

9.10.2024

Experiment 1

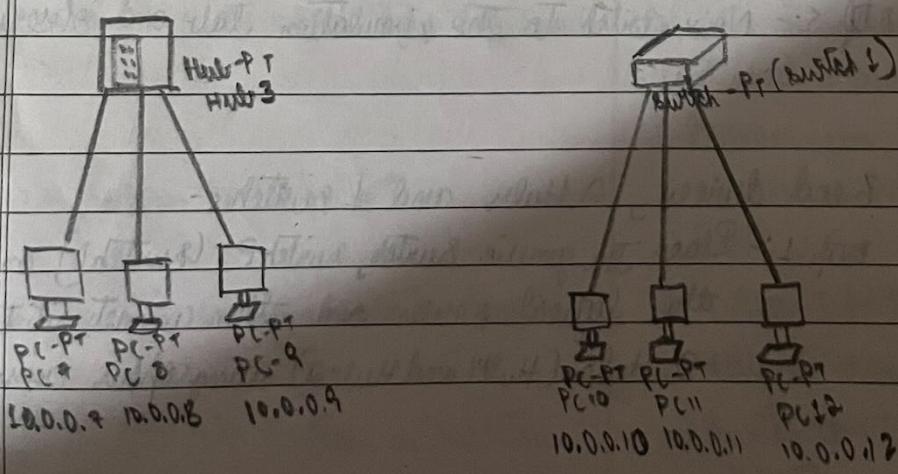
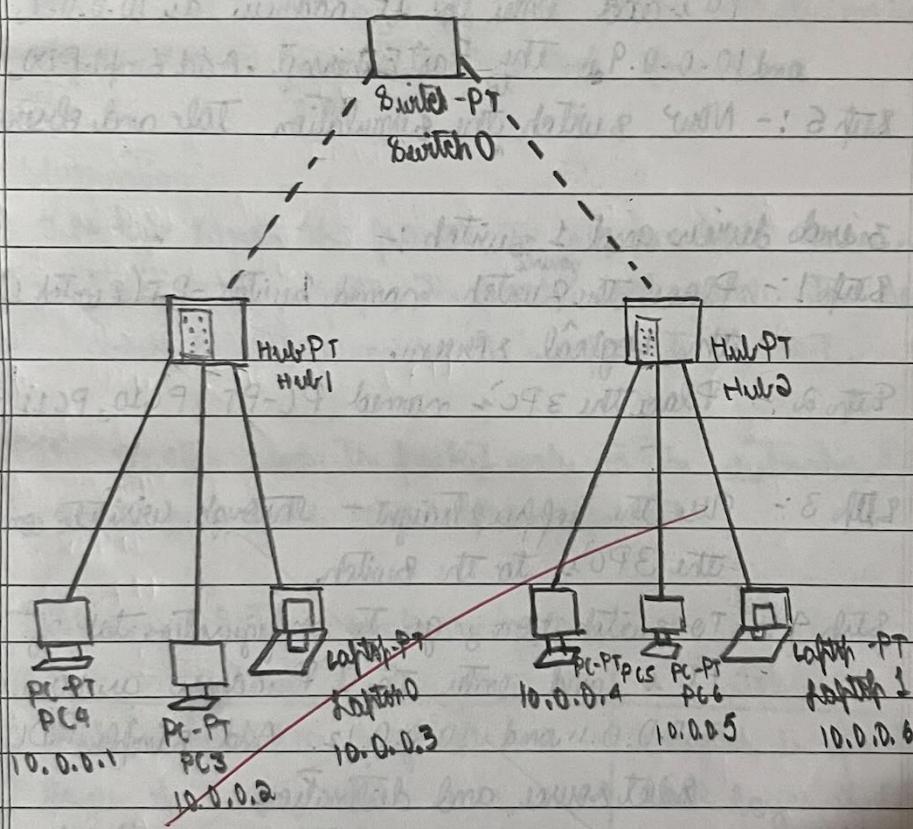
PAGE NO. _____
DATE : _____

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

Configurations :- 3 PC's, PC7, PC8, PC9 connected to Hub0

3 PC's, PC10, PC11, PC12 connected to Switch1

4 PCs, 2 Laptops connected to 2 Hubs which are connected to Switch.



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 Fast Ethernet. Add Simple PDU, select source and dest.

Step 5 :- Now switch to 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 (Hub₁ and Hub₂) using copper cross-over.

wire.

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

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

Step 4:- Go to the configuration tab of the PC's and enter its 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 PCs repeat the packet.
2. The switch sends the packet only to the destination PC which accepts and sends back acknowledgement.

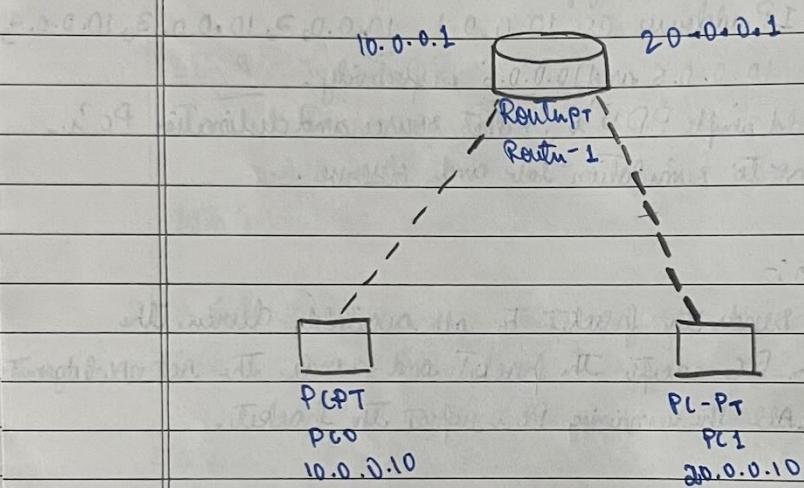
Difference between Hub and Switch:-

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, multi-cast and broadcast type transmission.
→ Hub has 4/12 ports	→ Switch can have 2 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.

Expt-2

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

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



Procedure :-

- 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:- Enter PC1 config \rightarrow 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

Expt

Net

Step 4:

General
PCPuring
of
Packet

16/10/22

Step 3:- Select switch and go To The CL1, execute The following command.

```
# enable
# config terminal
# interface fast Ethernet 1/0
# ip address 10.0.0.1 255.0.0.0
# no shutdown
# exit
```

notice PC0 and switch are successfully connected.

Now run for PC1

```
# interface fast Ethernet 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:-

Packet:-

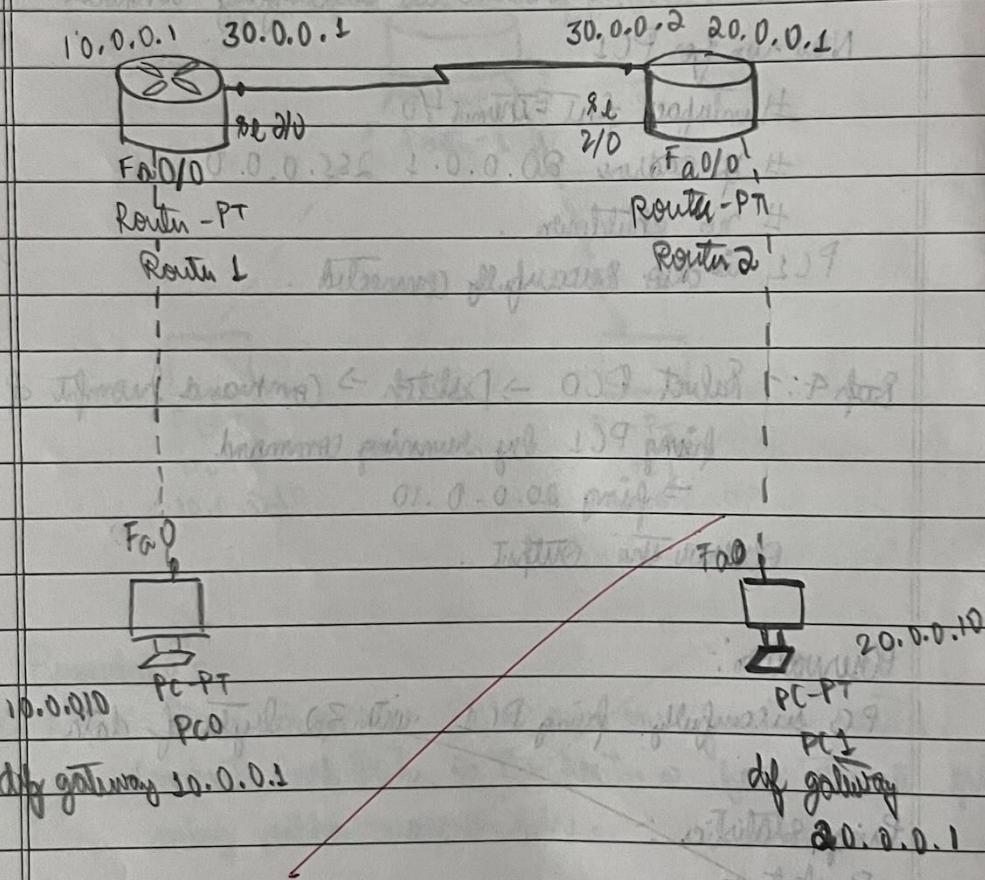
Bent=4, Received=4, Lost=0

16/10/24

Experiment 3

Objectives :- Configure default routes, static routes to the routers. To connect 2 PCs on 2 different networks using two routers.

Topology :-



Procedure:-

- ① Connect two PCs to the two different routers using Ethernet cables.
- ② Connect two routers to each other using serial DCE.
- ③ Configure PC0 to IP 10.0.0.10 and default gateway 10.0.0.1 and PC1 to IP 20.0.0.10 and default gateway 20.0.0.1.

gateway 20.0.0.1.

③ Select router 0 → CLI and execute the following commands

> enable

config terminal

interface fast Ethernet 0/0

ip address 10.0.0.1 255.0.0.0

no shutdown

Notice PC 0 of router 0 are successfully connected

Select the same procedure for router 1 with ip address 20.0.0.1

255.0.0.0

Notice PC 1 of 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 shutdown

Go to router 1 → CLI config terminal and execute

interface serial 2/0

ip address 30.0.0.2 255.0.0.0

no shutdown

Notice router 0 & router 1 are successfully connected

⑤ Go to PC 0 → Desktop → Command prompt and run

Ping 20.0.0.10

Ping 30.0.0.2

Ping 30.0.0.1

and observe the outputs

Observation :-

- All PCs and routers are connected successfully.
 - Ping 20.0.0.10 and ping 30.0.0.2 are unsuccessful and show destination host unreachable. This happens because they are not neighbor network with 10.0.0.1.
 - Ping 30.0.0.1 will be successful.

23.10.24

100.05 with 23/60 1st stage of withdrawal with Dr. Tandy

the distance from Brimfield (16.9) to Pittsfield (10.0).

Obsidian stationary #

0.0.0.226 1.0.0.0c ~~multiple~~ 5.4

more than linear below 1.10 & I think it is

0-0-0.236 0-0-0.08 similar to #

Tulsa 4

Waves of human migration into Europe after

various types of biomass) < Fertilizer < animal waste (2)

It is very difficult to

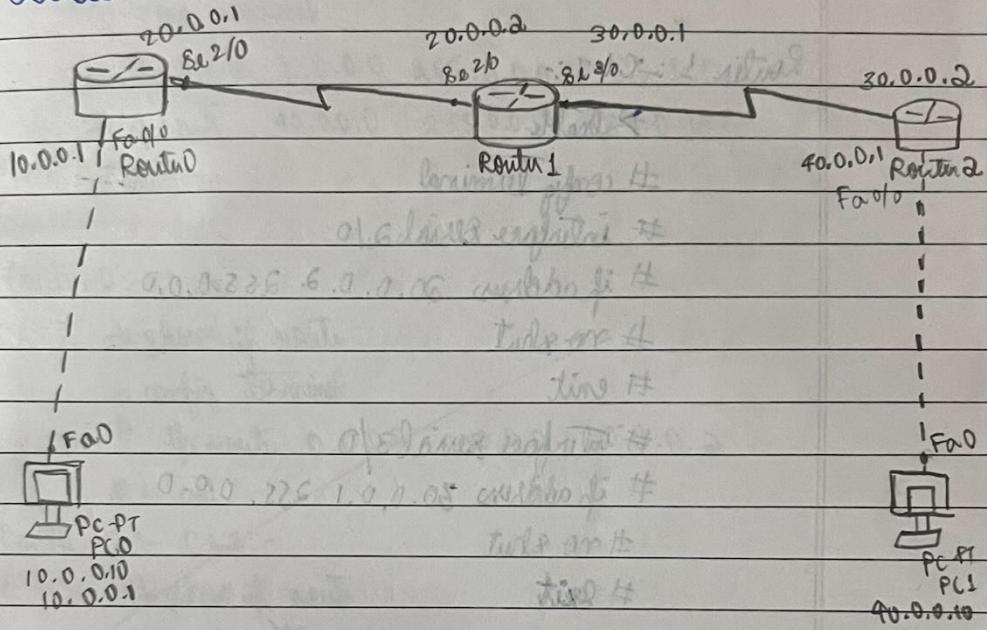
Digitized by srujanika@gmail.com

23.10.24

Experiment 4

Objective:- Configure one static routing and two default routing using three routers and 2 PCs.

Topology:-



Procedure:-

Step 1:- Place 2 PCs and 3 generic routers. Connect Po.0, Router 0 & PC's, Router 2 with copper cross over wire. Router 0, Router 1 & Router 1, Router 2 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

Router 0 → C1
Router 2 → Serial

Router 1 → Serial

config terminal

II interface FastEthernet0/0

ip address 20.0.0.1 255.0.0.0

no shut

exit

config terminal interface serial 2/0

ip address 20.0.0.1 255.0.0.0

no shut

exit

Router 1 :- CLI:-

> enable

config terminal

interface serial 2/0

ip address 20.0.0.2 255.0.0.0

no shut

exit

interface serial 3/0

ip address 30.0.0.1 255.0.0.0

no shut

exit

Router 2 :- CLI:-

> enable

config terminal

interface serial 2/0

ip address 30.0.0.2 255.0.0.0

no shut

exit

interface fast Ethernet 0/0

ip address 40.0.0.1 255.0.0.0

no shut

exit

Connection b/w (R_0, R_1) , (R_1, R_2) , (R_0, R_2) and $(P_{(1,2)}, R_2)$ will be

there

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 ass.0.0.0.0 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

Batch 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 connections are successful
- (ii) PCL → Desktop → Command prompt
Ping 40.0.0.10 is successful

Ping Statistics

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

(iii) IP routes in R1

- 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

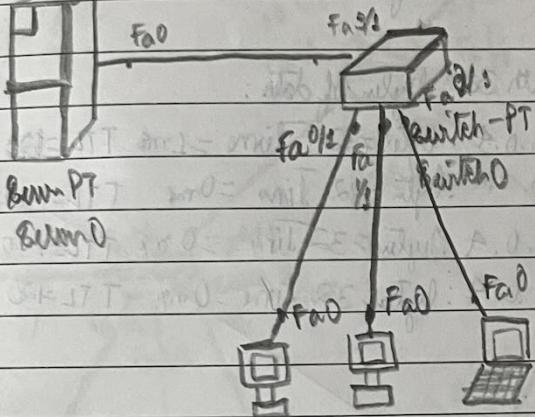
~~8x 10/2~~

Experiment 5

Objective :- Design a DHCP within LAN.

Topology :-

10.0.0.1 → J/P
10.0.0.2 → Gateway



Procedure:-

Step 1:- Place 3 PCs in 1 room and one Router and connect all end devices to the switch using copper straight wires.

Step 2:- Go To Super PT → Desktop → IP configuration

IP 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

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

Step 5 :- Pings 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

Time taken for round trip 1 ms

Initial configuration of PC0

10.0.0.1 - 192.168.1.1

Configured IP address & subnet mask

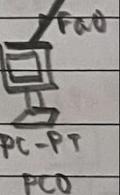
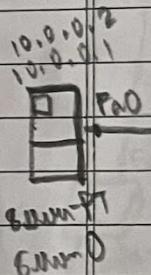
10.0.0.1 - 192.168.1.1

10.0.0.1 - 192.168.1.1

10.0.0.1 - 192.168.1.1

Objectiv

Topology



Procedure

Step 1 :-

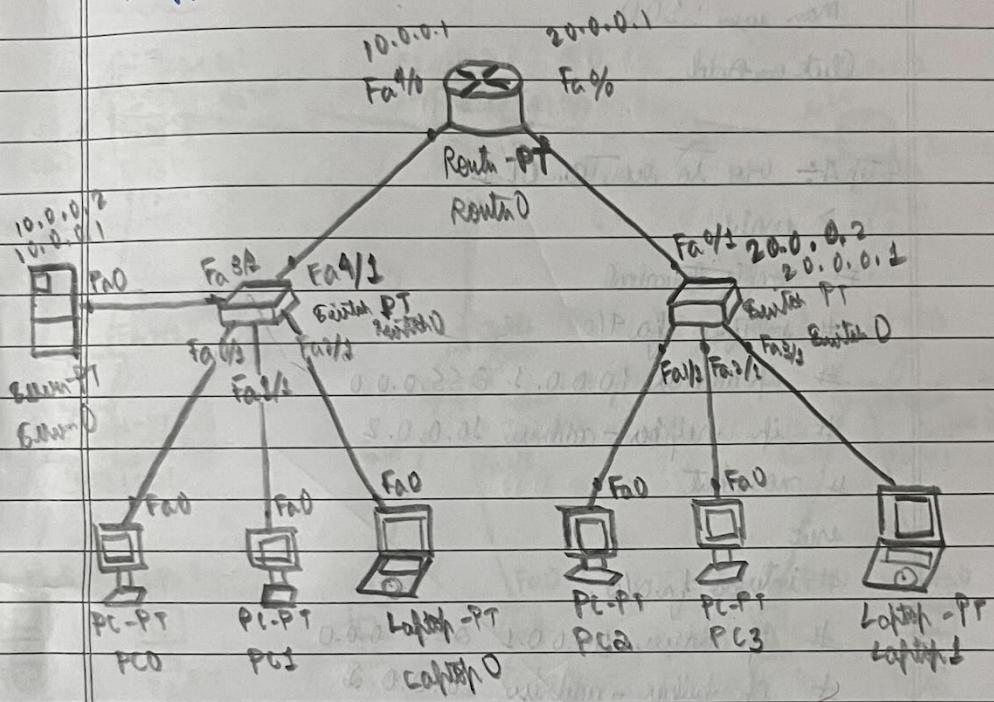
Step 2 :-

Step 3 :-

Experiment - 6

Objective :- Design DHCP outside LAN

Topology :-



Procedure :-

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

Step 2 :- Router \rightarrow Desktop \rightarrow IP Configuration

IP address - 10.0.0.2

Default gateway - 10.0.0.1

Step 3 :- Config \rightarrow Services \rightarrow DHCP, Turn service To ON

Pool name :- Switch 1

Default gateway :- 10.0.0.1

Start IP: 10.0.0.3

Mask size: 100

Click on Add.

Path name: switch 2

Default gateway: 20.0.0.1

Default start IP: 20.0.0.3

Mask size: 100

Click on Add

Step 4: Go To router CLI

> enable

config terminal

interface fa 0/0

ip address 10.0.0.1 255.0.0.0

ip helper-address 10.0.0.2

no shutdown

exit

interface fa 0/0

ip address 20.0.0.1 255.0.0.0

ip helper-address 10.0.0.2

no shutdown

All router switch connection go up

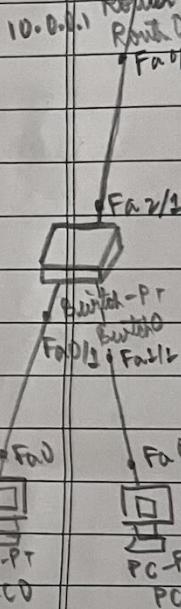
Step 5: Go to all 6 PCs and change IP configuration

from static IP DHCWP addresses will be automatically assigned

Step 6: Ping PC to PCs.

Observation: (1) All connections are successful

(2) All PCs are assigned.



Object

Topology

R1
Router
Rout C
Fa0/0

Fa2/1

Switch - P1

Switch - P2

Fa1/0

Fa1/1

PC - P1

PC - P2

Procedure

Step 1: - PD

Tru

Step 2: - Com

10.0

and

Experiment - 7 (B) (Method) - 2

Date 20.0.06.2017

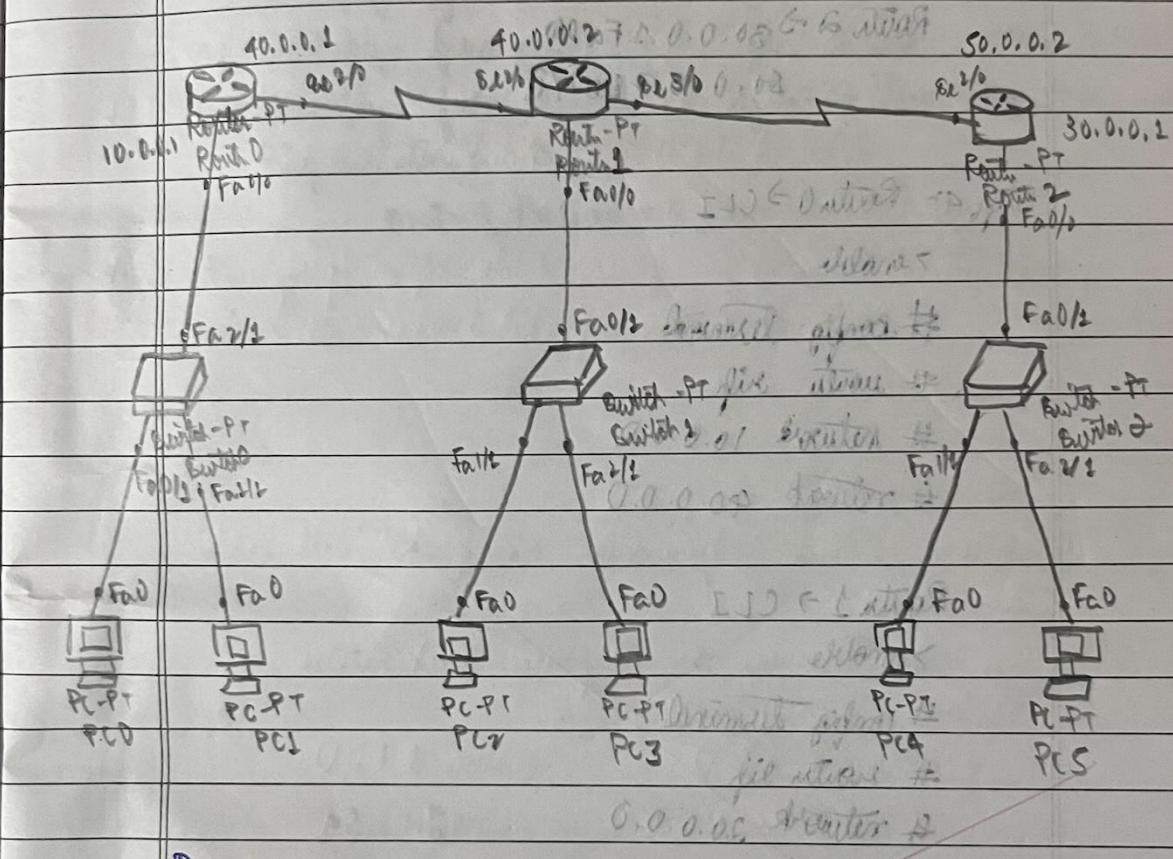
Objective :- Configure Routing Information Protocol in Router.

0.0.0.7 1.0.0.0.0.6 → Latency?

0.0.0.8 2.0.0.0.0.7

0.0.0.6 3.0.0.0.0.6

Topology :-



Procedure :-

Step 1:- Place 3 routers, 3 switches and 6 PCs and connect them using their respective wires as shown above.

I J J C S R T N A

Step 2:- Configure IP address 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 gateway 10.0.0.1, 20.0.0.1, 30.0.0.1

0.0.0.0.0 Router 1

0.0.0.0.0 Router 2

Step 3:- Configuring IP address of all Router

Router 0 \rightarrow 10.0.0.1 Fa0/0

Router 0 \rightarrow 10.0.0.1 8e2/0

Router 1 \rightarrow 20.0.0.1 Fa0/0

40.0.0.2 8e2/0

50.0.0.1 \rightarrow Fe3/0

workstation

Router 2 \rightarrow 30.0.0.1 Fa0/0

50.0.0.2 8e2/0

Step 4:- Router 0 \rightarrow CLI

> enable

config terminal

router rip

network 10.0.0.0

network 40.0.0.0

Router 1 \rightarrow CLI

> enable

config terminal

router rip

network 20.0.0.0

network 50.0.0.0

~~Configuring Router 0~~ Router 0 \rightarrow CLI
Router 0 \rightarrow 10.0.0.1 Fa0/0, 10.0.0.1 8e2/0

Router 2 \rightarrow CLI

> enable, do not suspension # interface -> 8e2/0

8.0.0.1 50.0.0.1, 50.0.0.1 8e2/0

config terminal, 0.0.0.0 0.0.0.0

router rip, 1.0.0.0 1.0.0.0, running, 10

network 30.0.0.0

network 50.0.0.0

BIT 5:-

Observation

1. All d

2. Ping 3

Ans 0.9.1.1

problem solved

Tested all

STT 10 minutes

Rin

done now go

20% extra

done

Ques 5: Show IP route 8 in any route and ping any 2 end devices

Observation :-

1. All devices are connected successfully.
2. Ping 30.0.0.3 from PC0, output -> 1000 bytes with 0.9 ms ping round trip time - 16 ms
3. reply from 30.0.0.3 bytes = 32 Time = 7 ms TTL = 125
4. reply from 30.0.0.3 bytes = 32 Time = 8 ms TTL = 125
5. reply from 30.0.0.3 bytes = 32 Time = 10 ms TTL = 125
6. reply from 30.0.0.3 bytes = 32 Time = 7 ms TTL = 125

Ring Statistics :-

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

~~Ring statistics information after configuration :-~~

and after taking no informative setting initial

initial value using the no command value will be

0.99 inactive

0.69 inactive

P.C. : STP

220 : STP

Experiment - 8

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

Procedure:-

Step 1 :- Construct the network topology.

Step 2 :- Create switches create a simple PDU source : PC0 and destination :- PC5. Go To simulation frame by switching it.

Step 3 :- click Auto capture / Play and select The packets when it reaches router 1 checks 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 here.

3

→ The value decreases as it passes each router

4 Router 0 :-

inbound PDU

TTL: 255

outbound PDU

TTL: 254

20/10/2024

7

→

→

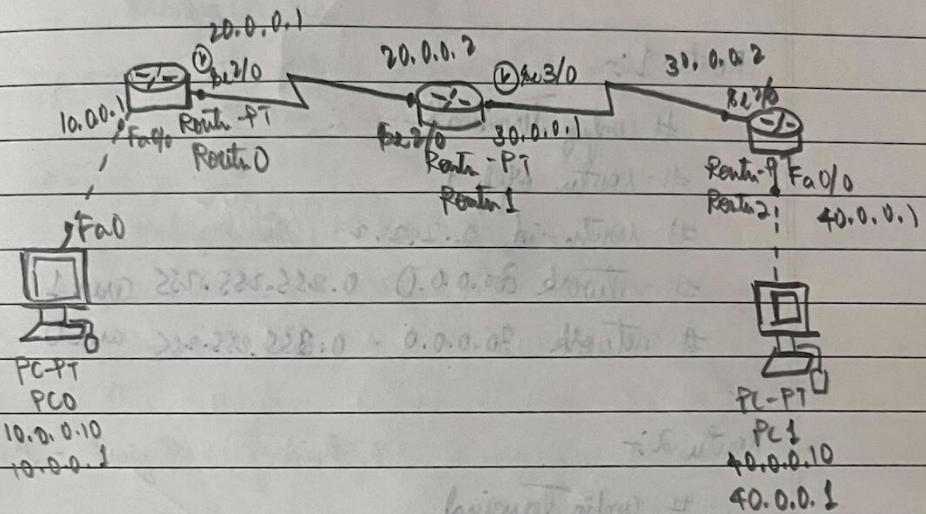
(config)
(config)
(config)

Experiment - 9

Objective:- Demonstration OSPF protocol

Procedure :- Topology :-

Step 1:-



Procedure :-

1 Step 1 :- Place 3 routers of 2PI 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 :-

(config-if) # 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 Router 2, in interface serial 2/0

(config-if) # encapsulation PPP

(config-if) # no shutdown .

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 30.0.0.0 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 loopback0
# ip add 172.16.1.252 255.255.0.0
# no shutdown
```

Router 1:-

```
(config-if) # interface loopback0
# ip add 172.16.1.253 255.255.0.0
# no shutdown
```

Route 2 :-

(config-if) # interface loopback 0

if add 172.16.1.254 285.255.0.0

no ~~Sheldan~~

~~8th~~ 6:-

Route :-

(Reconfig) + south offy 1

(using route) + area 1 virtual-link 2.2.2.2

exit

Route 1 :-

R₁ (config-Router) #1 router off

```
(config-router) # area 1 virtual-link 1.1.1.1
```

init station 1 bird name 3, 59 ± 0.017 (1)

After all this work, I am still in the tunnel.

Fig 9. - Pump PCs from PC07, no. 91, IT Test

59.19.1 nat 126

Observation:-

② All connections are successful.

② Pung 40.0.0.10

~~src_ip~~ from 90.0.0.10 depth: 320 Time = 7 ms TTL = 23

Reply from 40.0.0.10 length:32 Time=6ms TTL=125

Reply from 90.0.0.10 [depth: 32 Time = 7ms] TTL=125

depth from 40.0.0.10 depth: 32 Time = 7:29 T+L = 125

© 2023 McGraw-Hill Education. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has determined that any suppressed content does not materially affect the overall learning experience. McGraw-Hill Education reserves the right to remove additional content at any time if subsequent rights restrictions require it.

Ping statistics Received = 4 b6A data packets

$$\text{Rent} = 4 \rightarrow \text{lost} = 0 \text{ (0\% loss)}$$

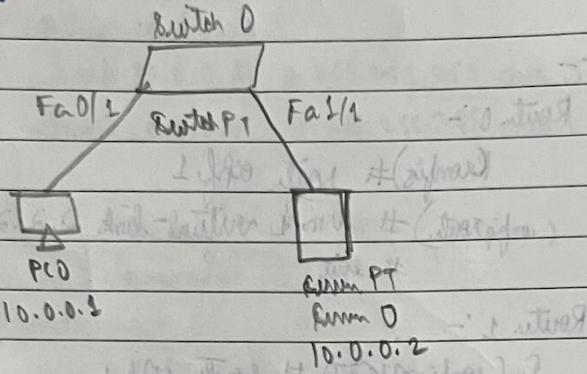
150° int. took bone, removed all C fibres & P int. & T int. intact (2)

89 115

alive in water etc.

Enp → 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 address of PC and the server. Also set DNS server for PC
- ④ Go To Server → 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.

Elevation
→ Encapsulating
entering
→ Configured

Observation:-

- Successfully accessed the server web page from PC by entering IP address (192.168.1.100) at night.
- Configured web server, DNS with a LAN.

11-4-17

- 00:00:00

Wch.

Set DNS

Content of

content.

in the URL

status is bad main in 192.168.1.100 & port 80 is open

Host number 255.255.255.255

status is present but blocked

and 192.168.1.100 is not taking status

at 92A

(so open in most of the time in browser)

status is block 92A port 80

www.192.168.1.100 is taking port 80 in 92A

but not in 192.168.1.100 so opening it may start

number of which two without need user status

(short)

for lots of port 80 but also 192.168.1.100 is not taking port 80

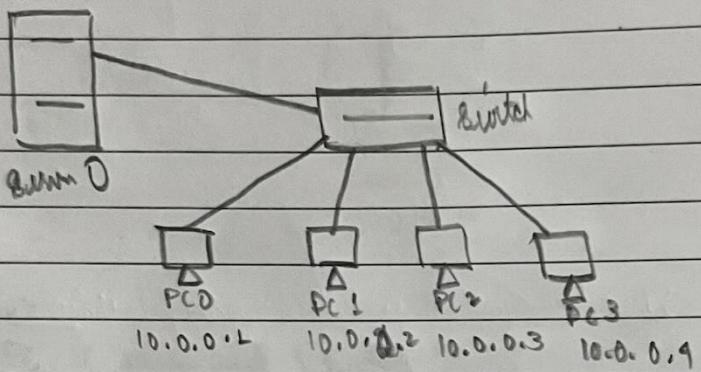
and not in 92A in spite of that no 192A

Exp - 11

Objectives:- To construct simple LAN and understand the

concept and operation of Address Resolution Protocol (ARP)

Topology:-



Procedure:-

- ① With 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 inspect tool to click on a PC to see the ARP table.
(Command in CLI for some in arp -a)
- ⑤ Initially ARP table is empty.
(Also in CLI of switch, the command show mac address-table can be given to every transaction to see how the switch learns from transaction and builds the address table).
- ⑥ Use the option in the simulation tool to go step by step so that the change in ARP can be clearly noted.

Observation :-

→ Burst an error. The media update the ARP table and when a new communication starts.

→ neighbor

→ neighbor

Number when we have started using network

→ It has been observed (i)

1.1.1.5

parent address #

1.1.1.6 address #

192.168.1.1 address #

0.0.0.255 1.1.1.1 address #

Indirect #

2.2.2.2 address #

1.1.1.1 #

0.0.0.0 address #

1.1.1.1 #

1.1.1.1 #

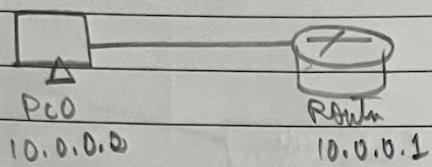
1.1.1.1 address #

Expt-12

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

(iii) Commands
ping 10
ping 20
Burst

Topology :-



* Password
* Password

(iv) Accessing
telnet

User A
Password
Y1
frame
12#
cool
AN

Observation
The admin result

(ii) Commands in Router :-

- > enable
- # config terminal
- # hostname R1
- # enable secret PS
- # interface fastethernet 0/0
- # ip address 10.0.0.1 255.0.0.0
- # no shutdown

Line vty 0 5

login

password p0

exit

> exit

WRT To bank manager.

(iii) Commands in PC

ping 10.0.0.1

ping result:-

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

* Password for user access verification in P0

* Password for enable in P1

(iv) Accessing switch CLI from PC

→ telnet 10.0.0.1

User Access Verification

Password:

r1 > ansley

password:

r1# show ip route

Codes

Area 0.0.0.1 has 2 entries (0.0.0.0/0 via 10.0.0.1, 10.0.0.1 via 10.0.0.1)

.

Observation :-

→ The admin in PC is not able to run commands and see the result from PC.

→ It means it is not having privilege to do so.

→ This shows that user can't run commands.

→ It is due to limitation. So i prefer in this task to

. switch function has certain word which

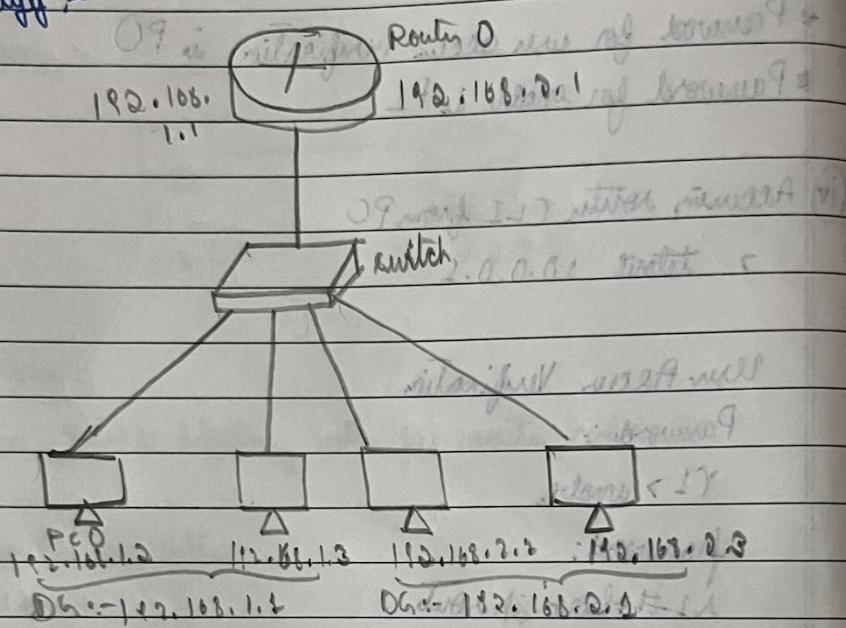
. always appear at starting of the command (like ip, show, config etc.)

(start with start this is same in HAV)

Enr - 13

Objectives :- To construct a VLAN and make the PCs communicate among a VLAN

Topology :-



Procedure :-

- ① Draw a topology with 4 PCs, router and 1 switch as shown above.

→ Assign IP address as shown in the topology.

→ Go to the switch → VLAN tab to configure the VLAN. (in VLAN no. and VLAN name, add it).

→ Select the interface i.e. port trunk 4/1 (from the switch from router) and make it trunk.

(VLAN trunking allows switch to forward frame from different VLAN's over a single link called Trunk).

→ To make switch understand VLAN.

↪ To config tab of switch, select VLAN database, enter the no and name of VLAN created.

(w) To (CLI)

Router (vdom) #

Router # config terminal

Router (config) # interface fastethernet 0/0.1

Router (config) #

encapsulation dot1Q 102

ip address 192.168.2.1 255.255.255.0

no shutdown

exit

end

Observation:-

→ Proper trunk configuration is established To make VLAN work properly.

→ Ping from one VLAN to another works properly.

VLAN 1 contains all ports which are defined

VLAN 2 contains ports which are transferred to it as it is

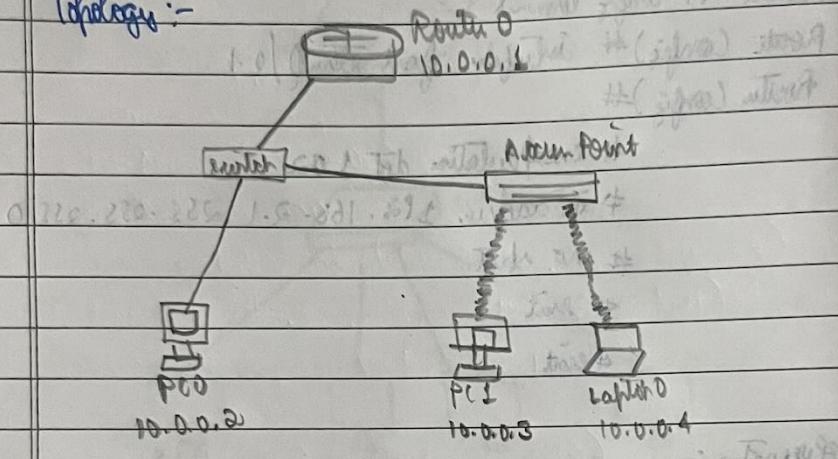
track jumbo cell size and other values

which are defined

Expt-14

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

Topology:-



Procedure:- At first take a wireless card in PC.

- ① Construct the above topology.
- Configuring PC0 and switch are normally done.
- ② Configure access point \rightarrow port \rightarrow SSID dynamic
 \rightarrow any Name (WLAN here)
- ③ Select WEP and give any 10 digit key
(#1234567890 here)

- ④ Configure PC1 and laptop with wireless standards.

- ⑤ In PC1,

Switch off the device. drag the existing PT-HUB-T-NM-14M to the component used in LTP. Drag WMP 300N within interface the empty port

Edit switch on the device.

- ⑥ In the config add a New configuration gateway (any IP)

- ⑦ Do similar in

- ⑧ Ping from laptop

Observation:-

- ① Router could change.
- ② Signal strength
- ③ Pairing is successful.

- ① In the config tab a new wireless interface would have been added. New configuration BSSID, WEP key, IP address and gateway (as normally done) to the device.
- ② Do similar in laptop.

③ Ping from my device to my other laptop. The result.

* Before taking config
"0.0.0.0.01" = configuration

Observation:-

- ① Device could connect to WLAN as long as they are in the network range.
- ② Signal strength decrease with increase in distance.
- ③ Ping is successful.

(Chennai, outside) ~~in~~ (inside room)

(Chennai, outside) ~~in~~ (inside room) - distance 10m

(Chennai, outside) ~~in~~ (inside room)

(Chennai, outside) ~~in~~ (inside room)

(Chennai, outside) ~~in~~ (inside room)

After taking config

"0.0.0.0.01" = configuration

"0.0.0.0.01" = configuration

ASAP → 0.0.0.0.01 Tab 1 = tab 2 more

(Chennai, outside) Tab 1. Tab 2 more

(Chennai, outside) Tab 1. Tab 2 more

: 1 distn

("Signal strength is same with") distn

(Chennai, outside) Tab 1. Tab 2 more = (Chennai, outside) Tab 1. Tab 2 more

(Chennai, outside) Tab 1. Tab 2 more = (Chennai, outside) Tab 1. Tab 2 more

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("Enter File Name :")

clientSocket.send(sentence.encode())

fileContent = clientSocket.recv(1024).decode()

print("\nFrom Server : " + sentence)

print(fileContent)

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, address = serverSocket.accept()

sentence = connectionSocket.recv(1024).decode()

Output :-

From in order

Requesting

Request sent

(Client)

Client in

Entered file

Received

Hello-World

file = open('sentence', 'r')

i = file.readline()

connectionSocket.send(s.encode())

print("I'm sent contents of " + sentence)

file.close()

connectionSocket.close()

Output :-

0001 = 109 words

server is online

Requesting for file : Test.txt

Request sent

Client is connected to server

Enter file name : Test.txt

Received response

Hello-World

(("8-10")) wrote. Test.txt

: Test.txt file is a file.

(("8-10")) wrote. Test.txt

:(Test.txt file is a file.)

20.9.2011 10:03:27

"Typing Test.txt

000.8 / - Test.txt

(("8-10")) wrote. Test.txt

(("8-10")) wrote. Test.txt

(("8-10")) wrote. Test.txt

: I didn't

understand. That's why I am writing this sentence

(8-10))

(("8-10")) wrote. Test.txt

2. Using UDP socket, 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 ("Enter file name : ")

client socket. sendto (Sentence, ("127.0.0.1", 12000))

(Server Name, Server Port))

file content, Server Address = client socket. recvfrom

print ("n Reply from server : n")

print (file content. decode ("utf-8"))

for i in file content :

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")

which 1:

Sentence, client Address = server socket. recvfrom

(2048)

Sentence = Sentence. decode ("utf-8")

file =

Com =

Enter bo

(2048) print ("")

print ("")

for

for

file .

Output :-
"server op :-
Server is Online
Hello Server

Client OP
Hello Client

GE NO :

TE :

To

SERIN

file

DREAM

"lf-8")
out))

2. New
from
(2048)

PAGE NO :

DATE :

file = open("sentences", "r")

con = file.read(2048)

sum socket . sendto (byt (con,"lf-8"), client address)
print ("Content of", end = '')
print (content)
for i in sentence:
print (8th(i), end = '')
file.close() (2) time = 2048

Output :-

"sum 016
sum in 016"

full sum

Client O/P

Hello Client

1=22,09

777700=3,03

(Client side) initial - it - no buffer - 06

(Client side) initial - it - no buffer - 06

2 "Hello world" = 12,00

it - no + initial - it -

note - (it : 00 - it - initial) - initial - no + initial - it

: 11 > (00 - it - initial - it -) -

why note

- no + initial - it -

[H - 100 - it - initial - initial - it -]

[it - initial - initial - initial - it -]

[initial - initial - initial - it -]

(initial - initial - initial - it -)

New from
(2048)

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

```
def crc_ccitt_16_bitstream (bitstream: str, 
    poly: int = 0x1021, init_crc: int = 0xffff)
    → int:
        crc = init_crc
        for bit in bitstream:
            crc ^= int(bit) << 1
            if crc & 0x8000:
                crc = (crc << 1) ^ poly
            else:
                crc <<= 1
        return crc
```

```
def append_crc_16_bitstream (bitstream: str)
    → str:
        crc = crc_ccitt_16_bitstream (bitstream)
        crc_hex = f'{crc:016x}'[2:]
        bitstream += crc_hex
```

```
def verify_crc_bitstream (bitstream_with_crc: str) → bool:
    if len(bitstream_with_crc) < 16:
        return False
    data, received_crc =
        bitstream_with_crc[:-16], bitstream_with_crc[-16:]
    calculated_crc =
        crc_ccitt_16_bitstream (data)
    return received_crc == calculated_crc
```

int (crc)
output:-
Enter bitstream
111101
The Transmitted
Enter the received
No error in data

return calculated - $\text{rc} = \frac{-\text{sum of first } t}{\text{int (second - rc, 2)}}$

Output:-

Enter 4 message

111101

The transmitted message is : 1111110101011111

Enter the received message in binary : 111101

No error in data

the words

Passing - per

On - fig - think

:(Cause of error is - self)

water fig - think = self - word

: self - word = > self - self - self - self

word - self - self = + self - self

self

(" self - self - self - self = word , ticket ? ") ticket

Ticket of two words = word . ticket ? + ticket

- ticket ? - word

(" ticket

- word - ticket = - word

: - word - self - self - self - self - self

Chinn

- self - self

Passing - self -

On - fig - think

Passing - self -

On - fig - think

4. Leaky Buckets :-

def main()

$$\text{storage} = 0$$

$$\text{No of -queries} = 9$$

$$\text{bucket_size} = 10$$

$$\text{input - pkt - size} = 9$$

$$\text{input - pkt - size} = 1$$

for - in range (no of - queries):

$$\text{size - left} = \text{bucket_size} - \text{storage}$$

$$\text{if } \text{input - pkt - size} \leq \text{size - left}: \quad \text{if } \text{input - pkt - size} < \text{size - left}:$$

$$\text{storage} += \text{input - pkt - size}$$

else:

print(f "Packet loss = {input - pkt - size}")

print(f "Buffer Size = {storage} out of bucket

size = {bucket_size}")

$$\text{storage} -= \text{output - pkt - size}.$$

if __name__ == "__main__":

main()

Output :-

$$\rightarrow \text{No of - queries} = 9$$

$$\text{Bucket_size} = 10$$

$$\text{input - pkt - size} = 4$$

$$\text{output - pkt - size} = 1$$

\checkmark $2^7 / 12 / 2^4$