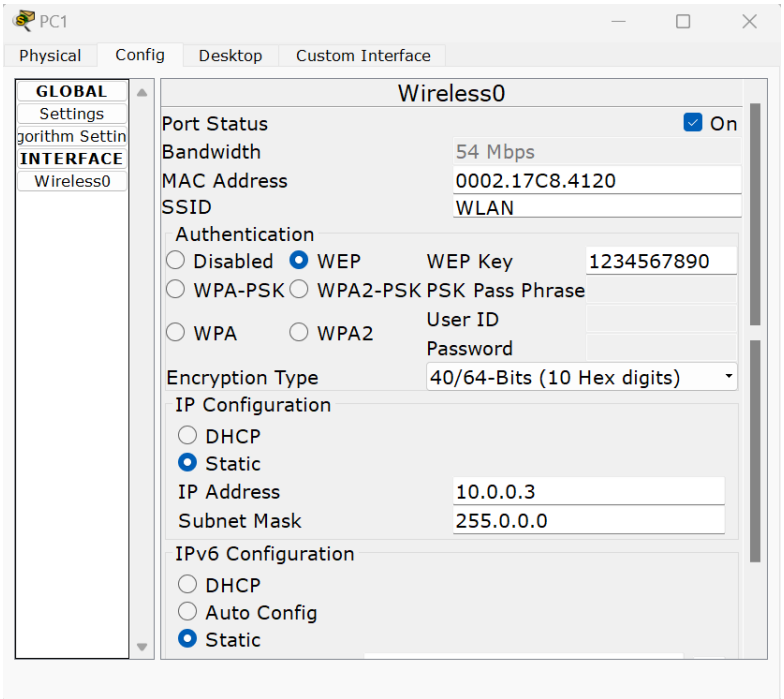
Laptop0 Physical port change:



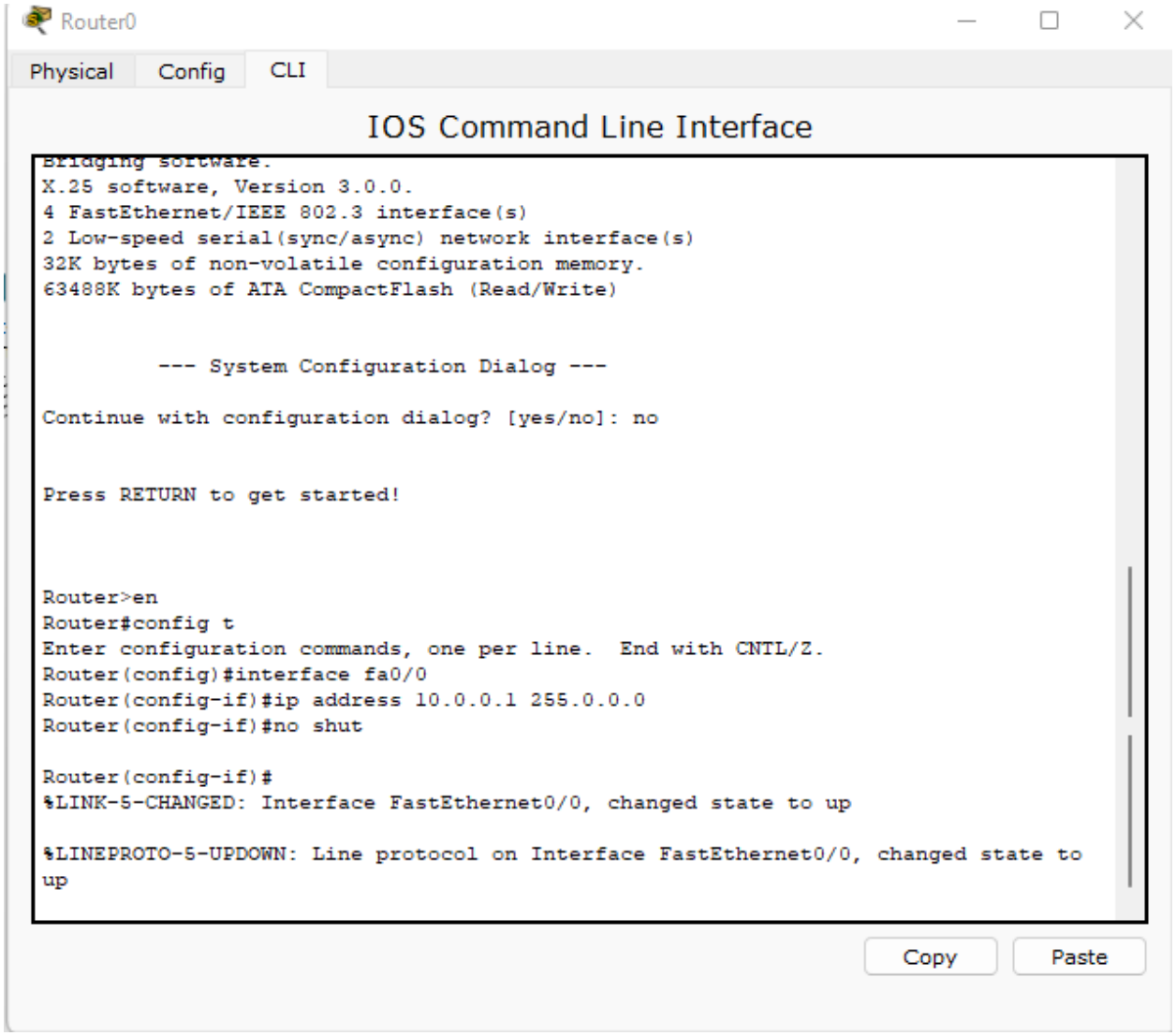
PC0 Physical port change:



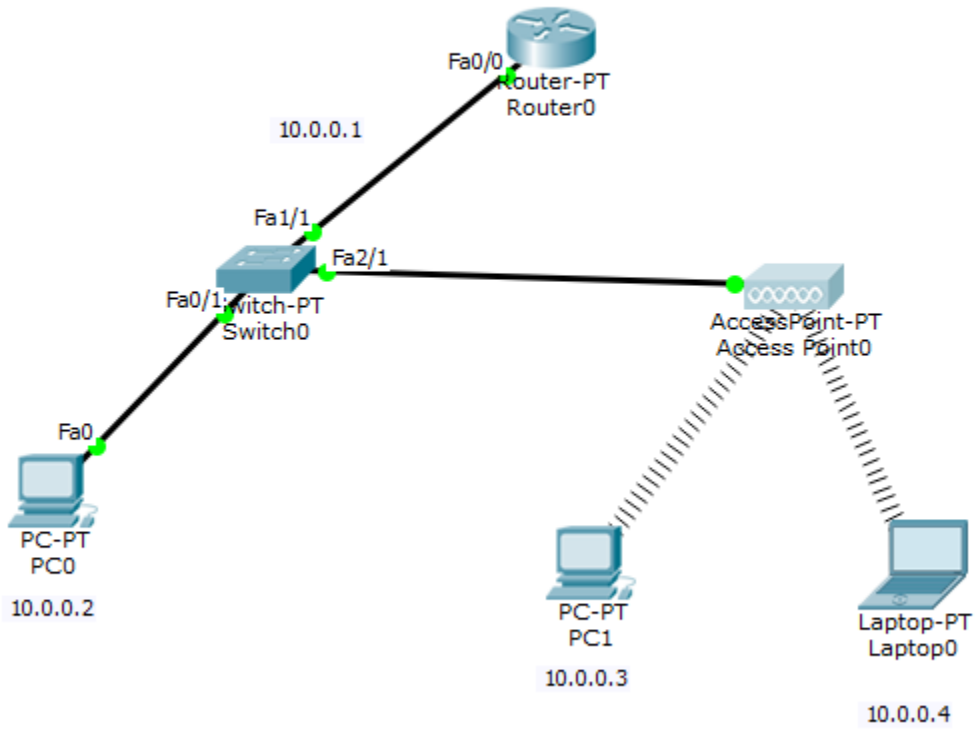
PC0 and Laptop0 Wireless configuration:



Router 0 CLI:

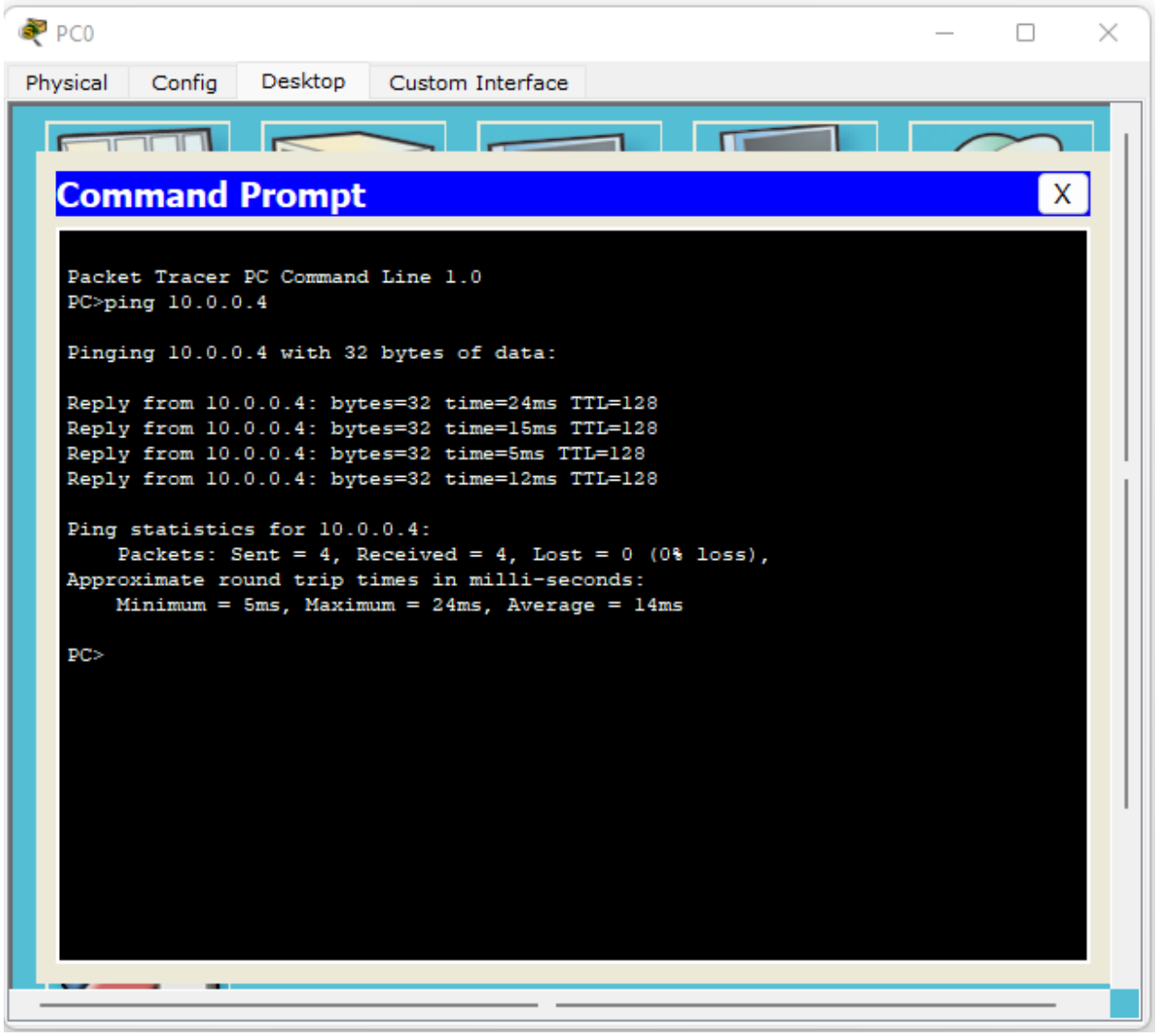

Router0
Physical
Config
CLI
IOS Command Line Interface
Bridging software.
X.25 software, Version 3.0.0.
4 FastEthernet/IEEE 802.3 interface(s)
2 Low-speed serial(sync/async) network interface(s)
32K bytes of non-volatile configuration memory.
63488K bytes of ATA CompactFlash (Read/Write)
--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: no
Press RETURN to get started!
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#no shut
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Copy
Paste

Final Topology:

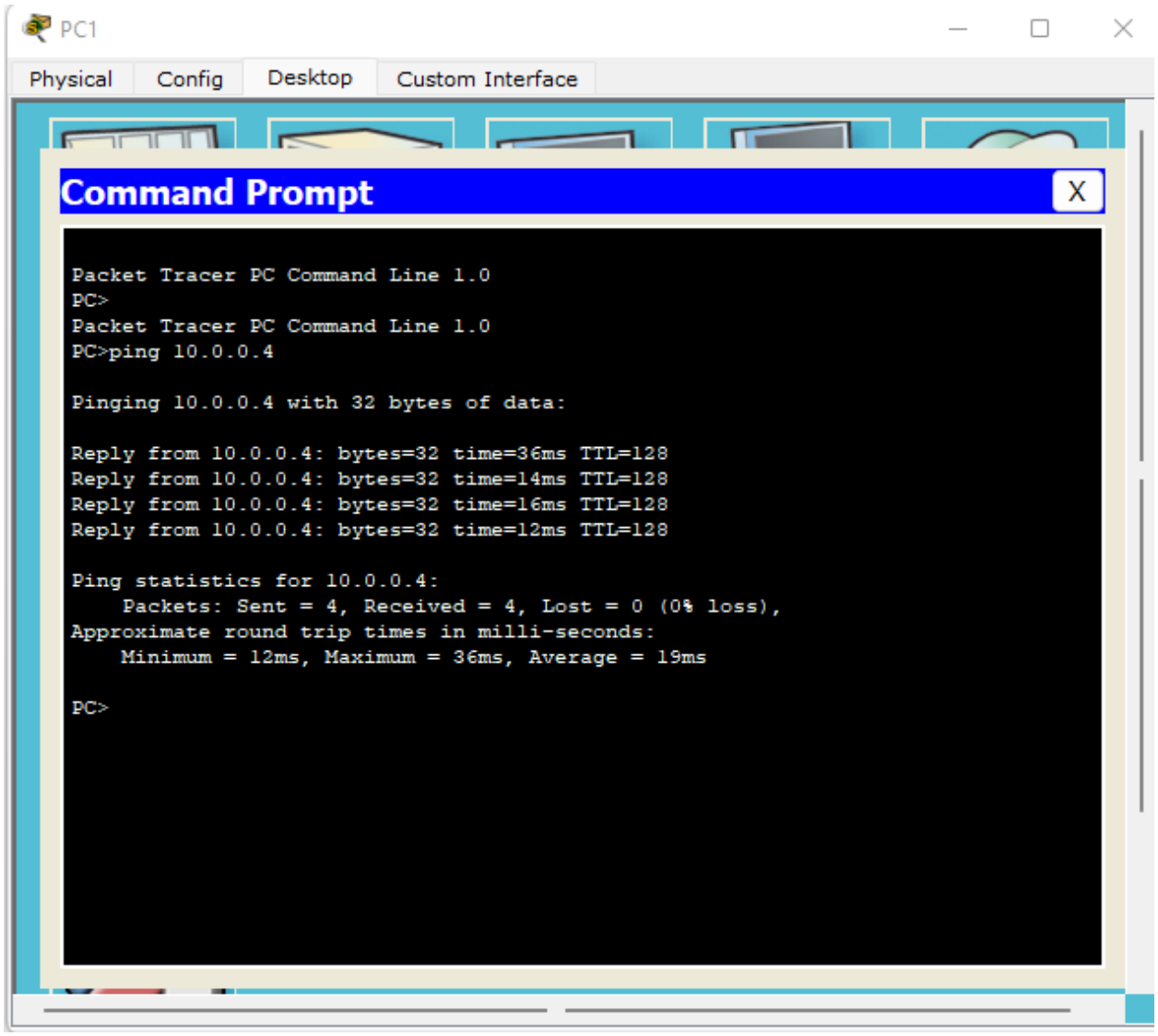


Ping Output :

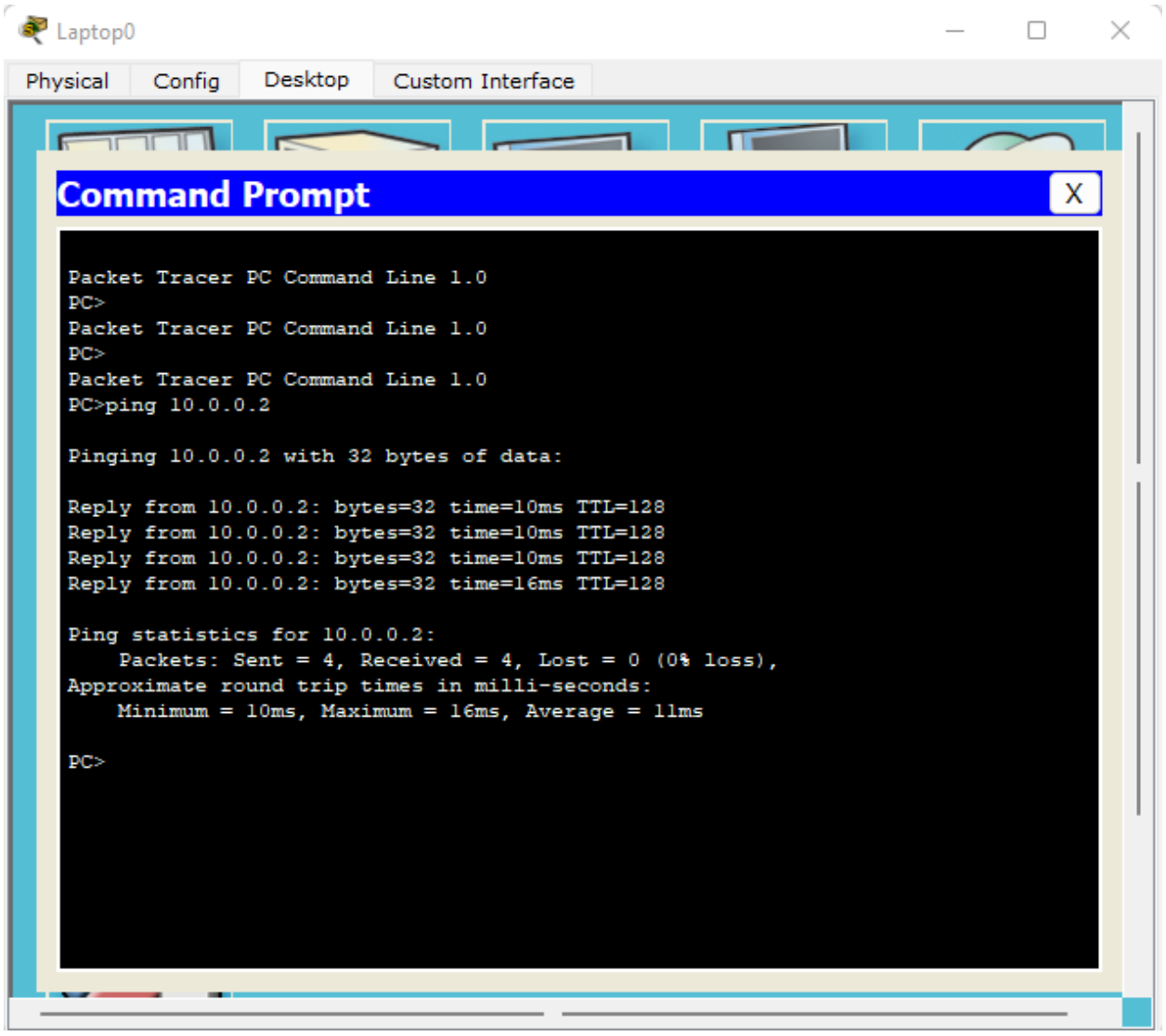
PC0 to Laptop0 :



PC1 to Laptop0 :



Laptop0 to PC0:

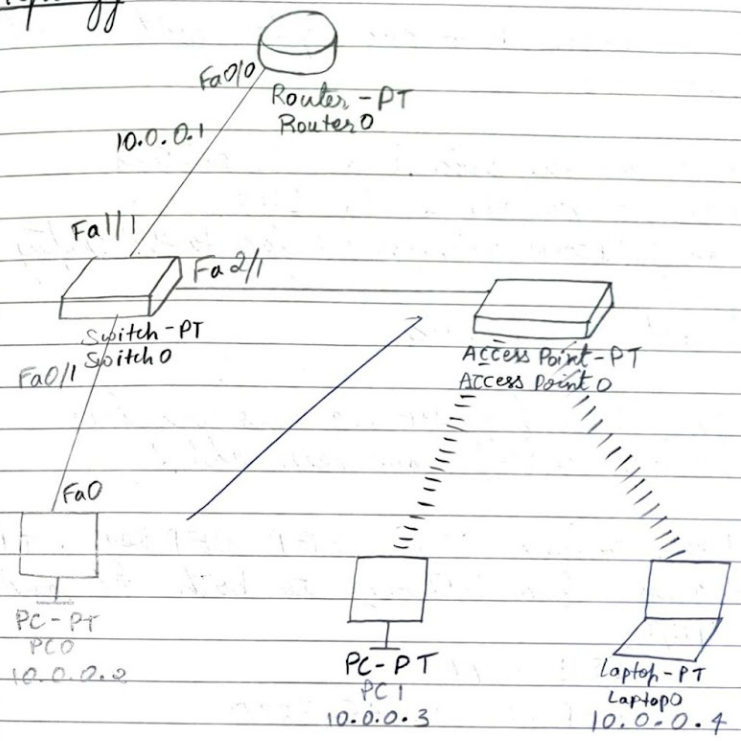


Observation :

Lab 11

Aim : To construct a WLAN and make the nodes communicate wirelessly.

Topology :



Procedure :-

1) Create the topology as shown above
2) Configure PC 3
   IP address : 10.0.0.2
   G
3) Configure Router 0. Set fastethernet 0/0 as 10.0.0.1 in usual method.

4) Configure the Access Point 1 → Port 1 in Config
→ SSID = WLAN
Select WEP and give any 10 digit hex ...
= 1234567890

5) To Configure PC4 and Laptop with Wireless standards.

- Switch off the device. Drag the existing PT-HOST-NM-1AM to the LHS.
- Drag WMP300N wireless interface to the empty port.
- Switch On the device.

6) In the config tab, a new wireless interface would have been added.

7) Now, configure SSID, WEP, WEP Key, IP address and Gateway to both PC and Laptop

SSID = WLAN

WEP key = 1234567890

Gateway = 10.0.0.1

Ping Output: (In PC0)

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=2ms TTL=128

Reply from 10.0.0.4 : bytes = 32 time = 15ms TTL = 128
Reply from 10.0.0.4 : bytes = 32 time = 5ms TTL = 128
Reply from 10.0.0.4 : bytes = 32 time = 12ms 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 = 5ms, Maximum = 24ms, Average = 14ms.

Observation :-

→ We can ping from every device to every other device successfully.

→ Thus we observe that the wireless connection is successfull.

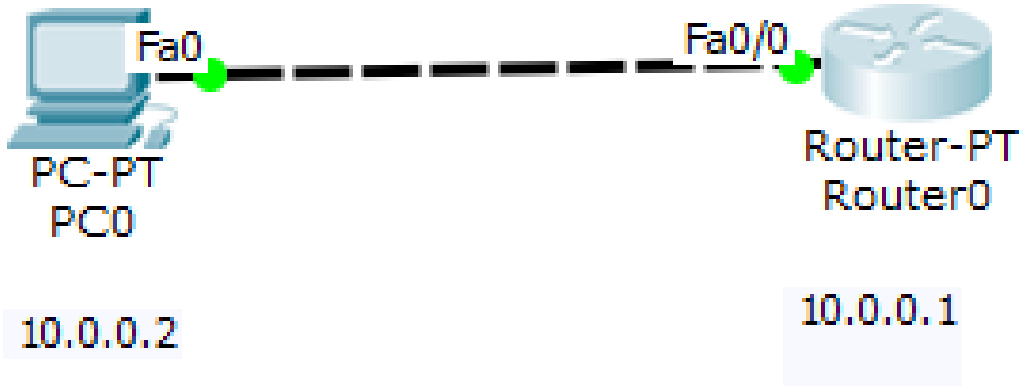
→ In the final topology, we observe stripped lines from access point to PC4 and Laptop0 indicating the establishment of wireless connection.

NLP
10/8/2023

Experiment 12

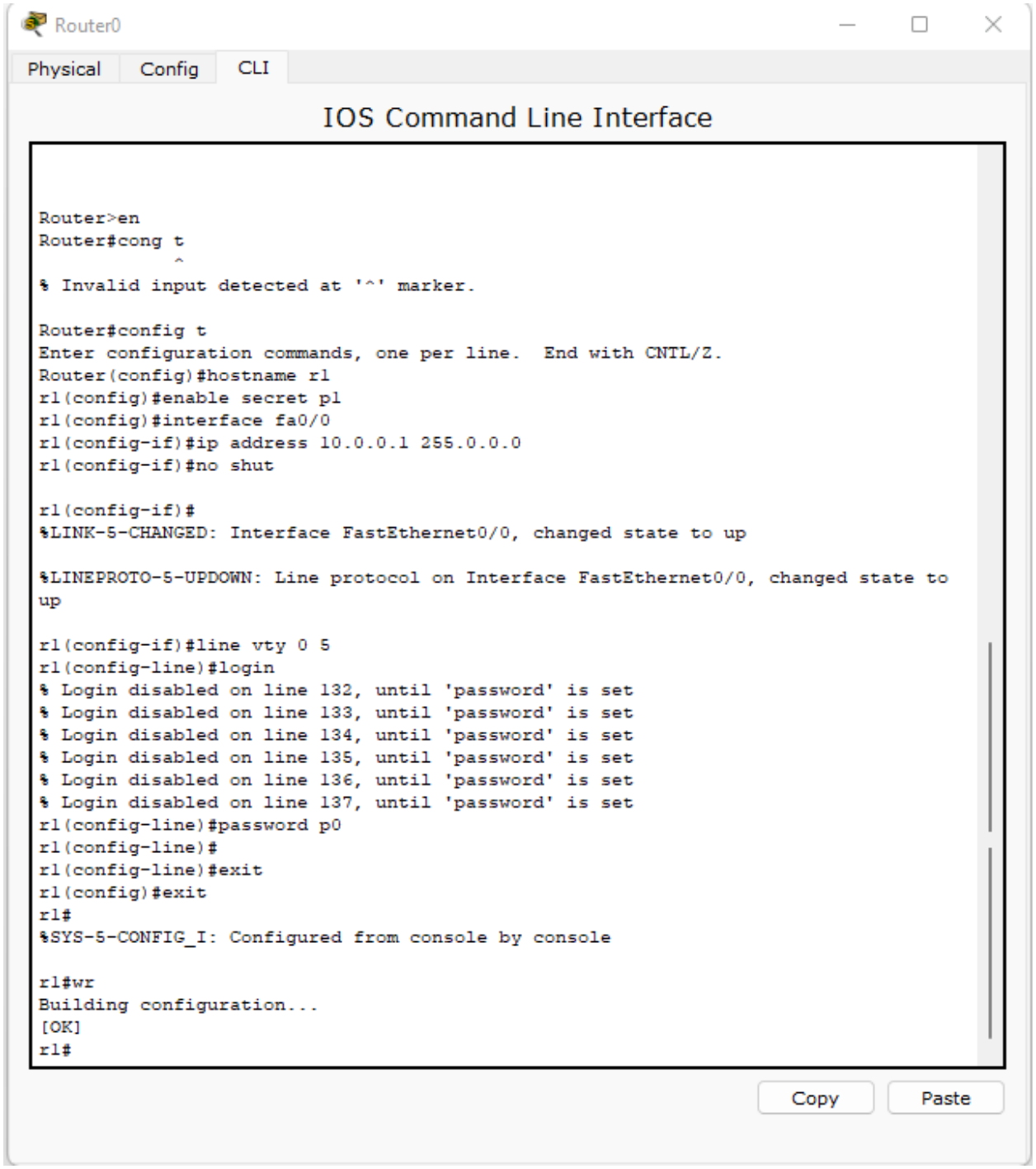
Aim : To understand the operation of TELNET by accessing the router in server room from a PC in IT office.

Topology:



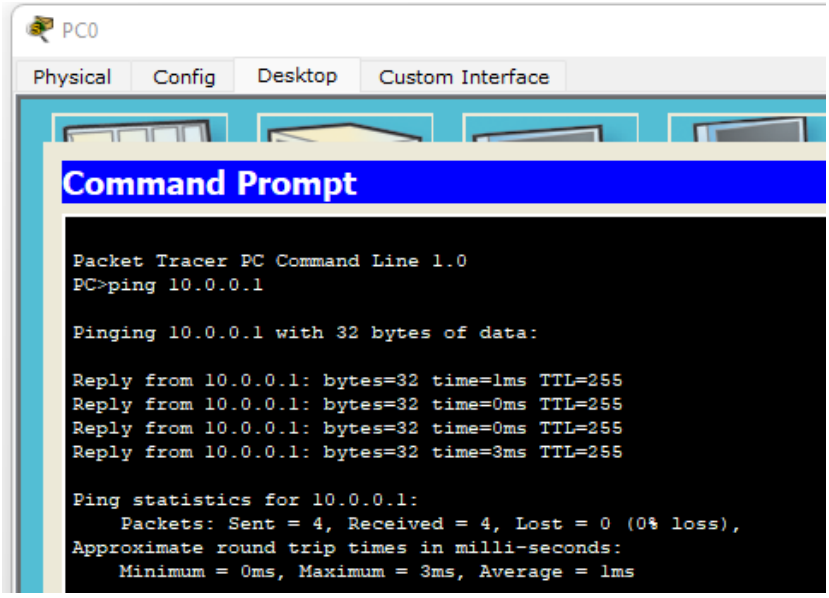
Configuration:

Router 0 CLI:

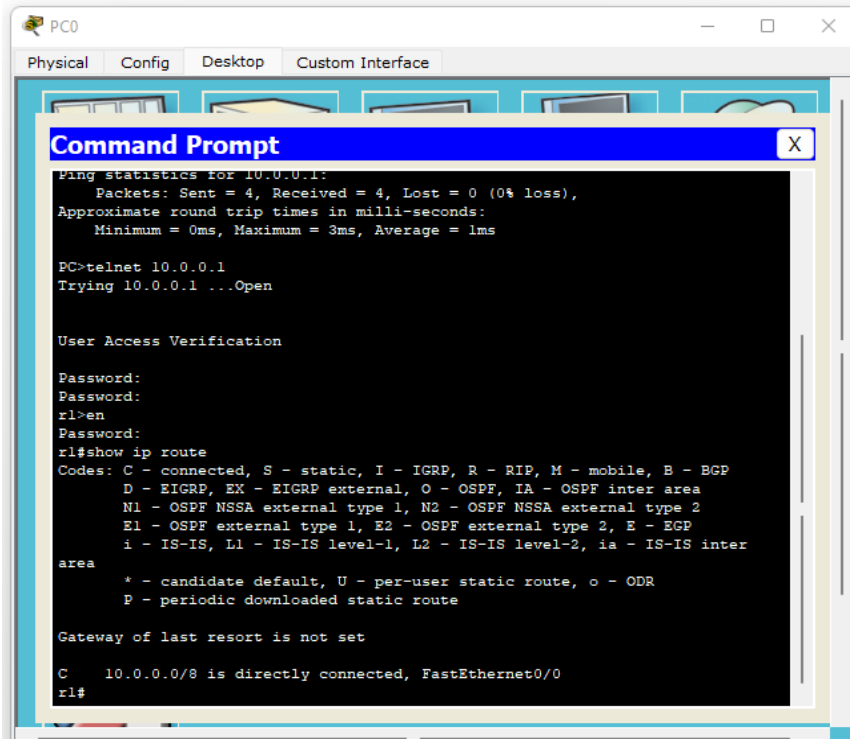


Ping Output:

PC0 to Router:



Accessing the router in server room from a PC in IT office:

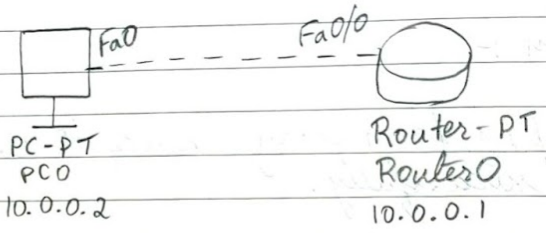


Observation :

Lab 12

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

Topology :-



Procedure :-

1) Create the topology as shown above.
Wire used - Copper cross over.

2) Configure the PC
IP address = 10.0.0.2
Gateway = 10.0.0.1

3) In Router0 CLI

```
Router > en
Router # config t
Router (config)#hostname r1
r1(config) # enable secret p1
r1(config) # int fa0/0
r1(config-if)#ip address 10.0.0.1 255.0.0.0
```

r1(config-if)#no shut
r1(config-if)# line vty 0 5 -- for 6 users

r1(config-line)# login
r1(config-line)# password p0
r1(config-line)# exit
r1(config)# exit

r1# wr
Building Configuration...

Ping Output and Command prompt of PC0

We can successfully Ping 10.0.0.1 from PC0.

In PC0 command prompt :-

PC> telnet 10.0.0.1
Trying 10.0.0.1 ... Open

User Access verification

Password: p0 -> won't be visible on screen
r1> en
Password: p1
r1# show ip route

C 10.0.0.0/8 is directly connected, FastEthernet0/0

Observation :-

→ We can observe that the admin in PC is able to run commands as run in router CLI and see the result from the PC.

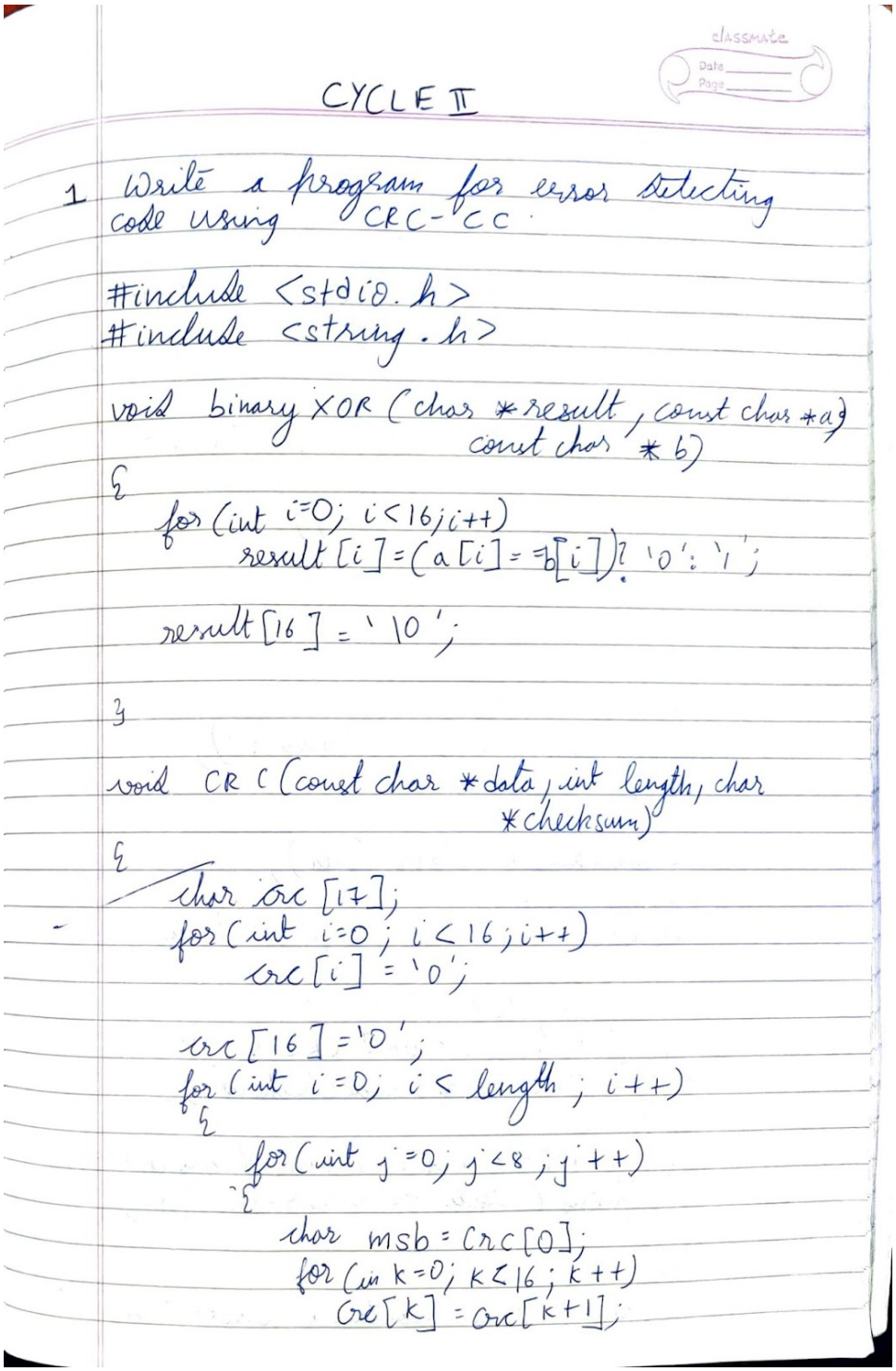
→ Thus, with help of TELNET, we access the router in server room from a PC.

16/8/2023.

Cycle - 2

Experiment 13

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



CYCLE II

1 Write a program for error detecting code using CRC-CC

```
#include <stdio.h>
#include <string.h>

void binaryXOR (char *result, const char *a,
                const char *b)
{
    for (int i=0; i<16;i++)
        result[i] = (a[i] == b[i])? '0': '1';

    result[16] = '\0';

}

void CRC (const char *data, int length, char
          *checksum)
{
    char crc[17];
    for (int i=0; i<16;i++)
        crc[i] = '0';

    crc[16] ='0';
    for (int i=0; i < length ; i++)
    {
        for (int j=0; j<8 ; j++)
        {
            char msb = crc[0];
            for (int k=0; k<16 ; k++)
                crc[k] = crc[k+1];
```

```
        crc[15] = '0';
        if (msb == '1')
        {
            char temp[17];
            binaryXOR(temp, crc, "1000100000010001");
            strcpy(crc, temp);
        }
    }
    crc[15] = [data[i] == '1' ? '1' : '0';
}
strcpy(checksum, crc);
}

void main()
{
    char data[100];
    printf("Enter data in binary : ");
    scanf("%s", data);

    int datalength = strlen(data);
    char checksum[17];
    calculateCRC(data, datalength, checksum);

    char receivedChecksum[17];
    printf("Enter received CRC : ");
    scanf("%s", receivedChecksum);

    if (strcmp(receivedChecksum, checksum) == 0)
        printf("Data is error-free \n");

    else
        printf("Data contains errors. \n");
```

return 0;
}

Output:

Enter data in binary : 1100101011100100 1
Calculated CRC : 1110100101110001
Entered received CRC = 1110100101110001
Data is error free.

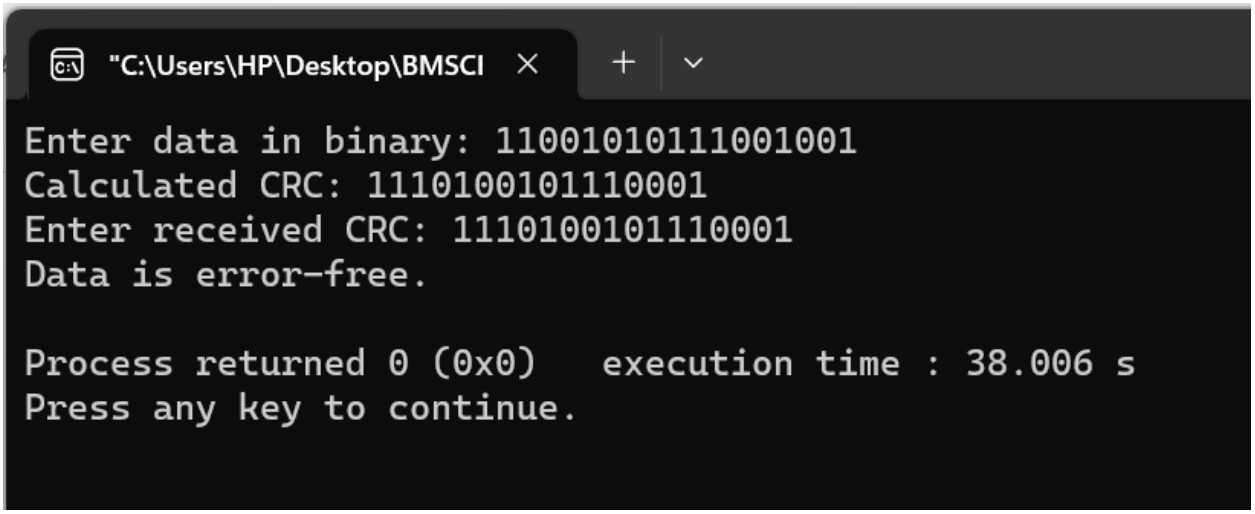
21/8/2023

Program:

```
#include <stdio.h>
#include <string.h>

// CRC-CCITT polynomial: x^16 + x^12 + x^5 + 1 (0x1021)
//#define CRC_POLY 0x1021

// Function to perform bitwise XOR on binary strings
void binaryXOR(char *result, const char *a, const char *b) {
    for (int i = 0; i < 16; i++) {
        result[i] = (a[i] == b[i]) ? '0' : '1';
    }
    result[16] = '\0';
}

// Function to calculate CRC-CCITT checksum
void calculateCRC(const char *data, int length, char *checksum) {
    char crc[17];
    for (int i = 0; i < 16; i++) {
        crc[i] = '0';
    }
    crc[16] = '\0';

    for (int i = 0; i < length; i++) {
        for (int j = 0; j < 8; j++) {
            char msb = crc[0];
            for (int k = 0; k < 16; k++) {
                crc[k] = crc[k + 1];
            }
            crc[15] = '0';

            if (msb == '1') {
                char temp[17];
                binaryXOR(temp, crc, "10001000000100001"); // CRC_POLY in binary
                strcpy(crc, temp);
            }
        }
        crc[15] = (data[i] == '1') ? '1' : '0';
    }
```

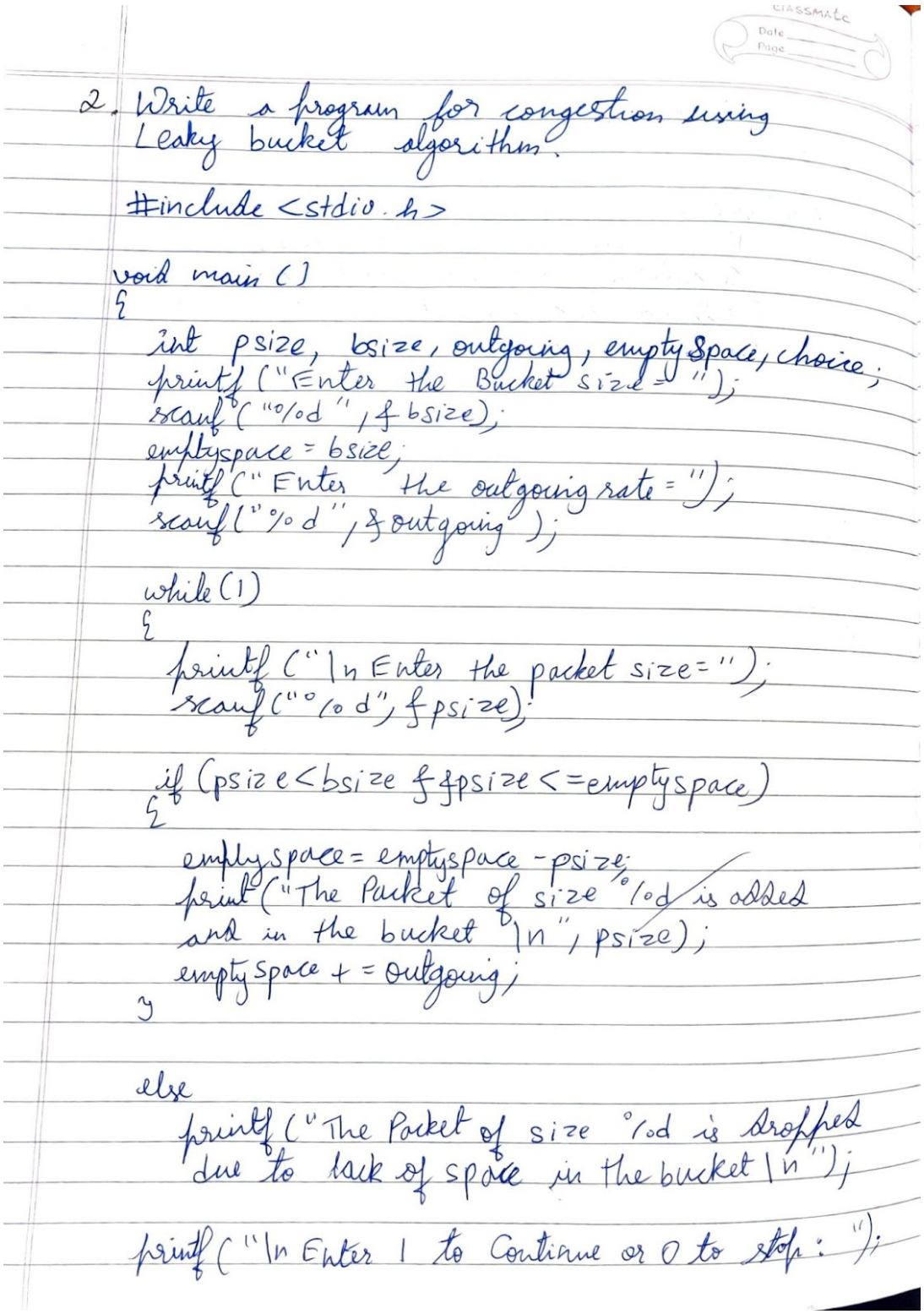
```
    strcpy(checksum, crc);
}

void main() {
    char data[100]; // Replace with your actual data
    printf("Enter data in binary: ");
    scanf("%s", data);

    int dataLength = strlen(data);
    char checksum[17];
    calculateCRC(data, dataLength, checksum);

    printf("Calculated CRC: %s\n", checksum);

    // Simulating error by changing a bit
    // data[2] ^= 0x01; // Uncomment this line to introduce an error

    // Verify the received data
    char receivedChecksum[17];
    printf("Enter received CRC: ");
    scanf("%s", receivedChecksum);

    if (strcmp(receivedChecksum, checksum) == 0)
        printf("Data is error-free.\n");

    else
        printf("Data contains errors.\n");
}
```

Output :


"C:\Users\HP\Desktop\BMSCI
Enter data in binary: 1100101011100 1001
Calculated CRC: 1110100101110001
Enter received CRC: 1110100101110001
Data is error-free.
Process returned 0 (0x0) execution time : 38.006 s
Press any key to continue.

Experiment 14

Aim : Write a program for congestion control using Leaky bucket algorithm.



2. Write a program for congestion using Leaky bucket algorithm.

```
#include <stdio.h>

void main()
{
  int psize, bsize, outgoing, emptySpace, choice;
  printf("Enter the Bucket size = ");
  scanf("%d", &bsize);
  emptyspace = bsize;
  printf("Enter the outgoing rate = ");
  scanf("%d", &outgoing);

  while(1)
  {
    printf("\n Enter the packet size=");
    scanf("%d", &psize);

    if (psize<bsize && psize<=emptyspace)
    {
      emptyspace = emptyspace - psize;
      print("The Packet of size %d is added
      and in the bucket \n", psize);
      emptyspace += outgoing;
    }

    else
      printf("The Packet of size %d is dropped
      due to lack of space in the bucket \n");

    printf("\n Enter 1 to Continue or 0 to stop: ");
```

```
scanf("%d", &choice);
if (choice == 0)
    break;
}
}
```

Output:-

Enter the bucket size : 5000
Enter the outgoing rate : 200

Enter the packet size : 300
The packet of size 3000 is added and in the bucket

Enter 1 to continue or 0 to stop : 1

Enter the packet size : 2000
The packet of size 2000 is added and in the bucket.

Enter 1 to continue or 0 to stop : 1

Enter the packet size : 1500
The packet of size 15000 is dropped due to lack of space in the bucket.

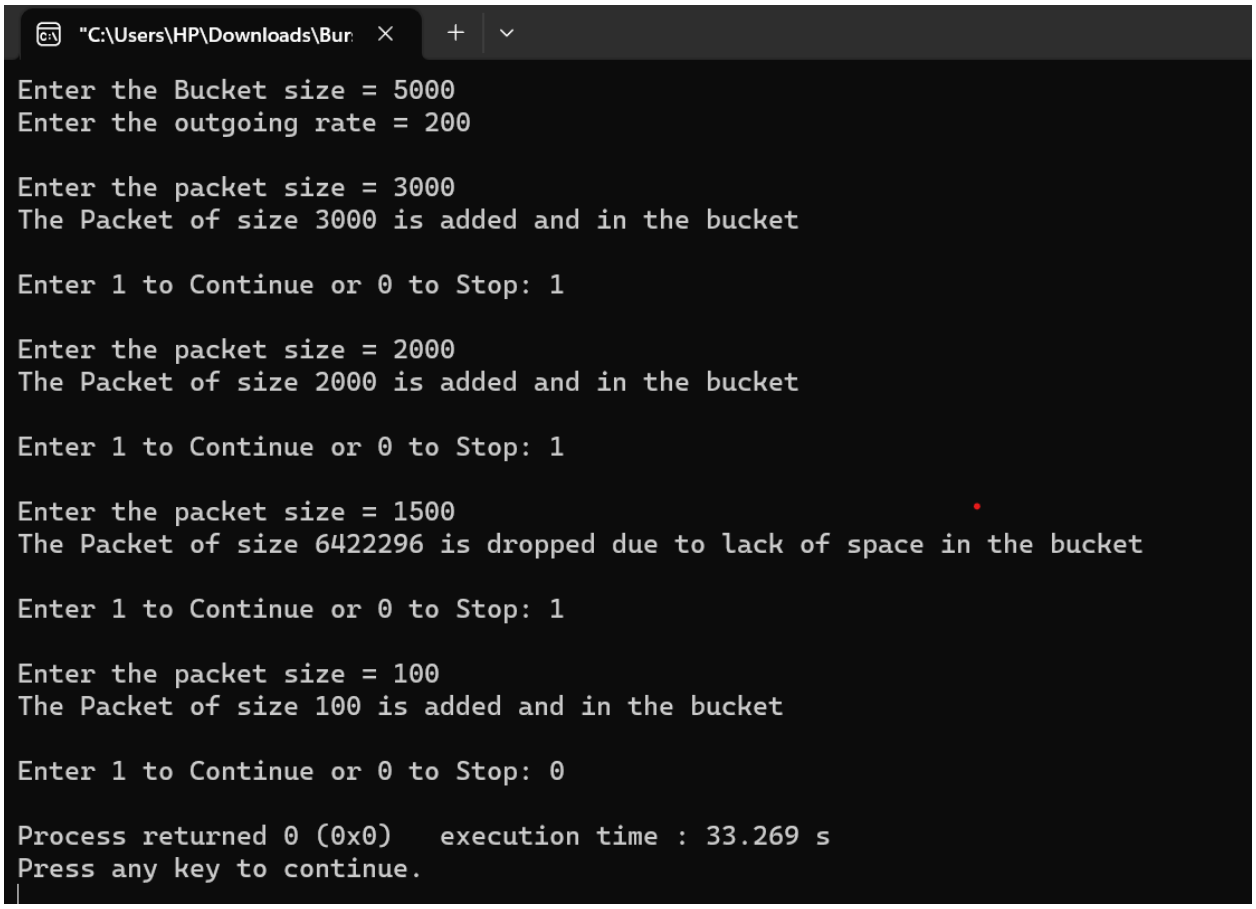
Enter 1 to continue or 0 to stop : 1

Enter the packet size : 100
The packet of size 100 is added and in the bucket.

Program:

```
#include<stdio.h>

void main()
{
   int psize,bsize,outgoing,emptyspace,choice;
   printf("Enter the Bucket size = ");
   scanf("%d",&bsize);
   emptyspace=bsize;
   printf("Enter the outgoing rate = ");
   scanf("%d",&outgoing);
   while(1)
   {
      printf("\nEnter the packet size = ");
      scanf("%d",&psize);

      if(psize<bsize&&psize<=emptyspace)
      {
         emptyspace=emptyspace-psize;
         printf("The Packet of size %d is added and in the bucket \n",psize);
         emptyspace+=outgoing;
      }

      else
      {
         printf("The Packet of size %d is dropped due to lack of space in the bucket\n");
      }

      printf("\nEnter 1 to Continue or 0 to Stop: ");
      scanf("%d",&choice);
      if(choice==0)
         break;

   }
}
```

Output :


"C:\Users\HP\Downloads\Bur
Enter the Bucket size = 5000
Enter the outgoing rate = 200
Enter the packet size = 3000
The Packet of size 3000 is added and in the bucket
Enter 1 to Continue or 0 to Stop: 1
Enter the packet size = 2000
The Packet of size 2000 is added and in the bucket
Enter 1 to Continue or 0 to Stop: 1
Enter the packet size = 1500
The Packet of size 6422296 is dropped due to lack of space in the bucket
Enter 1 to Continue or 0 to Stop: 1
Enter the packet size = 100
The Packet of size 100 is added and in the bucket
Enter 1 to Continue or 0 to Stop: 0
Process returned 0 (0x0) execution time : 33.269 s
Press any key to continue.

Experiment 15

Aim : Using TCP/IP 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.

3. Using TCP/IP 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.

ServerTCP.py

```
from socket import *
serverName = "127.0.0.1" → loopback address
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()
```

ClientTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
```

```
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
sentence = input("\n Enter file name: ")

clientSocket.send(sentence.encode())
fileContents = clientSocket.recv(1024).decode()
print('\nFrom Server: \n')
print(fileContents)
clinetSocket.close()
```

Procedure:

- Create 2 IDLE instances and write client and server files.
- Run server first and then the client.

Output:-

Server Instance :-

The server is ready to receive

Client Instance :-

Enter file name : ServerTCP.py

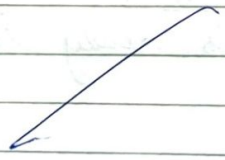
From Server :

The contents of ServerTCP.py is displayed here

Server Instance:-

The server is ready to receive

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



Program:

ServerTCP.py:

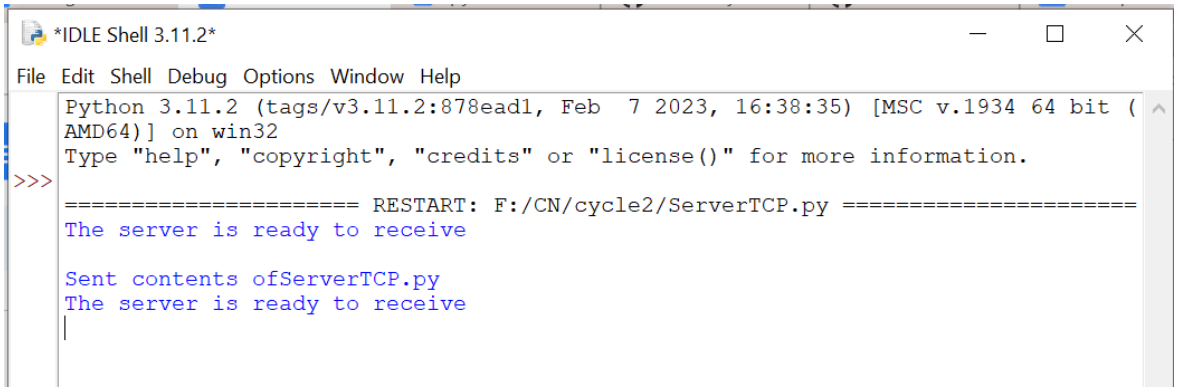
```
from socket import *
serverName="127.0.0.1"
serverPort=12000
serverSocket=socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
    print("The serve is ready to receive")
    connectionSocket,addr = serverSocket.accept()
    sentence=connectionSocket.recv(1024).decode()

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

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

ClientTCP.py:

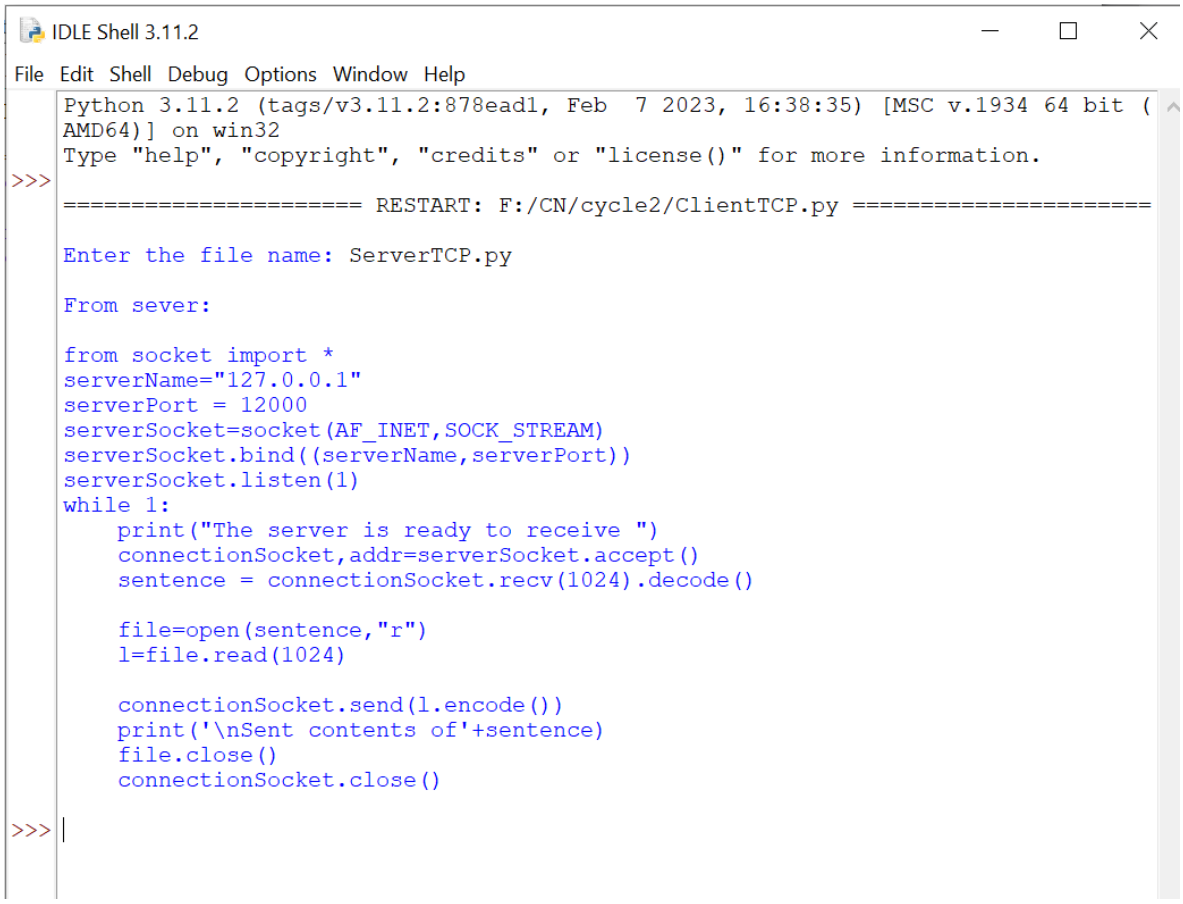
```
from socket import *
serverName='127.0.0.1'
serverPort=12000
clientSocket=socket(AF_INET,SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence=input("\nEnter file name: ")

clientSocket.send(sentence.encode())
filecontents=clientSocket.recv(1024).decode()
print('\nFrom Server:\n')
print(filecontents)
clientSocket.close()
```

Output :

Server instance:



Client instance:



Experiment 16

Aim : 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.

4. 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.

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('\n Sent contents of ', end = ' ')
    print(sentence)
    file.close()
```

ClientUDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clinetSocket = socket (AF_INET, SOCK_DGRAM)

sentence = input("\n Enter file name : ")

clinet
client Socket.sendto(bytes(sentence, "utf-8"),
                     (serverName, serverPort))

filecontents, serverAddress = client Socket.recvfrom
                                (2048)
print ('\n Reply from Server: \n')
print (filecontents.decode("utf-8"))

client Socket.close()
client socket.close()
```

Output :

Server Instance :-

The server is ready to receive

Client Instance :-

Enter file name : ServerUDP.py

Reply from Server:

contents of ServerUDP.py displayed here.

Server Instance :-

The server is ready to receive

Sent contents of ServerUDP.py
The server is ready to receive.

29/8/2023

Program:

ServerUDP.py:

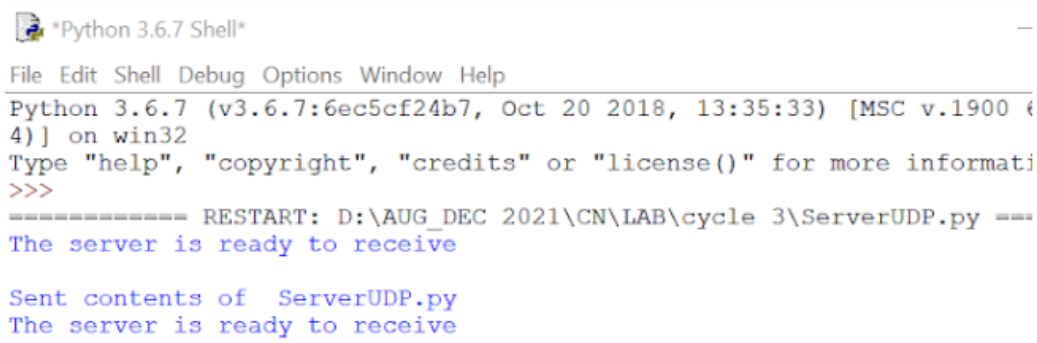
```
from socket import *
serverPort=12000
serverSocket=socket(AF_INET,SOCK_DGRAM)
serverSocket.bind(("127.0.0.1",serverPort))
print("The server is ready to receive")
while 1:
   sentence,clientAddress=serverSocket.recvfrom(2048)
   sentence=sentence.decode("utf-8")
   file=open(sentence,"r")
   con=file.read(2048)

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

   print('\nSent contents of',end='')
   print(sentence)
   file.close()
```

ClientUDP.py:

```
from socket import *
serverPort=12000
serverName="127.0.0.1"
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()
```

Output :

Server instance :



Client instance :

Experiment 17

Aim : Tool Exploration -Wireshark.

Wireshark

- Wireshark, a powerful open-source network protocol analyzer, or packet analyzer.
- It is used for network troubleshooting, analysis, Software and communication protocol development and education.
- It is used to track packets so that each one is filtered to meet our specific needs. It is also used by network security engineers to examine security problems.

Functionality of Wireshark :-

- It lets the user put network interface controllers that support promiscous mode into that mode, so they can see all the traffic visible on that interface.
- Packet capture and filtering:
  Primary function of wireshark lies in capturing network packets from various interfaces. Its flexible filtering options enable users to capture specific types of traffic based on protocol, source/destination addresses and even keywords within packet payloads.
- Real time Analysis:
  Wireshark's real time monitoring capability is invaluable for observing ongoing network activities

- This feature helps in detecting sudden traffic spikes, unusual protocol behaviour, and unauthorized network usage.
- Protocol Analysis:
  It decrypts encrypted protocols offering insights into secure communication method.
- Packet Reconstruction: Allows reassembling of fragmented packets.
- Statistical Information: Presents statistical analysis of captured data.
- Color-coded visualization: Employs color-coded packets to indicate various aspects such as error.
- Customizable Display: This tool offers a customizable interface where users can choose which fields to display & how to arrange them.

Procedure:

1) In the 1st window, select ethernet.
2) Fitles TCP or any required protocol.
3) Click on it, new window opens.
4) Dropdown: Transmission Control Protocol.

Scr Port: 62148, Dst Port: 443, Seq: 2,
Ack: 65, Len: 6

5) This is available in the previous window in the left split of screen.

6) Clicking on dropdown of it, clicking on any of them highlights its counterport in right split side of screen.

7) In cmd, type > ipconfig

~~RES~~

Result :-

Windows IP configuration

Ethernet adapter Ethernet :

Connection-specific DNS suffix :

Link-local IPv6 Address . . . : fe80::he78:ffa9:
ed25:e329:b3

IPv4 Address . . . : 10.129.2.83

Subnet Mask . . . : 255.255.0.0

Default-Gateway . . . : 10.127.0.11