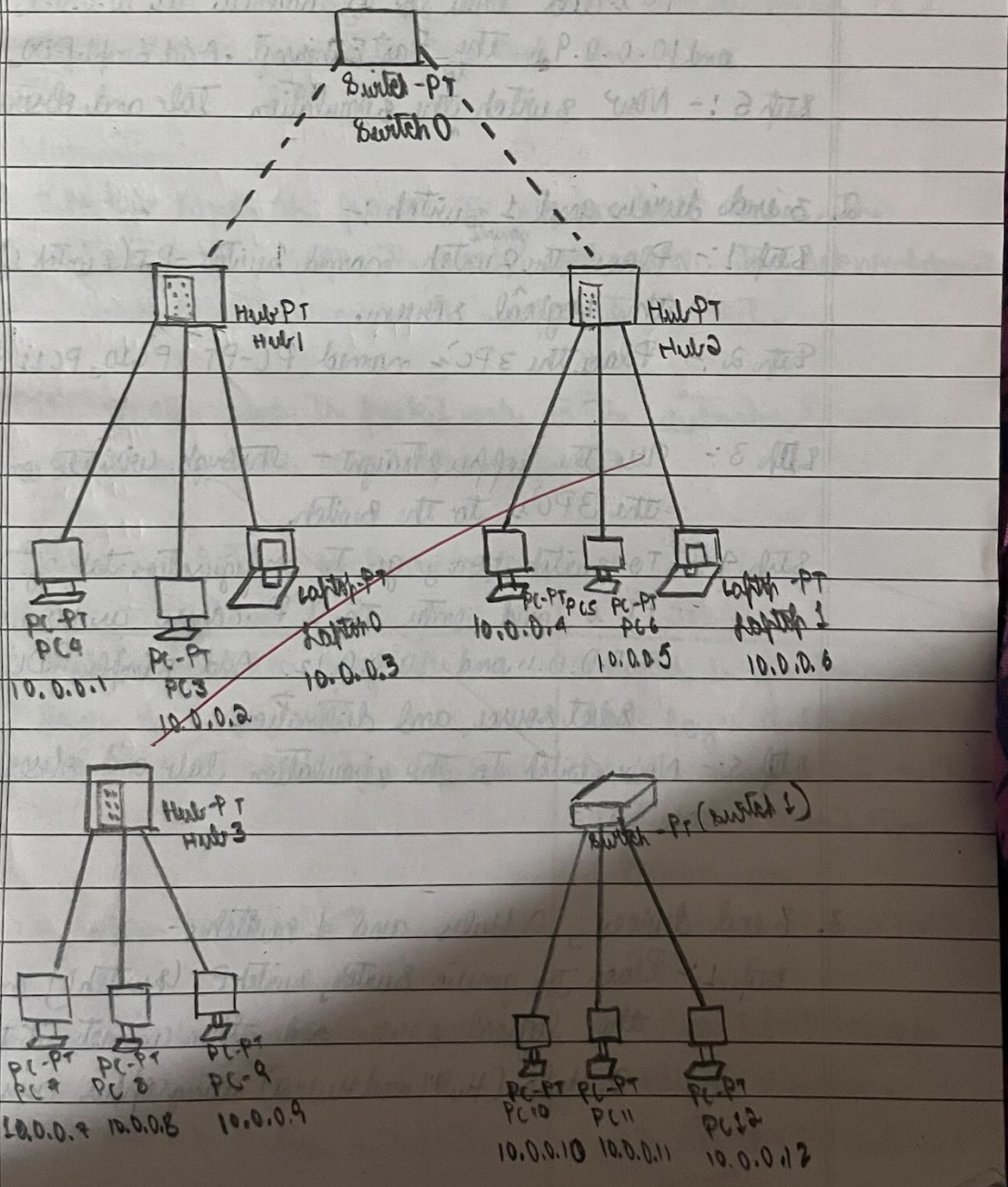
PAGE NO :
DATE :

9.10.2024

Experiment 1

Objective :- To create simple network consisting Transmission of simple PDU using hub and switch as connecting devices.

Configurations :- 3 PC's, PC_7, PC_8, PC_9 connected to Hub 0
3 PC's, PC_{10}, PC_{11}, PC_{12} connected to Switch 1
4 PC's, 2 Laptops connected to 2 Hubs which are connected to Switch.



PAGE NO :
DATE :

Procedure:-

1. 3 end devices and 1 hub:-

Step 1:- Place the Hub named Hub-PT (Hub 3) on the Logical screen.

Step 2:- Place the 3 PC's named PC-PT PC7, PC8, PC9.

Step 3:- Use the copper straight-through wire to connect the 3 PC's to the Hub.

Step 4:- To switch it on, go to the configuration tab of the PC's and enter the IP address as 10.0.0.7, 10.0.0.8, and 10.0.0.9 in the FastEthernet. Add simple PDU, select source and dest.

Step 5:- Now switch the simulation tab and observe.

2. 3 end devices and 1 switch:-

Step 1:- Place the generic Switch named Switch-PT (Switch 1) on the Logical screen.

Step 2:- Place the 3 PC's named PC-PT PC10, PC11, PC12

Step 3:- Use the copper straight-through wire to connect the 3 PC's to the Switch.

Step 4:- To switch it on, go to configuration tab of the PC's and enter the IP address as 10.0.0.10, 10.0.0.11 and 10.0.0.12. Add simple PDU, select source and destination.

Step 5:- Now switch to the simulation tab and observe.

3. 3 end devices, 2 Hubs and 1 switch:-

Step 1:- Place the generic Switch, switch-PT (Switch 0) on the logical screen and then connect it to 2 Hubs (Hub 1 and Hub 2) using copper cross-over

PAGE NO :
DATE :

wire.

Step 2:- Place 4 PC's named PC1, PC2, PC3, PC4 and 2 Laptop's Laptop 1, Laptop 2.

Step 3:- Use the copper straight - Through wire to connect the 4 PC's and 2 Laptops to Hubs 1 and 2.

Step 4:- Go to the configuration tab of the PC's and enter the IP address as 10.0.0.1, 10.0.0.2, 10.0.0.3, 10.0.0.4, 10.0.0.5 and 10.0.0.6 respectively.

Step 5:- Add simple PDU's, Select source and destination PC's.

Step 6:- Go to simulation tab and observe.

Observation:-

1. The hub sends the packet to all available devices. The destination PC accepts the packet and sends the acknowledgement back. All the remaining PC's reject the packet.

2. The switch sends the packet only to the destination PC which accepts and sends back acknowledgement.

Difference between Hub and Switch:-

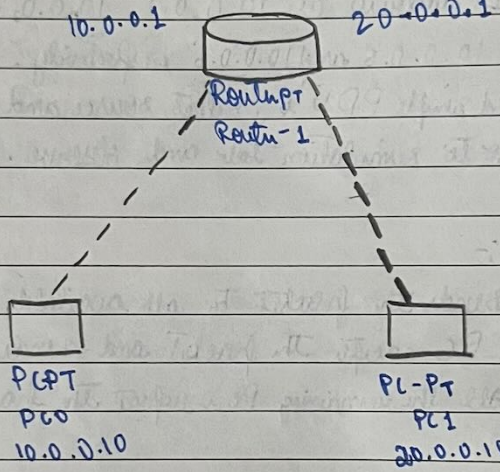
Hub	Switch
→ Hub is operated on physical layer of OSI model.	→ Switch is operated on Data link layer of OSI model.
→ Hub is a broadcast type transmission.	→ Switch is a unicast, multicast and broadcast type transmission.
→ Hubs have 4/12 ports	→ Switch can have 24 to 48 ports.
→ There is only one collision domain.	→ Different ports have own collision domain.
→ Hub can't be used as a repeater.	→ Switch can be used as a repeater.

PAGE NO :
DATE :

Exp-2

Objective :- To create simple network consisting of 2 PC's connected to the router facilitating communication between the two PC's through routers.

Topology :- 2 PC's are connected to the router using copper cross-over.



Procedure :-

1) Step 1 :- Place 2 generic PC's and one generic router. Connect both PC's to the router's fast ethernet ports using copper-crossover wire.

Step 2 :- Select PC1 config → FastEthernet 0 set IP address as 10.0.0.10 and default gateway 10.0.0.1. Similarly for PC2 IP 20.0.0.10 and default gateway 20.0.0.1

PAGE NO :
DATE :

Step 3:- Select router and go to the CLI, execute the following commands.

```
> enable
# config Terminal
# interface FastEthernet 0/0
# ip address 10.0.0.1 255.0.0.0
# no shutdown
# exit
```

Notice PC0 and router0 are successfully connected.

Now run for PC1

```
# interface fastEthernet 1/0
# ip address 20.0.0.1 255.0.0.0
# no shutdown.
```

PC1 is also successfully connected.

Step 4:- Select PC0 → Desktop → Command Prompt & ping PC1 by running command

→ ping 20.0.0.10

Observe the output.

Observation:-

PC successfully ping PC1 with 32 bytes of data

Ping statistics :-

Packets :-

Sent = 4, Received = 4, Lost = 0

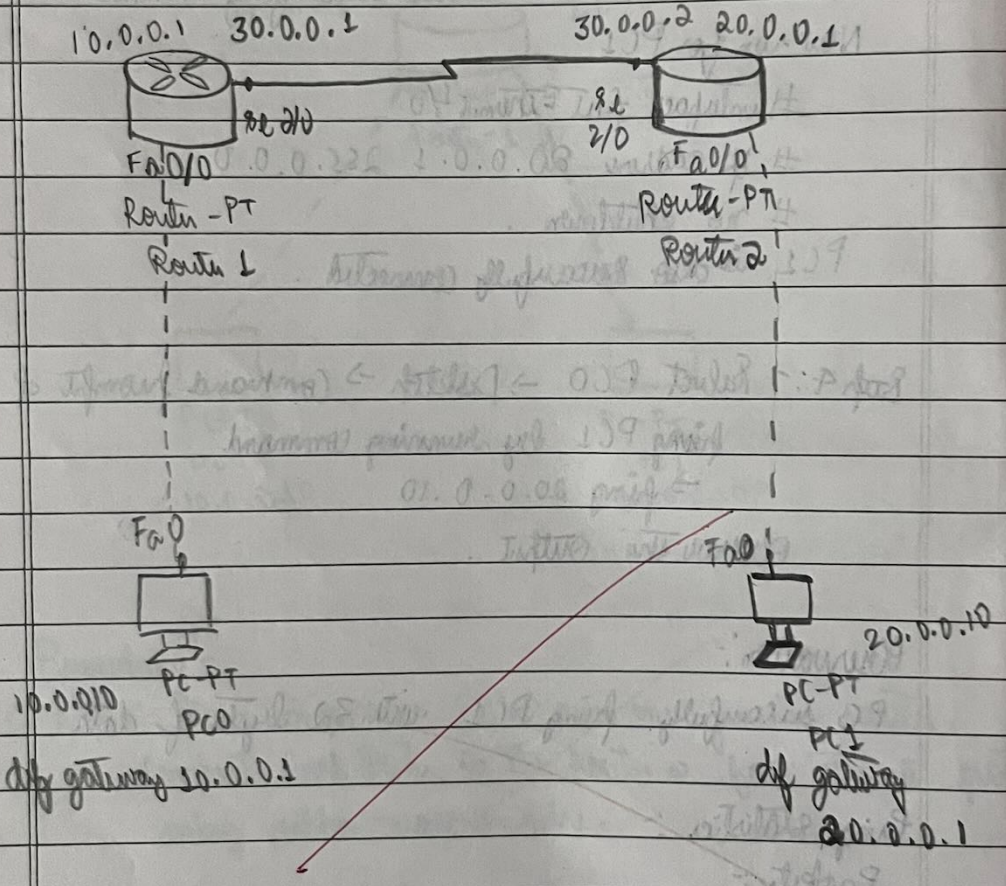
16/01/24

PAGE NO :
DATE :

Experiment 3

Objective :- Configure default routes, static routes to the routers. To connect 2 PC's on 2 different networks using two routers.

Topology :-



Procedure:-

1. Connect two PC's to the two different routers using copper cross-over. Connect two routers to each other using serial DCE.

2. Configure PC 0 to ip 10.0.0.10 and default gateway 10.0.0.1 and PC 1 to ip 20.0.0.60 and default

gateway 20.0.0.1.

③ Select router 0 → CLI and execute the following commands
> enable
config terminal
interface FastEthernet 0/0
ip address 10.0.0.1 255.0.0.0
no shut
notice PC0 & router 0 are successfully connected
Repeat the same procedure for router 1 with ip address 20.0.0.1 255.0.0.0
notice PC1 & router 1 are also successfully connected.

④ Go to router 0 → CLI config terminal and execute the following
interface serial 2/0
ip address 30.0.0.1 255.0.0.0
no shut
go to router 1 → CLI config terminal and execute
interface serial 2/0
ip address 30.0.0.2 255.0.0.0
no shut
notice router 0 & router 1 are successfully connected

⑤ Go to pc0 → Desktop → Command prompt and run
Ping 20.0.0.10
Ping 30.0.0.2
Ping 30.0.0.1
and observe the outputs

PAGE NO :
DATE :

23.10.24

Observation :-

1. All PC's and routers are connected successfully
2. Ping 20.0.0.10 and ping 30.0.0.10 are unsuccessful and show destination host unreachable. This happens because they are not neighbour network with 10.0.0.1
3. Ping 30.0.0.2 will be successful.

23/10/24

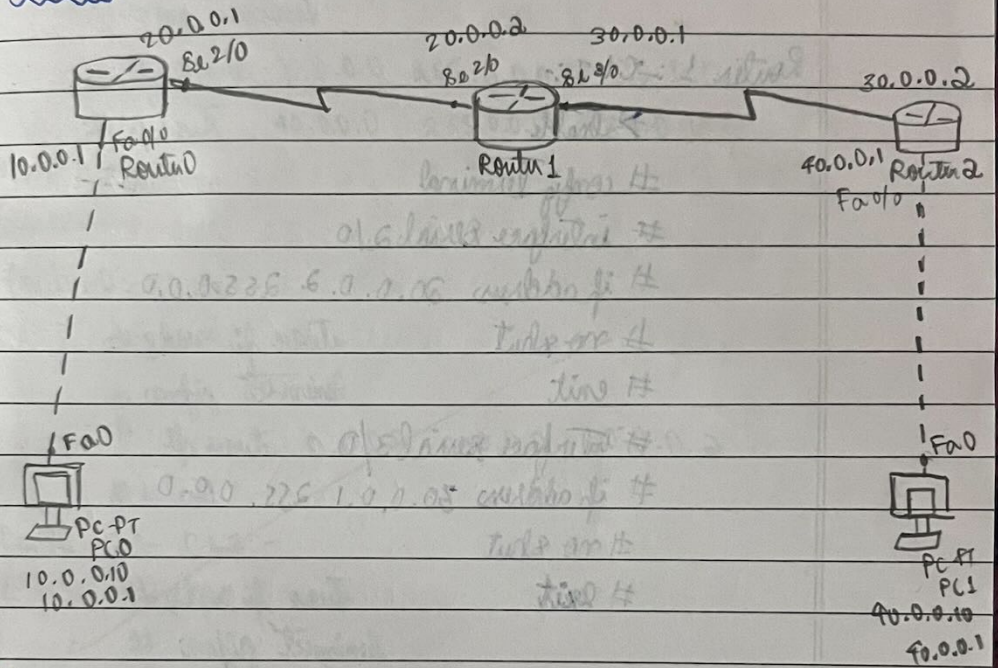
PAGE NO :
DATE :

23.10.24

Experiment 4

Objective:- Configure one static routing and two default routing using three routers and 2 PC's.

Topology :-



Procedure:-

Step 1:- Place 2 PC's and 3 generic routers. Connect PC0 Router0 & PC1, Router2 with copper cross over wire. Router0, Router1 & Router1, Router2 with serial DCE.

Step 2:- Config PC0 to ip 10.0.0.10 & gateway 10.0.0.1
Config PC1 to ip 40.0.0.10 & gateway 40.0.0.1

Router0 → CLI

Step 3:- Router > enable
config terminal
interface FastEthernet0/0

PAGE NO :
DATE :

if address 10.0.0.1 255.0.0.0
no shut
exit
config terminal interface serial 2/0
if addrss 20.0.0.1 255.0.0.0
no shut
exit

Router 1 :- CLI :-

> enable
config terminal
interface serial 2/0
if address 20.0.0.2 255.0.0.0
no shut
exit
interface serial 3/0
if address 30.0.0.1 255.0.0.0
no shut
exit

Router 2 :- CLI :-

> enable
config terminal
interface serial 2/0
if address 30.0.0.2 255.0.0.0
no shut
exit
interface FastEthernet 0/0
if address 40.0.0.1 255.0.0.0
no shut
exit

Connection b/w (R0, R1), (R1, R2), (PC1, R0) and (PC2, R2) will be

PAGE NO :
DATE :

Turn green.

Step 4:-

Router 1:- CLI :-

show ip route
C 20.0.0.0
C 30.0.0.0
config terminal
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
exit

Router 0 :- CLI :-

show ip route
config terminal
ip route 0.0.0.0 0.0.0.0 20.0.0.2
exit

Router 2 :- CLI :-

show ip route
config terminal
ip route 0.0.0.0 0.0.0.0 30.0.0.1
exit

Step 5:-

Show IP route in the router
Ping 40.0.0.10 from 10.0.0.10 and vice versa.
~~Ping 10.0.0.10~~

Observation :-

(i) All the connection are successful
(ii) PC1 → Desktop → command prompt
Ping 40.0.0.10 is successful

PAGE NO :
DATE :

Ping statistics
Sent = 4, Received = 4, Lost = 0 (0% loss)

(iii) IP route in R_1

S 10.0.0.0 via 20.0.0.1
C 20.0.0.0 directly
C 30.0.0.0 directly
S 40.0.0.0 via 30.0.0.2

(iv) Ping 40.0.0.10
Reply from 40.0.0.10 : bytes = 32 time = 8 ms TTL = 125
Reply from 40.0.0.10 : bytes = 32 time = 6 ms TTL = 125
Reply from 40.0.0.10 : bytes = 32 time = 9 ms TTL = 125
Reply from 40.0.0.10 : bytes = 32 time = 7 ms TTL = 125

?

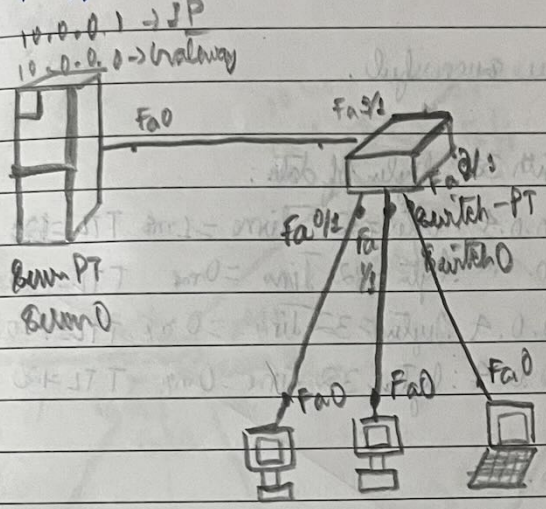
23/10/24

PAGE NO :
DATE :

Experiment 5

Objective :- Design a DHCP within LAN.

Topology :-



Procedure :-

Step 1 :- Place 3 PCs, 1 Server and one Switch and connect all end devices to the switch using copper straight wire.

Step 2 :- Go to Server PT → Desktop → IP configuration

I/p address - 10.0.0.1

Default Gateway - 10.0.0.0

Step 3 :- In Server pt go to config → SERVICES → DHCP

Turn service to On

Pool name :- Switch 1

Default Gateway :- 10.0.0.0

Start IP : 10.0.0.3

Max no of users :- 100

Add

PAGE NO :
DATE :

Step 4 :- Go to each PC desktop → IP configuration and change IP config from static to DHCP. The IP addresses will be assigned automatically.

Step 5 :- Ping from PC0 to PC3.

Observation :-

① All connections are successful.

② 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 = 120

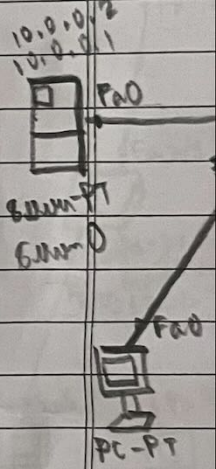
Reply from 10.0.0.4 : bytes = 32 Time = 0ms TTL = 120

Reply from 10.0.0.4 : bytes = 32 Time = 0ms TTL = 120

Reply from 10.0.0.4 : bytes = 32 Time = 0ms TTL = 120

Ping Statistics for 10.0.0.4

Packets : Sent = 4 ; Received = 4 , Lost = 0

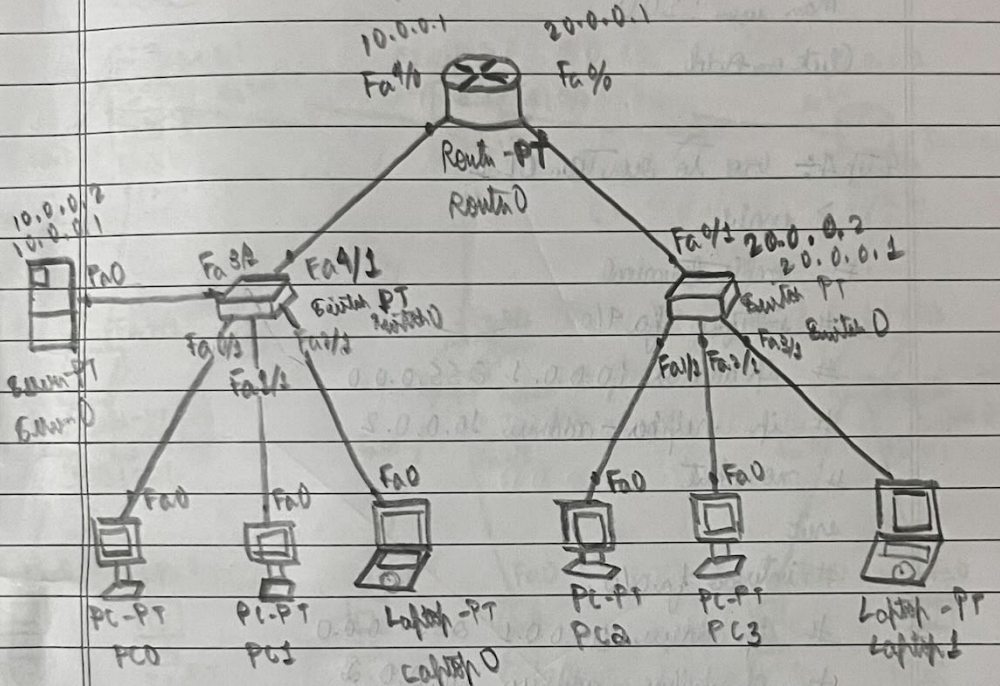


PAGE NO :
DATE :

Experiment-6

Objective :- Design DHCP outside LAN

Topology :-



Procedure :-

Step 1 :- Place 6 PC's, 2 switch, 1 server, 1 router and connect them as shown in the figure.

Step 2 :- Server → Desktop → IP configuration
IP address - 10.0.0.2
Default Gateway - 10.0.0.1

Step 3 :- Config → Service → DHCP, Turn service to ON
Pool name :- Switch 1
Default gateway :- 10.0.0.1

PAGE NO :
DATE :

Start IP: 10.0.0.3
Man asm: 100
→ Click on Add.

Pool name: switch 2
Default Gateway: 20.0.0.1
Start IP: 20.0.0.3
Man asm: 100
Click on Add

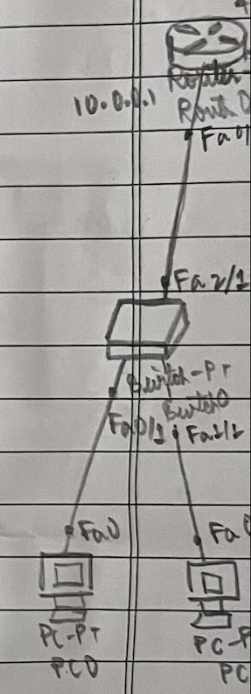
Step 4:- Go To router CLI
> enable
config terminal
interface fa 4/0
ip address 10.0.0.1 255.0.0.0
ip helper-address 10.0.0.2
no shut
exit
interface fa 0/0
ip address 20.0.0.1 255.0.0.0
ip helper-address 10.0.0.2
no shut

All router switch connections go up

Step 5:- Go To all 6 PC's and change IP configuration from static To DHCP addresses will be automatically assigned.

Step 6:- Ping PC0 To PC6.

Observation:- (1) All connection are successful
(2) All the PC's are assigned.

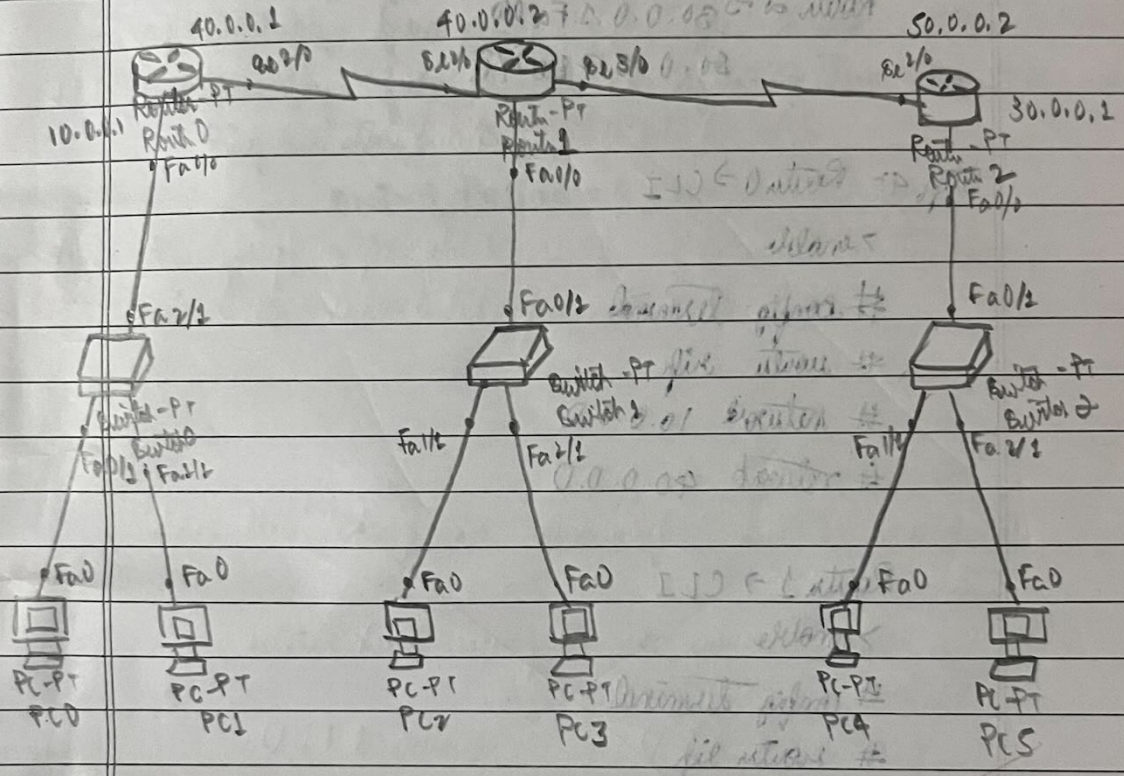


PAGE NO :
DATE :

Experiment - 7

Objective:- configure Routing Information Protocol in Router.

Topology :-



Procedure:-

Step 1:- Place 3 routers, 3 switches and 6 PC's and connect them using their respective cables as shown above.

Step 2:- Configure IP addresses for all end devices 10.0.0.2, 10.0.0.3, 20.0.0.2, 20.0.0.3, 30.0.0.2, 30.0.0.3 and gateways 10.0.0.1, 20.0.0.1, 30.0.0.1

PAGE NO :
DATE :

Step 3:- Configure IP address of all routers

Router 0 → 10.0.0.1 Fa0/0

→ 40.0.0.1 Se2/0

Router 1 :- 20.0.0.1 Fa0/0

40.0.0.2 → Se2/0

50.0.0.1 → Se3/0

Router 2 → 30.0.0.1 Fa 0/0

50.0.0.2 Se2/0

Step 4:- Router 0 → CLI

> enable

config terminal

router rip

network 10.0.0.0

network 40.0.0.0

Router 1 → CLI

> enable

config terminal

router rip

network 20.0.0.0

network 40.0.0.0

network 50.0.0.0

Router 2 → CLI

> enable

config terminal

router rip

network 30.0.0.0

network 50.0.0.0

PAGE NO :
DATE :

Step 5:- show IP route in any router and ping any 2 end devices

Observation:-

1. All devices are connected successfully.
2. Ping 30.0.0.3 from PC0,

Reply from 30.0.0.3 bytes=32 time=7ms TTL=125
Reply from 30.0.0.3 bytes=32 time=8 ms TTL=125
Reply from 30.0.0.3 bytes=32 time=10ms TTL=125
Reply from 30.0.0.3 bytes=32 time=7ms TTL=125

Ping statistics:-

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

PAGE NO :
DATE :

Experiment - 8

Objective :- Demonstrate the TTL / life of a packet.

Procedure :-

Step 1 :- Construct the previous topology.

Step 2 :- Create a simple PDU source : PC0 and destination :- PC5. Go to simulation panel by switching it.

Step 3 :- Click Auto capture / Play and select the packet when it reaches router 0 check the value of TTL inbound PDU, outbound PDU.

Step 4 :- Continue to press Auto capture / play and check the same when packet reaches all routers.

Observation :-

⇒ Routers contain information on packet till level 3

⇒ The value decreases as it passes each router

⇒ Router 0 :-

inbound PDU	outbound PDU
TTL : 255	TTL : 254

29/11/24

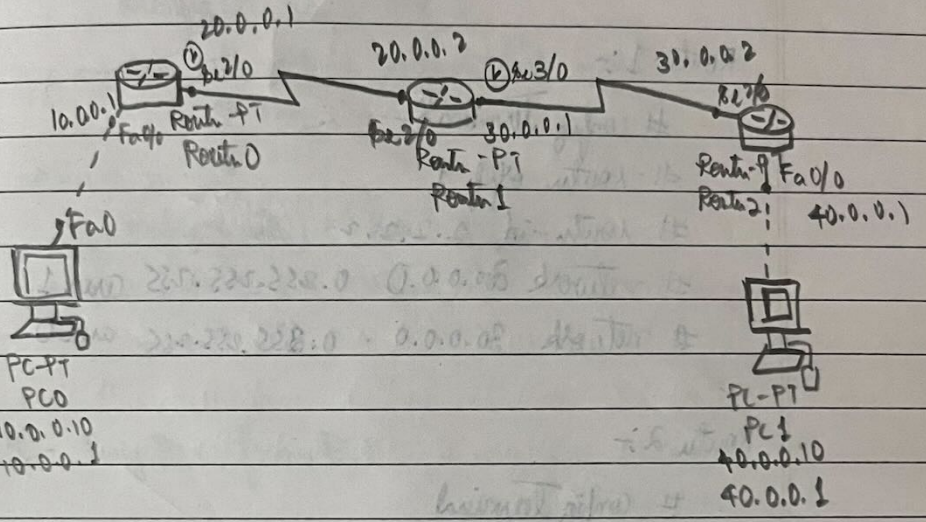
PAGE NO :
DATE :

Experiment - 9

Objective :- Demonstration OSPF protocol

Procedure :- Topology :-

Step 1 :-



Procedure ;-

→ Step 1 :- Place 3 routers & 2 pc's and connect them as shown in the topology.

→ Step 2 :- Configure IP address & gateway according to the topology.

→ Step 3 :- For all serial connection if the clock icon is displayed, in the corresponding routers run :-

interface serial 2/0
(config-if)# encapsulation ppp
(config-if)# clock rate 64000
(config-if)# no shutdown.

if there is no clock, for instance in Se 2/0 of router 2,

interface serial 2/0
(config-if)# encapsulation PPP
(config-if)# no shutdown.

PAGE NO :
DATE :

Step 4 :-

Router 0 :-

config terminal
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

Router 1 :-

config terminal
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

Router 2 :-

config terminal
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

Step 5 :-

Router 0 :-

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

Router 1 :-

(config-if) # interface loopback 0
ip add 172.16.1.253 255.255.0.0
no shutdown

Step 6 :

Step 7 :

Observat
① All conn
② Ping

PAGE NO :
DATE :

Router 2 :-

(config-if) # interface loopback 0
ip add 172.16.1.254 255.255.0.0
no shutdown.

step 6 :-

Router 0 :-

Rconfig)# router ospf 1
(config-router)# area 1 virtual-link 2.2.2.2
exit

Router 1 :-

R1(config-router)# router ospf 1
(config-router)# area 1 virtual-link 1.1.1.1
exit

Step 7 :- Ping PC1 from PC0

Observation :-

(1) All connections are successfull.

(2) Ping 40.0.0.10

reply from 40.0.0.10 bytes : 32 time = 7ms TTL = 125
reply from 40.0.0.10 bytes : 32 time = 6ms TTL = 125
reply from 40.0.0.10 bytes : 32 time = 7ms TTL = 125
reply from 40.0.0.10 bytes : 32 time = 7ms TTL = 125

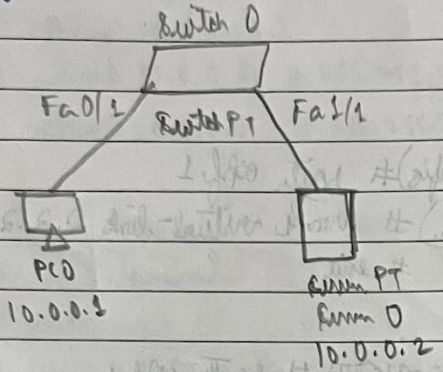
Ping statistics Received = 4
Sent = 4 → lost = 0 (0% loss)

PAGE NO :
DATE :

Exp → 10

Aim :- Configure Web Server, DNS with a LAN

Topology :-



Procedure :-

1. Place 1 PC, 1 server and 1 switch
2. Connect PC with the switch and server with the switch.
3. Set the IP addresses of PC and the server. Also set DNS server for PC
4. Go To Services → HTTP in the server To change the content of web page
5. Go To PC → Desktop → Web Browser.
6. In the URL, type IP address of server To get the content.
7. To configure, go To Server → DNS and give Name, type and address click Add.
8. Again go To the PC → Desktop → Web browser and in the URL type the address.
9. Now the webpage is visible.

PAGE NO :
DATE :

Observation :-

→ Successfully accessed the server web page from PC by entering IP address.

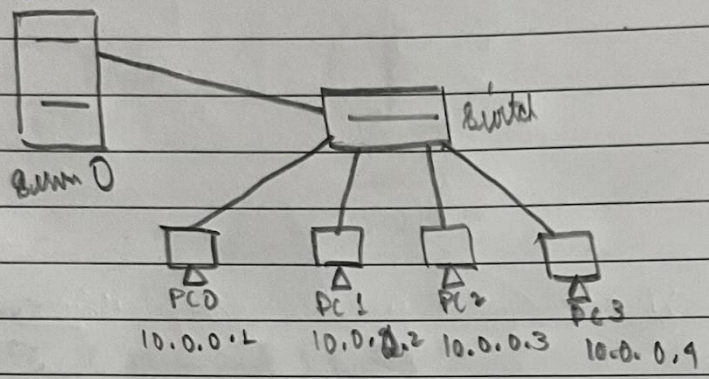
→ Configured web server, DNS with a LAN.

PAGE NO :
DATE :

Exp-11

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

Topology:-



Procedure:-

1. Create a topology of 4 PCs, a server and a switch.
2. Assign IP address to all.
3. Connect them through a switch.
4. Use the simple tool to click on a PC to see the ARP table.
   (command in CLI for same is arp -a)
5. Initially ARP table is empty.
   (Also in CLI of switch, the command show mac address-table can be given on every transaction to see how the switch learns from transaction and builds the address table).
6. Use the capture in the simulation tool to go step by step so that the change in ARP can be clearly noted.

PAGE NO :
DATE :

Observation :-

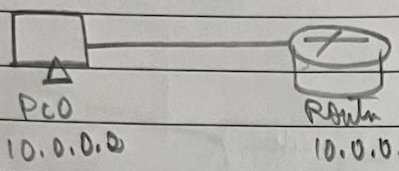
→ Router as well as the node update the ARP table as and when a new communication start.

PAGE NO :
DATE :

Exp-12

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

Topology :-



Procedure :-

* Make a simple topology with one router and one PC.

(ii) Commands in Router :-

> enable
config terminal
hostname R1
enable secret P1
interface fastethernet 0/0
ip address 10.0.0.1 255.0.0.0
no shut

line vty 0 5
login
password p0
exit
> exit
wr To save changes.

PAGE NO :
DATE :

(iii) Commands in PC

ping 10.0.0.1

ping results :-

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

* Password for user access verification in P0
* Password for enable in P1

(iv) Accessing router CLI from PC

> telnet 10.0.0.1

User Access Verification

Password :

R1 > enable

Password :

R1 # show ip route

Codes

Area

Observation :-

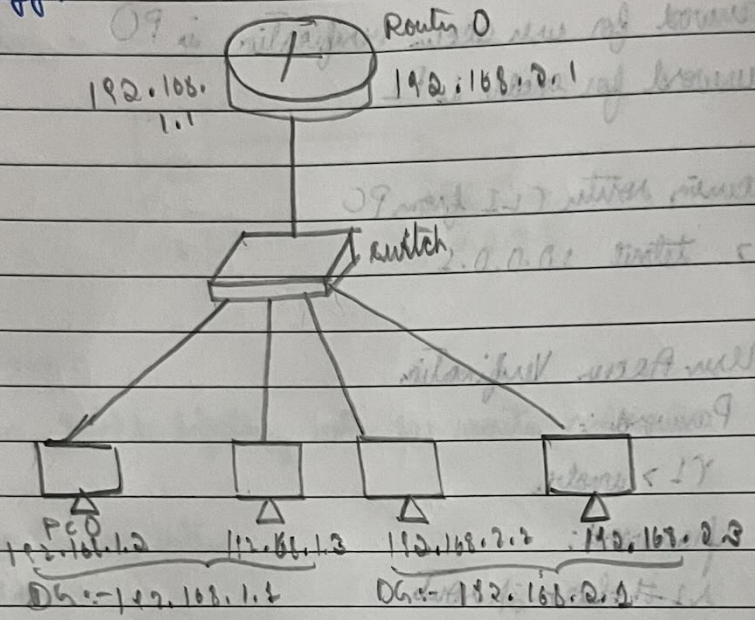
The admin in PC is ~~used~~ able to run commands and see the result from PC.

PAGE NO :
DATE :

Exp-13

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

topology :-



Procedure :-

(1) Create a topology with 4 PC's, router and 1 switch as shown above.

⇒ Assign IP address as shown in the topology.

⇒ Go to the switch ⇒ VLAN Database to configure the VLAN. Give VLAN no and VLAN name, add it.

⇒ Select the interface i.e. fastethernet 4/1 (near the switch from router) and make it trunk.
(VLAN trunking allow switches to forward frames from diff VLAN's over a single link called trunk)

PAGE NO :
DATE :

→ To make router understand VLAN.
Go to config tab of router, select VLAN database, write the no and name of VLAN created.
Go to CLI
Router (vlan) # exit
Router # config terminal
Router (config) # interface fast ethernet 0/0.1
Router (config) #
encapsulation dot1Q 2
ip address 192.168.2.1 255.255.255.0
no shut
exit
exit

Observation:-
→ Proper trunk configuration is established to make VLAN work properly.

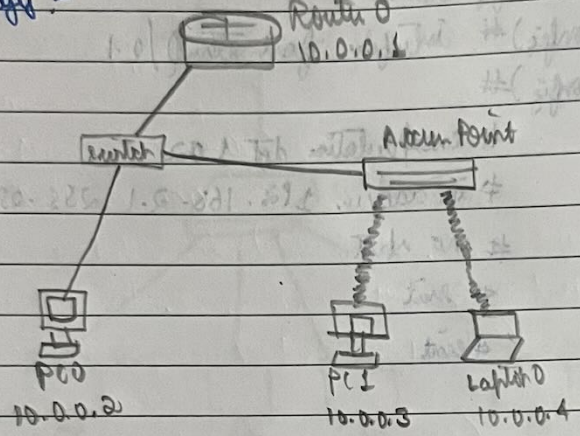
⇒ Ping from one VLAN to another works properly.

PAGE NO :
DATE :

Expt: 14

Objective :- To construct WLAN and make the nodes communicate wirelessly.

Topology :-



Procedure :-

- Construct the above topology.
  Configure PC0 and router as normally done.
- configure access point 1 -> port -> SSID Name
  -> any Name (WLAN here)
  here
- Select WEP and give any 10 digit key
  (eg 1234567890 here)
- Configure PC1 and laptop with wireless standards.
- In PC1,
  Switch off the device. drag the existing PT-HOST-NM-1AM to the component used in LHS. Drag WMP300N wireless interface to the empty port.
  Switch on the device.

- In the config
  added. Now co
  gateway. Co
- Do similar in
- Ping from any

Observation :-

- Device could
  range.
- Signal strength
- Ping is succes

PAGE NO :
DATE :

- In the config tab a new wireless interface would have been added. Now configure SSID, WEP key, IP address and gateway (as normally done) to the device.
- Do similar in laptop.
- Ping from any device to any other device the result.

Observation:-

- Device could connect to WLAN as long as they are in the network range.
- Signal strength decrease with increase in distance.
- Ping is successful.

PAGE NO :
DATE :

Part - B :-

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

Client TCP.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()
```

Server TCP.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()
```

Output :-

server is ready
Requesting
Request sent

Client is
Enter file
Received
Hello World

PAGE NO :
DATE :

```
file = open(sentence, "r")
l = file.read(1024)
connection socket.send(l.encode())
print("\n Sent contents of" + sentence)
file.close()
connection socket.close()
```

Output :-

Server is online
Requesting for file : test.txt
Request sent

Client is connected to server
Enter file name : Test.txt
Received response
Hello World.

PAGE NO :
DATE :

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

Client UDP.py

```
from socket import *
server Name = "127.0.0.1"
server Port = 12000
client Socket = socket (AF_INET, SOCK_DGRAM)
sentence = input ("\n Enter the file name: ")
client Socket.sendto (bytes (sentence, "utf-8"),
                      (server Name, server Port))

file contents, server Address = client Socket.recv
                                from (2048)
print ("\n Reply from server: \n")
print (file contents.decode ("utf-8"))
# for i in file contents:
  # print (str(i), end = '')
client socket.close()
client Socket.close()
```

Server UDP.py

```
from socket import *
server Port = 12000
server Socket = socket (AF_INET, SOCK_DGRAM)
server Socket.bind ("127.0.0.1", server Port)
print ("The server is ready to receive")
while 1:
    sentence, client Address = server Socket.recvfrom
                               (2048)
    sentence = sentence.decode ("utf-8")
```

Output:-

```
// Server O/P
Server is Online
Hello Server

// Client O/P
Hello Client
```

PAGE NO :
DATE :

```
file = open(filename, "a")
con = file.read(2048)

server_socket.sendto(bytes(con, "utf-8"), client_address)
print("\n Sent content of", end='')
print(filename)
# for i in sentence:
# print(str(i), end='')
file.close()
```

Output :-
// server O/P
Server is Online
Hello Server

// Client O/P
Hello Client

PAGE NO :
DATE :

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

```
def crc_ccitt_16_bitstream(bit_stream: str,
    poly: int = 0x1021, init_crc: int = 0xFFFF)
-> int:
    crc = init_crc
    for bit in bit_stream:
        crc ^= int(bit) << 15
        for _ in range(1):
            if crc & 0x8000:
                crc = (crc << 1) ^ poly
            else:
                crc <<= 1
            crc &= 0xFFFF
    return crc

def append_crc_to_bitstream (bit_stream: str)
-> str:
    crc = crc_ccitt_16_bitstream (bit_stream)
    crc_bits = f"{crc:016b}"
    return bit_stream + crc_bits

def verify_crc_bitstream( bitstream_with_crc: str) -> bool:

    if len (bit_stream_with_crc) < 16:
        return False
    data, received_crc =
    bit_stream_with_crc[:-16]
    bit_stream_with_crc[-16:]
    Calculated_crc =
    crc_ccitt_16_bitstream(data)
```

Output:-
Enter the message
111101
The Transmitted
Enter the
No error in

PAGE NO :
DATE :

return calculated_crc==
int (received_crc, 2)

Output:-
Enter the message
111101
The Transmitted message is : 11111110101010111
Enter the received message in binary : 111101
No error in data

PAGE NO :
DATE :

4. Leaky Bucket :-

```
def main()
    storage = 0
    no_of_queue = 4
    bucket_size = 10
    input_pkt_size = 4
    output_pkt_size = 1

for _ in range(no_of_queue):

    size_left = bucket_size - storage

    if input_pkt_size <= size_left:
        storage += input_pkt_size
    else:
        print(f"Packet loss = {input_pkt_size}")

    print(f"Buffer size = {storage} out of bucket
                                size = {bucket_
                                size}")

    storage -= output_pkt_size.

if __name__ == "__main__":
    main()
```

Output :-

→ No of queue = 4
Bucket size = 10
input pkt size = 4
output pkt size = 1