

LAB 1 – 9/10/2024

9/10 Date / /201

Expt. #1 :

3. ~~Two~~ PCs & Router

10.0.0.1 20.0.0.1
Router-PT Router 0
Fa0 Fa1/0
Def Gateway 10.0.0.1
(for this side PCs)
Def Gateway 20.0.0.1
(for this side PCs)

PC - PT
PC0
10.0.0.10

PC - PT
PC1
20.0.0.10

Aim — To connect 2 PCs belonging to different networks using a router.

Topology — 2 PCs are connected to each other using copper cross-over cable through the router

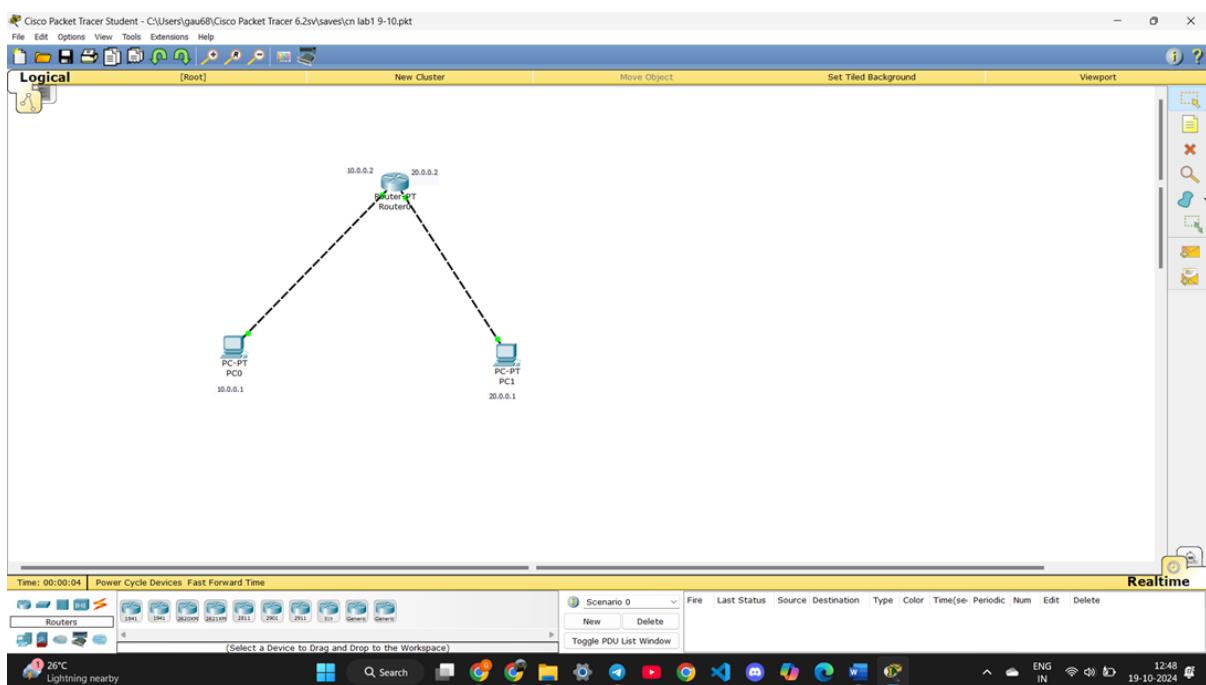
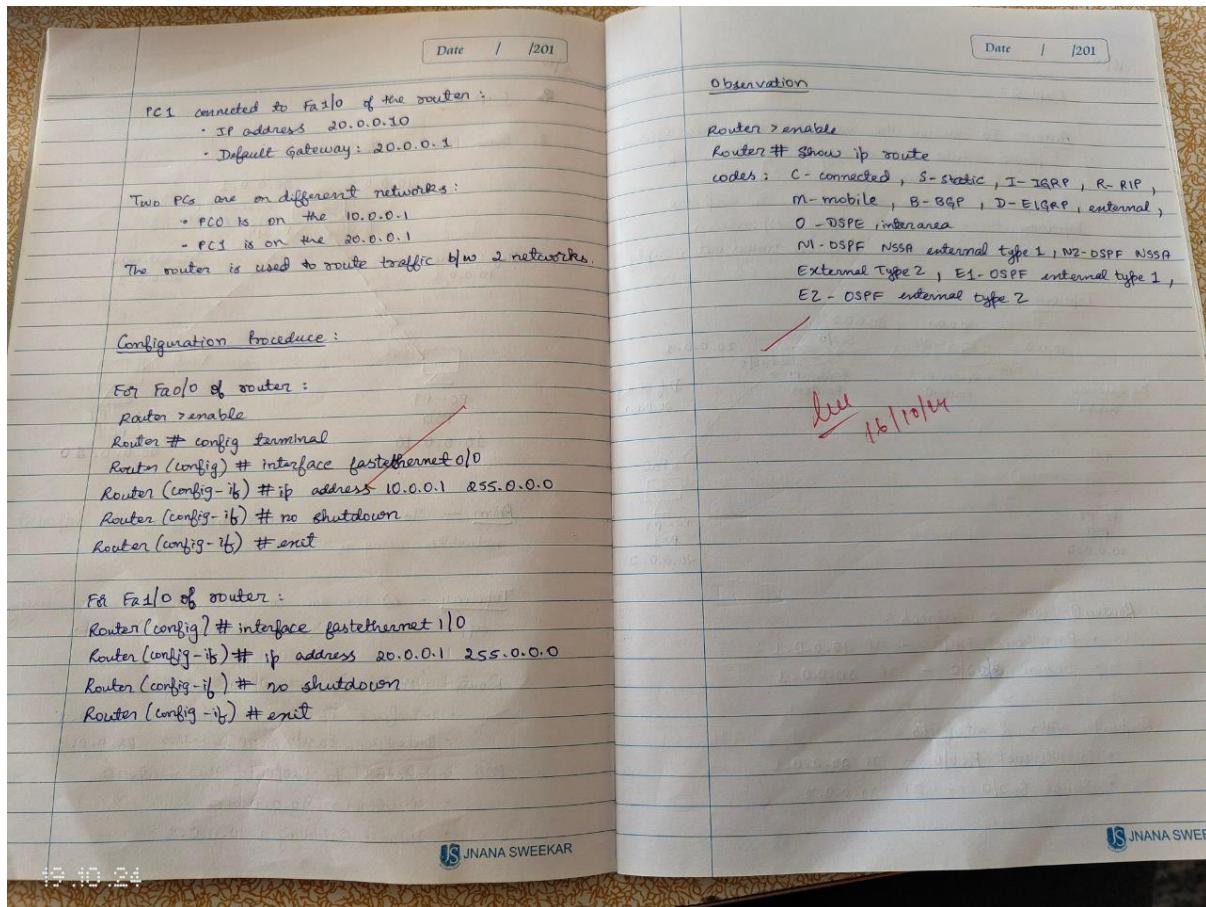
Router (Router 0) with two interfaces:

- Interface Fa0/0 : IP address 10.0.0.1
- Interface Fa1/0 : IP address 20.0.0.1

PC0 connected to Fa0/0 of the router:

- IP address 10.0.0.10
- Default Gateway : 10.0.0.1

JNANA SWEETKAR



Router0

Physical Config CLI

IOS Command Line Interface

```
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#no shutdown

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

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

Router(config-if)#exit
Router(config)#interface fastethernet 1/0
Router(config-if)#ip address 20.0.0.2 255.0.0.0
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

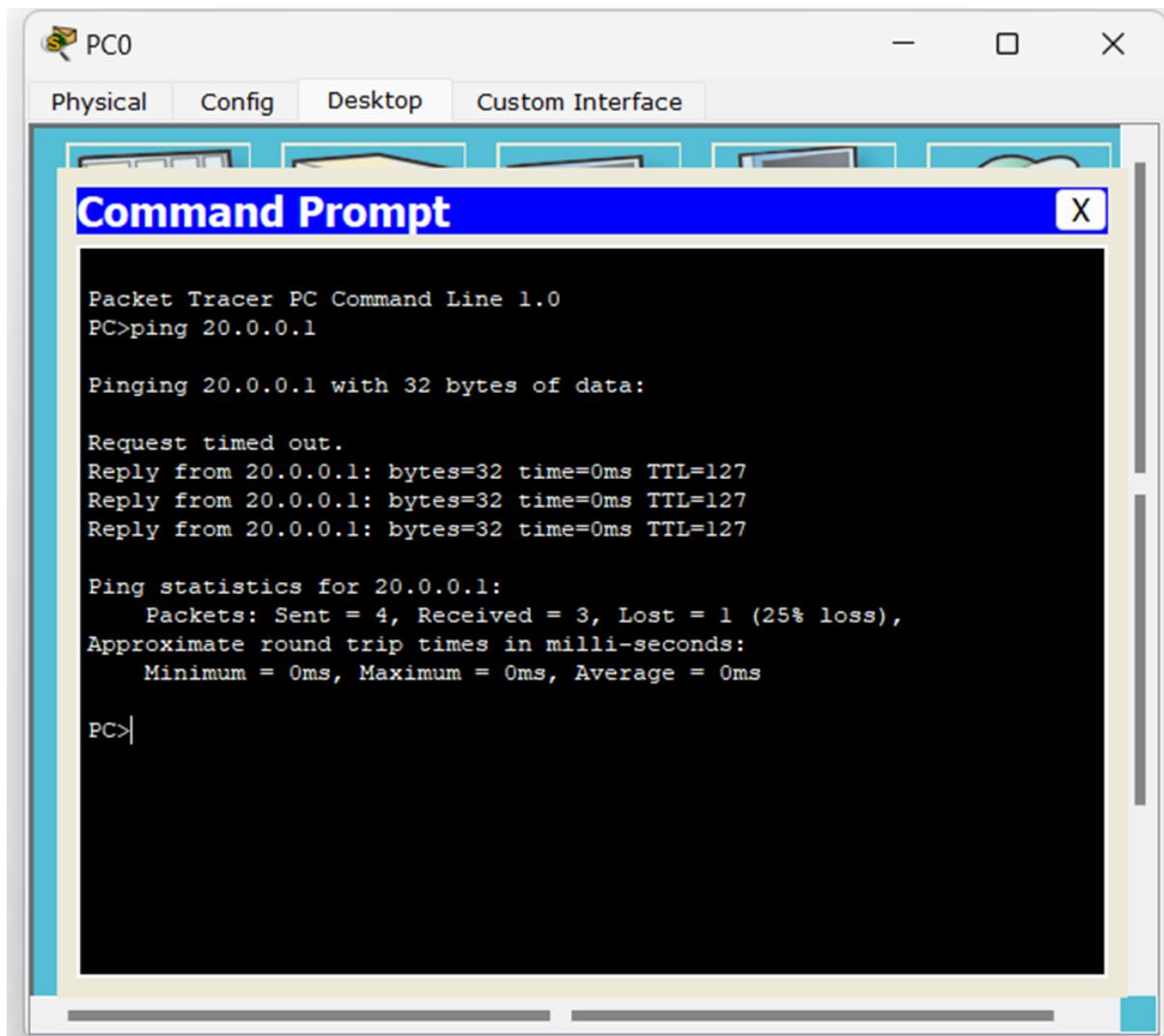
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

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

C    10.0.0.0/8 is directly connected, FastEthernet0/0
C    20.0.0.0/8 is directly connected, FastEthernet1/0
Router#
```

Copy Paste



LAB 2 – 16/10/2024

Date / /201

Expt 2 :

Aim - To observe the various error messages displayed while connecting 2 routers of different networks.

Configure IP address to routers & explore topology these messages - ping responses, destination unreachable, request timed out, reply.

Topology -

Router0 with 2 interfaces :

- Fastethernet Fa0/0 - IP 10.0.0.1
- Serial S2/0 - IP 30.0.0.1

Router1 with 2 interfaces :

- Fastethernet Fa0/0 - IP 20.0.0.1
- Serial S2/0 - IP 30.0.0.2

Date / /201

PC0 connected to Fa0/0 of Router0 :

- IP 10.0.0.2
- Default gateway 10.0.0.1 (For all PCs connected to the line 10.0.0.0)

PC1 connected to Fa0/0 of Router0 :

- IP 20.0.0.2
- Default Gateway 20.0.0.1 (For all PCs connected to the line 20.0.0.0)

PC0 is on the network 10.0.0.1 & PC1 is on the network 20.0.0.1. 2 routers used to communicate.

Procedure -

For Router0 :

```

Router>enable
Router# config terminal
Router(config)# interface fastethernet 0/0
Router(config-if)# ip address 10.0.0.1 255.0.0.0
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface serial 2/0
Router(config-if)# ip address 30.0.0.1 255.0.0.0
Router(config-if)# no shutdown
Router(config-if)# exit
    
```

||| for Router1 also.

JS JNANA SWEETKAR

Date / /201

Observation -

PC> For PC0 : In Command Prompt

PC> ping 30.0.0.2

Destination host unreachable. [Error]

∴ For Router 0 :-

Router>enable

Router# config terminal

Router(config)# ip route 20.0.0.0 255.0.0.0 30.0.0.2

Router(config)# exit

Router# show ip route

For Router 1 :-

Router>enable

Router# show ip route

Router# config terminal

Router(config)# ip route 10.0.0.0 255.0.0.0 30.0.0.1

Router(config)# exit

Router# show ip route

S 10.0.0.0 [1/0] via 30.0.0.1

Router# exit

PC0 command Prompt :-

PC > ping 30.0.0.2

Pinging 30.0.0.2 with 32 bytes of data:

Reply from 30.0.0.2: bytes=32 time=1ms TTL=254

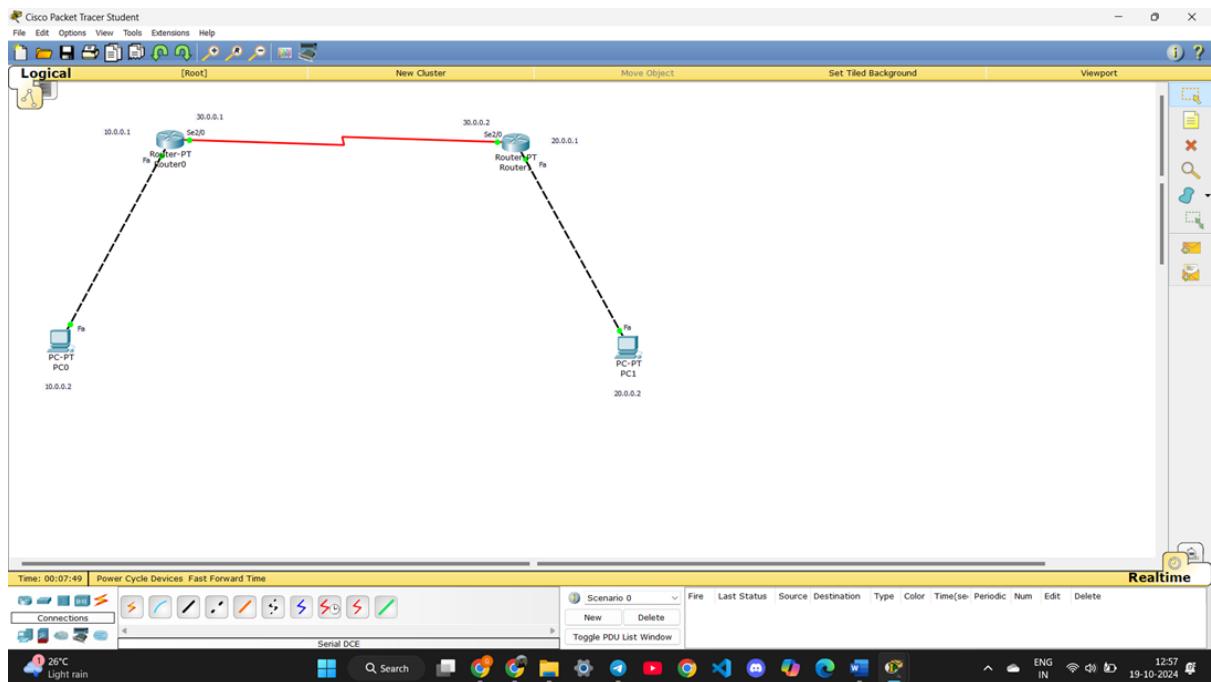
4 times, total

Ping statistics for 30.0.0.2 :

Packets: Sent=4, Received=4, Lost=0 (0.0% loss)

JNANA SWEEKAR

19.10.24



Router0

Physical Config CLI

IOS Command Line Interface

```
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fastethernet 0/0
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

Router(config-if)#exit
Router(config)#interface serial 2/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#no shut

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

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state
to up
|
```

Copy Paste



Physical Config CLI

IOS Command Line Interface

```
Router>enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 20.0.0.1
% Incomplete command.
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#no shut

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

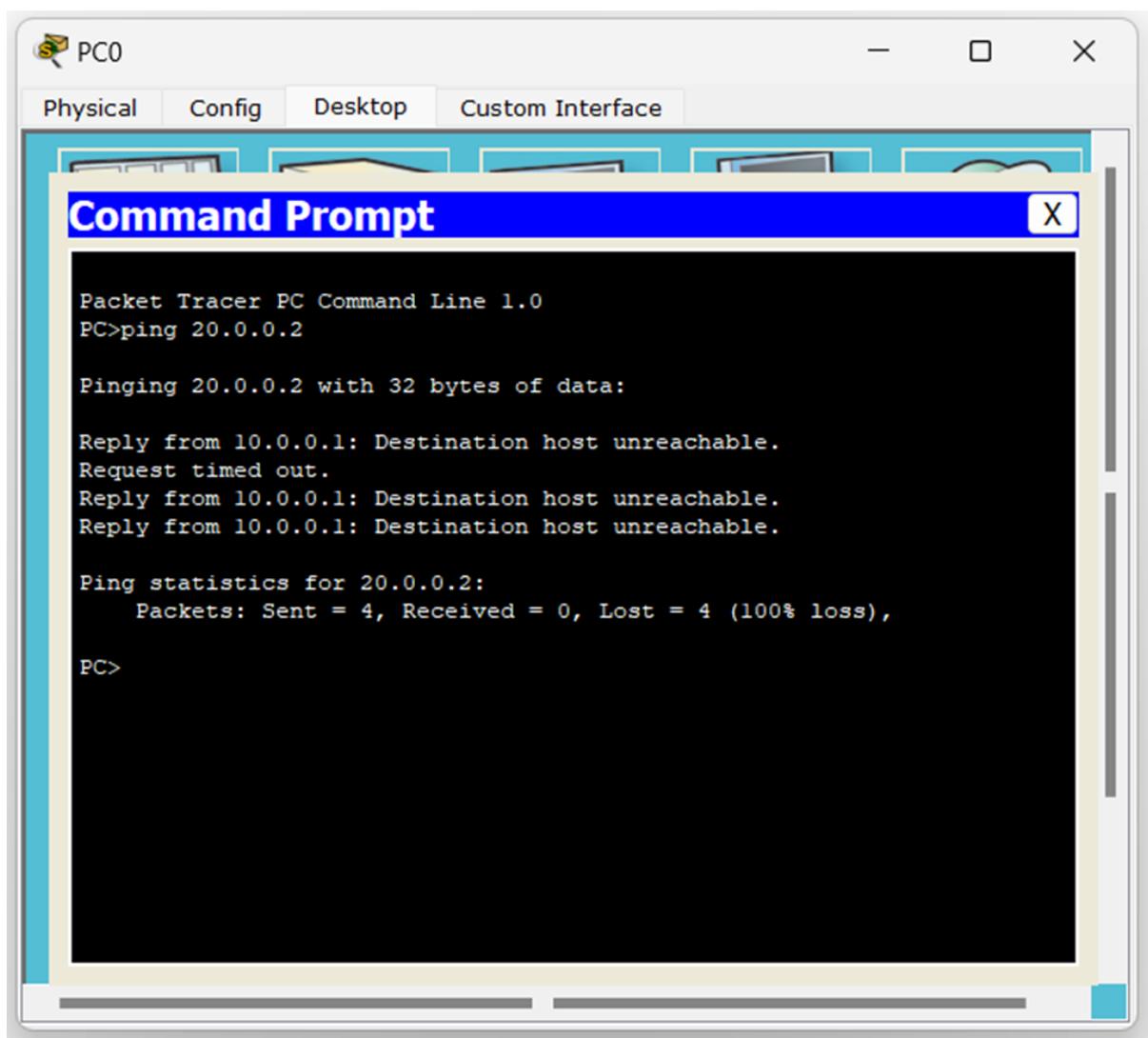
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed
state to up

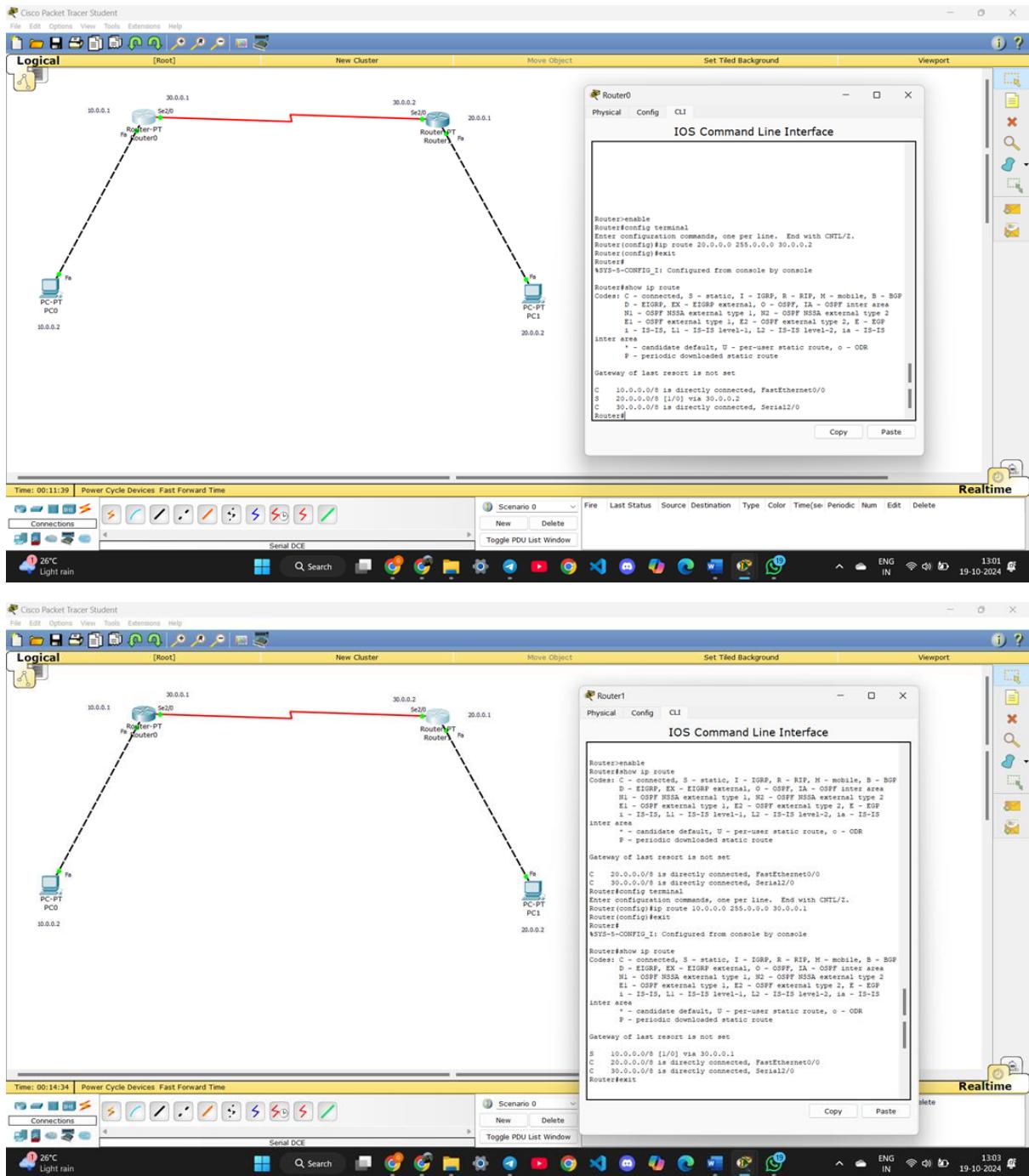
Router(config-if)#exit
Router(config)#interface serial2/0
Router(config-if)#ip address 30.0.0.2 255.0.0.0
Router(config-if)#no shut

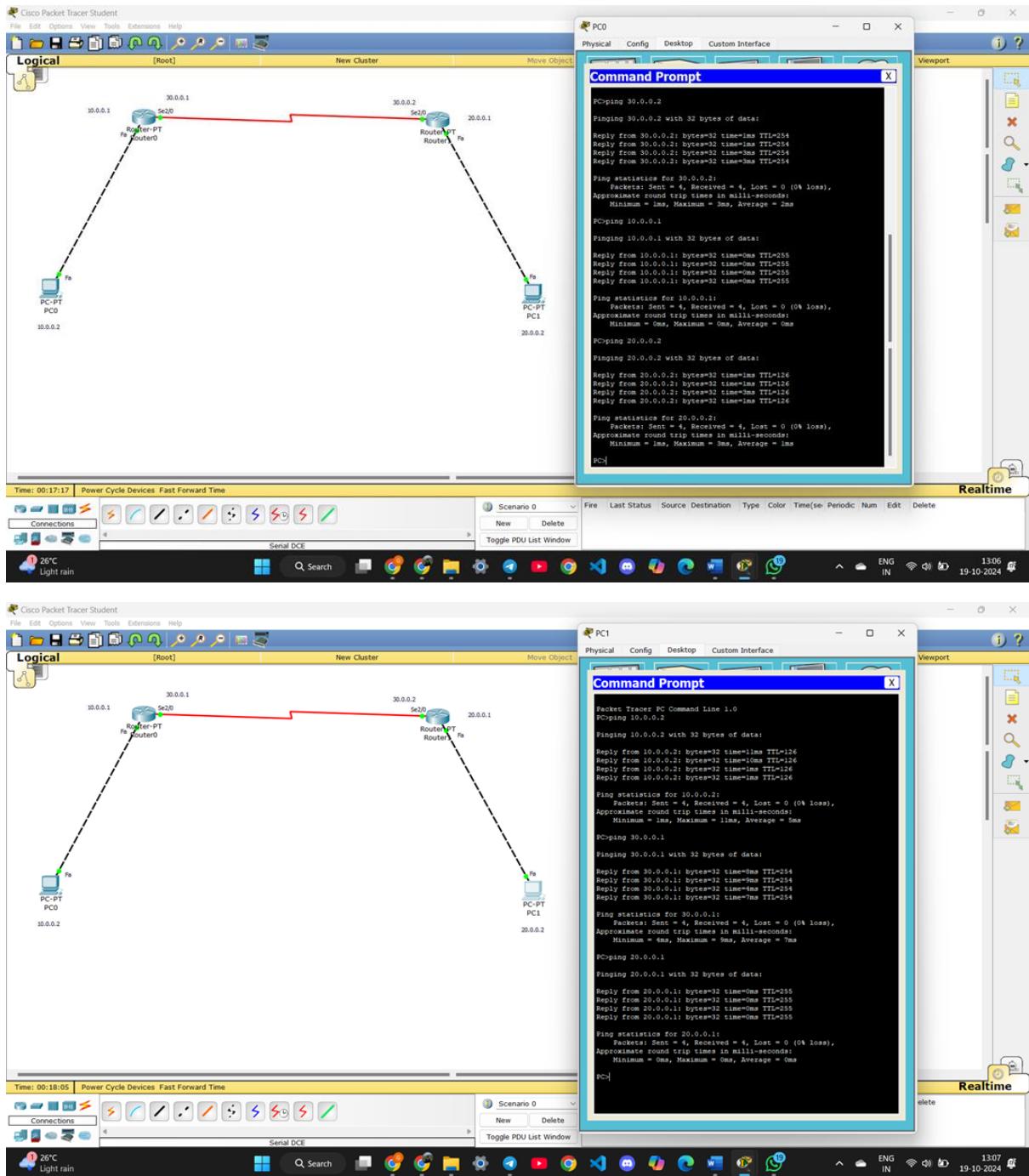
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

Router(config-if)#exit
Router(config)#

```







Expt. 3 – 23/10/2024

Date / /201

Expt. 3 :

Aim – Configure Default Route, Static Route to the Router.

Topology – 3 Routers with 2 PCs at opp. ends

R2 :
Fastethernet 0/0 – 40.0.0.1
Serial 2/0 – 30.0.0.2

R3 :
~~Fastethernet 2/0 – 20.0.0.2~~
Serial 3/0 – 30.0.0.1

PC0 is on the network 10.0.0.0 &
PC1 is on the network 40.0.0.0.

Procedure :

1. Configure 2 PCs properly
2. Configure 3 Routers according to their correct interfaces – Fastethernet & Serial.
3. Default route R1 :


```
# ip route 10.0.0.0 255.0.0.0 20.0.0.1
# ip route 40.0.0.0 255.0.0.0 30.0.0.2
```
4. Static route R0 :


```
# ip route 0.0.0.0 0.0.0.0 20.0.0.2
```
5. Static route R2 :


```
# ip route 0.0.0.0 0.0.0.0 30.0.0.1
```

JNANA SWEET

Date / /201

Observation:

R0 :

show ip route
C 10.0.0.0/8 is directly connected, FastEthernet 0/0
C 20.0.0.0/8 is directly connected, serial2/0
S* 0.0.0.0/0 [1/0] via 20.0.0.2

R2 :

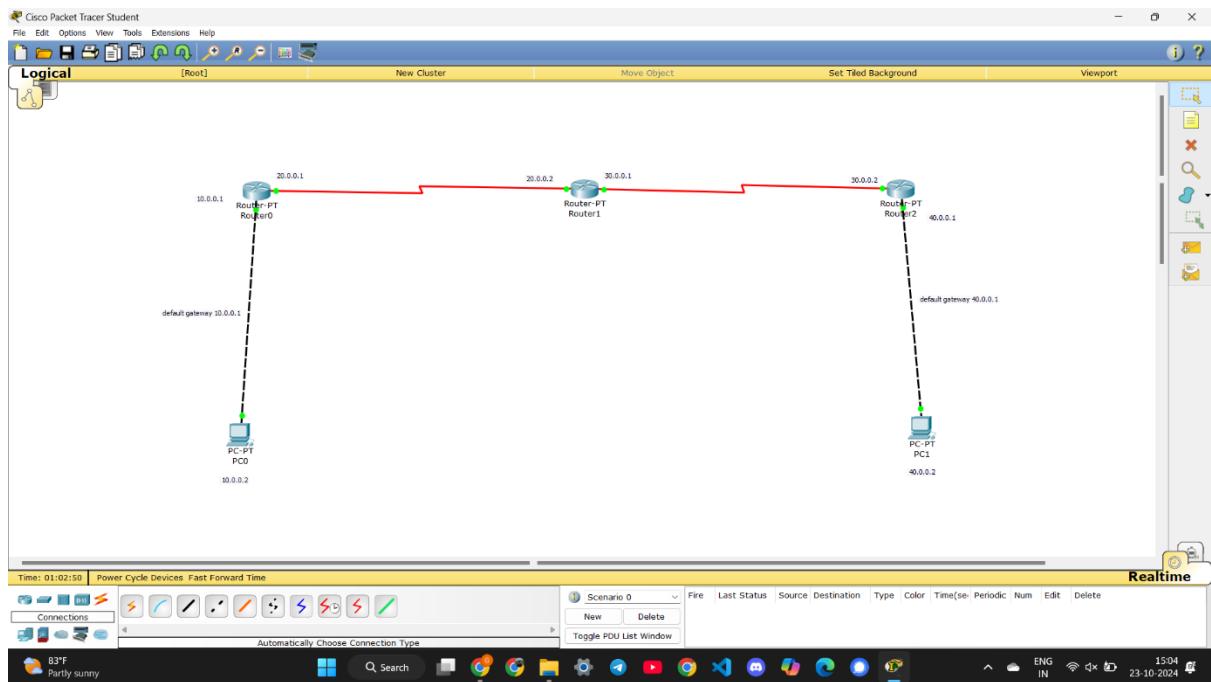
show ip route
C 30.0.0.0/8 is directly connected, ~~Serial~~ 2/0
C 40.0.0.0/8 is directly connected, FastEthernet 0/0
S* 0.0.0.0/0 [1/0] via 30.0.0.1

~~PC~~ PC0 :

PC > ping 40.0.0.2

Packets : Sent = 4 , Received = 4 , Lost = 0 (0% loss)

See
23/10/24



Router0

Physical Config CLI

IOS Command Line Interface

```
--- System Configuration Dialog ---  
Continue with configuration dialog? [yes/no]: n  
  
Press RETURN to get started!  
  
Router>enable  
Router#config terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)#interface fastethernet 0/0  
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  
exit  
Router(config)#exit  
Router#  
%SYS-5-CONFIG_I: Configured from console by console  
exit
```

Copy Paste

Router0

Physical Config CLI

IOS Command Line Interface

```
Press RETURN to get started.

Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#no shut

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

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state
to up
```

Copy Paste

 Router1

Physical Config CLI

IOS Command Line Interface

Press RETURN to get started.

```
Router>
Router>enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface serial 3/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#no shut
Router(config-if)#exit
Router(config)#interface serial 2/0
Router(config-if)#ip address 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)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state
to up

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

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

[Copy](#)

[Paste](#)

Router2

Physical Config CLI

IOS Command Line Interface

```
Copyright (c) 1986-2005 by cisco Systems, Inc.
Compiled Wed 27-Apr-04 19:01 by miwang

PT 1001 (PTSC2005) processor (revision 0x200) with 60416K/5120K bytes of memory
.
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)

--- System Configuration Dialog ---

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

Press RETURN to get started!

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

Router>
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 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)#interface
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

% Incomplete command.
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 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)#

```

Router1

Physical Config CLI

IOS Command Line Interface

Press RETURN to get started.

```
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 not set

C    20.0.0.0/8 is directly connected, Serial2/0
C    30.0.0.0/8 is directly connected, Serial3/0
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip route 10.0.0.0 255.0.0.0 20.0.0.1
Router(config)#ip route 40.0.0.0 255.0.0.0 30.0.0.2
Router(config)#exit
Router#
$SYS-5-CONFIG_I: Configured from console by console

Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

S    10.0.0.0/8 [1/0] via 20.0.0.1
C    20.0.0.0/8 is directly connected, Serial2/0
C    30.0.0.0/8 is directly connected, Serial3/0
S    40.0.0.0/8 [1/0] via 30.0.0.2
Router#
```

Copy Paste

Router0

Physical Config CLI

IOS Command Line Interface

```
% Invalid input detected at '^' marker.

Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 20.0.0.2
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

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 20.0.0.2 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 20.0.0.2
Router#
```

[Copy](#)

[Paste](#)



Router2

Physical Config CLI

IOS Command Line Interface

```
Router#enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 30.0.0.1
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
s
% Ambiguous command: "s"
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 30.0.0.1 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 30.0.0.1
Router#
```

 Router1

- □ ×

Physical Config CLI

IOS Command Line Interface

```
S 40.0.0.0/8 [1/0] via 30.0.0.2
Router#
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

S 10.0.0.0/8 [1/0] via 20.0.0.1
C 20.0.0.0/8 is directly connected, Serial2/0
C 30.0.0.0/8 is directly connected, Serial3/0
S 40.0.0.0/8 [1/0] via 30.0.0.2
Router#
```

PC0

Physical Config Desktop Custom Interface

Command Prompt

```
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.2

Pinging 40.0.0.2 with 32 bytes of data:

Request timed out.
Reply from 40.0.0.2: bytes=32 time=2ms TTL=125
Reply from 40.0.0.2: bytes=32 time=5ms TTL=125
Reply from 40.0.0.2: bytes=32 time=2ms TTL=125

Ping statistics for 40.0.0.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 5ms, Average = 3ms

PC>ping 30.0.0.2

Pinging 30.0.0.2 with 32 bytes of data:

Reply from 30.0.0.2: bytes=32 time=11ms TTL=253
Reply from 30.0.0.2: bytes=32 time=9ms TTL=253
Reply from 30.0.0.2: bytes=32 time=8ms TTL=253
Reply from 30.0.0.2: bytes=32 time=9ms TTL=253

Ping statistics for 30.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 11ms, Average = 9ms

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=9ms TTL=254
Reply from 20.0.0.2: bytes=32 time=1ms TTL=254
Reply from 20.0.0.2: bytes=32 time=9ms TTL=254
Reply from 20.0.0.2: bytes=32 time=1ms TTL=254

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 9ms, Average = 5ms

PC>|
```

PC1

Physical Config Desktop Custom Interface

Command Prompt X

```
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=4ms TTL=125
Reply from 10.0.0.2: bytes=32 time=3ms TTL=125
Reply from 10.0.0.2: bytes=32 time=2ms TTL=125
Reply from 10.0.0.2: bytes=32 time=2ms TTL=125

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 4ms, Average = 2ms

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=3ms TTL=253
Reply from 20.0.0.1: bytes=32 time=2ms TTL=253
Reply from 20.0.0.1: bytes=32 time=9ms TTL=253
Reply from 20.0.0.1: bytes=32 time=8ms TTL=253

Ping statistics for 20.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 9ms, Average = 5ms

PC>ping 30.0.0.1

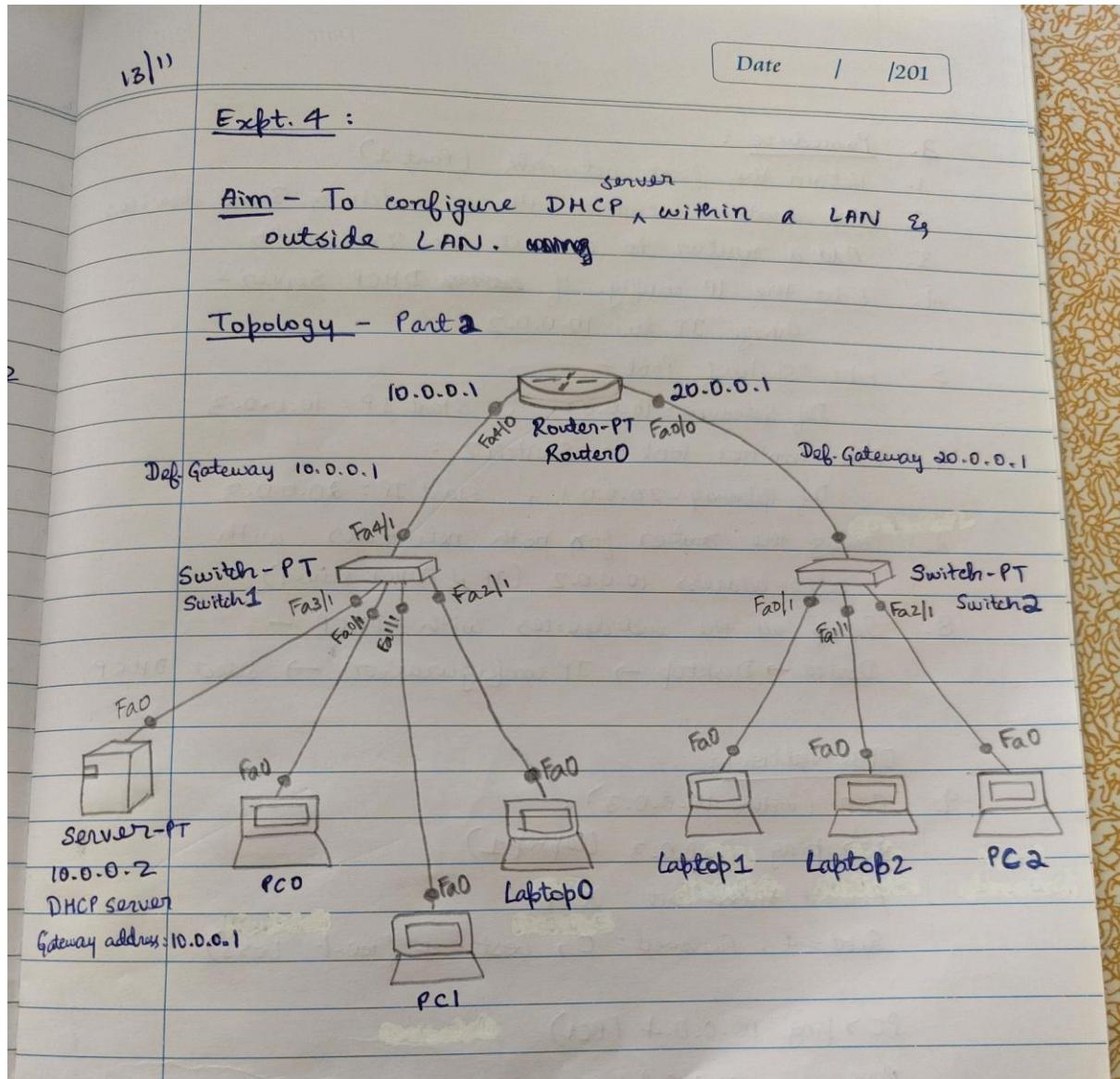
Pinging 30.0.0.1 with 32 bytes of data:

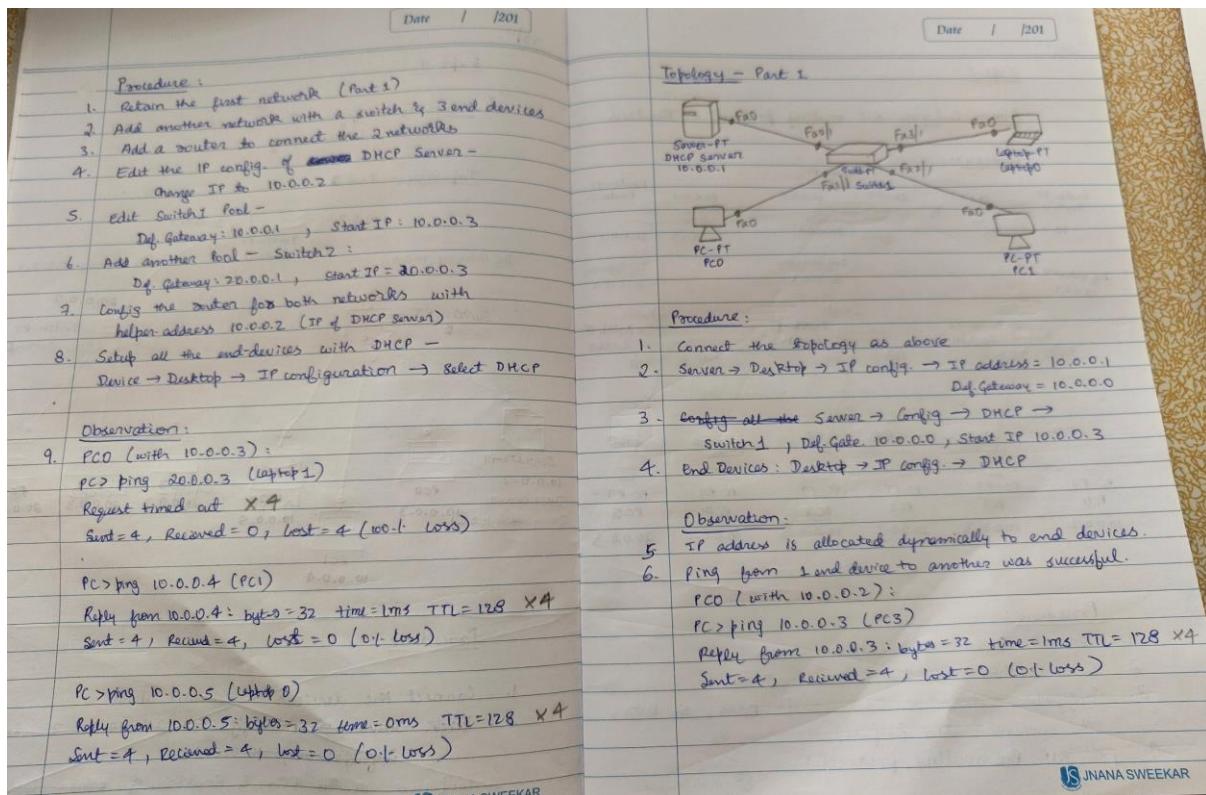
Reply from 30.0.0.1: bytes=32 time=10ms TTL=254
Reply from 30.0.0.1: bytes=32 time=9ms TTL=254
Reply from 30.0.0.1: bytes=32 time=1ms TTL=254
Reply from 30.0.0.1: bytes=32 time=16ms TTL=254

Ping statistics for 30.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 16ms, Average = 9ms

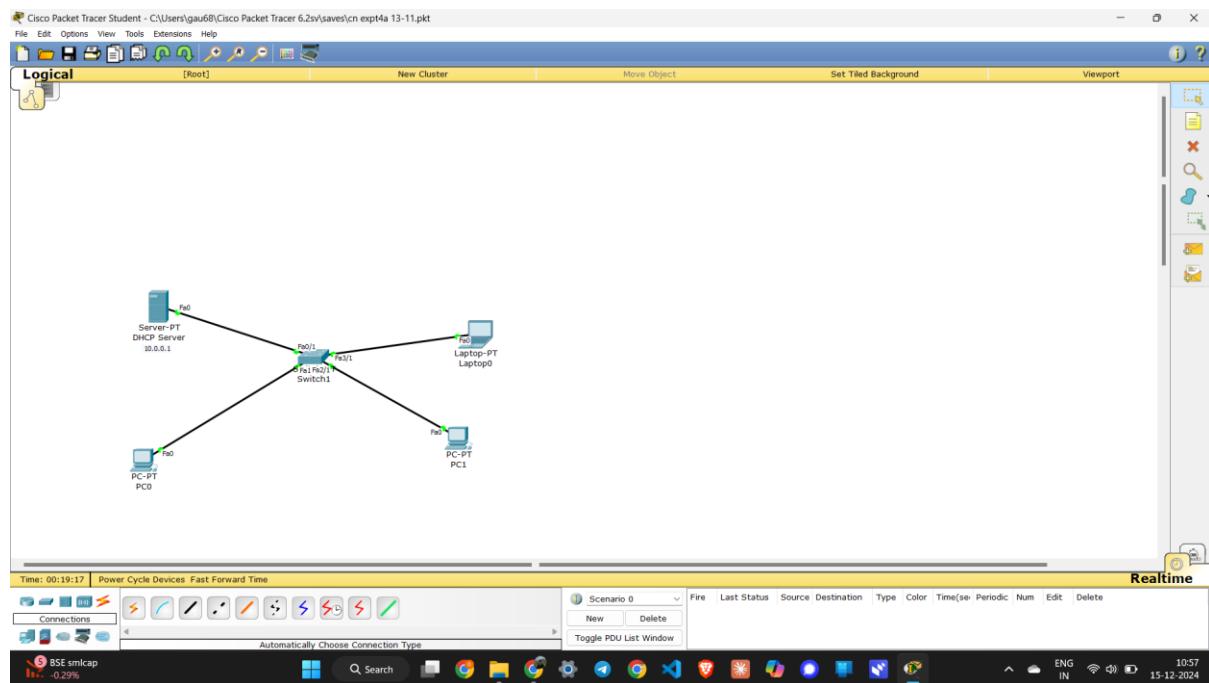
PC>
```

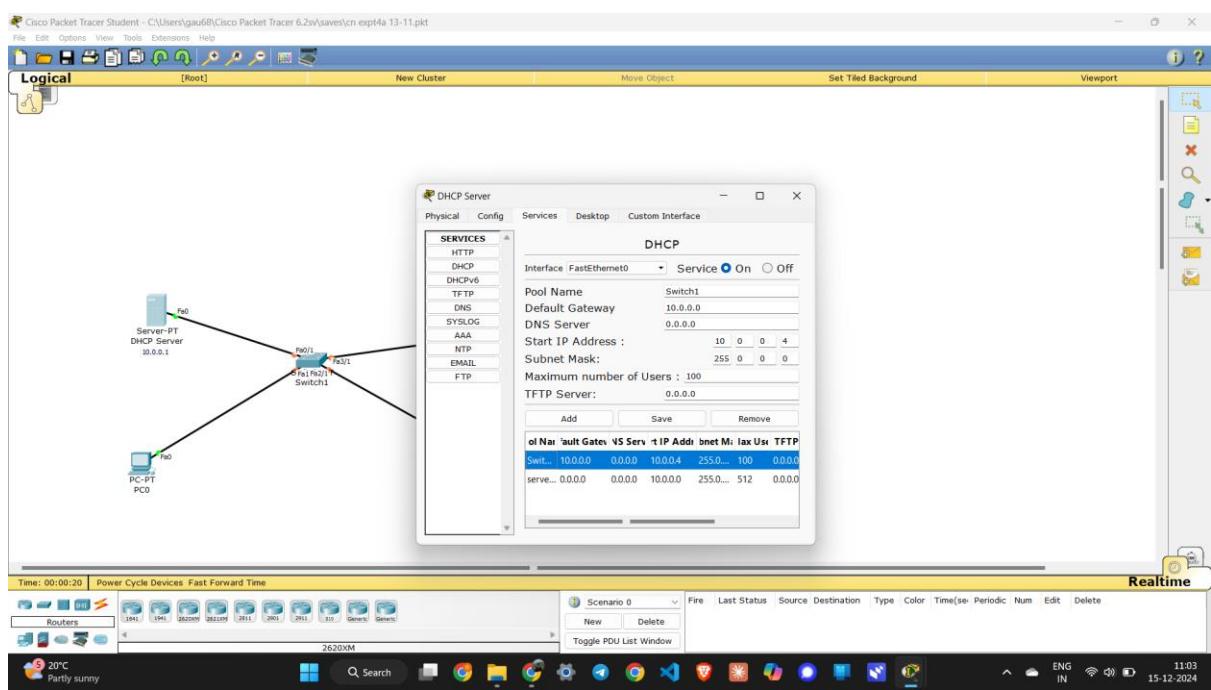
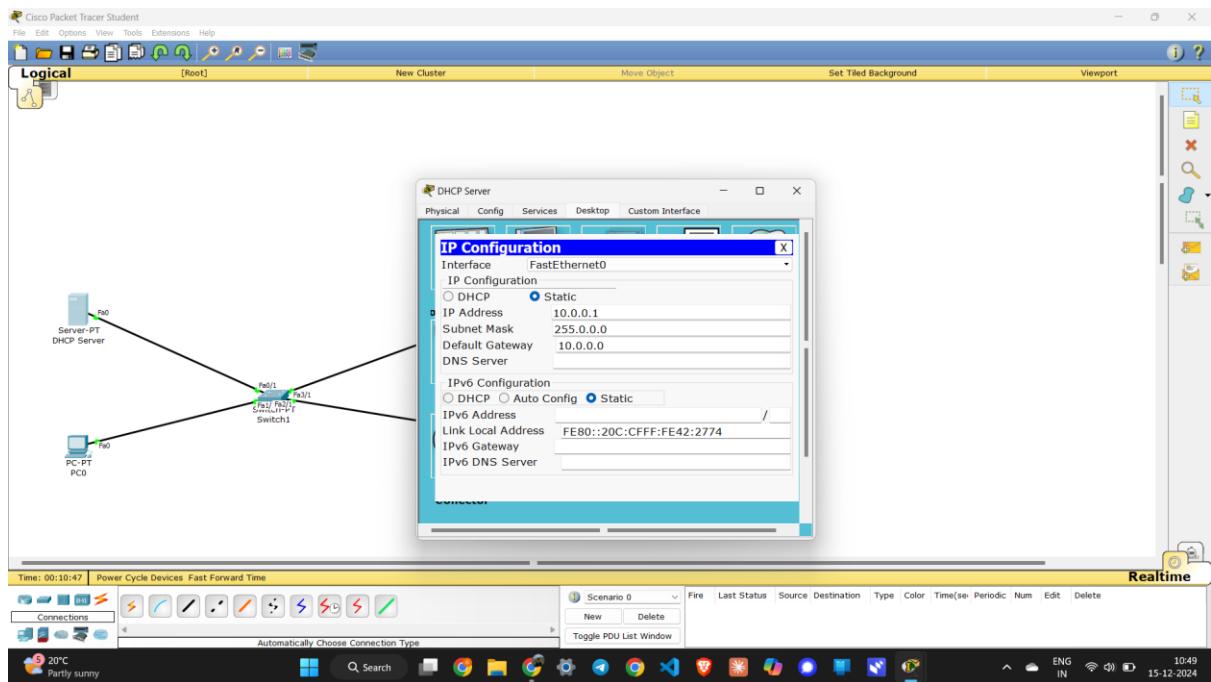
Expt. 4 – 13/11/2024

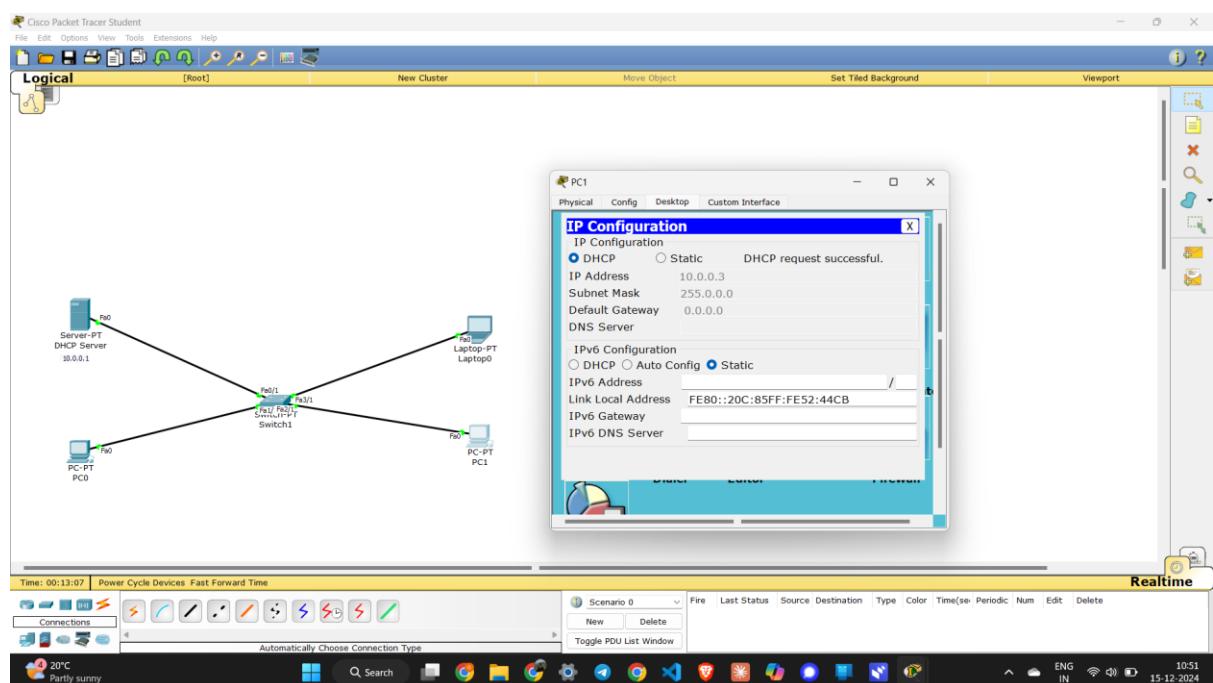
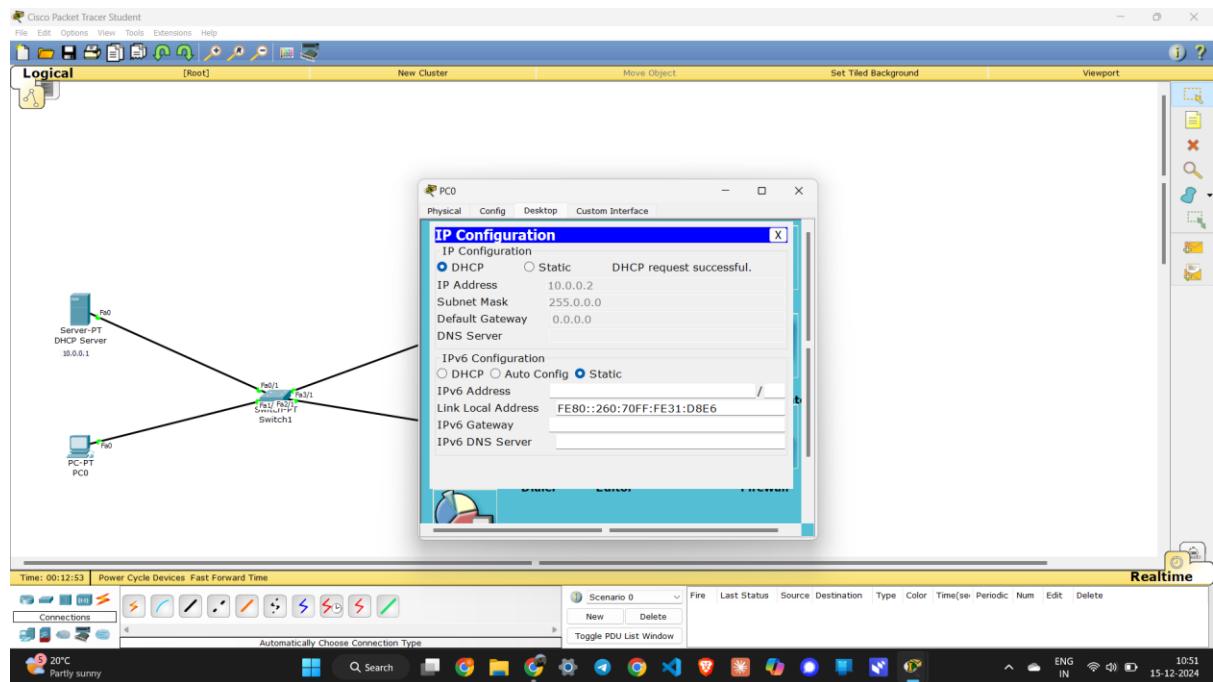


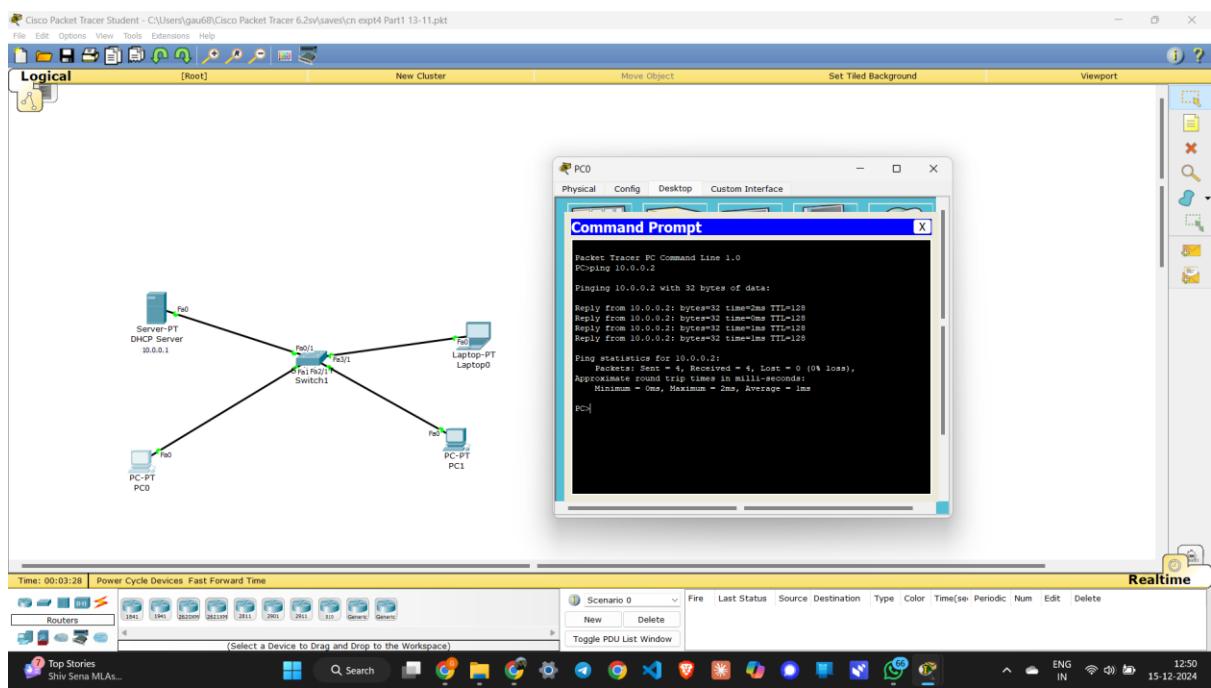
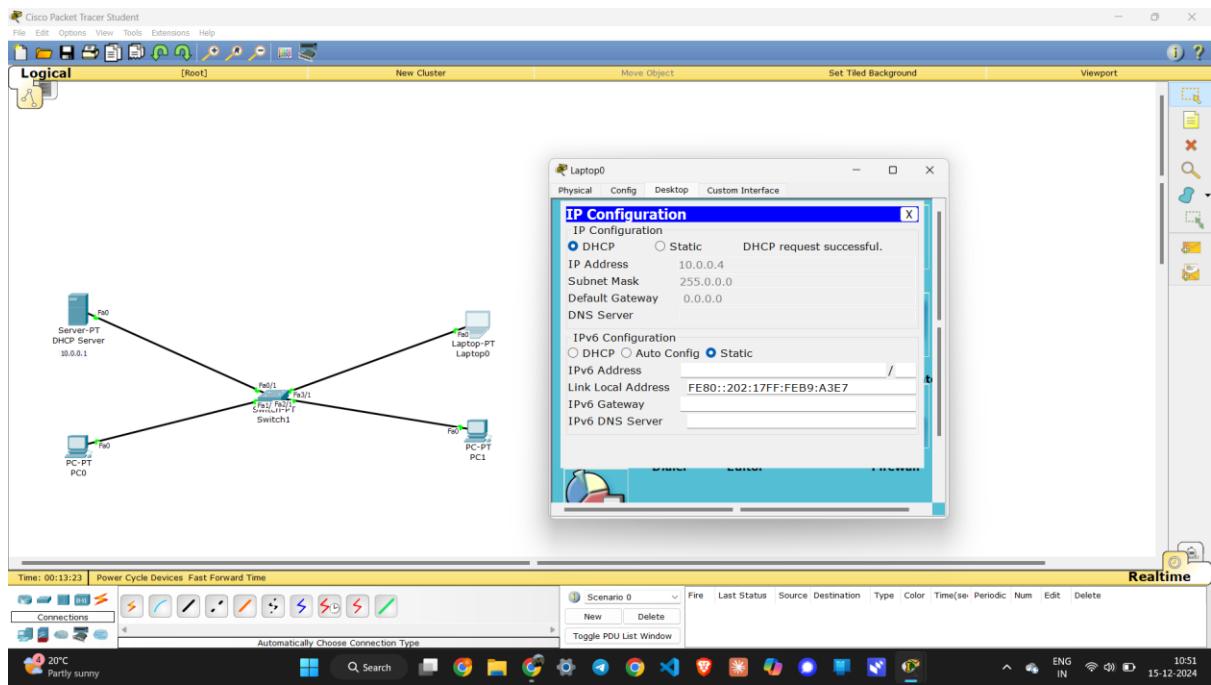


Part1:

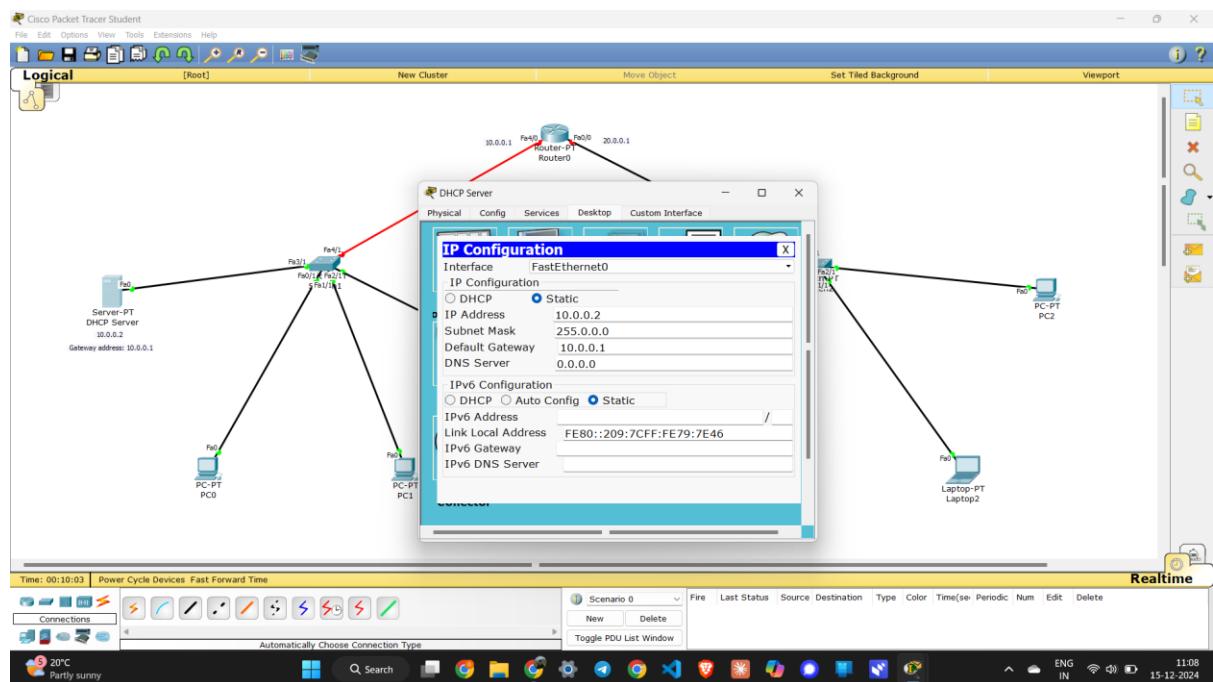
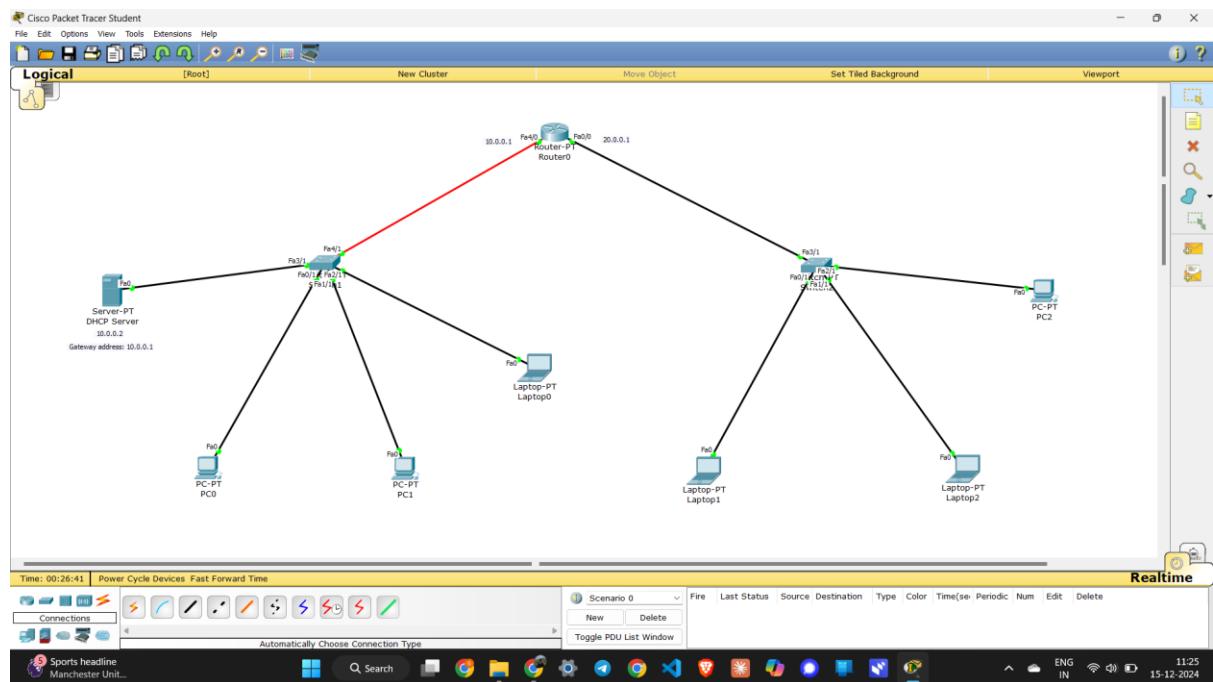


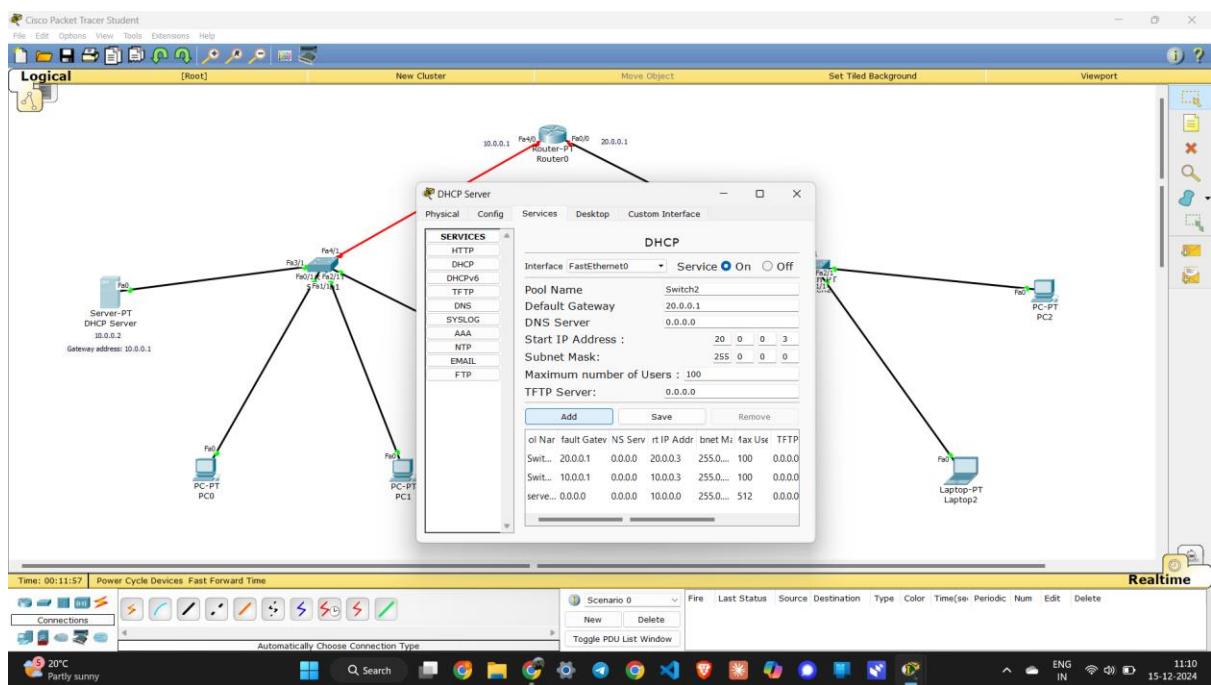
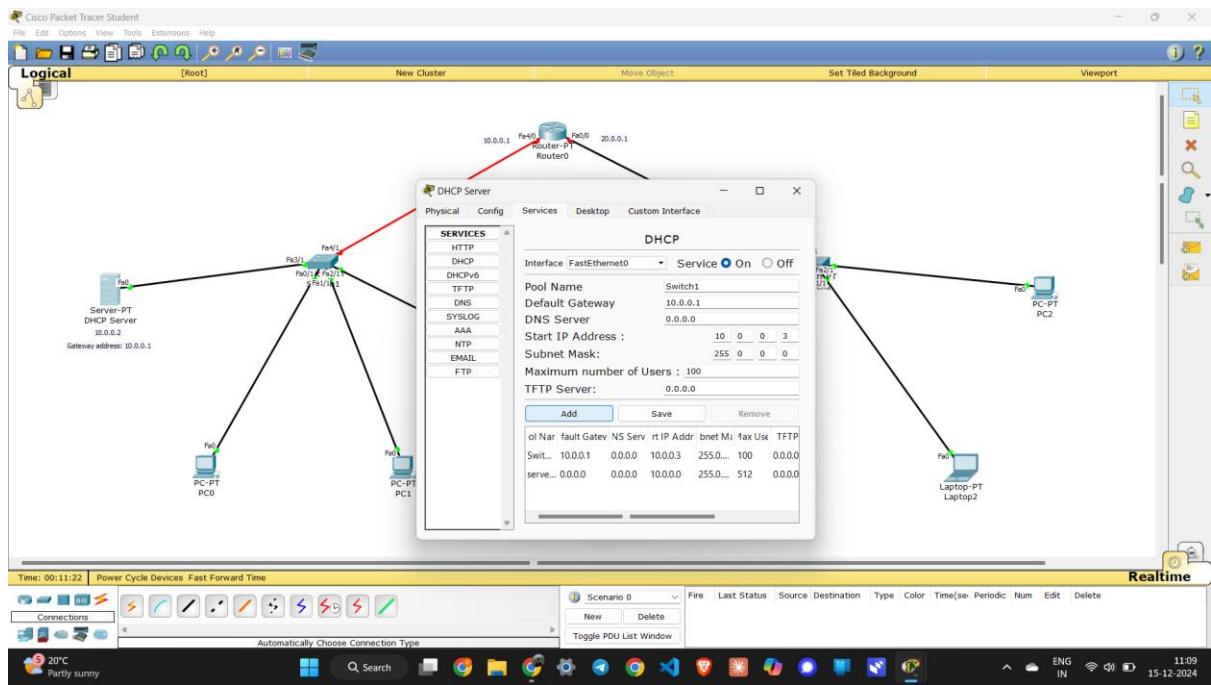


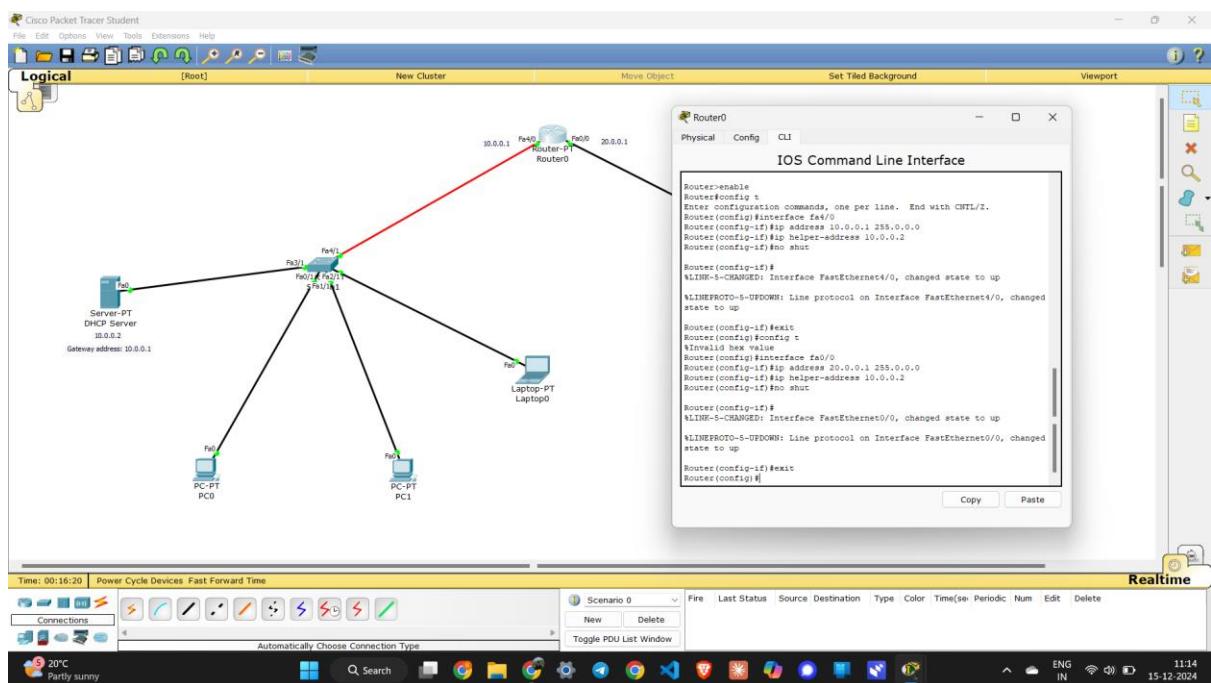
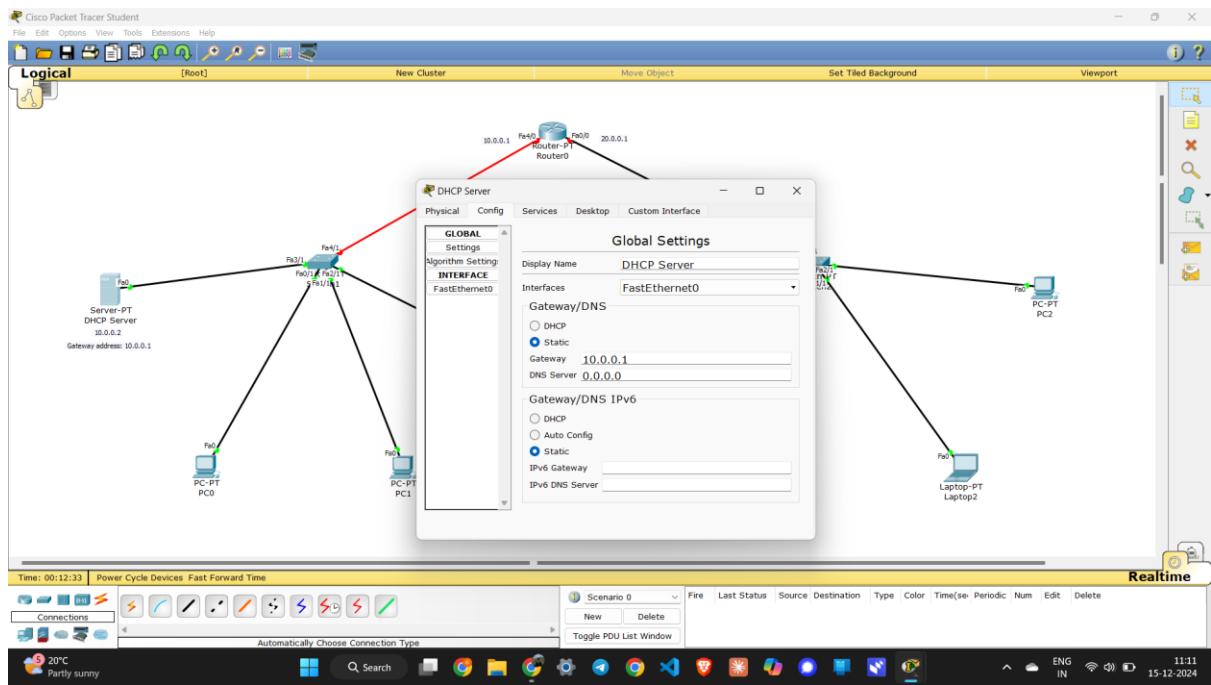




Part2:







Router0

Physical Config CLI

IOS Command Line Interface

```
Router>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fa4/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#ip helper-address 10.0.0.2
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet4/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet4/0, changed
state to up

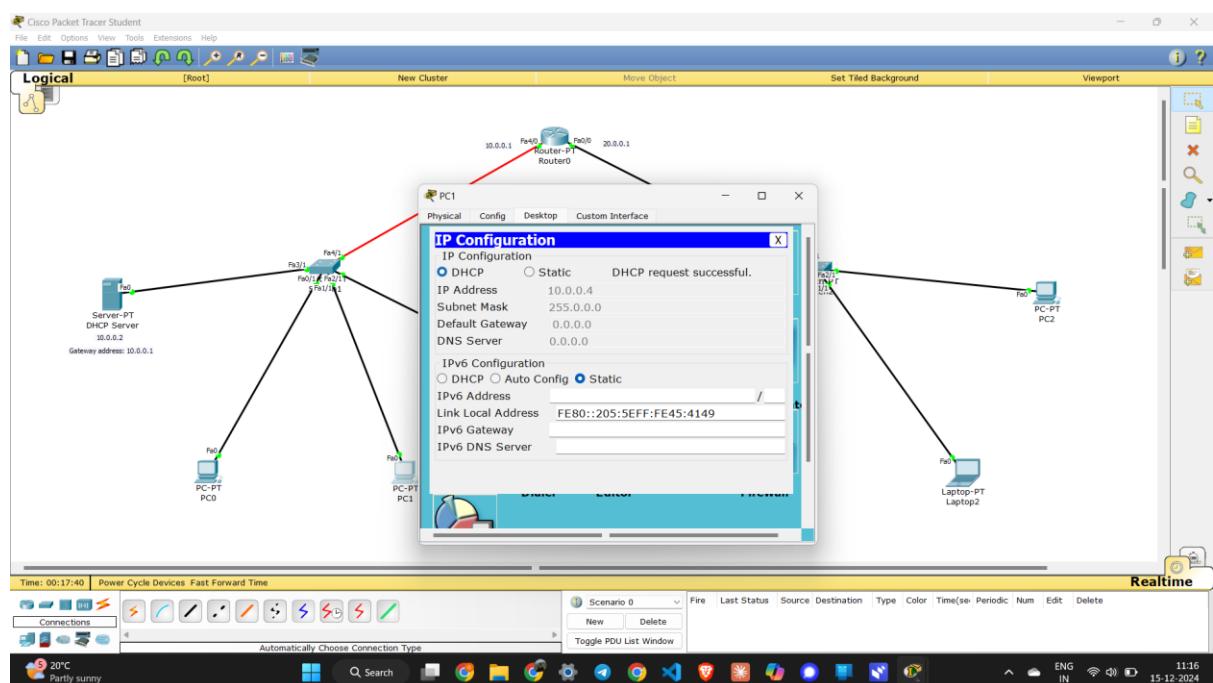
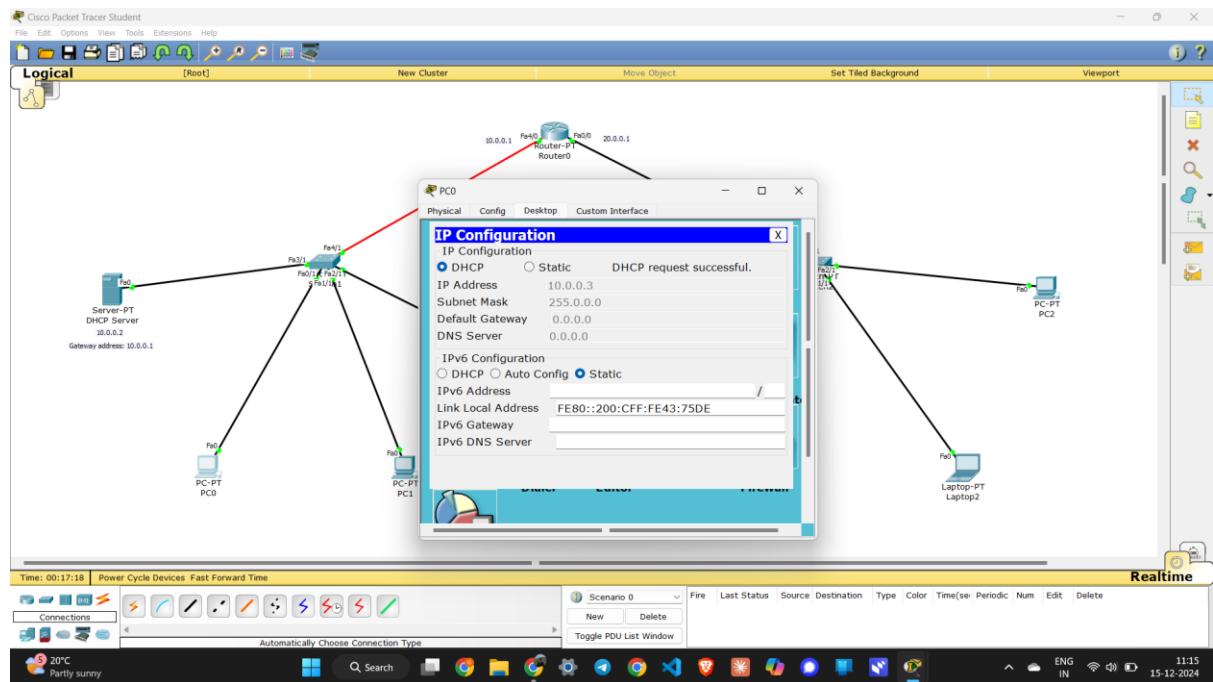
Router(config-if)#exit
Router(config)#config t
%Invalid hex value
Router(config)#interface fa0/0
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#ip helper-address 10.0.0.2
Router(config-if)#no shut

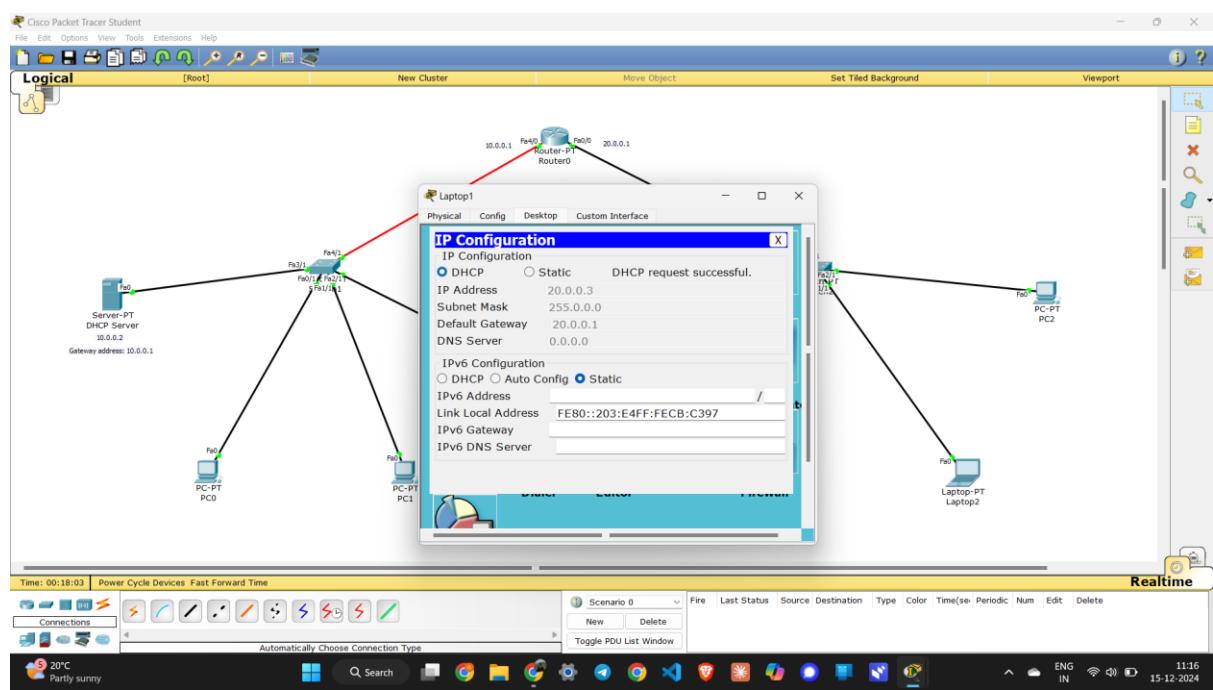
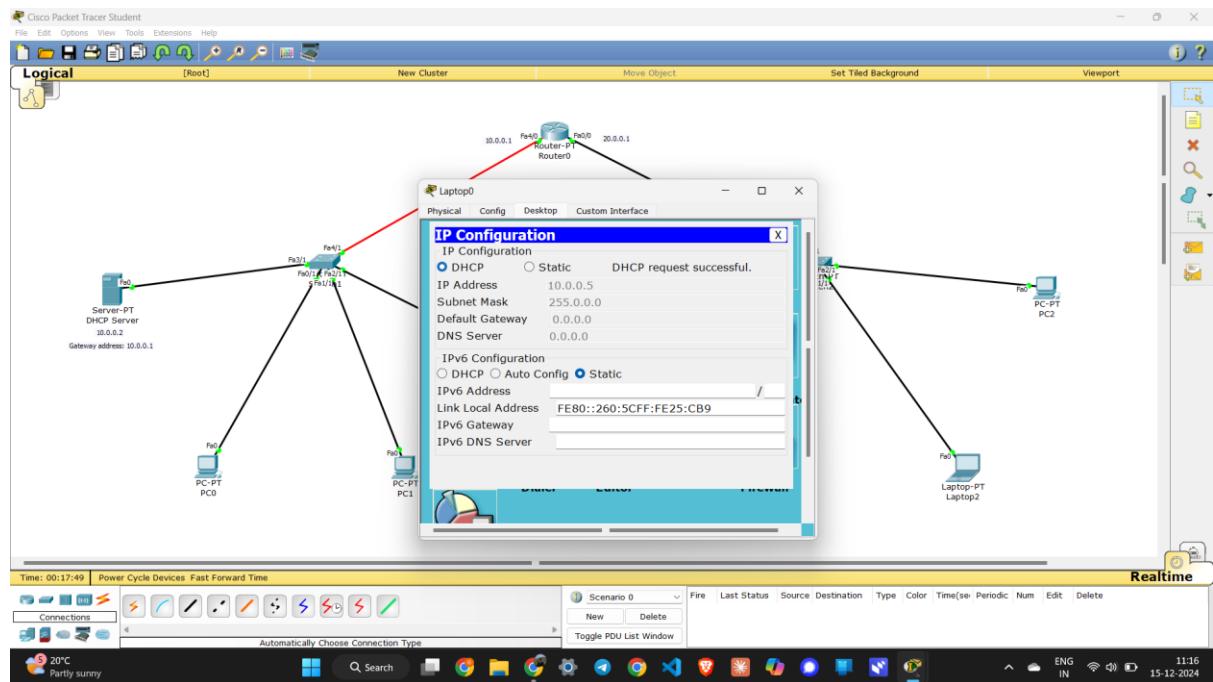
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

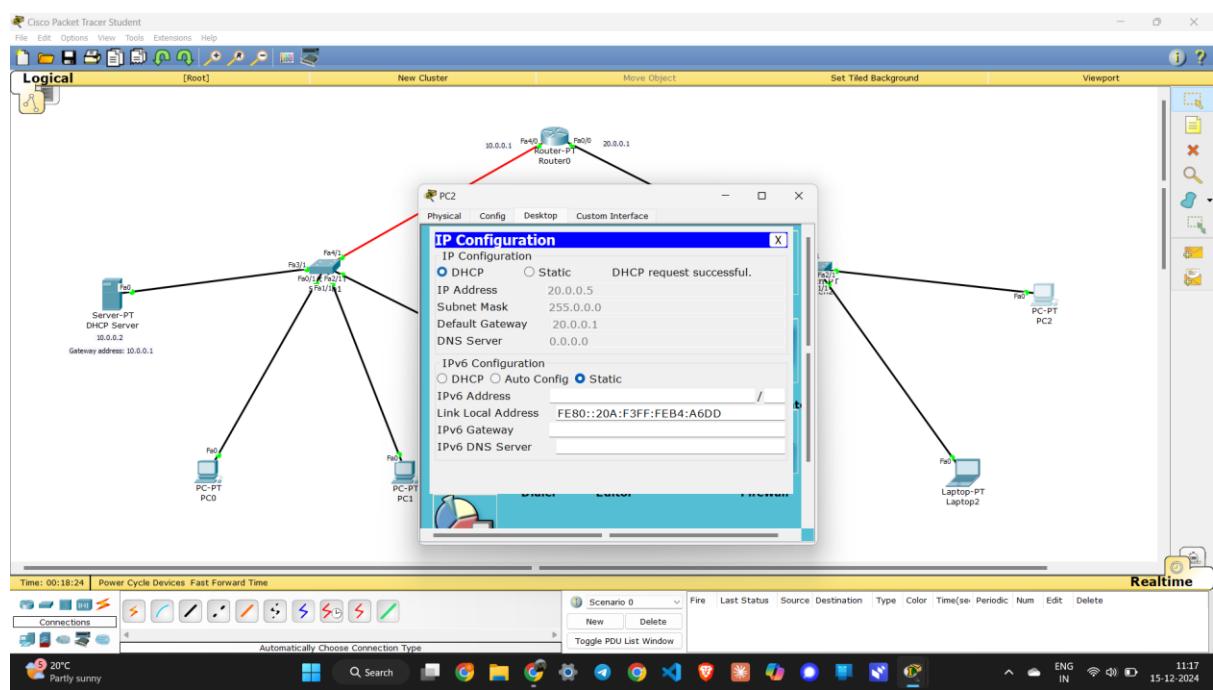
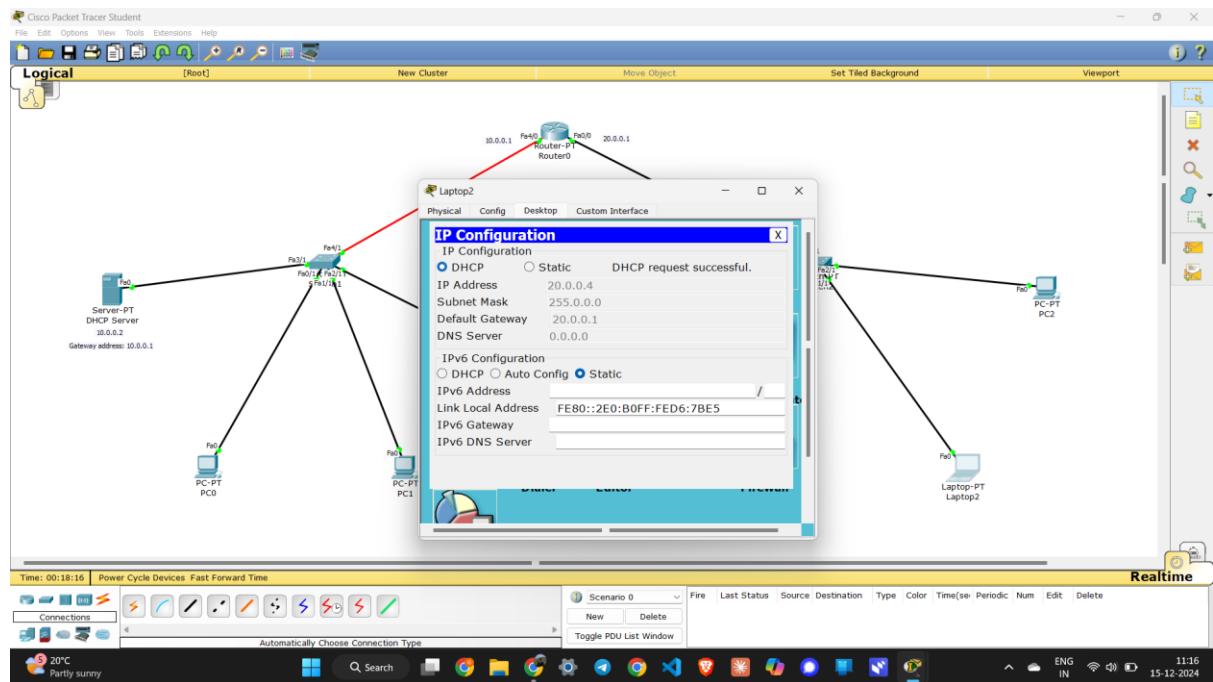
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed
state to up

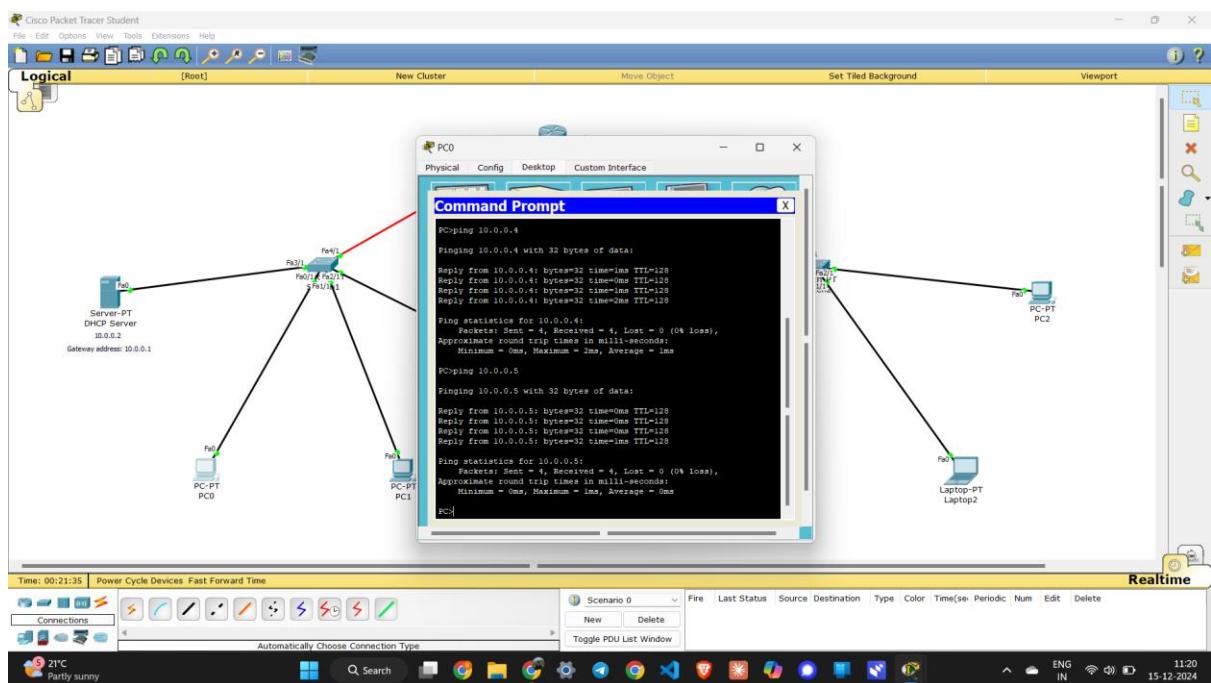
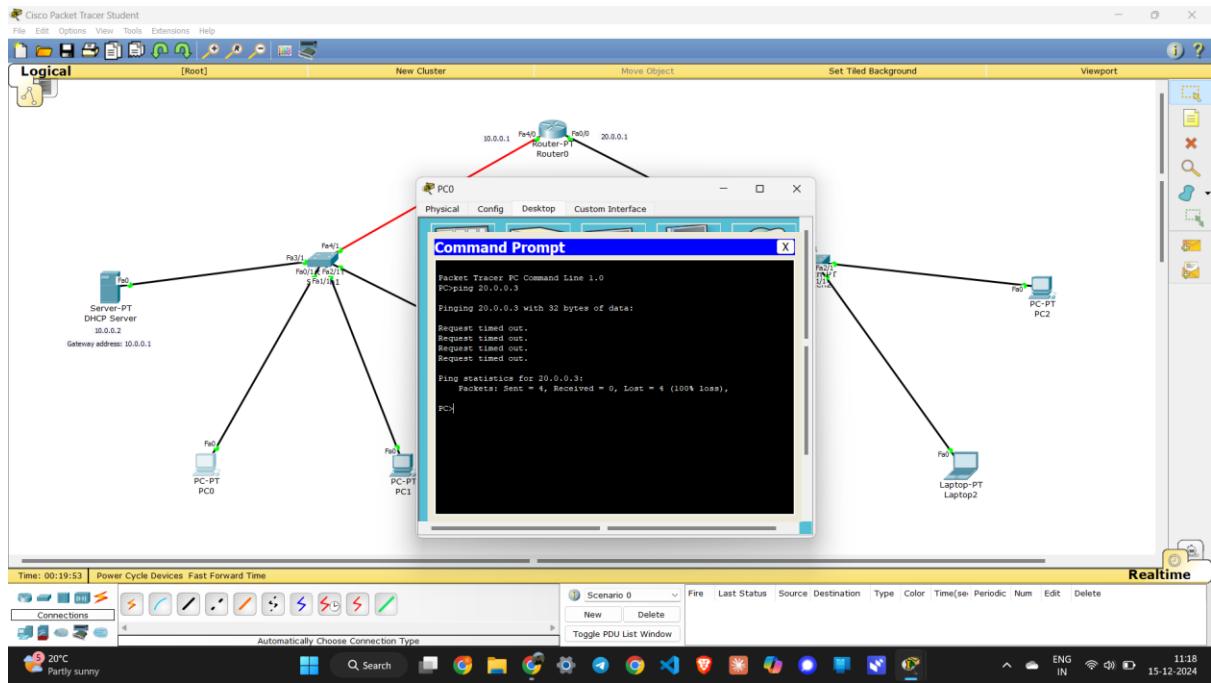
Router(config-if)#exit
Router(config)#

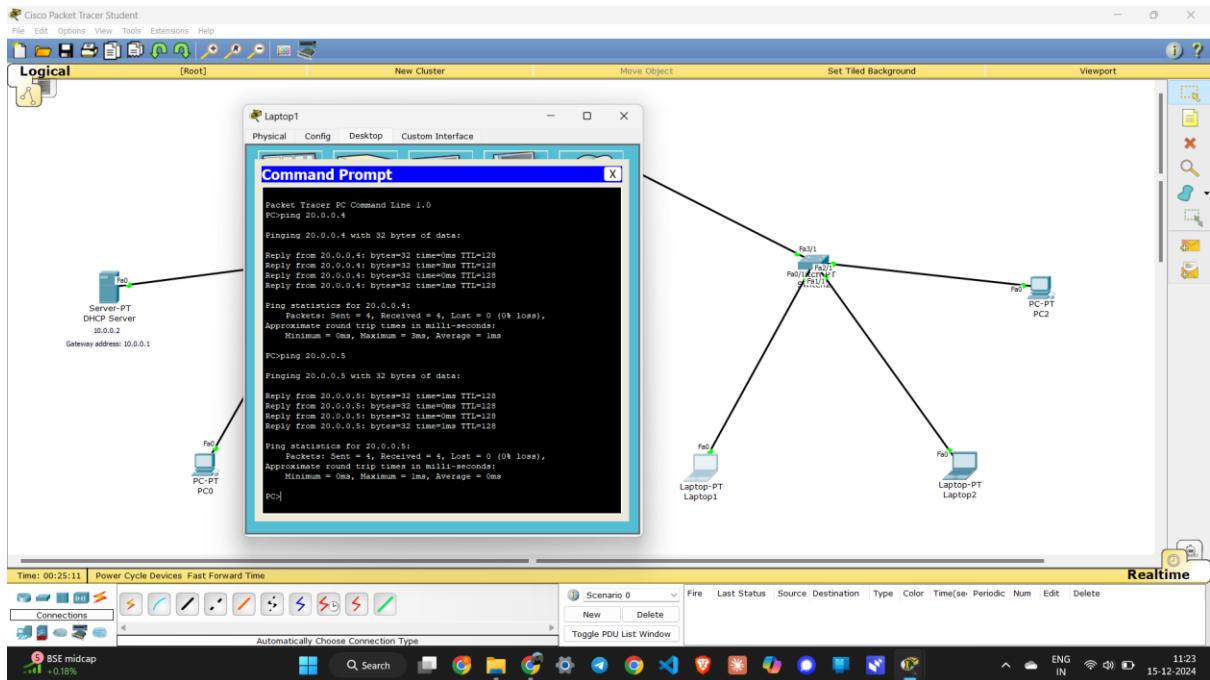
```











Expt. 5 – 20/11/2024

Date / /201

Expt. 5:

Aim – Configure RIP routing protocol in Routers

Topology –

In Router0 CLI:
Router > enable
Router# config terminal
Router (config) # router rip
network 10.0.0.0
network 40.0.0.0

In Router1 CLI:
Router (config) # router rip
network 20.0.0.0
network 40.0.0.0
network 50.0.0.0

In Router2 CLI:
Router (config) # router rip
network 30.0.0.0
network 50.0.0.0

Test Connectivity:
PC0 > ping 30.0.0.1
Sent = 4 Loss = 0 100%

Observation:
The routers communicate with each other & share a common routing table. Once RIP is installed/activated in routers, every router shares its routing protocol with its neighbours. Hence in iterations, every router will know about all info that

JNANA SWEEKAR

their neighbours are connected to.

Demonstrating TTL:

1. Select Simulation → Simple PDU
2. Select Source & Destination PCs
3. Press Auto Capture / Play.
4. ~~stop~~ Press the same button repeatedly such that the packet stops at every router & check the Inbound PDU & Outbound PDU for each router. We can notice that initial PDU is ~~is~~ 255, i.e., for Router 0.

Router 0 :

Inbound TTL = 255 Outbound TTL = 254

Router 1 :

Inbound TTL = 254 Outbound TTL = 253

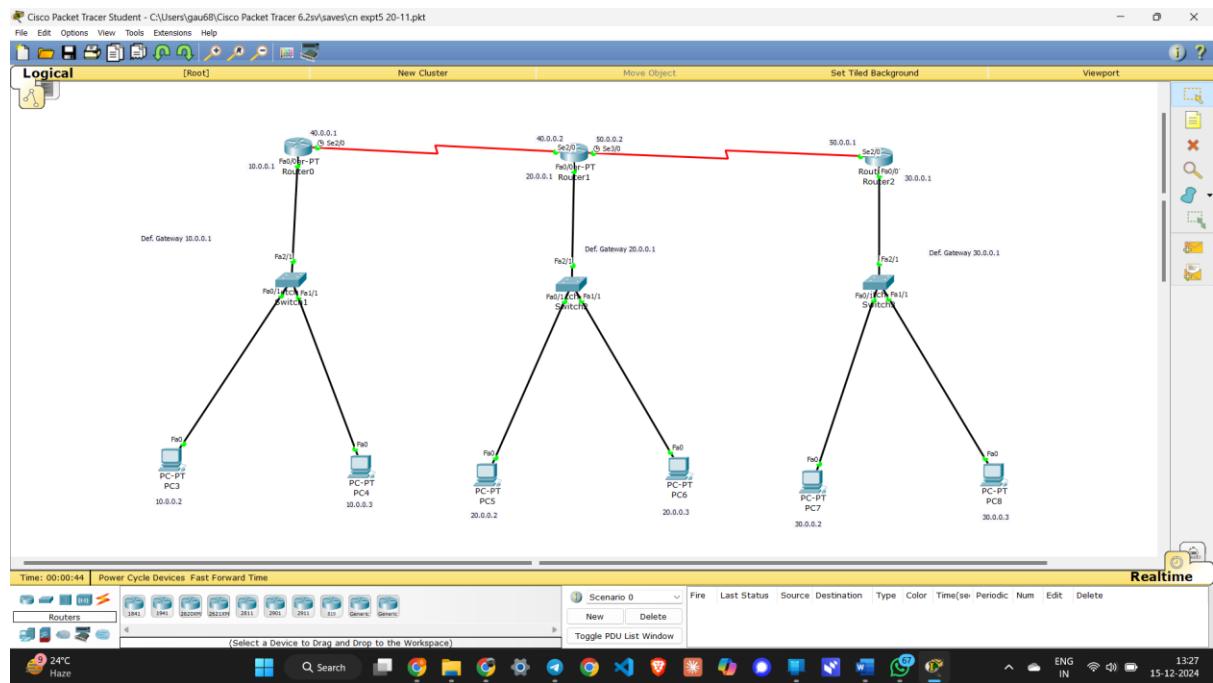
Router 2 :

Inbound TTL = 253 Outbound TTL = 252

Observation:

1. TTL of a packet decreases by 1 at each router hop to prevent infinite loops.
If the TTL becomes 0 (i.e., reaches 0), the router discards the packets & sends an ICMP (Internet Control Message Protocol) "Time Exceeded" msg. back to the sender.
2. TTL value decreases after moving from router to router. But stays constant for PC to Switch & switch to router.

Topology:



Configure Network:

The image displays two separate windows, each titled "IOS Command Line Interface".

Router0 Window:

```
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial2/0
Router(config-if)#ip address 40.0.0.1 255.0.0.0
Router(config-if)#no shut

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

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

Router1 Window:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 2/0
Router(config-if)#ip address 40.0.0.2 255.0.0.0
Router(config-if)#no shut

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

Router(config-if)#exit
Router(config)#ip address 40.0.0.2 255.0.0.0
%LINEPROTO-5-UPDOWN: Line protocol on Interface Seinterface serial 2/0
Router(config-if)#exit
Router(config)#interface serial 3/0
Router(config-if)#ip address 50.0.0.1 255.0.0.0
Router(config-if)#no shut

%LINK-5-CHANGED: Interface Serial3/0, changed state to down
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#interface fastethernet 0/0
```

Router1

Physical Config CLI

IOS Command Line Interface

```

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

Router#interface fastethernet 0/0
^
% Invalid input detected at '^' marker.

Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 20.0.0.1 255.0.0.0
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)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#

```

Copy Paste

Router2

Physical Config CLI

IOS Command Line Interface

```

Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 3/0
Router(config-if)#ip address 50.0.0.2 255.0.0.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up
exit
Router(config)#inte
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state
to up
% Incomplete command.
Router(config)#interface fastethernet 0/0
^
% Invalid input detected at '^' marker.

Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 30.0.0.1 255.0.0.0
Router(config-if)#no shut

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

```

Copy Paste

Configure Routing:

Router 0:

```
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 40.0.0.0
Router(config-router)#network 10.0.0.0
Router(config-router)#End
```

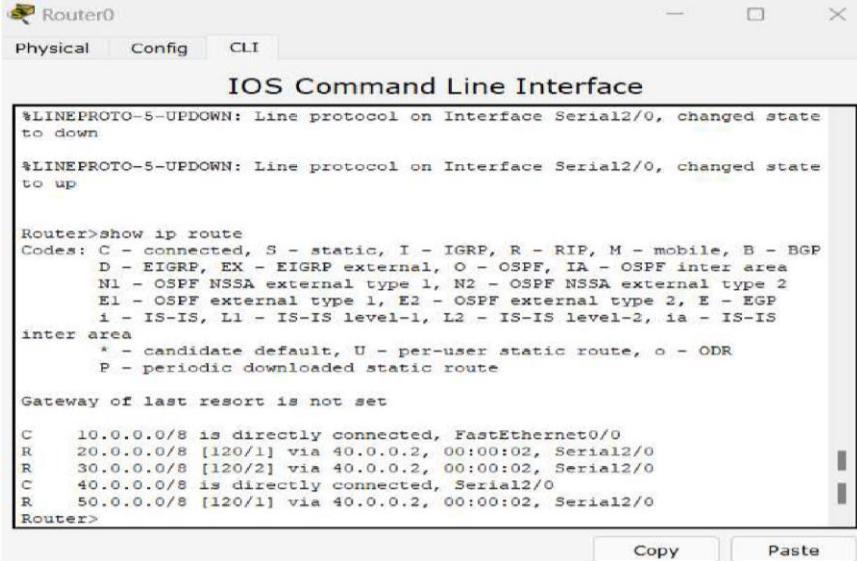
Router 1:

```
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 40.0.0.0
Router(config-router)#network 20.0.0.0
Router(config-router)#network 50.0.0.0
Router(config-router)#network 10.0.0.0
Router(config-router)#exit
Router(config)#End
```

Router 2:

```
Router>enable
Router#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 50.0.0.0
Router(config-router)#network 30.0.0.0
Router(config-router)#End
```

Routing:



Router0

Physical Config CLI

IOS Command Line Interface

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
      inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/0
R    20.0.0.0/8 [120/1] via 40.0.0.2, 00:00:02, Serial2/0
R    30.0.0.0/8 [120/2] via 40.0.0.2, 00:00:02, Serial2/0
C    40.0.0.0/8 is directly connected, Serial2/0
R    50.0.0.0/8 [120/1] via 40.0.0.2, 00:00:02, Serial2/0
Router>
```

Copy Paste



Router1

Physical Config CLI

IOS Command Line Interface

```
to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

Router>
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
      inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

R    10.0.0.0/8 [120/1] via 40.0.0.1, 00:00:12, Serial2/0
C    20.0.0.0/8 is directly connected, FastEthernet0/0
R    30.0.0.0/8 [120/1] via 50.0.0.2, 00:00:16, Serial3/0
C    40.0.0.0/8 is directly connected, Serial2/0
C    50.0.0.0/8 is directly connected, Serial3/0
Router>
```

Copy Paste

Router2

Physical Config CLI

IOS Command Line Interface

```
Router>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS
      inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

R    10.0.0.0/8 [120/2] via 50.0.0.1, 00:00:25, Serial3/0
R    20.0.0.0/8 [120/1] via 50.0.0.1, 00:00:25, Serial3/0
C    30.0.0.0/8 is directly connected, FastEthernet0/0
R    40.0.0.0/8 [120/1] via 50.0.0.1, 00:00:25, Serial3/0
C    50.0.0.0/8 is directly connected, Serial3/0
Router>
```

Test Connectivity:

From PC0, ping PC4 –

```
Command Prompt

Pinging 20.0.0.3 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.3: bytes=32 time=12ms TTL=126
Reply from 20.0.0.3: bytes=32 time=6ms TTL=126
Reply from 20.0.0.3: bytes=32 time=3ms TTL=126

Ping statistics for 20.0.0.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 12ms, Average = 7ms

PC>ping 30.0.0.2

Pinging 30.0.0.2 with 32 bytes of data:
Request timed out.
Reply from 30.0.0.2: bytes=32 time=2ms TTL=125
Reply from 30.0.0.2: bytes=32 time=2ms TTL=125
Reply from 30.0.0.2: bytes=32 time=11ms TTL=125

Ping statistics for 30.0.0.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 11ms, Average = 5ms

PC>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:
Request timed out.
Reply from 20.0.0.2: bytes=32 time=9ms TTL=126
Reply from 20.0.0.2: bytes=32 time=6ms TTL=126
Reply from 20.0.0.2: bytes=32 time=7ms TTL=126

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 9ms, Average = 7ms

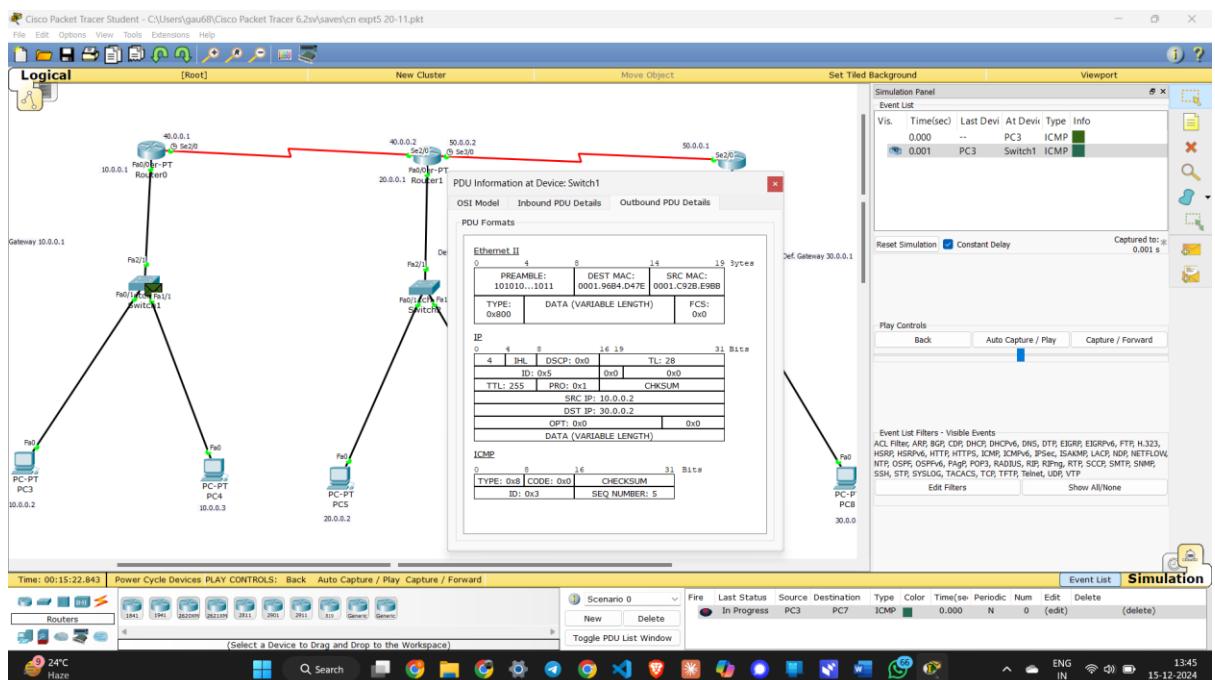
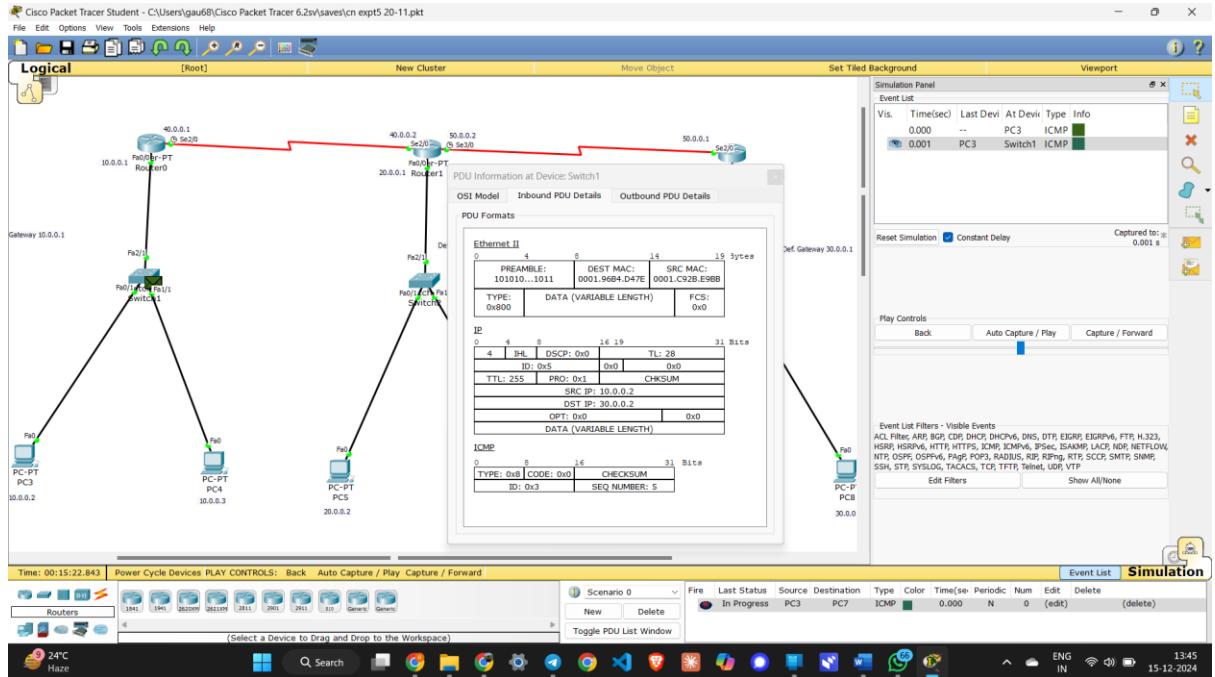
PC>ping 30.0.0.3

Pinging 30.0.0.3 with 32 bytes of data:
Request timed out.
Reply from 30.0.0.3: bytes=32 time=15ms TTL=125
Reply from 30.0.0.3: bytes=32 time=2ms TTL=125
Reply from 30.0.0.3: bytes=32 time=2ms TTL=125

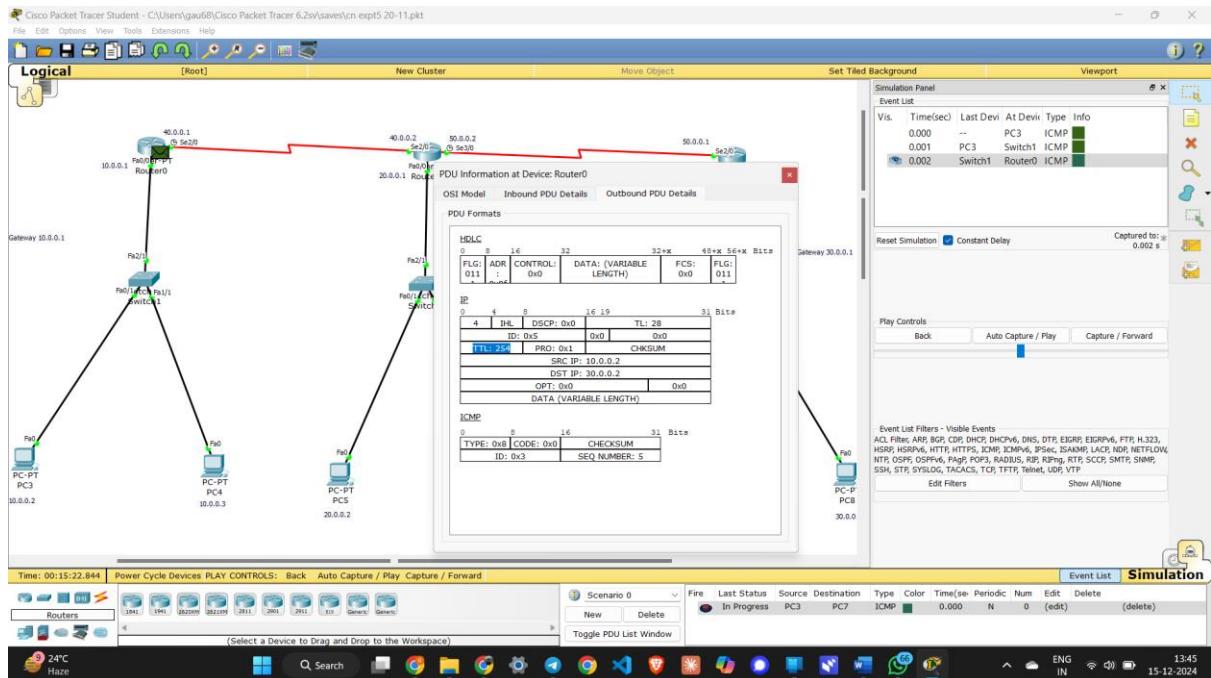
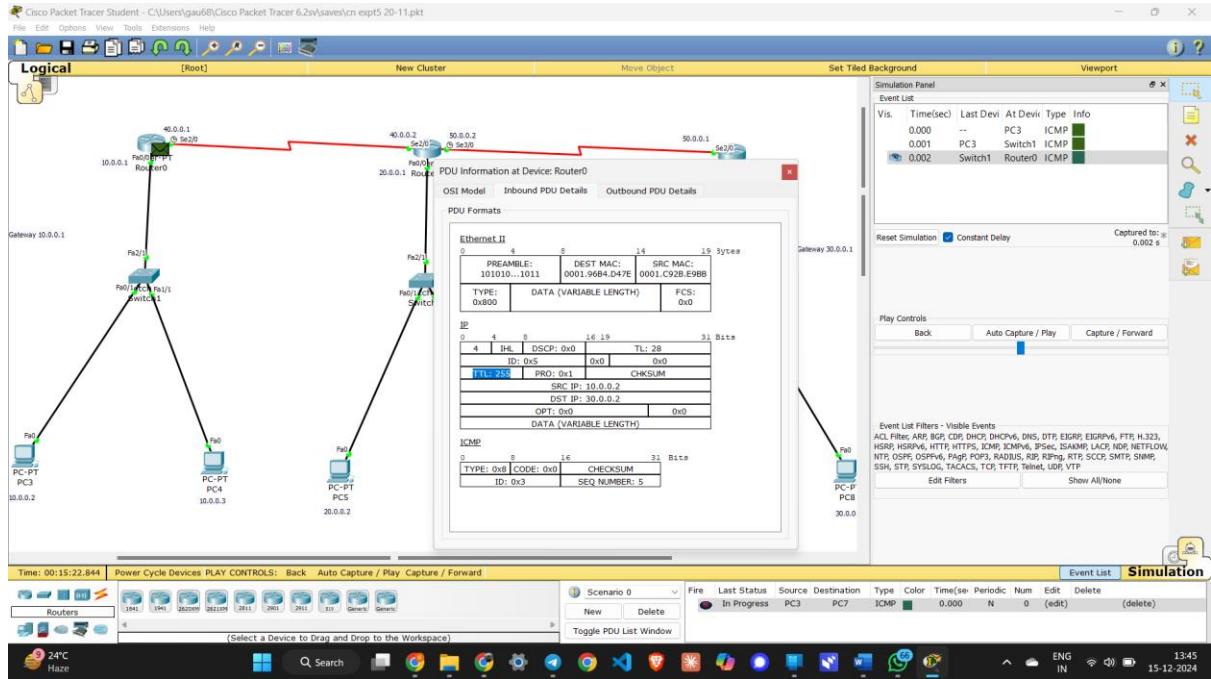
Ping statistics for 30.0.0.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 15ms, Average = 6ms
```

Demonstrating TTL:

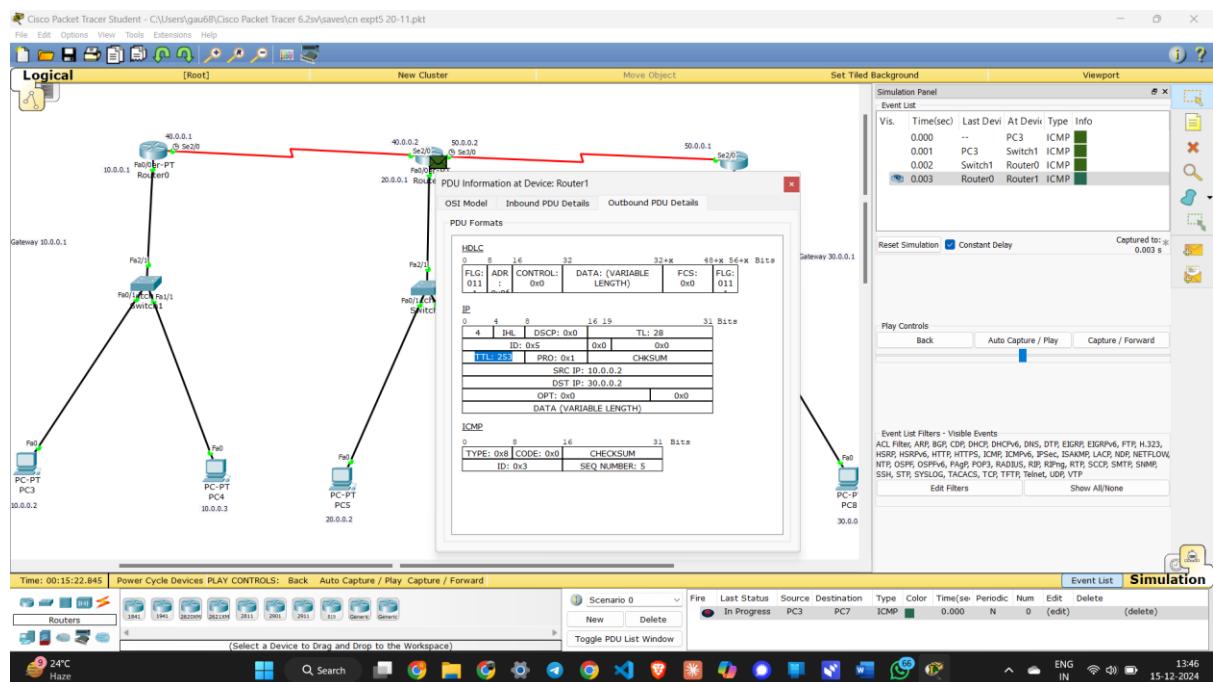
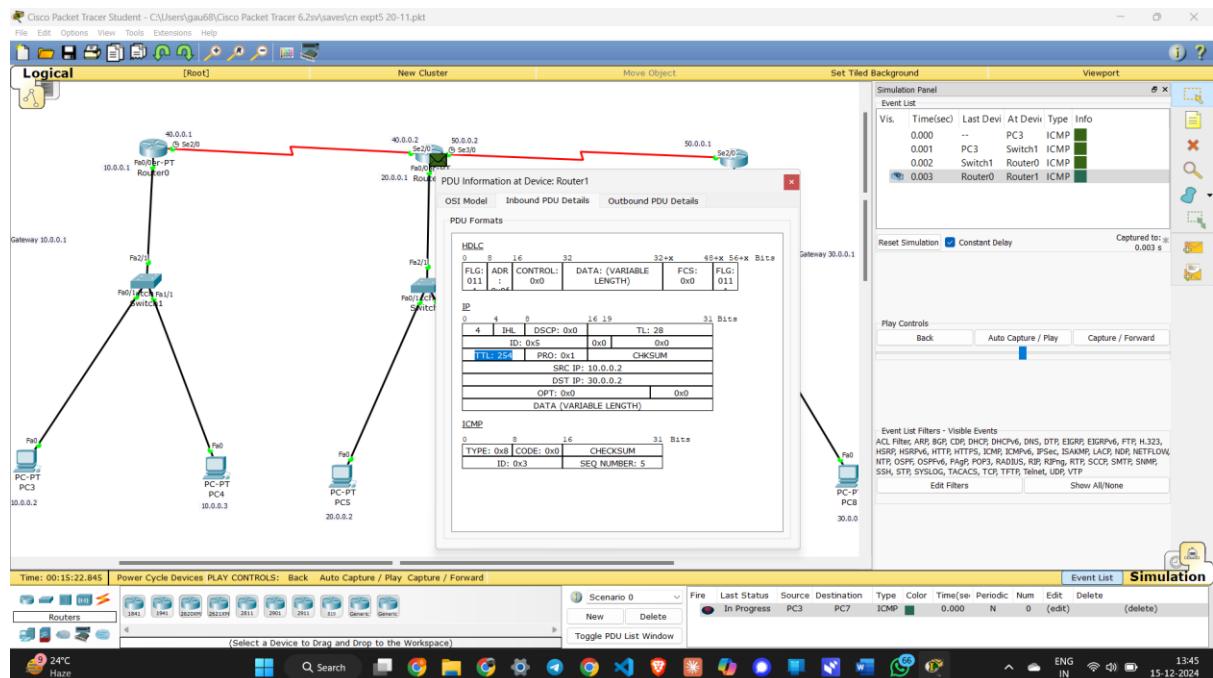
Switch1:



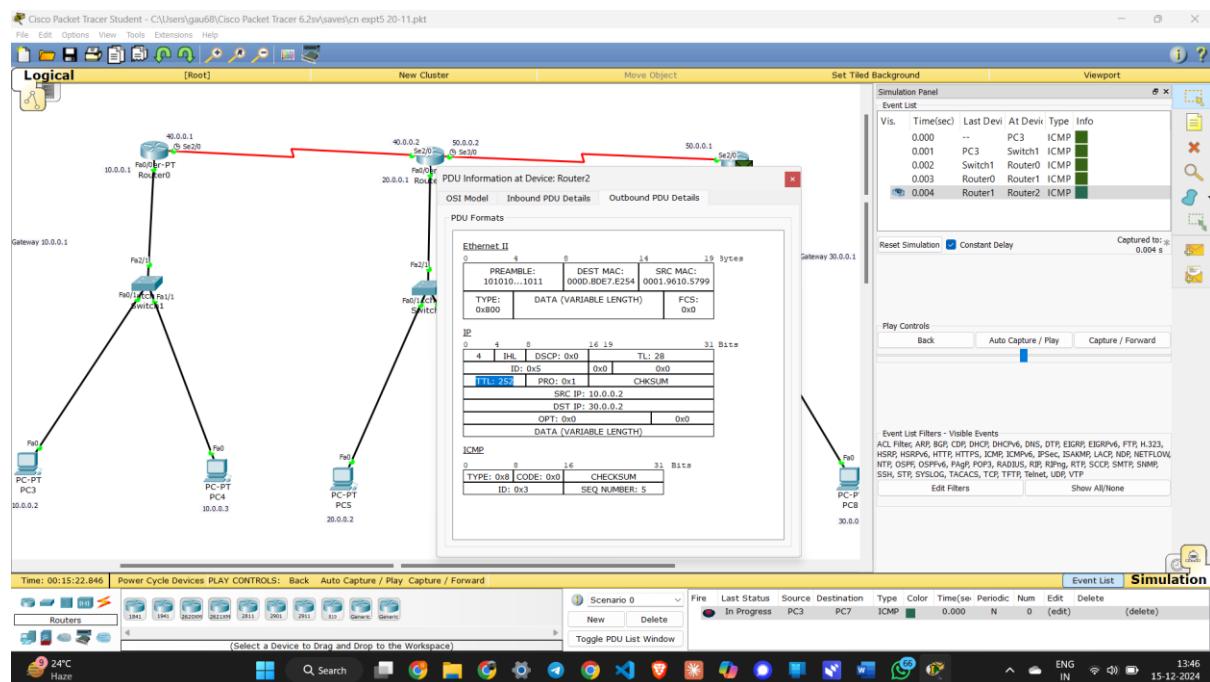
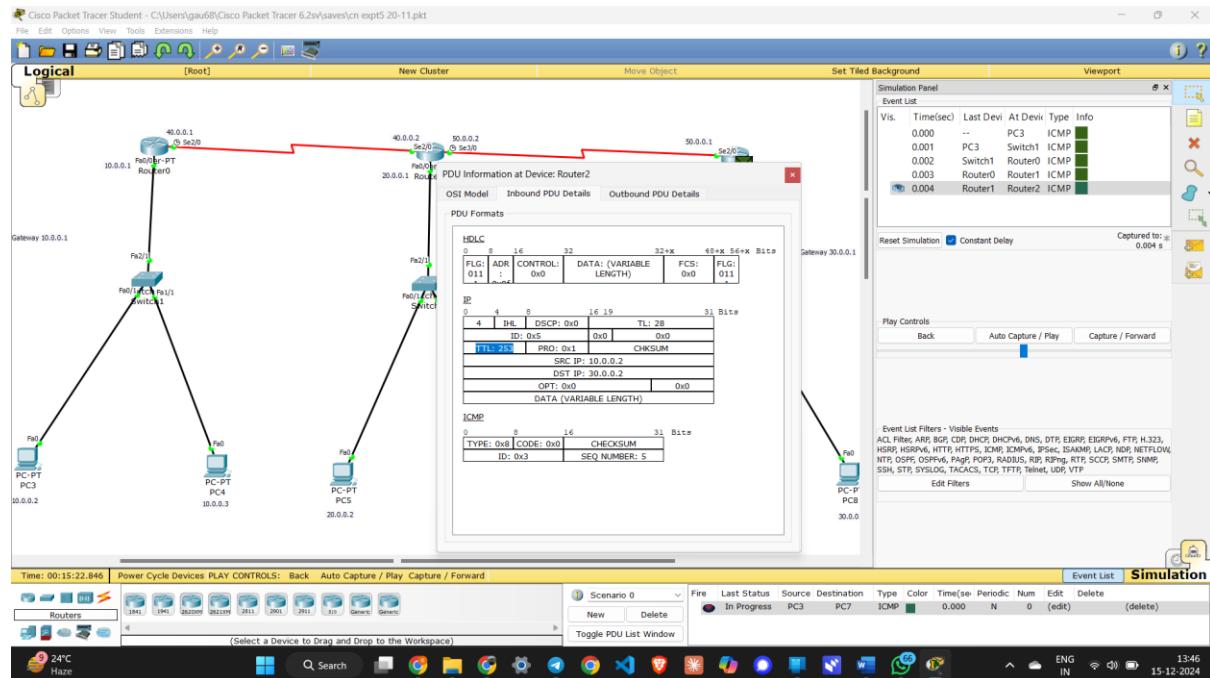
Router0:



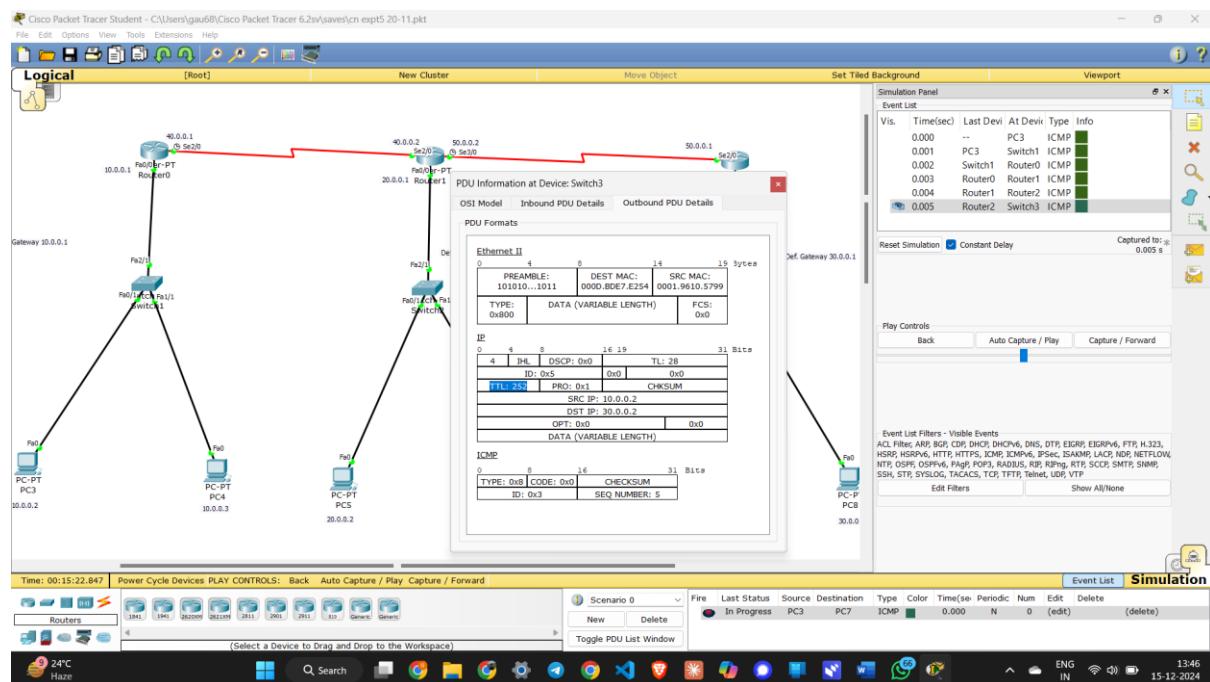
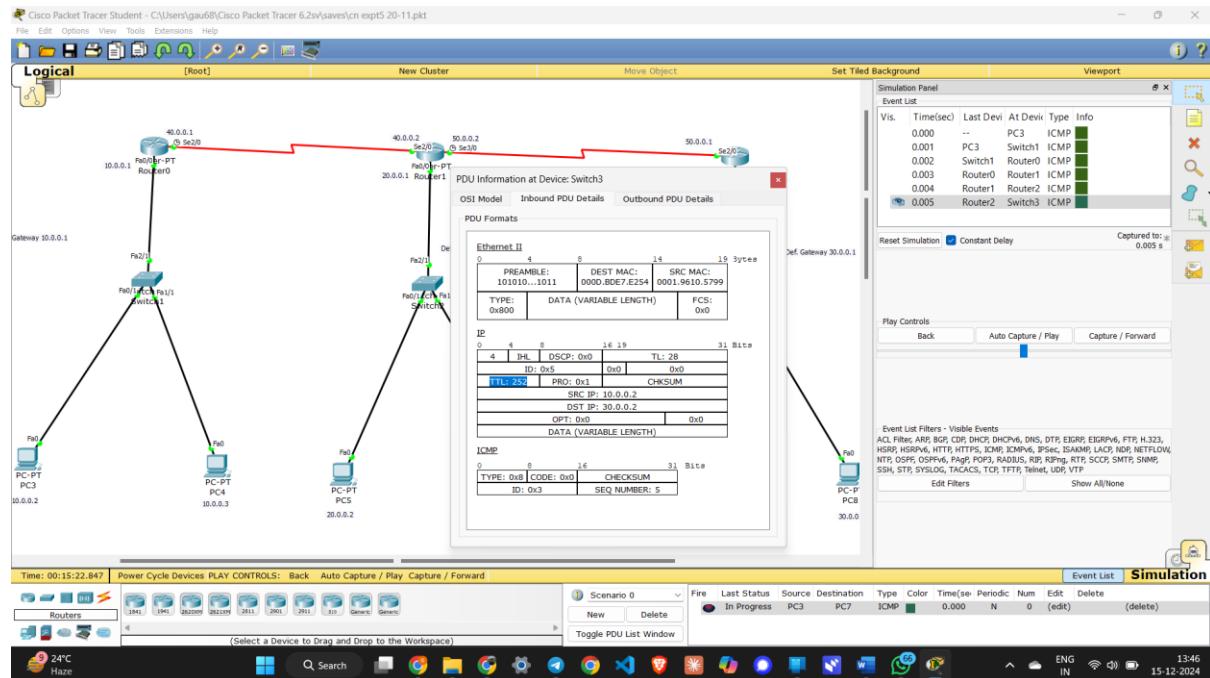
Router1:



Router2:



Switch3:



Expt. 6 – 27/11/2024

Date / /201

27/11

Expt. 6 :

Aim - OSPF Routing Protocol to Connect Areas

Topology -

Router-PT
Router0

20.0.0.1

20.0.0.2

10.0.0.1

Def. Gateway
10.0.0.1

Area 1

Router-PT
Router1

30.0.0.1

30.0.0.2

40.0.0.1

40.0.0.2

Def. Gateway
40.0.0.1

Area 0

Router-PT
Router2

FastE0

PC-PT
PC0

PC-PT
PC1

Procedure -

1- Configure IP addresses of all the components.

2- R0:

enable

config t

interface serial 2/0

encapsulation PPP

clock rate 64000

no shut

exit

R1:

enable

config t

interface serial 2/0

encapsulation PPP

no shut

exit

JS JNANA SWEEKAR

15.12.24

<p>Date / /201</p> <pre> R1: enable config t interface serial 2/0 encapsulation ppp no shut exit interface serial 3/0 encapsulation ppp clock rate 64000 no shut exit </pre> <p>3. Configure OSPF routing protocol:</p> <pre> R0: enable config t router ospf 1 router-id 1.1.1.1 network 10.0.0.0 0.255.255.255 area 3 network 20.0.0.0 0.255.255.255 area 1 exit </pre> <p><u>R1:</u></p> <pre> router ospf 1 router-id 2.2.2.2 network 20.0.0.0 0.255.255.255 area 1 network 30.0.0.0 0.255.255.255 area 0 exit </pre> <p><u>R2:</u></p> <pre> router ospf 1 router-id 3.3.3.3 network 30.0.0.0 0.255.255.255 area 0 network 40.0.0.0 0.255.255.255 area 2 exit </pre> <p>JNANA SWEEKAR</p>	<p>Date / /201</p> <p>4. Check routing table of R0:</p> <pre> Router# show ip route </pre> <p>5. Configure loopback address to routers:</p> <p><u>R0:</u></p> <pre> enable config t interface loopback 0 ip add 172.16.1.252 255.255.0.0 no shut </pre> <p><u>R1:</u></p> <pre> interface loopback 0 ip add 172.16.1.253 255.255.0.0 no shut </pre> <p><u>R2:</u></p> <pre> interface loopback 0 ip add 172.16.1.254 255.255.0.0 no shut </pre> <p>6. Check routing table of R1:</p> <pre> Router# show ip route </pre> <p>7. Create virtual link between R0 & R1 to connect area3 & area0</p> <p><u>R0:</u></p> <pre> enable config t router ospf 1 area 1 virtual-link 2.2.2.2 "message comes" exit </pre> <p><u>R1:</u></p> <pre> enable config t router ospf 1 area 1 virtual-link 1.1.1.1 "message comes" exit </pre> <p>JNANA SWEEKAR</p>
--	---

15.12.24

Date / /201

- B. R1 & R2 get updates about area3 - check routing table of R2 now:
- R2# show ip route
- O IA 20.0.0.0/8 via 30.0.0.1, Serial 2/0
 - C 40.0.0.0/8 is directly connected, Fa0/0
 - O IA 10.0.0.0/8 via 30.0.0.1, Serial 2/0
 - C 30.0.0.0/8 is directly connected, Serial 2/0

9. Check connectivity b/w host 10.0.0.2 to 40.0.0.2

Pc0 > ping 40.0.0.2

Pinging 40.0.0.2 with 32 bytes of data:

Reply from 40.0.0.2: bytes=32 time=2ms TTL=125

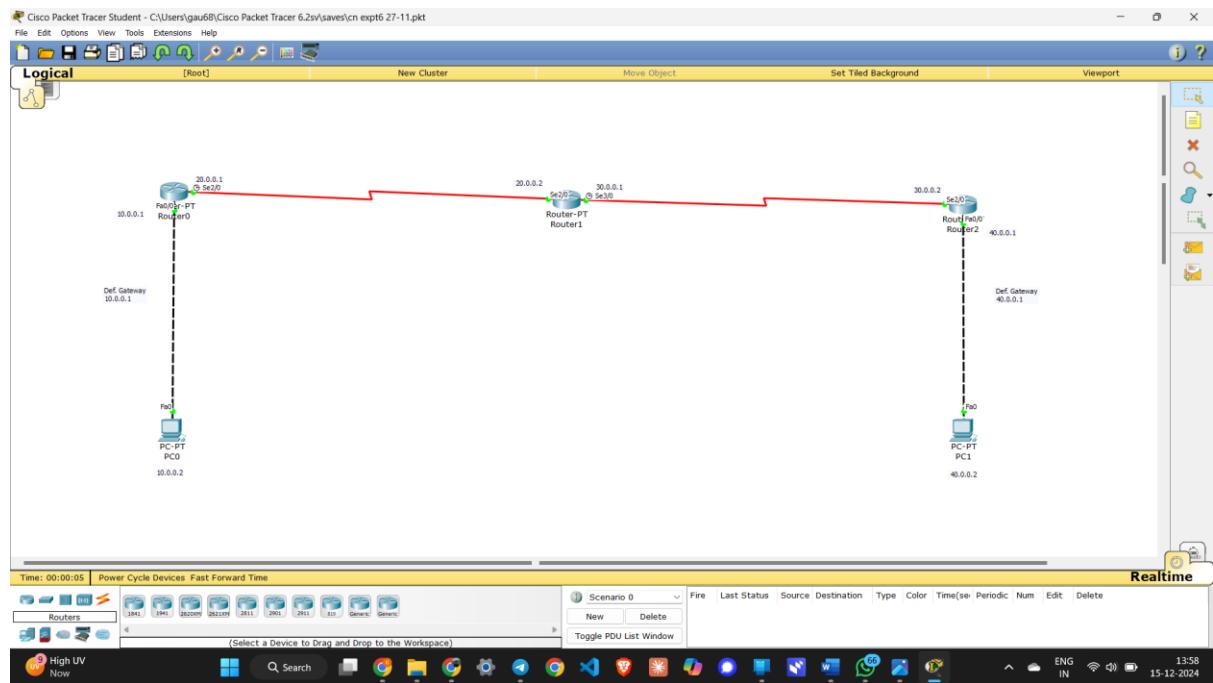
||
||
||

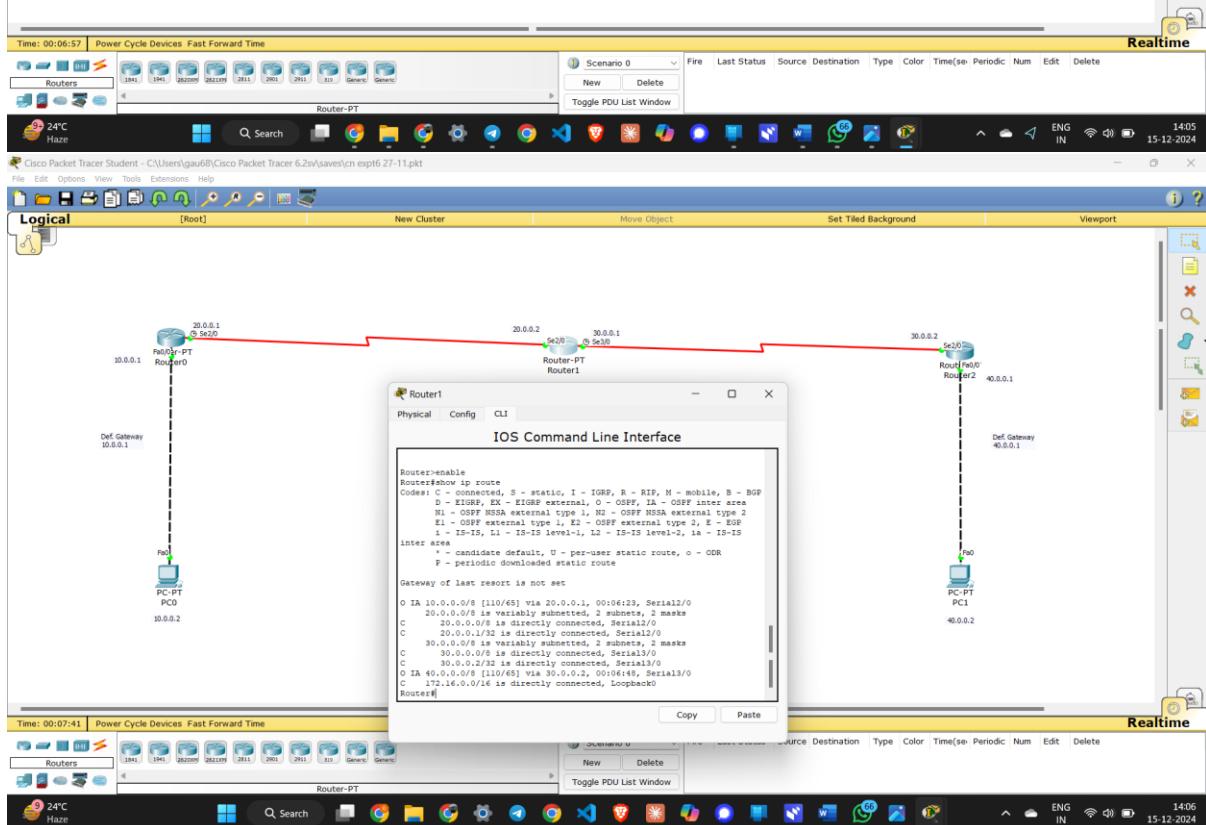
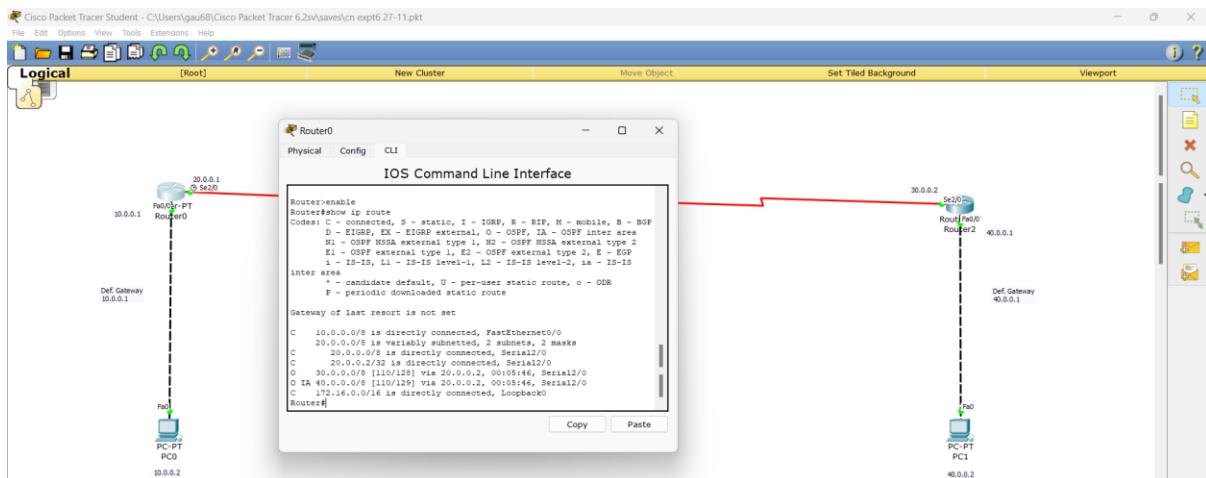
Ping statistics for 40.0.0.2:

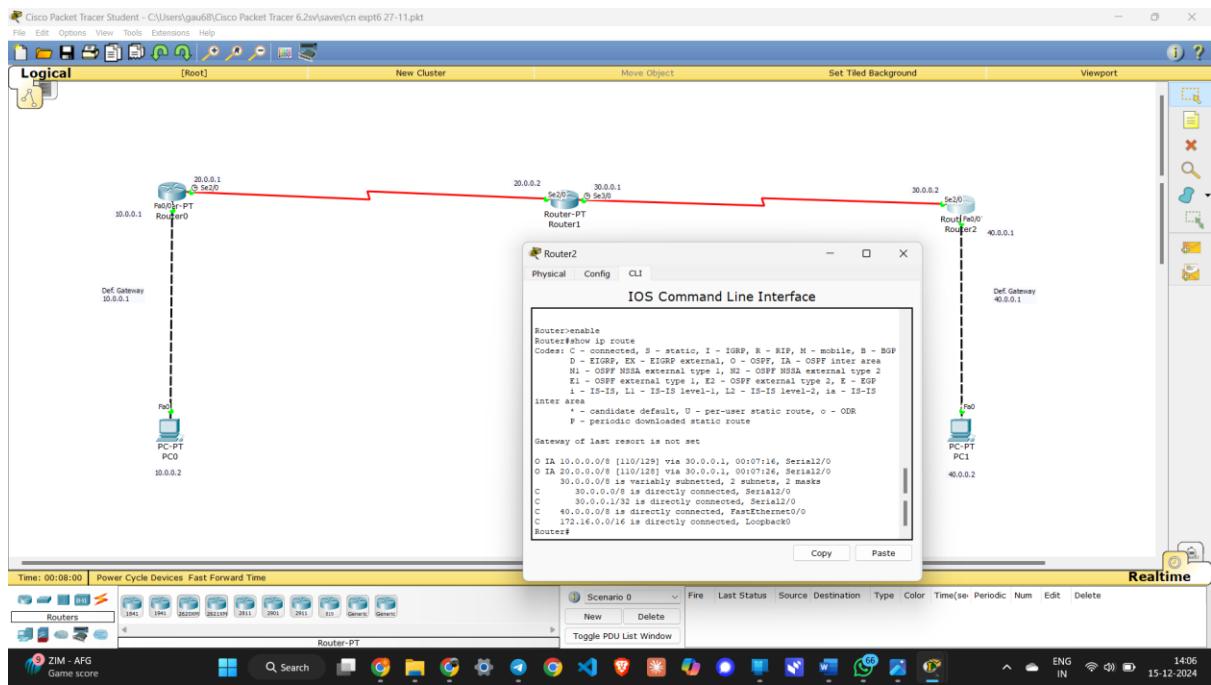
Packets: sent=4, Received=4, Loss=0 (0% loss)

15.12.24

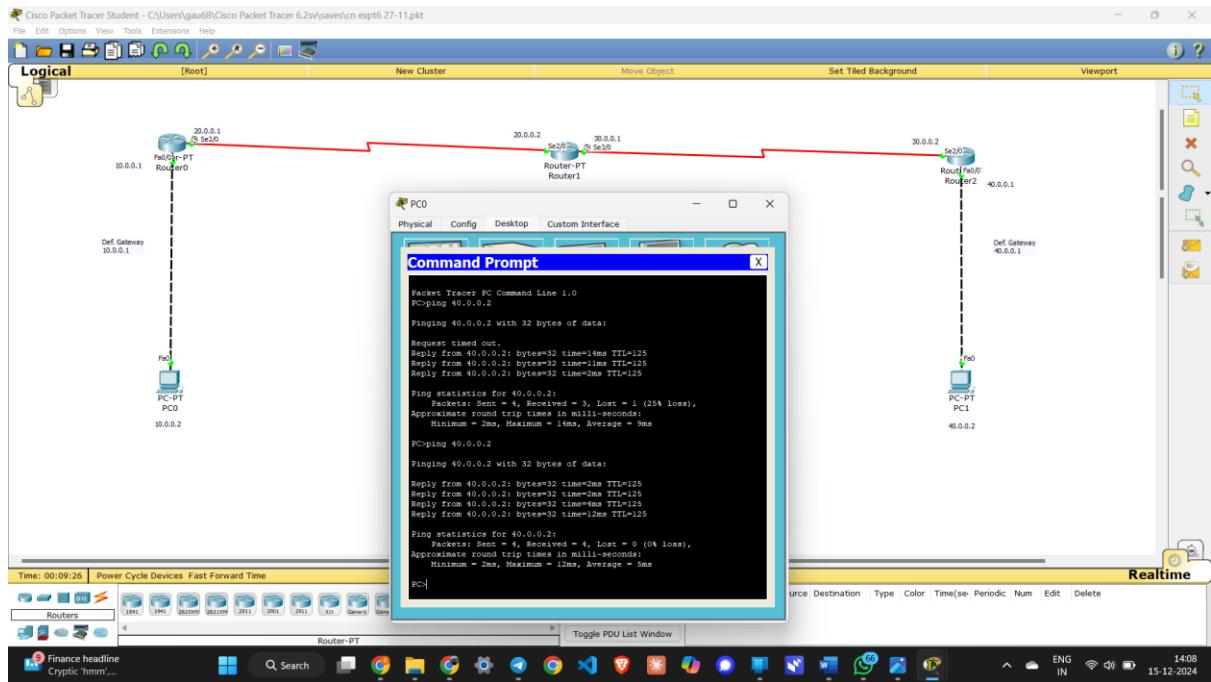
Topology:







Pinging:



Expt. 8 – 18/12/2024

Date / /201

18/12

Expt. 8 :

Aim : To ~~configure~~ configure Web Server, DNS within a LAN.

Topology :

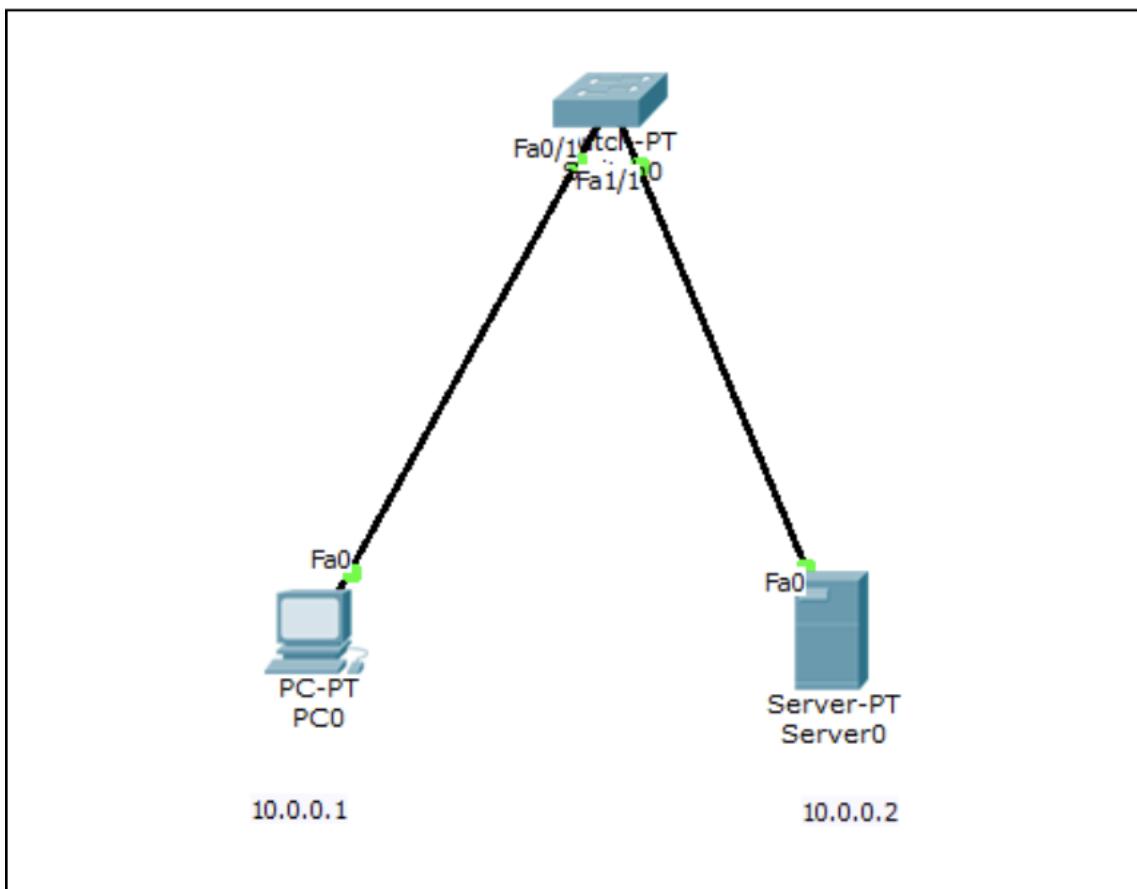
Procedure :

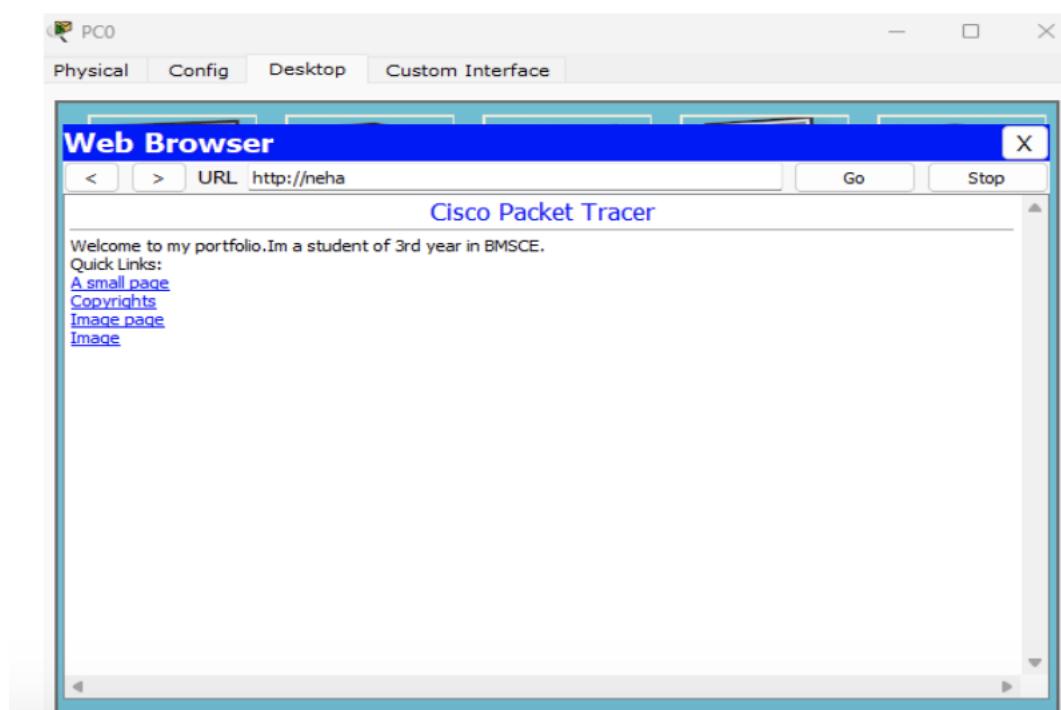
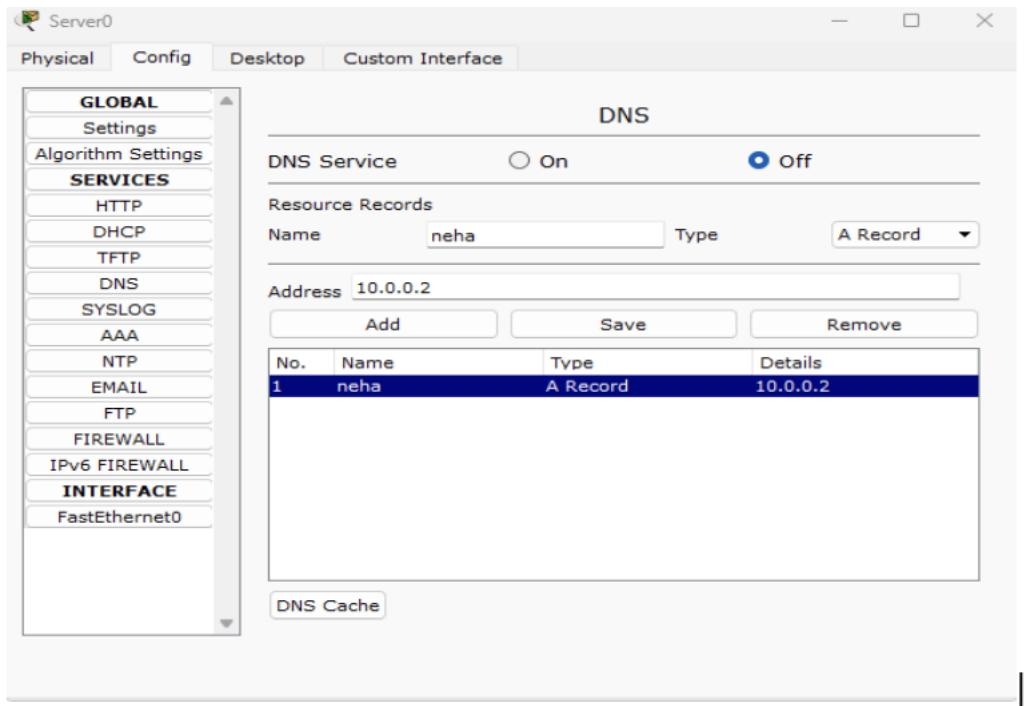
1. Set up the LAN as per the topology mentioned above & configure the devices.
2. Go to Server → Service → DNS :
Name: bmsce (Domain name)
Address: 10.0.0.2
Add the mapping of domain name to address
3. Go to PC → Config → Global → Settings →
DNS Server: 10.0.0.2
[The server that provides the DNS mapping]
4. Go to PC → Desktop → web Browser
Type the URL : http://bmsce

Date / /201

Observation:

1. Webpage hosted by the Server were visible on the browser.
2. The DNS was successful in mapping the domain name to the IP address.
3. DNS server is a server that contains a Domain name : IP address mapping to which the end devices send requests to map the Name to IP address.





Expt. 9 – 18/12/2024

Date / /201

Expt. 9 :

Aim: To construct simple LAN & understand the concept & operation of ARP.

Topology:

Procedure:

1. Create a topology as shown above
2. Configure the PC's & the Server
3. Click on Inspect mode (Icon), then click on the end devices to open ARP tables.
4. Send data packet from any end device, say server to other end devices, say 10.0.0.3 PC.
5. Open simulation mode to capture each step of data transfer.

Date / /201

Observations:

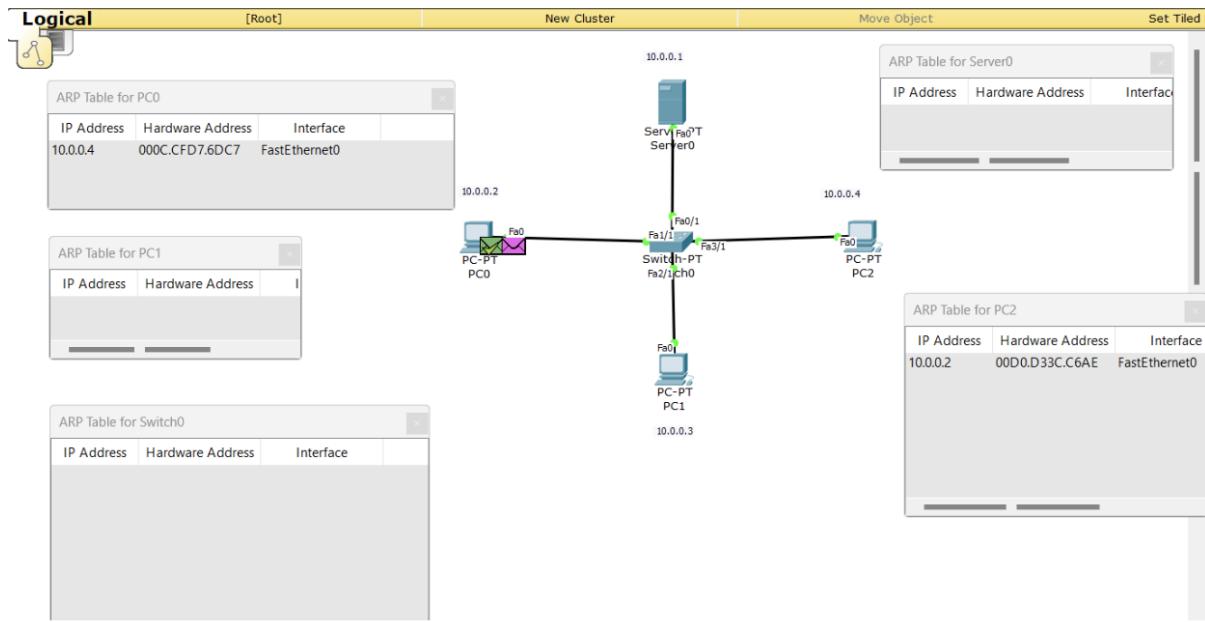
1. The ARP tables of all end devices are initially empty.
2. When the data packet from Server arrives at the switch, since the Server MAC address is unknown, it sends a broadcast message to all devices.
3. The device with the IP address present in the destination address of the data packet responds to the message.
4. The server & the PC update their ARP tables matching IP address to MAC address.
5. Over time, the ARP tables grow as data packets are sent.
6. The MAC table of the switch which was initially empty updates its MAC table gradually too.

ARP table for 10.0.0.4

X

IP address	Hardware address	Interface
10.0.0.3	0001.C726.47E5	Fastethernet0

7. Similarly, other ARP tables are updated.



Simulation Panel

Event List

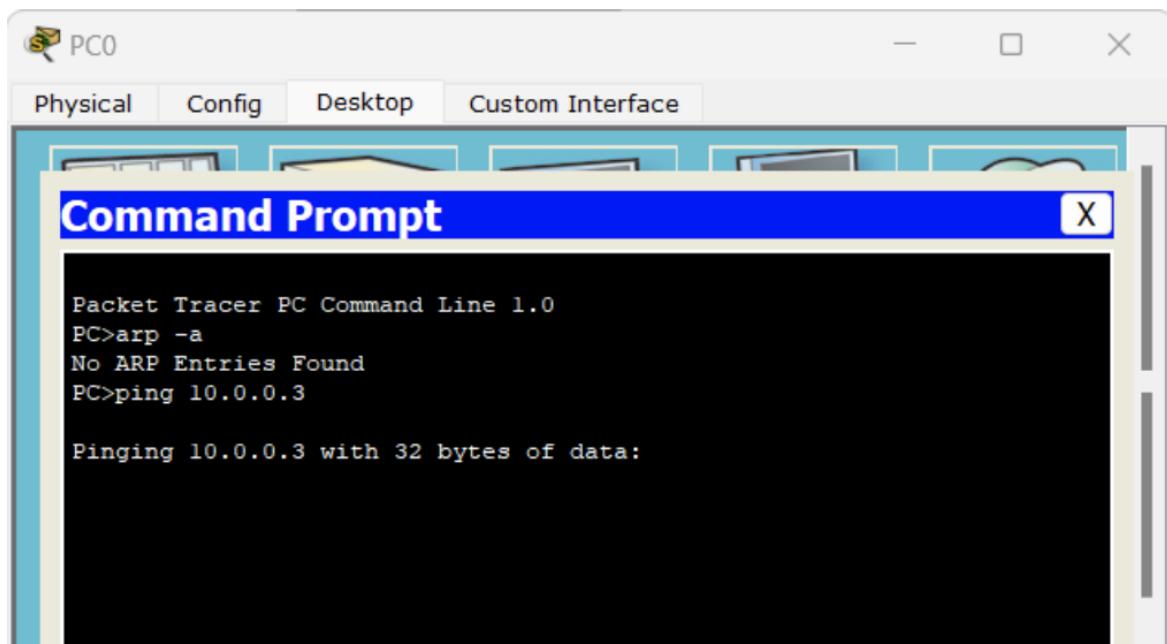
Vis.	Time(sec)	Last Devi	At Devic	Type	Info
	0.001	PC0	Switch0	ARP	
	0.002	Switch0	Server0	ARP	
	0.002	Switch0	PC1	ARP	
	0.002	Switch0	PC2	ARP	
	0.003	PC2	Switch0	ARP	
👁	0.004	Switch0	PC0	ARP	
👁	0.004	--	PC0	ICMP	

Controls

- Reset Simulation**
- Constant Delay** (checkbox checked)
- Captured to: 0.004 s**

Play Controls

- Back**
- Auto Capture / Play** (highlighted)
- Capture / Forward**



Switch:

The screenshot shows a terminal window for a switch. The command entered is 'show mac address-table'. The output displays the MAC Address Table:

Vlan	Mac Address	Type	Ports
1	0009.7c3c.0719	DYNAMIC	Fa2/1
1	000c.cfd7.6dc7	DYNAMIC	Fa3/1
1	0090.2b9d.194b	DYNAMIC	Fa0/1
1	00d0.d33c.c6ae	DYNAMIC	Fa1/1

Switch>

At the bottom right of the terminal window are 'Copy' and 'Paste' buttons.

Expt. 10 – 18/12/2024

01
Date / /201
18/12/24

Expt. 10:

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

Topology:

PC
10.0.0.1

Router
10.0.0.2

Procedure:

1. Create a topology as given above & configure the devices.
2. Commands in Router:
Router > enable
Router# config terminal
Router(config)# hostname R1
enable secret 1234
interface fastethernet 0/0
Router(config-if)# ip address 10.0.0.2 255.0.0.0
no shut
line vty 0 3
Router(config-line)# login
login disabled on line 194, until 'password' is set
R1(config-line)# password 4321
R1(config-line)# exit
user access verification password

JS JNANA SWEETAR

03.01.25

Date / / 201

Date / / 201

Observations :

R1 (config) # exit
R1 # who

Building configuration ...
[OK]

Note : try o 3 : First ~~for~~ virtual terminal
lines for Telnet access

In PC : command prompt

- First try pinging to see if the devices are
connected

PL > telnet 10.0.0.2
Trying 10.0.0.2 ... open

User Access Verification

password : 4321

password : 4321

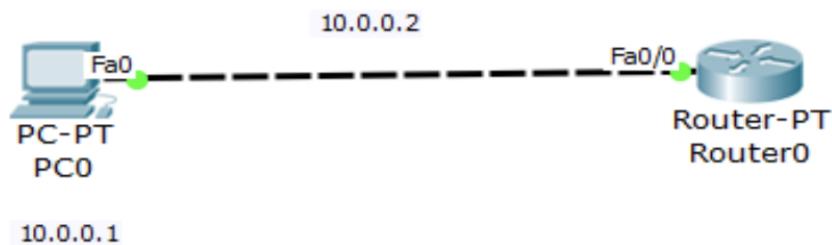
R1 > enable

password : 1234

R1 # show ip route

C 10.0.0.0/8 is directly connected, Fasteth-
ernet 0/0

R1 #



Router0

Physical Config CLI

IOS Command Line Interface

```
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>enable
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname r1
r1(config)#enable secret neha
r1(config)#interface fastethernet 0/0
r1(config-if)#ip address 10.0.0.2 255.0.0.0
r1(config-if)#no shut

r1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
```

Router0

Physical Config CLI

IOS Command Line Interface

```
rl(config-if)#line vty 0 3
rl(config-line)#login
% Login disabled on line 132, until 'password' is set
% Login disabled on line 133, until 'password' is set
% Login disabled on line 134, until 'password' is set
% Login disabled on line 135, until 'password' is set
rl(config-line)#login password chitral
^
% Invalid input detected at '^' marker.

rl(config-line)#login
% Login disabled on line 132, until 'password' is set
% Login disabled on line 133, until 'password' is set
% Login disabled on line 134, until 'password' is set
% Login disabled on line 135, until 'password' is set
rl(config-line)#password chitral
rl(config-line)#exit
rl(config)#exit
rl#
%SYS-5-CONFIG_I: Configured from console by console

rl#wr
Building configuration...
[OK]
rl#
```

Copy Paste

Open command prompt of PC0: ping 10.0.0.2

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=0ms TTL=255

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

PC>telnet 10.0.0.2
Trying 10.0.0.2 ...Open

User Access Verification

Password:
rl>neha
Translating "neha"...domain server (255.255.255.255)
* Unknown command or computer name, or unable to find computer address
```

PC0

Physical Config Desktop Custom Interface

Command Prompt

```
rl>
rl>exit

[Connection to 10.0.0.2 closed by foreign host]
PC>telnet 10.0.0.2
Trying 10.0.0.2 ...Open

User Access Verification

Password:
Password:
rl>enable
Password:
rl#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/0
rl#
```

Expt. 11 – 18/12/2024

8
9
10
11
12
Expt. 11 :

Date / / 201

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

Topology:

Router 1841

192.168.1.1 192.168.2.1

Fa0/0 Ethen 6/1

Fa1/1 Fa2/1 Fa3/1

Switch

PC PC PC PC

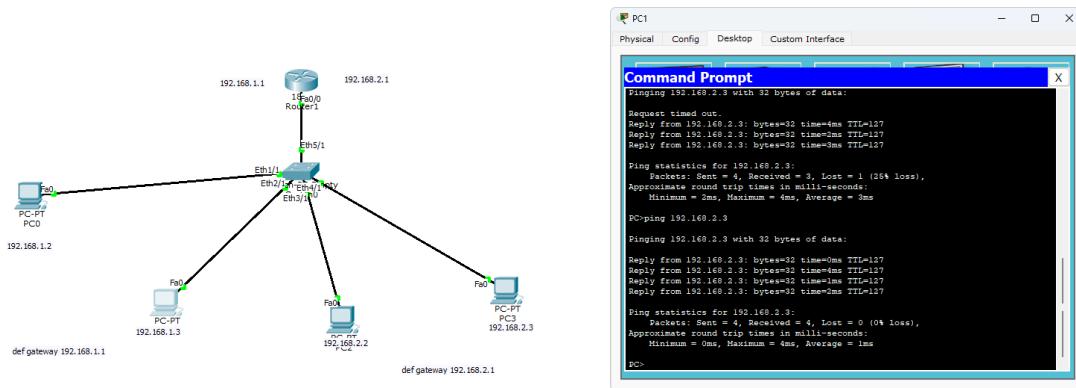
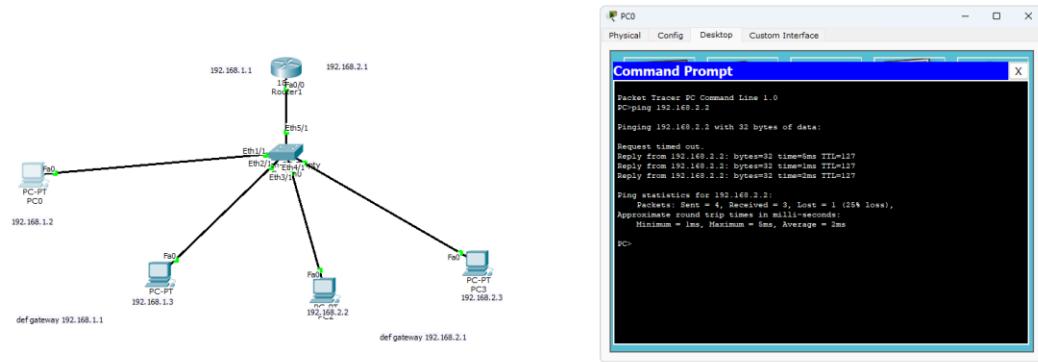
192.168.1.2 192.168.1.3 192.168.2.2 192.168.2.3

Procedure:

1. Place a 1841 router, a switch & 4 PCs
2. Connect the 4 PCs to the switch via fastethernet
3. Since only 4 fastethernet ports are available on the switch, we have to add ~~an~~ an ethernet port.
4.
 - Switch off the power button of switch
 - Add the ethernet port to switch
 - Switch on the power button
 - Connect the router to switch via Ethernet 6/1
5. In the switch, go to config tab &
- Select VLAN Database

JS JNANA SWEEKAR

<p>Date / /201</p> <p>1. Give VLAN number, say 2 - Give VLAN name, say, "ceise" - Add it to the Database.</p> <p>6. Select the Switch: - Go to config - Go to Ethernet 0/1, which is connected to Router - Make it the trunk</p> <p>7. Configure the PCs as shown in the topology</p> <p>8. Select Switch: - Go to config - Go to Fastethernet 2/1 - Set VLAN number as 2, i.e., ceise - Similarly set VLAN 2 for Fa3/0 interface</p> <p>9. Configure the Router: Router(config)# interface fa0/0 Router(config-if)# ip address 192.168.1.1 255.255.255.0 Router(config-if)# no shut Router(config-if)# exit</p> <p>Now to configure the router's VLAN interface: Router(config)# interface Fa 0/0.1 Router(config-subif)# encapsulation dot1q 2 # ip address 192.168.2.1 255.255.255.0 Router(config-subif)# no shut Router(config-subif)# exit</p> <p>10. Ping devices within the same VLAN is to devices of different VLAN.</p>	<p>Date / /201</p> <p>Observations:</p> <p>1. When devices are pinged within same VLAN: - Pinging 192.168.1.3 from 192.168.1.2 - The data packet doesn't go to router - The switch forwards the packet without the need of the router</p> <p>2. When a device ping a device of another VLAN: - Pinging 192.168.2.3 from 192.168.1.2 - The data packets journey is as follows: 192.168.1.2 → Switch → Router 192.168.2.3 ← Switch</p> <p>3. VLANs divide a single switch into multiple logical switches - Devices in one VLAN can't directly communicate with devices in another VLAN without without a router</p> <p>4. Traffic Isolation: - Each VLAN maintains its own broadcast domain - Broadcasts sent by devices in one VLAN do not reach devices in another VLAN.</p> <p>5. VLAN trunking allows switches to forward frames from different VLANs over a single link called trunk - This is done by adding an additional header information called tag to the ethernet frame - VLAN tagging</p>
--	---



Expt. 12 – 18/12/2024

Date / /201

Date / /201

18/12

Expt LAB NO 12:
WLAN

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

Topology:

```

graph LR
    Router[Router 10.0.0.2] --- Switch[Switch]
    Switch --- PC1[PC 10.0.0.1]
    Switch --- PC3[PC 10.0.0.3]
    Switch --- Laptop4[Laptop 10.0.0.4]
    Switch --- AP[Access Point 20000]
  
```

Procedure :

- 1] Create the topology as given above & configure the devices.
- 2] Configure Access Point:
 Click Access Point → config → Part 1.
 Select : WEP
 set key : 1234567890

Topology after Wireless Configuration:

```

graph LR
    Router[Router 10.0.0.2] --- Switch[Switch]
    Switch --- PC1[PC 10.0.0.1]
    Switch --- PC3[PC 10.0.0.3]
    Switch --- Laptop4[Laptop 10.0.0.4]
    Switch --- AP[Access Point 20000]
    Router -.-> PC1
    Router -.-> PC3
    Router -.-> Laptop4
    Router -.-> AP
  
```

3. Configure PC & Laptop with wireless standards:
 - Switch off device
 - Drag the existing PT-HOST-NM-JAM to the component listed in the LHS of Physical
 - Drag WMP300N wireless interface to the empty port.
 - Switch on the device.
4. In the config tab, a new wireless interface was added.
5. Configure the device by entering SSID, WEP, WEP Key, IP address & Gateway.
6. Ping from every device to every other device to check for connectivity.

JS JNANA SWEEKAR

08.01.25

Observation:

1. We were able to ping from every device to every other device.

2. Access Point :

Creates bridge b/w wired & wireless devices.

- SSID Broadcasting : Announces the wireless network's name (SSID) to allow devices to connect using WEP, WPA or WPA2

3. WMP300N wireless interface :

- Wireless network adapter that enables devices to communicate with access point using wireless signals.

4. Pinging: 10.0.0.1 to 10.0.0.3 :

10.0.0.1 → Switch → Access Point → 10.0.0.3

- This is after the ARP tables are updated after broadcasting.

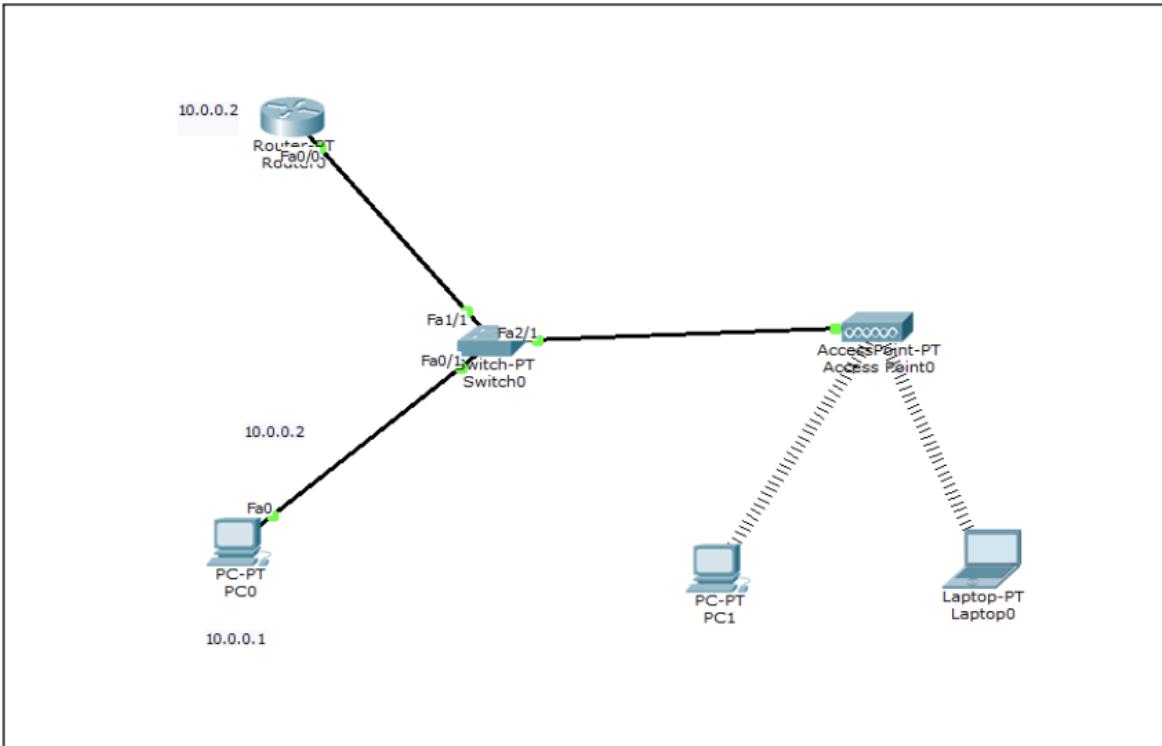
5. Pinging = 10.0.0.3 to 10.0.0.1 :

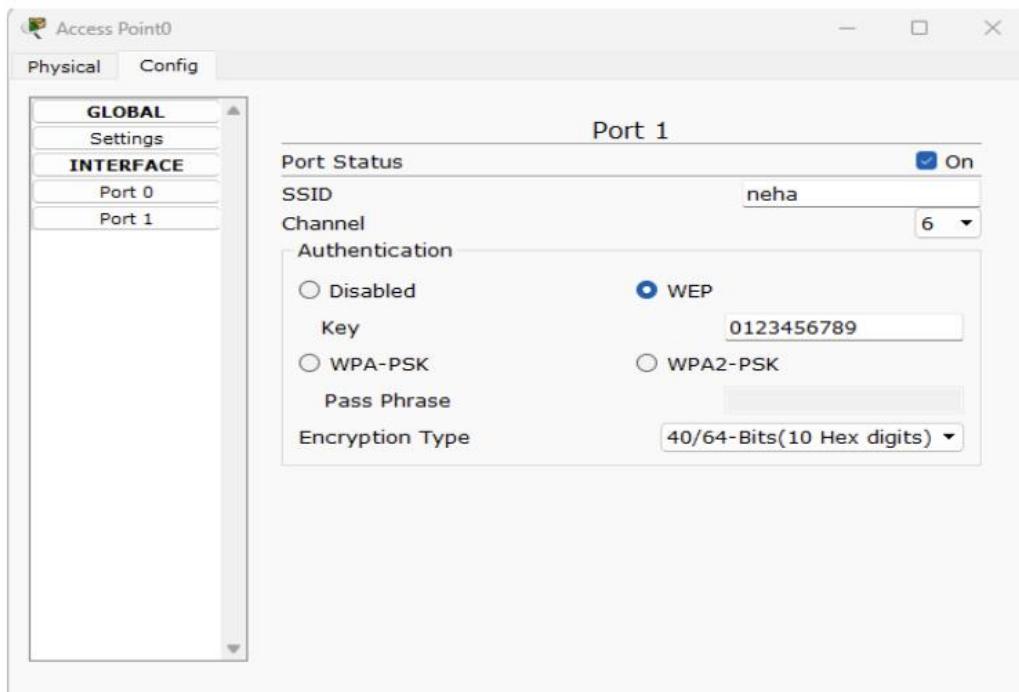
10.0.0.3 → Access Point → Switch → 10.0.0.1

6. Pinging = 10.0.0.3 to 10.0.0.4 :

10.0.0.3 → Access Point → 10.0.0.4

7. Every device is now connected to every other device in the WLAN





Router0

Physical Config CLI

IOS Command Line Interface

```
--- System Configuration Dialog ---  
Continue with configuration dialog? [yes/no]: no  
  
Press RETURN to get started!  
  
Router>enable  
Router#config terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)#interface fastethernet 0/0  
Router(config-if)#ip address 192.168.1.1 255.255.255.0  
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  
  
Router(config-if)#exit  
Router(config)#
```

Copy

Paste

Physical Config Desktop Custom Interface

Command Prompt

```
Packet Tracer PC Command Line 1.0  
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=20ms TTL=128  
Reply from 10.0.0.3: bytes=32 time=9ms TTL=128  
Reply from 10.0.0.3: bytes=32 time=6ms TTL=128  
Reply from 10.0.0.3: bytes=32 time=8ms 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 = 6ms, Maximum = 20ms, Average = 10ms  
  
PC>ping 10.0.0.4  
  
Pinging 10.0.0.4 with 32 bytes of data:  
  
Reply from 10.0.0.4: bytes=32 time=18ms TTL=128  
Reply from 10.0.0.4: bytes=32 time=10ms TTL=128  
Reply from 10.0.0.4: bytes=32 time=7ms TTL=128  
Reply from 10.0.0.4: bytes=32 time=11ms 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 = 7ms, Maximum = 18ms, Average = 11ms
```

Expt. 13 – 21/12/2024

CYCLE - 2

Date / /201

Date / /201

202

Expt. 13 :

Aim : Error detecting code using CRC = CCITT (16 bits)

```

def max(a, b):
    result = []
    for i in range(1, len(b)):
        if a[i] == b[i]:
            result.append('0')
        else:
            result.append('1')
    return ''.join(result)

def moddiv(dividend, divisor):
    pick = len(divisor)
    tmp = dividend[0:pick]
    while pick < len(dividend):
        if tmp[0] == '1':
            tmp = not(dividend, tmp) +
                  dividend[pick]
        else:
            tmp = not('0' * pick, tmp) +
                  dividend[pick]
    remainder = tmp[0:pick]
    dividend = tmp[pick:]
    return remainder, dividend

```

def encode(data, key):
 l_key = len(key)
 appended_data = data + '0' * (l_key - 1)
 remainder = moddiv(appended_data, key)
 codeword = data + remainder
 print("Remainder", remainder)
 print("EncodedData (Data + remainder)", codeword)
 return codeword

def decode(data, encoded_data, key):
 remainder = moddiv(encoded_data, key)
 print("Remainder after decoding:", remainder)
 if '1' not in remainder:
 print("No error detected in received data")
 else:
 print("Error detected in received data")
 data = "101001000100"
 key = "101"
 encoded_data = encode(data, key)
 decoded_data = decode(encoded_data, key)
 print(decoded_data)

Output:

remainder = 1
 encoded_data (data + remainder) = 10010010001001001
 Remainder after decoding = 000
 No error detected in received data.

checkword = tmp
 return checkword

Code:

```
#include <stdio.h>
#include <string.h>
#define N strlen(gen_poly)
char data[28], gen_poly[10], check[28];
int data_len, i, j;
void XOR() {
    for (j = 0; j < N; j++) {
        check[j] = (check[j] == gen_poly[j]) ? '0' : '1';
    }
}
void crc() {
    for (i = 0; i < N; i++) {
        check[i] = data[i];
    }
    do {
        if (check[0] == '1') {
            XOR();
        }
        for (j = 0; j < N - 1; j++) {
            check[j] = check[j + 1];
        }
        check[j] = data[i++];
    } while (i <= data_len + N - 1);
}
void receiver() {
    printf("\nData received: ");
    scanf("%s", data);
    crc();
    for (i = 0; i < N - 1; i++) {
```

```
if (check[i] == '1') {
break;
}
}

if (i < N - 1) {
printf("\nERROR!");
} else {
printf("\nNO ERROR!");
}
}

5
1

int main() {
printf("\nEnter data: ");
scanf("%s", data);
printf("\nEnter generator: ");
scanf("%s", gen_poly);
data_len = strlen(data);

// Append N-1 zeros to the data
for (i = data_len; i < data_len + N - 1; i++) {
data[i] = '0';
}

data[data_len + N - 1] = '\0'; // Null-terminate the string
printf("\nData with padded 0's: %s", data);
crc();
printf("\nCheck sum: ");
for (i = 0; i < N - 1; i++) {
printf("%c", check[i]);
}

// Append checksum to data
```

```
for (i = data_len; i < data_len + N - 1; i++) {  
    data[i] = check[i - data_len];  
}  
  
data[data_len + N - 1] = '\0'; // Null-terminate the string  
  
printf("\nFinal data to be transmitted: %s", data);  
  
receiver();  
  
return 0;  
}
```

Output:

Output

```
Enter data: 1001  
  
Enter generator: 101  
  
Data with padded 0's: 100100  
Check sum: 11  
Final data to be transmitted: 100111  
Data received: 100110  
  
ERROR!  
  
==== Code Execution Successful ===
```

Expt. 14 – 21/12/2024

Date / /201

Expt. 14:

Aim: Congestion Control using Leaky Bucket Algorithm.

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#define NOF_PACKETS 5
int rand(int a) {
    int m = (random() % 10) + a;
    return m == 0 ? m : rand(a);
}
longint random(void);
int main() {
    int packets[NOF_PACKETS], i, d, b_size, o_rate,
        p_size, op = 0, p, p_time, op;
    for (i = 0; i < NOF_PACKETS; i++)
        packets[i] = random() % 100;
    for (i = 0; i < NOF_PACKETS; i++) {
        printf("In %d bytes (%d bytes)\n", i, packets[i]);
        if (packets[i] > b_size)
            printf("REJECTED\n");
        else {
            p = p + packets[i];
            if (p > o_rate) {
                p = o_rate;
                op = op + p - o_rate;
            }
            else
                op = op + p;
        }
    }
    printf("Total packet size: %d\n", op);
}

```

Date / /201

(b -bytes) is greater than bucket capacity (b -bytes) - PACKET-REJECTED packet-size[i] b_size)

else

printf ("In Bucket capacity exceeds - d-Packets REJECTED !") ;

else {

$p_sy_amt = \text{Packet_sy}[i]$;

printf ("In Incoming packet amt - d",
"packet_sy[i]);

printf ("In Bytes remaining to transmit: "d",
"p_sy_rm);

while ($p_sy_rm > 0$) {

- sleep();
- $p(p_sy_rm);$

If ($p_sy_rm \leq 0$ -rate)

$op = p_sy_rm$, $p = p_sy_rm = 0$

else

printf (Packet size, op);

else

printf ("No packets to transmit");

}

}

JNANA SWEEKAR

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h> // for sleep function
#define NOF_PACKETS 5

// Function to simulate sending packets
void send_packet(int packet_size, int output_rate) {
    while (packet_size > 0) {
        int sent = (packet_size < output_rate) ? packet_size : output_rate;
        printf("Packet of size %d Transmitted---", sent);
        packet_size -= sent;
        printf("Bytes Remaining to Transmit: %d\n", packet_size);
        sleep(1); // Simulate time delay between packets
    }
}

int main() {
    int output_rate, bucket_size, incoming_packet_size;
    int i, packet_size[NOF_PACKETS];
    // Input number of packets and their sizes
    for(i = 0; i < NOF_PACKETS; i++) {
        packet_size[i] = rand() % 100; // Random packet size between 0 and 99
        printf("packet[%d]:%d bytes\n", i, packet_size[i]);
    }
    printf("Enter the Output rate:");
    scanf("%d", &output_rate);
    printf("Enter the Bucket Size:");
    scanf("%d", &bucket_size);
    for(i = 0; i < NOF_PACKETS; i++) {
        printf("\nIncoming Packet size: %d\n", packet_size[i]);
        if(packet_size[i] > bucket_size) {
```

```

printf("Incoming packet size (%dbytes) is Greater than bucket capacity
(%dbytes)-PACKET REJECTED\n", packet_size[i], bucket_size);

55

continue;

}

printf("Bytes remaining to Transmit: %d\n", packet_size[i]);

send_packet(packet_size[i], output_rate);

}

return 0;

}

```

Output:

Output Clear

```

packet[0]:83 bytes
packet[1]:86 bytes
packet[2]:77 bytes
packet[3]:15 bytes
packet[4]:93 bytes
Enter the Output rate:30
Enter the Bucket Size:85

Incoming Packet size: 83
Bytes remaining to Transmit: 83
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 53
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 23
Packet of size 23 Transmitted---Bytes Remaining to Transmit: 0

Incoming Packet size: 86
Incoming packet size (86bytes) is Greater than bucket capacity (85bytes)-PACKET REJECTED

Incoming Packet size: 77
Bytes remaining to Transmit: 77
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 47
Packet of size 30 Transmitted---Bytes Remaining to Transmit: 17
Packet of size 17 Transmitted---Bytes Remaining to Transmit: 0

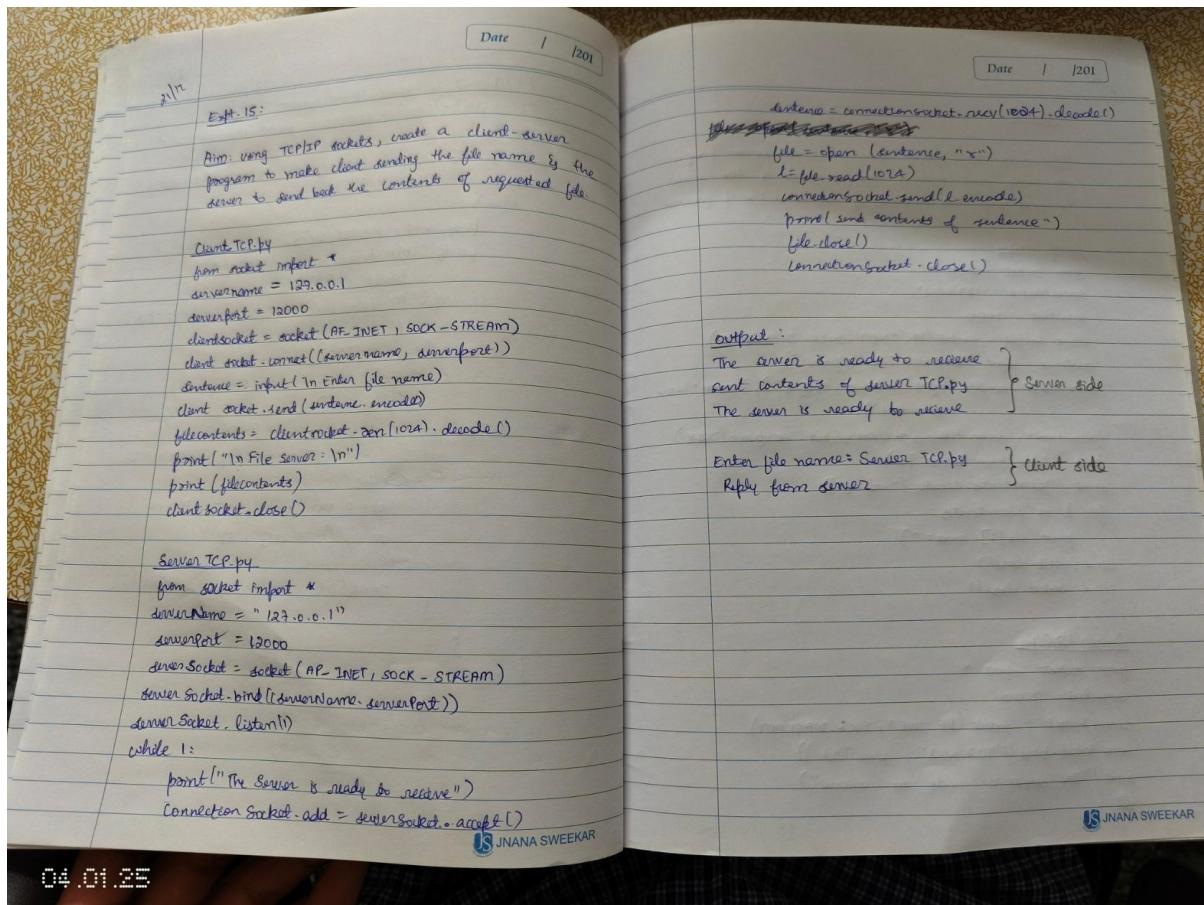
Incoming Packet size: 15
Bytes remaining to Transmit: 15
Packet of size 15 Transmitted---Bytes Remaining to Transmit: 0

Incoming Packet size: 93
Incoming packet size (93bytes) is Greater than bucket capacity (85bytes)-PACKET REJECTED

*** Code Execution Successful ***

```

Expt. 15 – 21/12/2024



04.01.25

Code:

ServerTCP.py

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

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("\n From Server: \n")
```

```
print(filecontents)
clientSocket.close()
```

Output:

The screenshot shows a code editor with two tabs: "ClientTCP.py" and "ServerTCP.py". The "ClientTCP.py" tab is active, displaying the following code:

```
from socket import *
serverName='127.0.0.1'
serverPort=12888
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("\n From Server: \n",filecontents)
print(filecontents)
clientSocket.close()
```

The "ServerTCP.py" tab shows the server's code:

```
from socket import *
serverName="127.0.0.1"
serverPort=12888
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()
```

A terminal window at the bottom shows the execution of the programs:

```
(base) bhug@Bhuvanas-MacBook-Pro ~ LAB % python ServerTCP.py
The server is ready to receive
Sent contents ofServerTCP.py
The server is ready to receive
(base) bhug@Bhuvanas-MacBook-Pro ~ LAB % python ClientTCP.py
Enter file name :ServerTCP.py
From Server:
from socket import *
serverName="127.0.0.1"
serverPort=12888
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()
(base) bhug@Bhuvanas-MacBook-Pro ~ LAB %
```

The screenshot shows a terminal window with two tabs: "ServerTCP.py" and "ClientTCP.py".

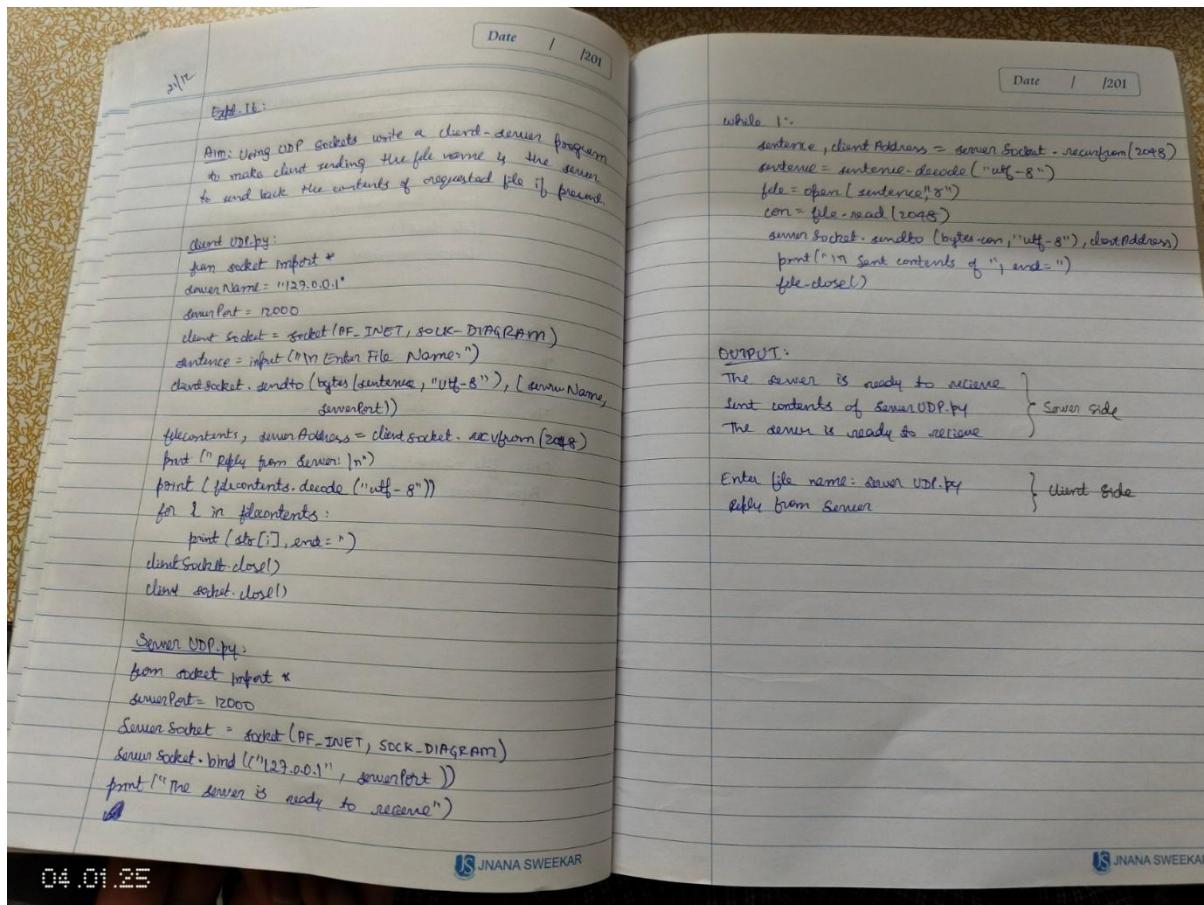
ServerTCP.py:

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

ClientTCP.py:

```
(base) bhug@Bhuwanas-MacBook-Pro AI LAB % python ClientTCP.py
Enter file name :ServerTCP.py
From Server:
from socket import *
serverName="127.0.0.1"
serverPort=12000
serverSocket=socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1)
while 1:
    print("The server is ready to receive")
    connectionSocket,addr=serverSocket.accept()
    sentence=connectionSocket.recv(1024).decode()
    file=open(sentence,"r")
    l=file.read(1024)
    connectionSocket.send(l.encode())
    print("\n Sent contents of "+sentence)
    file.close()
    connectionSocket.close()
```

Expt. 16 – 21/12/2024



Code:

ServerUDP.py

```
from socket import *
serverPort=12000
serverSocket=socket(AF_INET,SOCK_DGRAM)
serverSocket.bind(("127.0.0.1",serverPort))
while 1:
    print("The server is ready to receive")
    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 "+sentence)
    file.close()
```

ClientUDP.py

```
from socket import *
serverName="127.0.0.1"
serverPort=12000
clientSocket=socket(AF_INET,SOCK_DGRAM)
sentence=input("\n Enter File Name:")
clientSocket.sendto(bytes(sentence,"utf-8"),(serverName,serverPort))
filecontents,serverAddress=clientSocket.recvfrom(2048)
print("\n Reply from server: \n")
print(filecontents.decode("utf-8"))
clientSocket.close()
```

Output:

```
ClientUDP.py ServerUDP.py
ServerUDP.py > ...
2 serverPort=12000
3 serverSocket=socket(AF_INET,SOCK_DGRAM)
4 serverSocket.bind(("127.0.0.1",serverPort))
5 while 1:
6     print("The server is ready to receive")
7     sentence,clientAddress=serverSocket.recvfrom(2048)
8     sentence=sentence.decode("utf-8")
9     file=open(sentence,"r")
10    con=file.read(2048)
11    serverSocket.sendto(bytes(con,"utf-8"),clientAddress)
12    print("\n Sent contents of "+sentence)
13    file.close()
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
(base) bhug@Bhuvanas-MacBook-Pro AI LAB % python ClientUDP.py
The server is ready to receive
Sent contents of ServerUDP.py
The server is ready to receive
zsh +
```

```
(base) bhug@Bhuvanas-MacBook-Pro AI LAB % python ClientUDP.py
Enter File Name:ServerUDP.py
Reply from server:
from socket import *
serverPort=12000
serverSocket=socket(AF_INET,SOCK_DGRAM)
serverSocket.bind(("127.0.0.1",serverPort))
while 1:
    print("The server is ready to receive")
    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 "+sentence)
    file.close()
(base) bhug@Bhuvanas-MacBook-Pro AI LAB %
```

```
ClientUDP.py ServerUDP.py
ClientUDP.py > [e] serverAddress
1 from socket import *
2 serverName="127.0.0.1"
3 serverPort=12000
4 clientSocket=socket(AF_INET,SOCK_DGRAM)
5
6 sentence=input("\n Enter File Name:")
7
8 clientSocket.sendto(bytes(sentence,"utf-8"),(serverName,serverPort))
9
10 filecontents,serverAddress=clientSocket.recvfrom(2048)
11 print("\n Reply from server: \n")
12 print(filecontents.decode("utf-8"))
13 clientSocket.close()
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
(base) bhug@Bhuvanas-MacBook-Pro AI LAB % python ClientUDP.py
The server is ready to receive
Sent contents of ServerUDP.py
The server is ready to receive
zsh +
```

```
(base) bhug@Bhuvanas-MacBook-Pro AI LAB % python ClientUDP.py
Enter File Name:ServerUDP.py
Reply from server:
from socket import *
serverPort=12000
serverSocket=socket(AF_INET,SOCK_DGRAM)
serverSocket.bind(("127.0.0.1",serverPort))
while 1:
    print("The server is ready to receive")
    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 "+sentence)
    file.close()
(base) bhug@Bhuvanas-MacBook-Pro AI LAB %
```

Expt. 17 – 21/12/2024

Date / /201

Expt. 17: Tool Exploration - Wireshark

Key Features:

1. **Packet Capture** = Captures live network traffic from various interfaces (e.g. ethernet, wifi)
2. **Protocol Analysis** = supports hundreds of protocols (e.g. TCP, UDP, HTTP, FTP)
3. **Filtering** = offers powerful filters to isolate specific packets or traffic types.
4. **Visualization** = Displays packet details with hierarchical layers (ethernet, IP, TCP/UDP)

Common Filters:

- http : show only HTTP Traffic
- TCP Port == 80 : show traffic on TCP Port 80
- IP address == 192.168.1.1 : show packets to or from a specific IP address
- UDP : show only UDP Traffic