Lab-2

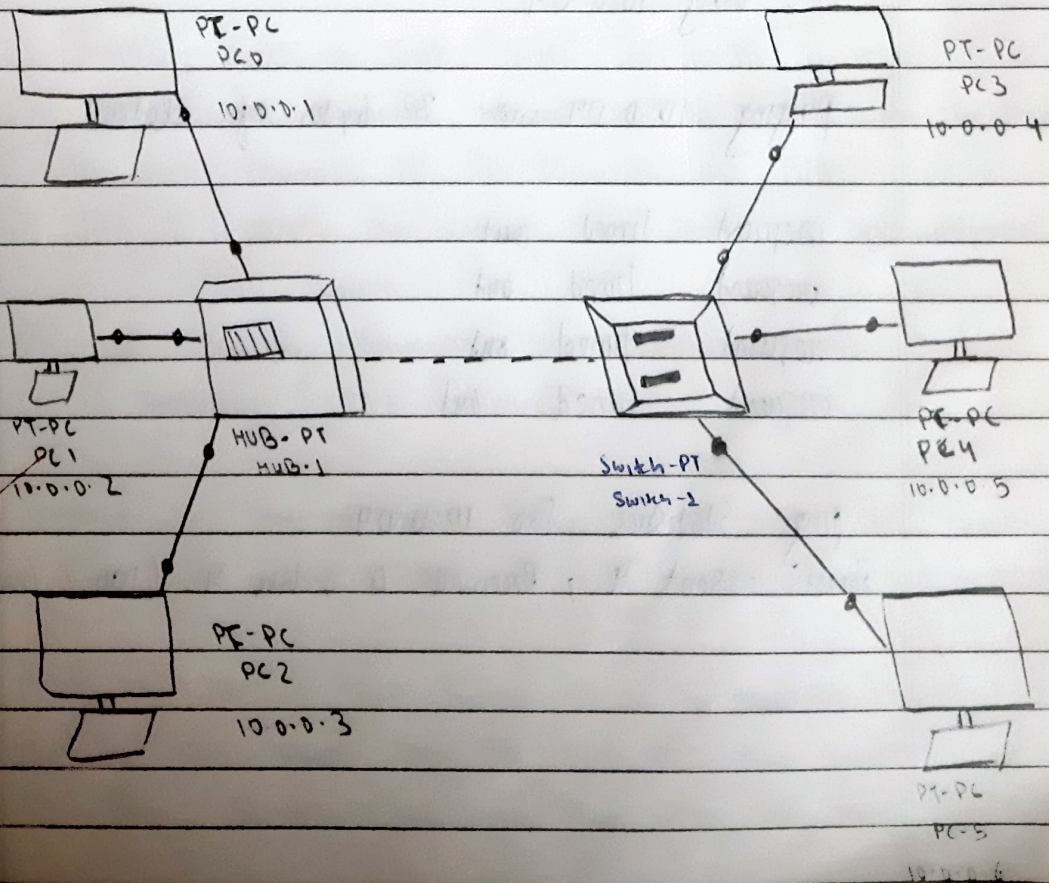
Create a topology and Simulate a simple PDU form source to destination using simple hub and Switch as connecting domains'.

Step 1 : Select end devices and add generic switch and hub to workspace. Add 6 PC-Pt

Step 2 : Make Connections using Copper Straight Cable

Step 3 : open each pc configuration window and change the IP address to 10.0.0.1 10.0.0.2, 10.0.0.3, 10.0.0.4, 10.0.0.5, 10.0.0.6

Step 4 : Save



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=12ms TTL=128
reply from 10.0.0.4 : bytes=32 time=6ms TTL=128
reply from 10.0.0.4 : bytes=32 time=6ms TTL=128
reply from 10.0.0.4 : bytes=32 time=6ms TTL=128

ping statistics for 10.0.0.4
packets: sent=4, received=4, lost=0 (0% loss)

approximate round trip time in milli-seconds
Minimum: 6ms, Maximum=12ms, Average=7ms

PC> Ping 10.0.04

Pinging 10.0.04 with 32 bytes of data:

request timed out.
request timed out.
request timed out.
request timed out.

ping statistics for 10.0.04:
Packets: Sent=4, Received=0, Lost=4 (100% loss)

Scenario 1 : Sending btw PC's Connected to Hub
if message is sent from PC1 to PC2

1) Simple PDU is sent from PC1 to hub.
2) Hub send the copies of the msg to PC0. PC2 and Switch.
3) The PC2 receives the message & sends back acknowledgement to the hub.
4) the hub further forward the acknowledgement to PC0, PC1 & Switch
5) The PC1 receives it and the transfer is complete.

Scenario 2: Sending between PC's connected to Switch if message is sent from PC3 to PC4.

1) Simple PDU is sent from PC3 to Switch
2) The Switch forwards the message to PC5, PC4 and the Hub
3) The Hub forwards it to PC0, PC1 and PC2
4) the PC4 accepts the message and sends an acknowledgement to switch.
5) The Switch forwards the acknowledgement and Similarly PC3 receives it

Scenario 3: Sending btw PC's connected to Switch and Hub. if msg is sent from PC0 to PC4

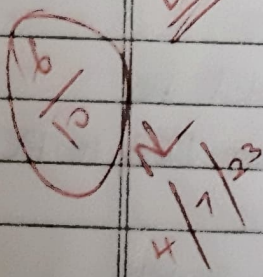
1) Simple PDU is sent from PC0 to ~~switch~~. Hub.
2) the Hub sends copy to PC1, PC2 and switch.
3) The Switch forwards the msg to PC3, PC4, PC5.

4) PC4 receives the msg and sends an acknowledgement to the Switch.

5) The Switch forwards the acknowledgment to the PC3, PC5 and the HUB.

6) Hub sends copy of the acknowlegment to PC0, PC1 and PC4.

Topology?



Steps:

1) Add 2 PCs & connect them to a generic router
2) Configure the PCs by setting their IP address to 10.0.0.1 & 20.0.0.1 respectively.
3) Set the gateway as 10.0.0.2 and 20.0.0.2 respectively
4) Go to cmd line interface in the router and enter 'no' for 'continue' with configuration dialog.
5) Type 'Config terminal' and enter twice
6) Type 'interface fastEthernet 0/0' & enter.
7) Type 'ip address 10.0.0.2 255.0.0.0'
8) Type 'no shut' & the connection is now established.
9) Repeat steps '4 to 8' for the other PC 20.0.0.1
10) Repeat steps '1 to 9' too PC3 and PC4.
11) Connect router 1 & router 2 via a router 3
12) All routers to router connection via Serial DTE and PC to route via Copper cross-over
13) Go to Router 1 'CLI' and type the following

```
enable ↵
Config terminal ↵
interface Serial 2/0 ↵
ip address 50.0.0.2 255.0.0.0 ↵
no shut ↵
```

14) Go to Router 3 'CLI' & type

```
enable ↵
Config t ↵
interface Serial 2/0 ↵
ip address 50.0.0.1 255.0.0.0 ↵
no shut ↵
```

15) Repeat steps (13) and (14) for Router 2 to 3

16) Go to Router 1 'CLI' and type
'show ip route'
It shows only the direct connections.

17) We statically connect router to the PCs by typing the following in the CLI: (for router 1).

ip route 30.0.0.0 255.0.0.0 50.0.0.1 ↵
ip route 40.0.0.0 255.0.0.0 40.0.0.1 ↵

(for router 2):

ip route 10.0.0.0 255.0.0.0 60.0.0.1 ↵
ip route 20.0.0.0 255.0.0.0 60.0.0.1 ↵

(for router 3):

~~(ip route 3):~~

ip route 10.0.0.0 255.0.0.0 50.0.0.1
ip route 20.0.0.0 255.0.0.0 50.0.0.1
ip route 30.0.0.0 255.0.0.0 60.0.0.1
ip route 40.0.0.0 255.0.0.0 60.0.0.1

18) Now data transfer btw PCs is successful.

19) Before statically connecting routers ping btw PCs NOT directly connected was unsuccessful.

PC > ping 30.0.0.1

pinging 30.0.0.1 with 32 bytes of data :

Reply from 20.0.0.2 : Destination host unreachable

" " " " " "

" " " " " "

Reply from 20.0.0.2 : Destination host unreachable

ping statistics for 30.0.0.1

packets : Sent = 4 , Received = 0 , lost = 4 (100% lost)

After statically defining the route :

PC > ping 40.0.0.1

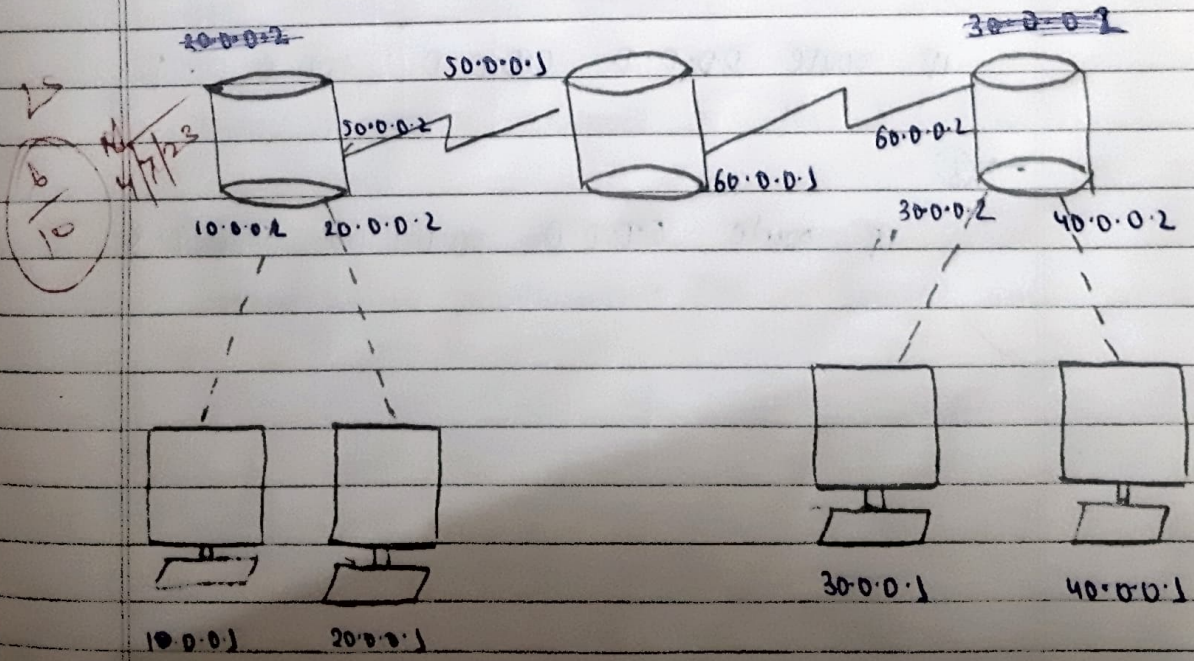
pinging 40.0.0.1 with 32 bytes of data :

Reply from 40.0.0.1 bytes = 32 times time. 2ms TTL: 125

" " " " " " " " " "

" " " " " " " " " "

" " " " " " " " " "



Default route

Lab - 03

Configure default route, static route to the routers

*) Connect devices as shown in Topology
*) Set ip address to all PC and routers

router
enable
Config t
interface fastEthernet Serial 0/0
ip address 10.0.0.1 255.0.0.0
no shutdown

*) Default route for the routers 1, routers 5 routers 3

→ (1)

enable
Config t
ip route 0.0.0.0 0.0.0.0 60.0.0.3

→ (5)

ip route 00.0.0.0 0.0.0.0 100.0.0.1

→ (3)

ip route 0.0.0.0 0.0.0.0 50.0.0.3

Static routing for router 0, router 6 router 4, router 2

ip route 20.0.0.0 255.0.0.0 60.0.0.1
ip route 30.0.0.0 255.0.0.0 70.0.0.1
ip route 40.0.0.0 255.0.0.0 70.0.0.1
ip route 50.0.0.0 255.0.0.0 70.0.0.1
ip route 80.0.0.0 255.0.0.0 70.0.0.1
ip route 100.0.0.0 255.0.0.0 70.0.0.1
ip route 110.0.0.0 255.0.0.0 70.0.0.1

Similarly for rest routers.

output :-

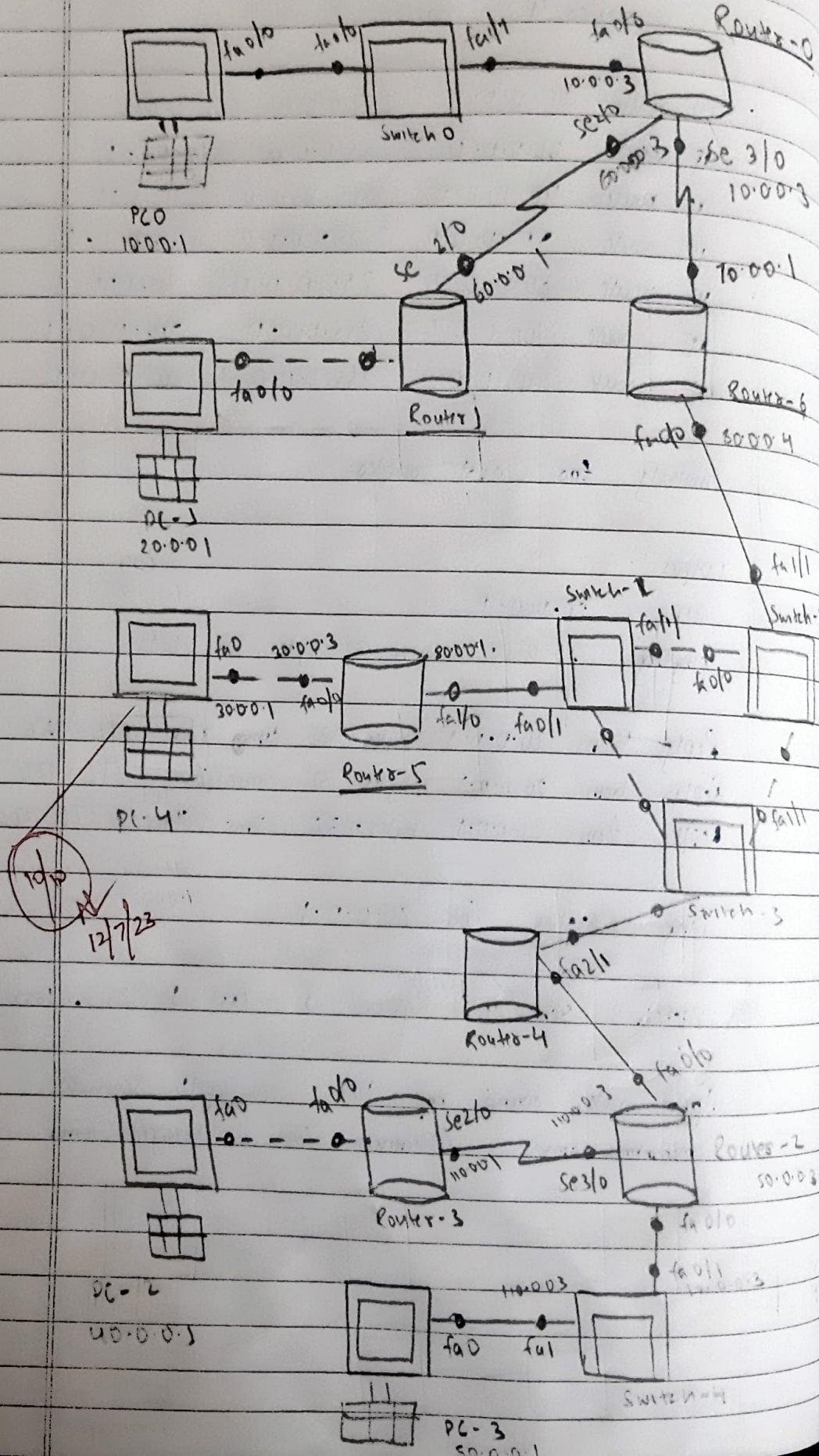
ping 20.0.0.1
pinging 20.0.0.1 with 32 bytes of data

Reply from 20.0.0.1 : bytes: 32 time: 7ms TTL: 126
Reply from 20.0.0.1 : bytes: 32 time: 6ms TTL 126
Reply from 20.0.0.1 bytes = 32 time: 7ms TTL 126

ping statistics for 20.0.0.1

Packets : Sent: 4, Received: 3, Lost: 1 (25% loss)

Approximate round trip time in milli seconds:
minimum: 6ms, Maximum: 7ms, Average: 6ms

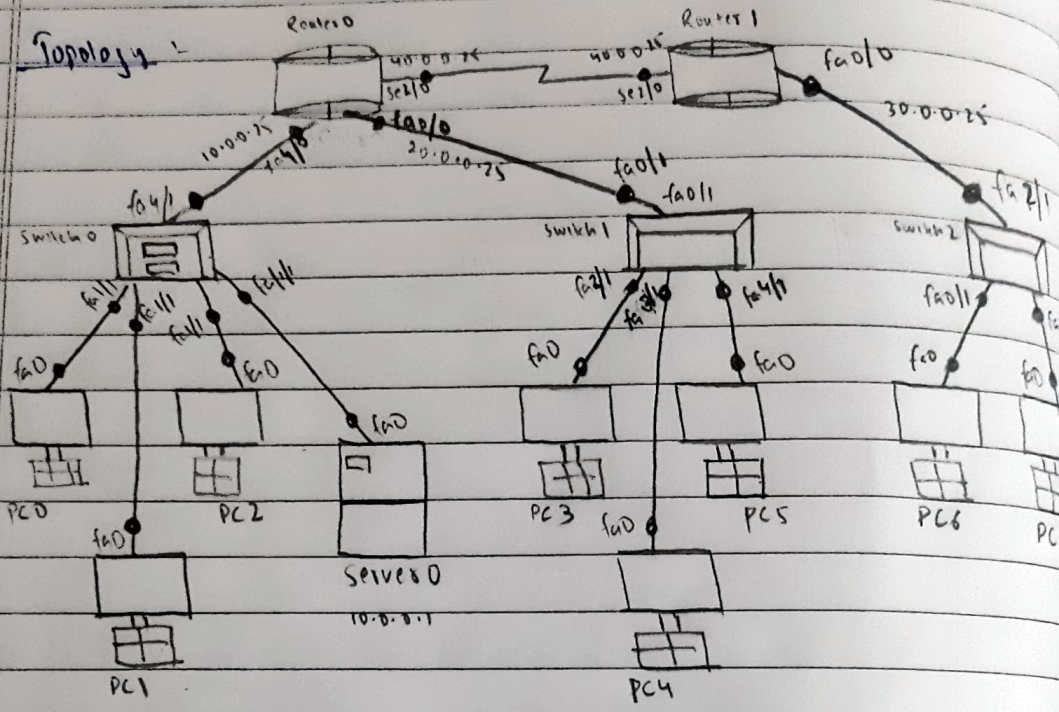

fa0/0
fa0/0
fa1/1
fa0/0
Router-0
10.0.0.3
Switch 0
se2/0
se 3/0
10.0.0.3
PC0
10.0.0.1
se 2/0
60.0.0.1
10.0.0.1
fa0/0
Router1
Router-6
fa0/0
80.0.0.4
PC-1
20.0.0.1
fa1/1
Switch-1
Switch-
fa0
30.0.0.3
80.0.0.1
fa0/1
fa0/0
fa1/0
fa0/1
fa0/0
Router-5
PC-4
fa0/0
12/7/23
fa1/1
Switch-3
fa2/1
Router-4
fa0/0
fa0
fa0
se2/0
110.0.0.3
110.0.0.1
Router-2
Se3/0
50.0.0.3
Router-3
fa0/0
fa0/1
PC-2
40.0.0.1
110.0.0.3
fa0
fa1
Switch-4
PC-3
50.0.0.1

SURYA Gold
Date Page

Lab - 4 14/07/23

Aim :- Configure DHCP within a LAN and outside LAN for DHCP within a LAN.

Topology :-



Procedure :-

Step 1 :- Drag and drop 3 pcs, 1 server and 1 switch and connect them.

~~Step 2~~ :- Create Topology :

In Server

Set IP address 10.0.0.1 (config - setting - gateway - 10.0.0.25)

Services → DHCP → On

Services → DHCP → default Gateway → 10.0.0.25

Service POOL

start IP address 10.0.0.2

Save

SURYA Gold
Date Page

On routers:

Set ip address
- Config t
- interface fa 0/0
- ip address 40.0.0.26 255.0.0.0
- no shut.

- Config t
- interface Serial 2/0
- ip address 30.0.0.26 255.0.0.0
- no shut.

Static routing for 10 and 20 network

- Config t
- ip route 10.0.0.0 255.0.0.0 30.0.0.25
- ip route 20.0.0.0 255.0.0.0 30.0.0.25
- no shut.

→ Setting helper address
- Config t
- Interface fastEthernet 0/0
- ip helper-address 10.0.0.1
- no shut.

10/10

Output :-

On any PC

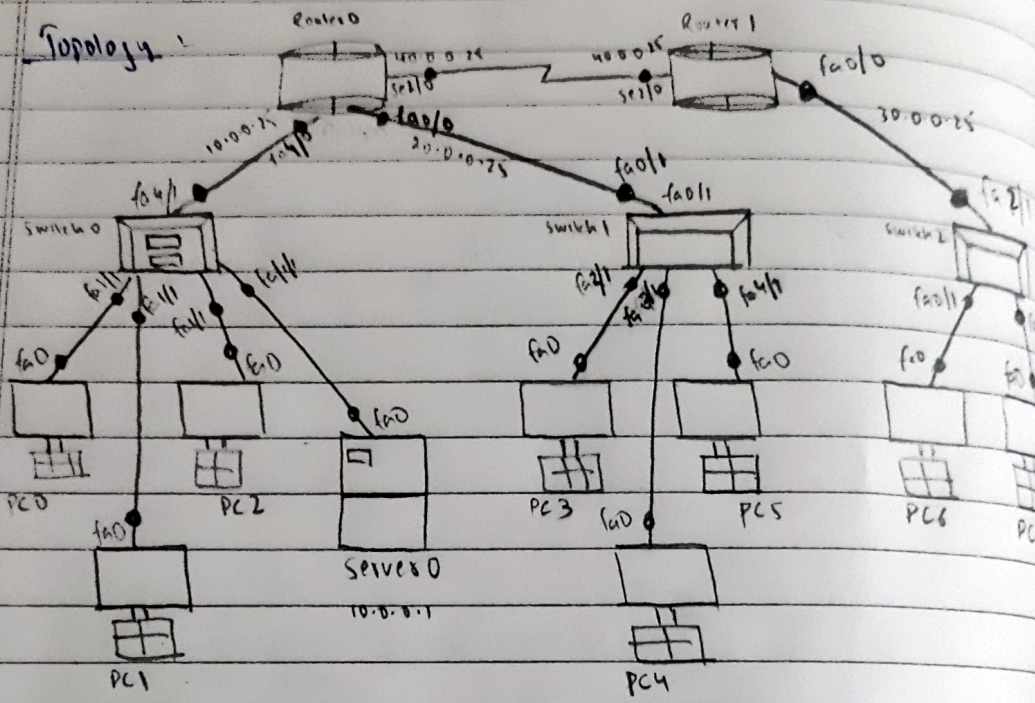
Desktop → IP configuration → DHCP

(Dynamic IP address is assigned to all PC by server)

Lab - 4 14/07/23

Aim :- Configure DHCP within a LAN and outside LAN for DHCP within a LAN.

Topology :-



Procedure :-

Step 1 :- Drag and drop 3 pcs, 1 server and 1 switch and connect them.

~~Step 2~~ :- Create Topology.

In Server

Set IP address 10.0.0.1 (config - setting - gateway - 10.0.0.25)

Services → DHCP → On

Services → DHCP → default Gateway → 10.0.0.25

Service POOL

start IP address 10.0.0.2

Save

Source POOL 1

gateway → 10.0.0.25

start ip address 20.0.0.2

add

Source POOL 2

gateway → 10.0.0.25

start ip address 40.0.0.2

add.

On routers 0 :-

Set two Network ip address

- Config t
- interface fastEthernet 4/0
- ip address 10.0.0.25 255.0.0.0
- no shut.

(Similarly 20.0.0.25)

- Config t
- interface fastEthernet 0/0
- ip address 10.0.0.1
- no shut.

→ Static route (for 40 ip)

- Config t

ip route 40.0.0.0 255.0.0.0 30.0.0.26

In router:

Set ip address
- Config t
- interface fa 0/0
- ip address 40.0.0.26 255.0.0.0
- no shut.

- Config t
- interface Serial 2/0
- ip address 30.0.0.26 255.0.0.0
- no shut.

Static routing for 10 and 20 network

- Config t
- ip route 10.0.0.0 255.0.0.0 30.0.0.25
- ip route 20.0.0.0 255.0.0.0 30.0.0.25
- no shut.

→ Setting helper address
- Config t
- interface FastEthernet 0/0
- ip helper-address 10.0.0.1
- no shut.

10/10

27/7/23

Output :-

In any PC

Desktop → IP configuration → DHCP

(Dynamic IP address is assigned to all PC by

Configure Web-Server, DNS within a LAN

Aim :- To Configure DNS server in LAN

Step 1 :- Create a Topology with 1PC, 1 Server, 1 Switch

Step 2 :- Configure the PC ip address - 10.0.0.10
set ip address and gateway.
Set ip address of Server. ip-address - 10.0.0.20

Step 3 :- Open web-browser in PC
URL → Enter Server IP address
Go back to Server.

Step 4 :- Turn the DNS service 'on' in the Service and add a name and address same ip address

Step 5 :- In HTTP go to index-HTML and using the normal HTML commands add name and USN.

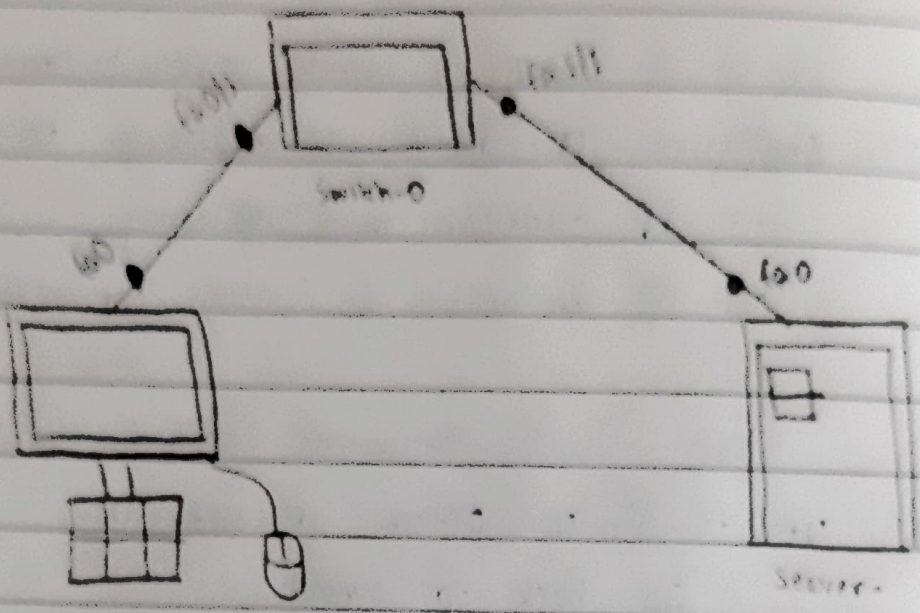
<H1> Rahul </H1>
<H1> 1BM21CS158 </H1>

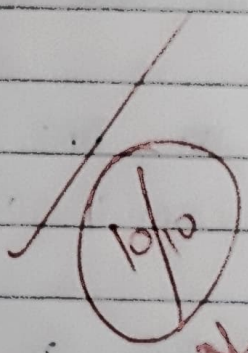
Step 6 :- Go to web browser in PC and enter the URL Same as name of website you saved in Server.

Output :- In web browser in PC and enter the name of Server and we can shown the index.Html page

Rahul
1BM21CS158

Observation :- We can enter name of website instead of its ipaddress

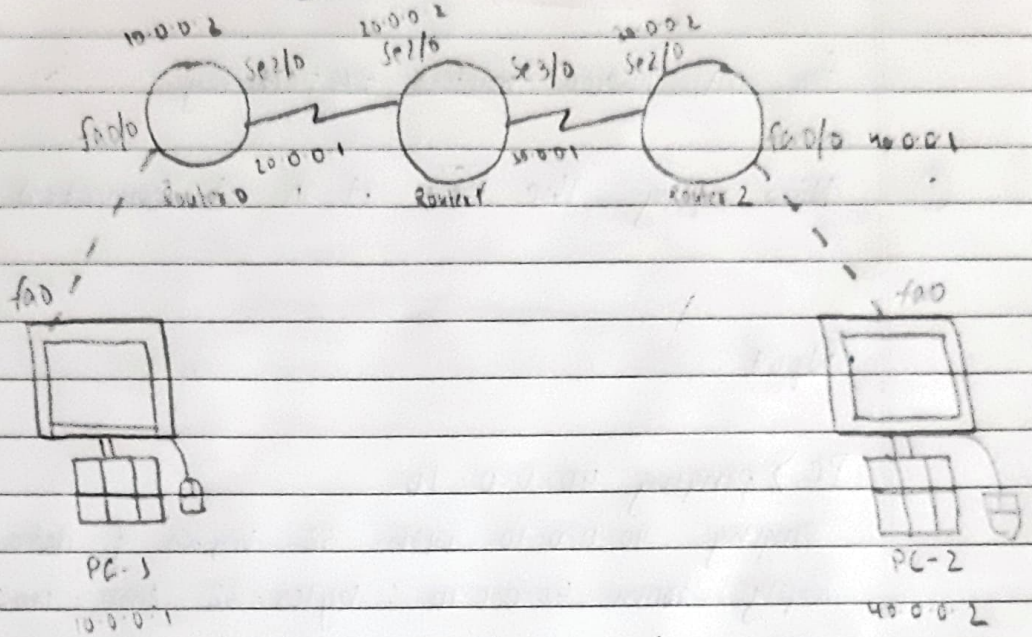




N
23/7/23

② Aim: Configure RIP routing protocol in routers.

Create Topology :-



Steps :-

① Create Topology as shown in the figure.
② Set the ip address for 2 pc's and the routers
③ Configure all the pc's and routers with each other using commands :-

interface Serial
ip address
encapsulation PPP
clock rate 64000

for all respectively.

Now Routes ip

network 10.0.0.10
network 10.0.0.20

for all three routers respectively.

(5) Now ping the PC's it is all connected to:

output :-

PC> pinging 40.0.0.10

pinging 40.0.0.10 with 32 bytes of data.
Reply from 40.0.0.10 : bytes : 32 time : 4ms TTL:12
" " " " " " " "
" " " " " " " "
" " " " " " " "

Pinging statistics for 40.0.0.10 :

packets : Sent : 4 ; Received : 4 ; Lost : 0 (0% loss)
Approximate round trip time in milliseconds :
Minimum : 4ms , Maximum : 7ms , Average : 4ms

Observation ?

8/10
25/4/22

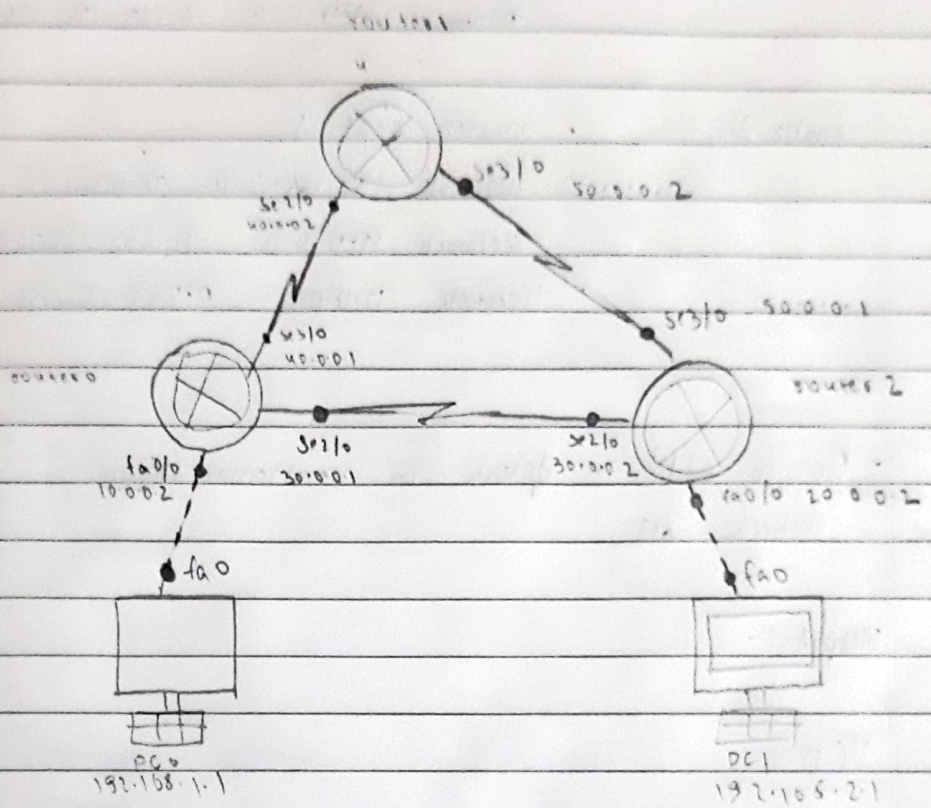
Lab : 07

SURYA Gold

Date Page

Configure OSPF routing protocol

Aim : Construct OSPF routing protocol.



① Create Topology.

② Configure IP and gateway

Router 0 :-

- enable
- Config t
- ~~Router~~ Ospf 1 ~~#~~ interface se 2/0
- network 192.168.1.0 0.0.0.255 area 0
- network 10.0.0.0 0.255.255.255 area 0
- network 12.0.0.0 0.255.255.255 area 0
- exit

Router 1 :-
- router ospf 1
- network 10.0.0.0 0.255.255.255 area 0
- network 11.0.0.0 0.255.255.255 area 0

Router 2 :-
- router ospf 1
- network 192.168.2.0 0.0.0.255 area 0
- network 11.0.0.0 0.255.255.255 area 0
- network 12.0.0.0 0.255.255.255 area 0

- Go to edit in filters in simulation mode
- uncheck all

output :-

PC0

Ping 192.168.2.1
Reply from 192.168.2.1 : bytes=32 time=2ms TTL=128
Reply from 192.168.2.1 : bytes=32 time=10ms TTL=128
Reply from 192.168.2.1 : bytes=32 time=12ms TTL=128
Reply from 192.168.2.1 : bytes=32 time=8ms TTL=128

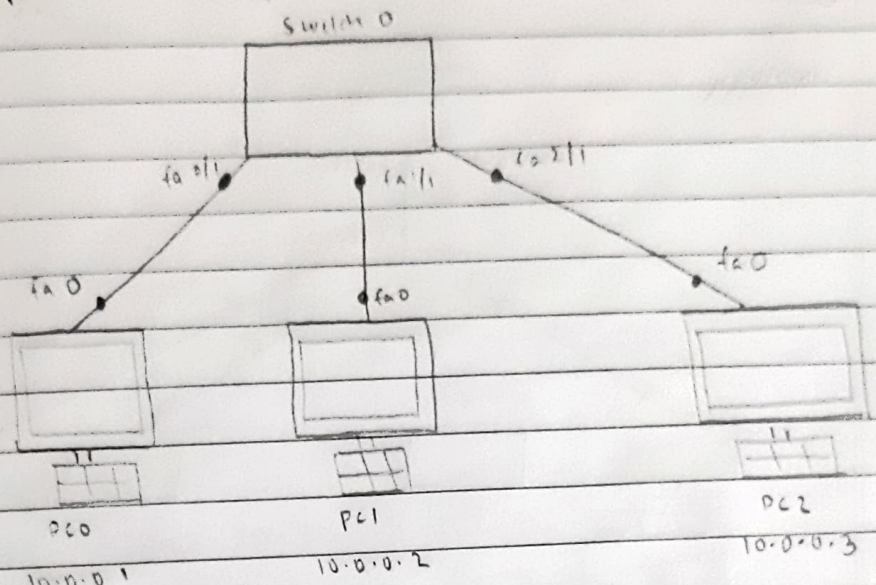
Ping statistics for 192.168.2.1:
Packets : Sent=4, Received=4 Lost=0 (0% loss)

Approximate round trip time in millisecond
minimum : 2ms maximum : 11ms Average : 8ms

Lab - 07

SURYA Gold
Date Page

Aim: Construct a lan and understand the concept of operation and Address Resolution protocol (ARP)



→ Construct a Topology

→ Configure IP

PC0 (CMD)

- arp -a
- ping 10.0.0.2
- arp -a
- arp -d

Arp command to see Arp table. initially the Arp table will be empty. The command show MAC address table can be given ~~every~~ every transaction to see how the situation recon from transaction and build address tbl

10/10

output :- * arp -a
No ARP entries found.

* ping 10.0.0.2
Reply from 10.0.0.2 bytes=32 time Dms TTL: 128
Reply from 10.0.0.2 bytes: 32 time<1ms TTL: 128
Reply from 10.0.0.2 bytes: 32 time: 0ms TTL: 128

N 1 | 9 | 23

* arp -a

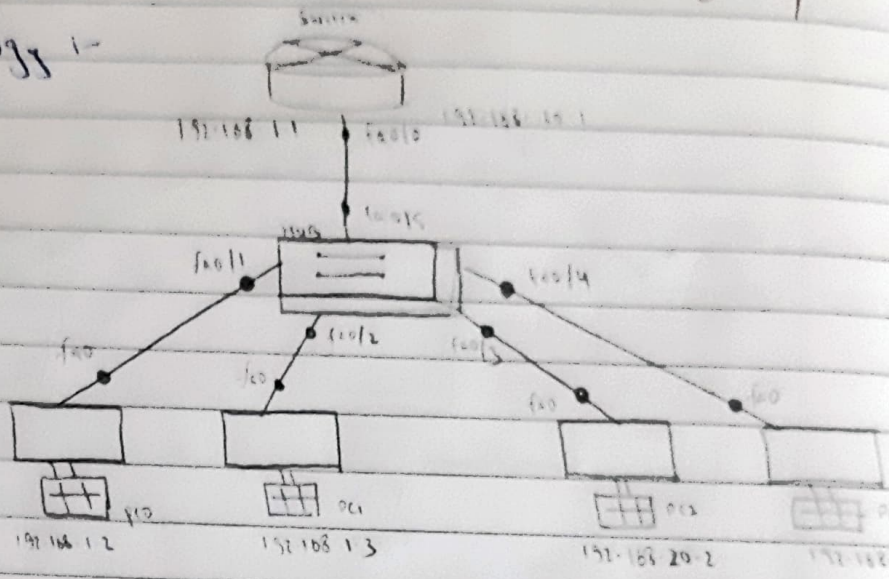
Internet address	physical address	Type
10.0.0.2	0002.16C5.9890	dynamic

lab. 8

SURYA Gold
Date Page

Aim :- To construct Simple LAN and understand concept and operation of TCP/IP

Topology :-



Create Topology of 4 pcs, one switch, one router. Configure the pcs with each other.
for PC0 & PC1 give gateway 192.168.1.1
for PC2 & PC3 give gateway 192.168.20.1

Select VLAN in database and put VLAN number and VLAN name for both Switch and router.

In VLAN :-

fastethernet → dropdown, select TRUNK
fastethernet 0/4 → Configure VLAN
fastethernet 0/3 → Configure VLAN

In Router :-

CLI of Router
Config t
interface fastEthernet 0/0.1

encapsulation dot 1q 2

ip address 192.168.20.1 255.255.255.0

no shut

exit

Output :-

ping 192.168.20.3

pinging 192.168.20.3 with 32 bytes of data

Reply from 192.168.20.3 : byte : 32 time : 0ms TTL : 127

Reply from 192.168.20.3 : byte : 32 time 0ms TTL : 127

Reply from 192.168.20.3 byte : 32 time 0ms TTL 127

Reply from 192.168.20.3 byte . 32 time 0ms TTL : 127

Ping stastics for 192.168.20.3

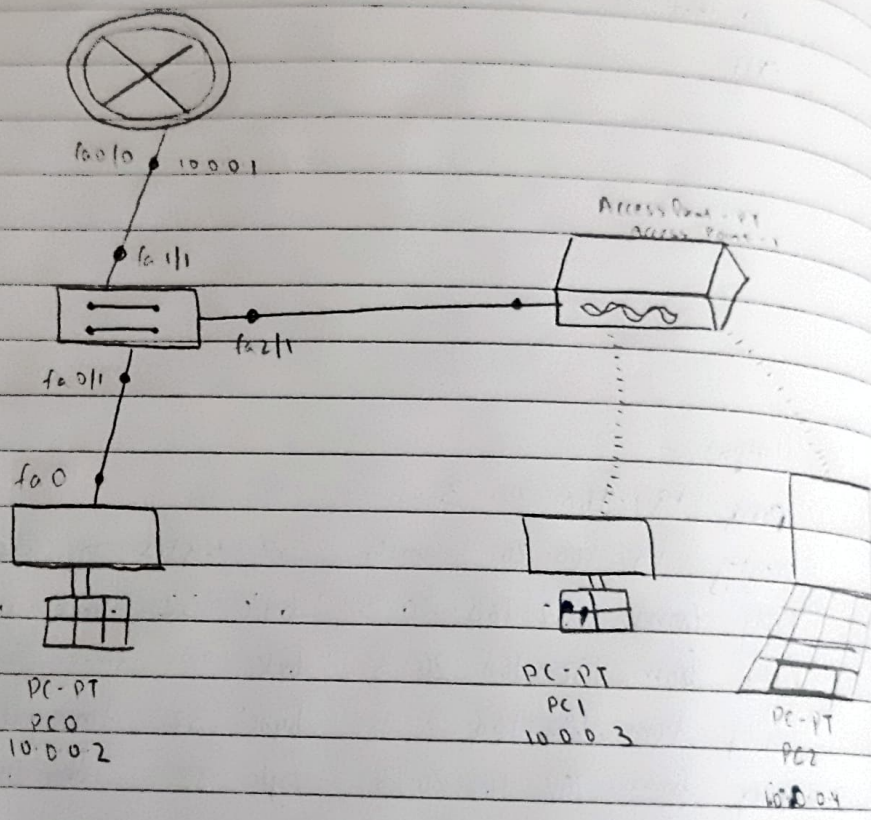
packets : Sent = 4, Received = 4 Lost : 0 (0% loss).

Aim : To construct VLAN and make the PC communication among VLAN

10/10

4/23

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



Procedure :-

Construct the topology

Set IP address and gateway of PC's

for access Point 0.

→ Input

SSID = WLAN

WEPKey = 1234567890

→ for PC1

Switch off dog ethernet PT-HOST-NM-1AM to the Component list & drag WMP 3 on to the by POO. Then switch on the device.

SURYA Gold
Date Page

In wireless 0;

SSID - WLAN

WEP key - 1234567890

IP address - 10.0.0.3

Gateway - 10.0.0.1

Repeat the same for laptop.

Switch off the device. Drag the existing PT-HOST-NM-1AM to component listed in LHS Drag WMP300N Wireless interface to empty port.

Switch on the device.

In the Config tab. a new wireless interface would have been added. Now Configure SSID, WEP, WEP key. Ip address and gateway as manually done to the device

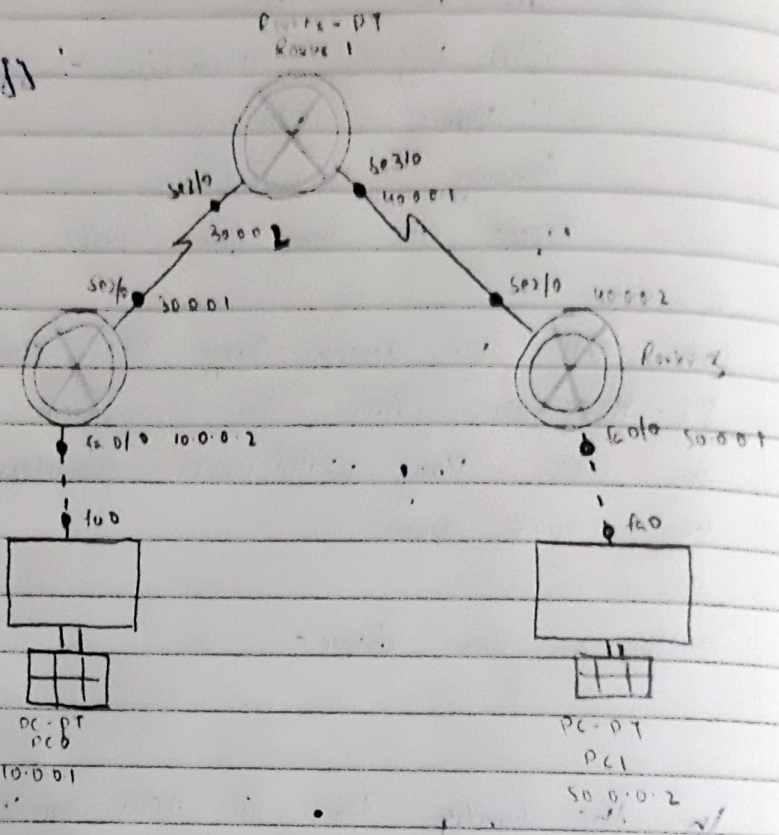
10/10

NL

SURYA Gold
Date Page

Aim: Demonstrate the TTL / life of a packet.

Topology:



Procedure:

1. Create a topology as shown above the 2 PCs and 3 routers.

2. Configure the IP address of PC0 and PC1 as 10.0.0.1 , 50.0.0.2 respectively.

3. Configure the IP address of routers using following commands.

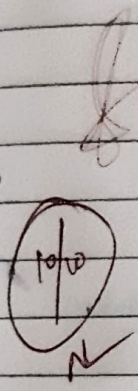
```
Config t
Interface fa 0/0
ip address 10.0.0.2 255.0.0.0
exit
```

4. Configure the source using default / static routing.

5. In Simulation mode, send a simple PDU from one PC to another.

6. Use Capture button to capture every transfer.

7. Click on PDU during every transfer to see the Inbound & outbound PDU details.

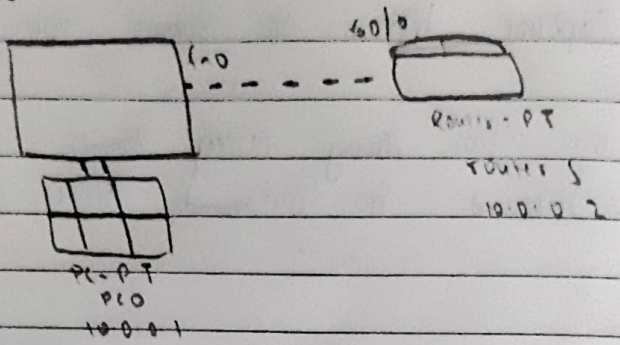


SURYA Gold

Lab-9

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

Topology:



Procedure:

Steps: Create a Topology using 1 PC and 1 router as shown above

Step 2: Set the IP address and gateway as 10.00.1 and 10.0.0.2 for the PC

Step 3: In router, go to CLI

```
Router > enable
Router # config t
Router (config t) # hostname R1
R1 (config) # enable secret P1
R1 (config-if) # ip address 10.0.0.2 255.0.0.0
R1 (config-if) # no shut
R1 (config-if) # line vty 0 5
R1 (config-line) # login
Login disabled on line 132, until password is set
R1 (config-line) # password p1
R1 (config-line) # exit
R1 (config) # exit
R1 # wr
```

Step 4: In Command prompt of PC,

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
Reply from 10.0.0.2 : bytes=32 time=0ms TTL=255
Reply from 10.0.0.2 : bytes=32 time=0ms TTL=255
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 time in milliseconds
Minimum = 0ms, Maximum = 0ms, Average = 0ms.

PC > telnet 10.0.0.2
Trying 10.0.0.2 open
User Access Verification
Password: (PO)
R1 > enable
Password: (PS)
R1 # show ip route
Codes: C - connected
C 10.0.0.0/8 is directly connected, FastEthernet 0/0

LS

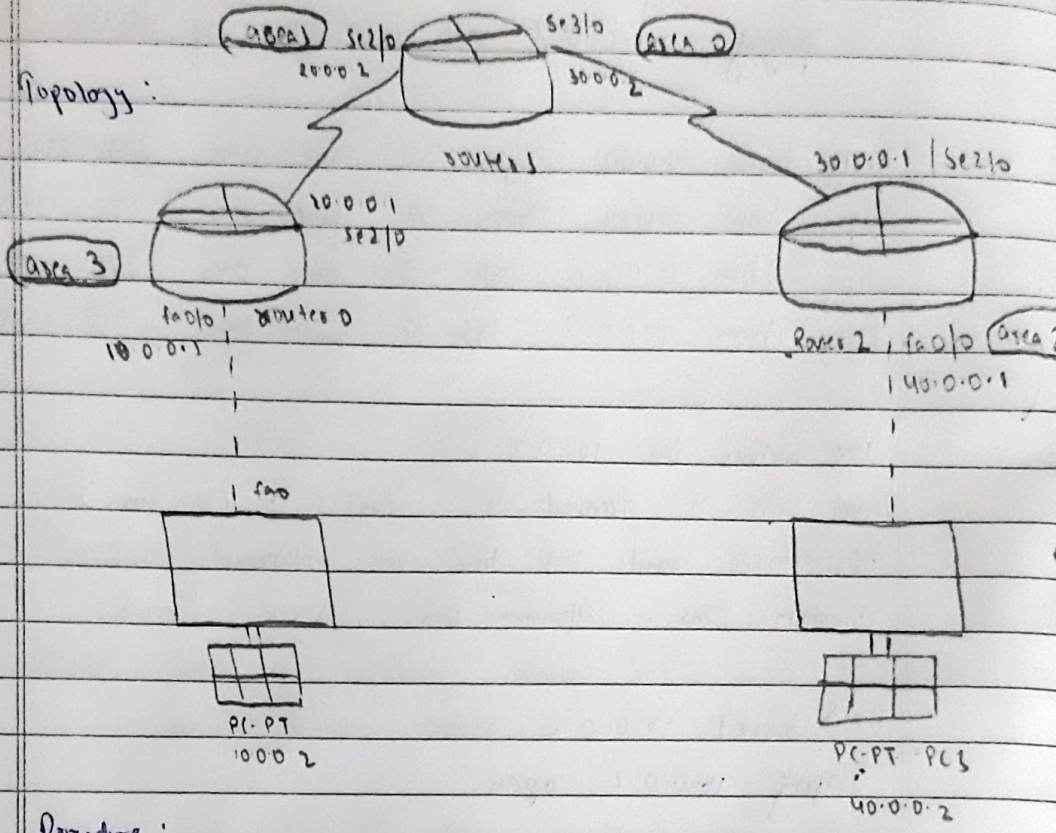
8/10

1/9/23

observation: using telnet protocol we can access the router from the PC which is connected to it.

Aim: To Configure OSPF routing protocol and Connected areas.

Topology:



Procedure:

Steps: Create a topology as shown above using 2 PCs and three routers.

Step 2: Configure ip address and gateway for PCs as 10.0.0.2 and 10.0.0.1 for PC0 and 40.0.0.2 and 40.0.0.1 for PC3 respectively.

Step 3: Configure ip address to all router interfaces

Router R0

R0 (config)# interface fastEthernet 0/0

R0 (config-if)# ip address 10.0.0.1 255.0.0.0

no shut

interface Serial 2/0

ip address 20.0.0.1 255.0.0.0

encapsulation PPP
clock rate 64000
no shut
exit

Similarly, configure for R1 and R2.

Step 4: Now, enable ip routing by configuring OSPF routing Protocol in all routers.

Router R0:

(config) # router ospf 1
router-id 1.1.1.1
network 10.0.0.0 255.255.255.0 area 3
network 20.0.0.0 255.255.255.0 area 1
exit

Similarly, Configure for R1 and R2.

Step 5: Now, check routing table of R0

Router # show ip route

C - Connected
O - OSPF
C - 10.0.0.0/8 is directly connected, fa 0/0
C - 20.0.0.0/8 is directly connected, Serial 2/0
O IA 40.0.0.0/8 via 20.0.0.2, 00:04:23, Serial 2/0
O - IA 30.0.0.0/8 via 20.0.0.2, 00:07:29, Serial 2/0

Here R1 knows area 0 Network 20.0.0.0 connected to R1 from R0, So R0 learns networks through this network.

Router (config) # router ospf 1, 1 ⇒ process id (1 - 65535)
There must be one interface up to keep ospf process - up
So, its better to configure loopback address to router,
It is a virtual interface never goes down once we configure

R0 (configure - if) # interface loopback 0
ip add 172.16.1.252 255.255.255.0
no shut.
Similarly, configure for R1 and R2.

Step 1: Now check routing table for R3
R3 # show ip route
Codes: O - Ospf C - Connected

O IA 20.0.0.0/8 via 30.0.0.2, 00:18:58, Serial 2/0
C 40.0.0.0/8 is directly connected, fast Ethernet 0/0
C 30.0.0.0/8 is directly connected, Serial 2/0

Here, R3 doesn't know about the area 3 So we have to create virtual link between R0 and R1.

Step 7: Create a virtual link between R0, R1, and by this we create a virtual link to connect area 3 to area 0.

SURYA Gold

Date _____ Page _____

In R0.

R0 (config) # router ospf 1

area 1 virtual-link 2.2.2.2

In R1.

R1 (config) # router ospf 1.

area 1 virtual link 1.1.1.1

Step 8: R1 and R2 get updates about Area 1 Now, check routing table R2.

R2 # show ip route

Codes: O - OSPF C - Connected

O IA 20.0.0.0/8 via 30.0.0.2, 00:01:56, Serial 2/0

C 40.0.00/8 is directly connected, FastEthernet 0/0

O IA 10.0.0.0/8 via 30.0.0.2, 00:01:56, Serial 2/0

C 30.0.0.0/8 directly Connected, Serial 2/0

Step 9: Ping PC1 from PC0.

In PC0,

PC> Ping 40.0.0.2

8/12

Pinging 40.0.0.2 with 32 bytes of data:

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

" " " " " time: 10ms TTL: 125

" " " " " time: 14ms TTL: 125

" " " " " time: 2ms TTL: 125

Packet: Sent: 4 Received: 4 Lost: 0 (0% loss)

Approximate round trip time in millisecond

Minimum: 2ms, Maximum: 14ms, Average: 7ms.

12/9/22

Cycle - 2

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

Program

```
#include <stdio.h>
char m[50], g[50], a[50], q[50], temp[50];
void caltans(int);
void xor(int);
void algram();
void shift();
void main()
{
    int n, i=0;
    char flag = 0;
    printf("Enter the frame bits:");
    while((ch=getc(stdin)) != '\n');
        m[i++] = ch;
        n = i;
    for(i=0; i<16; i++)
        m[n++] = '0';
        n[n] = '\0';

printf("message after appending 16 zeros : %s", m);
    for(i=0; i<16; i++)
        g[i] = '0';
    g[0] = g[4] = g[11] = g[16] = '1';
    g[17] = '\0';
printf("\n generator : %s \n", g);
    xor(n);
```

```
printf ("\n quotient %s", q);
Calloram (n);
printf ("\n transmitted frame %s", m);
printf ("\n Enter transmitted frame ");
Scanf ("%s", m);
printf ("CRC checking \n");
  crc(n);
printf ("\n\n last remainder %s", r);
  for (i=0; i<10; i++)
    if (r[i] != '0')
      flag = 1;
    else
      Continue;
  if (flag == 1)
    printf ("Error during transmission");
  else
    printf ("\n Received frame is correct");
}

Void crc (int n)
{ int i, j;
  for (i=0; i<n; i++)
    tmp[i] = m[i];
```

```
x[10] = m[17+i];
x[17] = '\0';
for (j=0; j<17; j++)
    temp[j] = x[j];
}
g[n-16] = '\0';
}

Void colram ()
{   int i, j;
    for (i=1; i<=16; i++)
      x[i-1] = (int)temp[i]-48) ^ ((int)g[i]-48)+48
}

Void shift()
{ int i;
    for (i=1; i<16; i++)
      x[i-1] = x[i];
}

Void coffen (int n)
{ int i, K=0;
    for (i=n-16; i<n; i++)
    m[i] = ((int)m[i]-48) ^ ((int)x[k++]-48]+48;
      m[i] = '\0';
}
```

Output: Enter bits = 1011

Message after appending 16 zeros : 1011 0000 0000 0000 0000

generator : 10010000010001

quotient : 1011

transmitted frame : 1011 1011 0001 0110 1011

last remainder : 0000 0000 0000 0000

Received frame is Correct.

```
} else {
    printf(" Dropped %d no of packets \n", incoming - (bucket_size - store));
    store = bucket_size;
}
printf(" After Outgoing %d packet left out %d in buffer
    \n", store, bucket_size);
    n--;
}
return 0;
}
```

Output: Enter bucket_size, outgoing rate and no of inputs: 20 10 2
Enter incoming packet size: 30
Incoming packet size: 30
Dropped 10 no of packets
Bucket buffer size 0 out of 20
After Outgoing 10 packets left out 20 in buffer

9/10

10/23

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

Code

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("In Enter file name : ")

ClientSocket.Send(Sentence.encode())
FileContents = ClientSocket.recv(1024).decode()
print("In From Server : \n")
print(FileContents)
ClientSocket.close()
```

ServerTCP.Py

```
from Socket import *
ServerName = "127.0.0.1"
ServerPort = 12000
ServerSocket = Socket(AF_INET, SOCK_STREAM)
ServerSocket.bind((ServerName, ServerPort))
ServerSocket.listen(1)
```

while 1:

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

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

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

bind() - method bind to specific IP and port so that it can listen to incoming req on the IP

listen()

client output:

```
>>> The server is ready to receive
```

Server output

```
Enter the filename: Server TCP.py
from Server
/*
    Content of the file will be displayed here
*/
```

Client UDP.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))

fileContent, ServerAddress = ClientSocket recvfrom(2048)
print ("\n Reply from Server \n")
print (fileContents.decode("Utf-8"))
    # for i in fileContents:
        # print (str(i), end = "")
        clientSocket.close()
        ClientSocket.close()
```

Server UDP.py

```
from Socket import *
ServerPort = 12000
```

while 1:

Sentence, clientAddress = serverSocket.recvfrom(2048)
Sentence = Sentence.decode("utf-8")
file = open(Sentence, "r")
Con = file.read(2048)

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

print('In Sent Content of', end = ' ')
print(Sentence)
~~for~~ # for i in sentence:
# print(str(i), end = '')
file.close()

output:

The Server is ready to receive
Sent Content of SeaverTCP.py
The Server is ready to receive

Server UDP output

Enter the file name: serverUDP.py
Reply from Server

/*
Content of the file
displayed well
*/