Manarat International University

Spring-2020

Lab Report

Computer Networking

Course Code - CSE410

Submitted to:

Zahurul Haque

Lecturer of MIU

Department of Computer Science and Engineering

Submitted by:

Farjana Sultana Samia

ID:1743CSE00584

Submission Date:

06-july-2021

Experiment no.1
Experiment name: Configuration of VLAN Network

Introduction

This experiment is about the VLAN network. VLAN stands for Virtual Local Area Network.

VLAN

VLAN in networking is a virtual extension of LAN. A LAN is a group of computer and peripheral devices which are connected in a limited area such as school, laboratory, home, and office building. It is a widely useful network for sharing resources like files, printers, games, and other applications. VLAN is a custom network which is created from one or more local area networks. It enables a group of devices available in multiple networks to be combined into one logical network.

How VLAN Works

- Here is step by step details of how VLAN works:
- VLANs in networking are identified by a number.
- A Valid range is 1-4094. On a VLAN switch, you assign ports with the proper VLAN number.
- The switch then allows data which needs to be sent between various ports having the same VLAN.
- Since almost all networks are larger than a single switch, there should be a way to send traffic between two switches.
- One simple and easy way to do this is to assign a port on each network switch with a VLAN and run a cable between them.

How I design the VLAN network

In my network I have taken two 2960 series switches, and each switch is connected with six pc's. They are pc0,pc1,pc2,pc3,pc4,pc5 .Switches are connected with the trunk connection.

In "switch 0" :

There are six pc's . Three of them are vlan 10 group and the other three pc's are group of vlan 20. Pc0,pc1,pc2 are a group of vlan 10 and cse_ashulia users . Their ip addresses are 192.168.10.2, 192.168.10.3, 192.168.10.4 respectively . Their network is 192.168.10.0 and the subnet mask is 255.255.255.0 . And their Ethernet connection to switch 0 is 0/1,0/2,0/3 respectively.

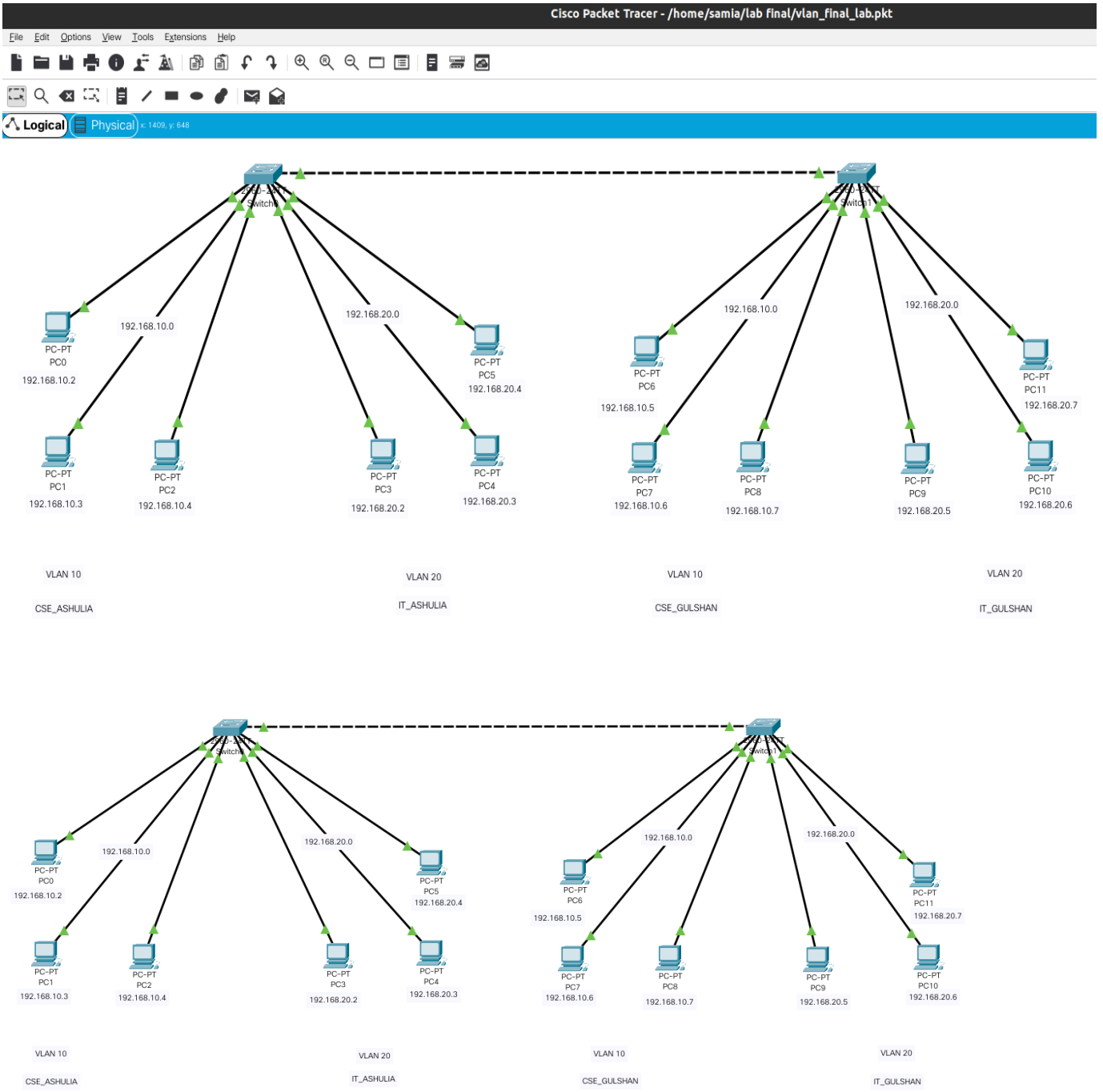
Another three switches pc3,pc4,pc5 are a group of vlan 20 and it_ashulia users . Their ip addresses are 192.168.20.2, 192.168.20.3, 192.168.20.4 respectively. Their network is 192.168.20.0 and the subnet mask is 255.255.255.0 . And their Ethernet connection to switch 0 is 0/4,0/5,0/6 respectively.

In "switch 1" :

There are also six pc's here . Three of them are vlan 10 group and the other three pc's are group of vlan 20. Pc6,pc7,pc8 are a group of vlan 10 and cse_ashulia users . Their ip addresses are 192.168.10.5, 192.168.10.6, 192.168.10.7 respectively . Their network is 192.168.10.0 and the subnet mask is 255.255.255.0 . And their Ethernet connection to switch 0 is 0/1,0/2,0/3 respectively.

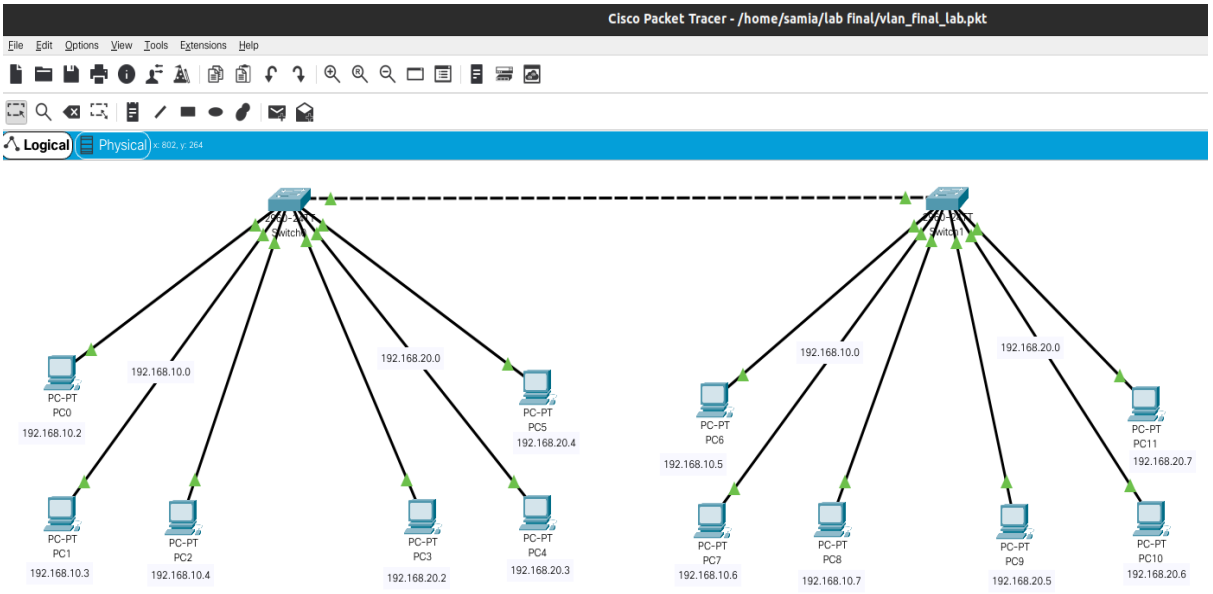
Another three switches pc9,pc10,pc11 are a group of vlan 20 and it_ashulia users . Their ip addresses are 192.168.20.5, 192.168.20.6, 192.168.20.7 respectively. Their network is 192.168.20.0 and the subnet mask is 255.255.255.0 . And their Ethernet connection to switch 1 is 0/4,0/5,0/6 respectively.

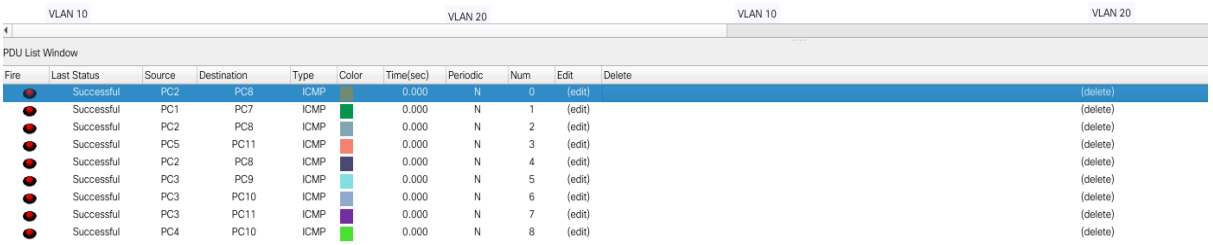
VLAN Network Design Screenshot



Result

cse_ashulia users can successfully send their message to cse_gulshan and vice versa. cse_ashulia and cse_gulshan wont send any message to it_ashulia and it_gulshan. Similarly , it_ashulia users can successfully send their message to it_gulshan and vice versa. it_ashulia and it_gulshan wont send any message to cse_ashulia and cse_gulshan.





Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC2	PC8	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC1	PC7	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC2	PC8	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC5	PC11	ICMP		0.000	N	3	(edit)	(delete)
	Successful	PC2	PC8	ICMP		0.000	N	4	(edit)	(delete)
	Successful	PC3	PC9	ICMP		0.000	N	5	(edit)	(delete)
	Successful	PC3	PC10	ICMP		0.000	N	6	(edit)	(delete)
	Successful	PC3	PC11	ICMP		0.000	N	7	(edit)	(delete)
	Successful	PC4	PC10	ICMP		0.000	N	8	(edit)	(delete)

The Command I have used for designing VLAN Network

“Switch 0”

```
Switch>en
Switch#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#hostname SW1

SW1(config)#vlan 10
SW1(config-vlan)#name cse_ashulia
SW1(config-vlan)#exit

SW1(config)#vlan 20
SW1(config-vlan)#name it_ashulia
SW1(config-vlan)#exit

SW1(config)#interface fastEthernet 0/1
SW1(config-if)#switchport access vlan 10
SW1(config-if)#exit

SW1(config)#interface fastEthernet 0/2
SW1(config-if)#switchport access vlan 10
SW1(config-if)#exit

SW1(config)#interface fastEthernet 0/3
SW1(config-if)#switchport access vlan 10
SW1(config-if)#exit

SW1(config)#interface fastEthernet 0/4
SW1(config-if)#switchport access vlan 20
SW1(config-if)#exit

SW1(config)#interface fastEthernet 0/5
SW1(config-if)#switchport access vlan 20
SW1(config-if)#exit

SW1(config)#interface fastEthernet 0/6
SW1(config-if)#switchport access vlan 20
SW1(config-if)#exit

SW1(config)#interface fastEthernet 0/7
SW1(config-if)#switchport mode trunk
SW1(config-if)#exit
```

“Switch 1”

```
Switch>en
Switch#config t
Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#hostname SW2

SW2(config)#vlan 10
SW2(config-vlan)#name cse_gulshan
SW2(config-vlan)#exit

SW2(config)#vlan 20
SW2(config-vlan)#name it_gulshan
SW2(config-vlan)#exit

SW2(config)#interface fastEthernet 0/1
SW2(config-if)#switchport access vlan 10
SW2(config-if)#exit

SW2(config)#interface fastEthernet 0/2
SW2(config-if)#switchport access vlan 10
SW2(config-if)#exit

SW2(config)#interface fastEthernet 0/3
SW2(config-if)#switchport access vlan 10
SW2(config-if)#exit

SW2(config)#interface fastEthernet 0/4
SW2(config-if)#switchport access vlan 20
SW2(config-if)#exit

SW2(config)#interface fastEthernet 0/5
SW2(config-if)#switchport access vlan 20
SW2(config-if)#exit

SW2(config)#interface fastEthernet 0/6
SW2(config-if)#switchport access vlan 20
SW2(config-if)#exit
```

Experiment no.2
Experiment name: Configuration of RIP Network

Introduction

This experiment is about the RIP network. RIP stands for Routing Information Protocol.

RIP

Routing Information Protocol (RIP) is a protocol that routers can use to exchange network topology information. It is characterized as an interior gateway protocol, and is typically used in small to medium-sized networks. A router running RIP sends the contents of its routing table to each of its adjacent routers every 30 seconds. When a route is removed from the routing table, it is flagged as unusable by the receiving routers after 180 seconds, and removed from their tables after an additional 120 seconds.
Routing Information Protocol (RIP) is a dynamic routing protocol which uses hop count as a routing metric to find the best path between the source and the destination network.

How RIP Works

RIP requires information about locally available networks. On the first step, we add this information and activate the RIP routing protocol on routers of the network. Once configured and activated, each router sends the routing update out of all active interfaces every 30 seconds.

Each router also receives routing updates from its neighboring routers. A routing update contains the entire routing table of the sending router. Routers compare the received routing tables with their routing tables. If they find any new route in the received routing tables, they add them to their routing tables.

In the next routing update, routers advertise the updated routing tables. Over time, as each router learns more routes, they advertise about those routes as well. By the end of the process, all routers know about all routes.

How I design the RIP network

In this network I have used two generic routers. Each router consists of one 2950-24 series switch and each switch connected with five users or pc's . Two routers are connected with trunk connection and routers network is 192.168.50.0 and their subnet mask is 255.255.255.252 .

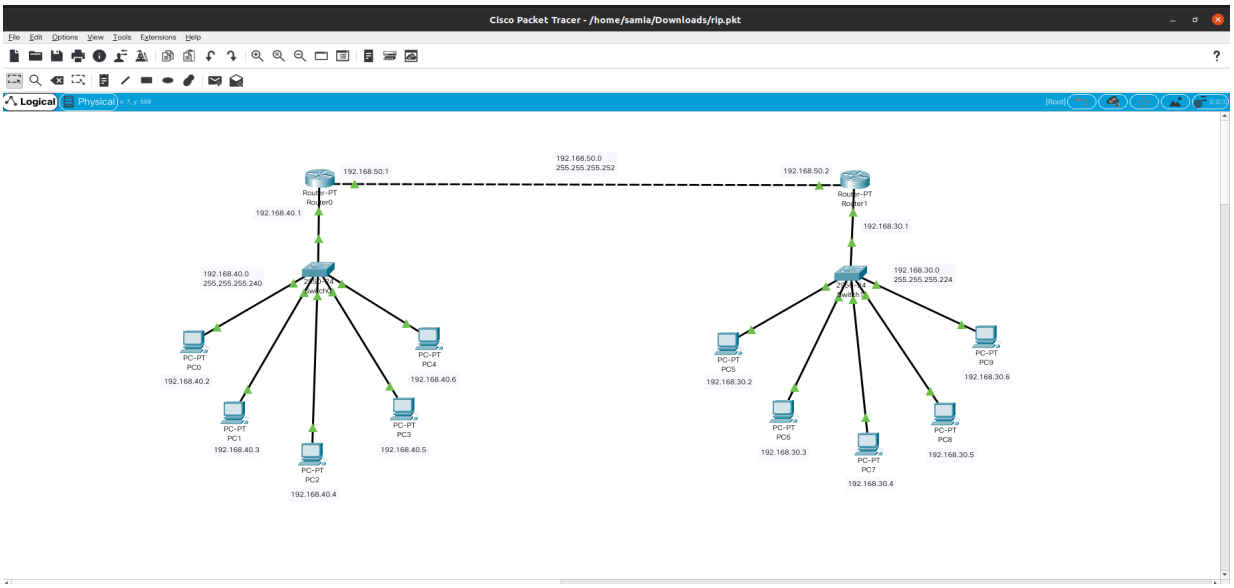
For "Router 0"

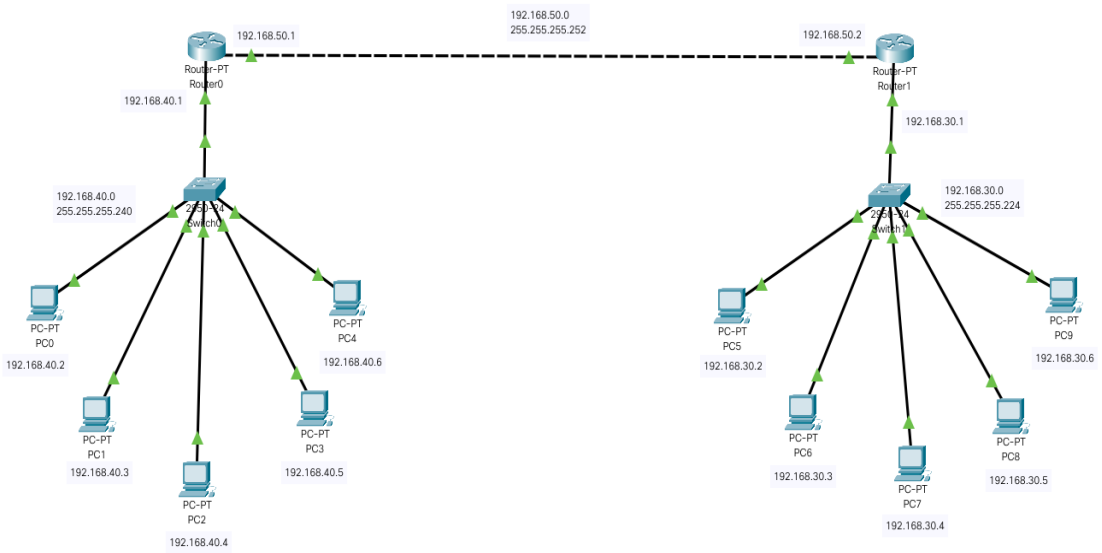
The ip address of "router 0" is 192.168.50.1 and the "switch 0" ip address is 192.168.40.1 . switch 0 is connected with five pc's . They are pc0,pc1,pc2,pc3,pc4. Their ip addresses are 192.168.40.2, 192.168.40.3, 192.168.40.4, 192.168.40.5, 192.168.40.6 respectively. Their network is 192.168.40.0 and the subnet mask is 255.255.255.240 And their Ethernet connection to switch 0 is 0/2,0/3,0/4,0/5,0/6 respectively.

For "Router 1"

The ip address of "router 1" is 192.168.50.2 and the "switch 1" ip address is 192.168.30.1 . switch 0 is connected with five pc's . They are pc5,pc6,pc7,pc8,pc9. Their ip addresses are 192.168.30.2, 192.168.30.3, 192.168.30.4, 192.168.30.5, 192.168.30.6 respectively. Their network is 192.168.30.0 and the subnet mask is 255.255.255.224 And their Ethernet connection to switch 1 is 0/2,0/3,0/4,0/5,0/6 respectively.

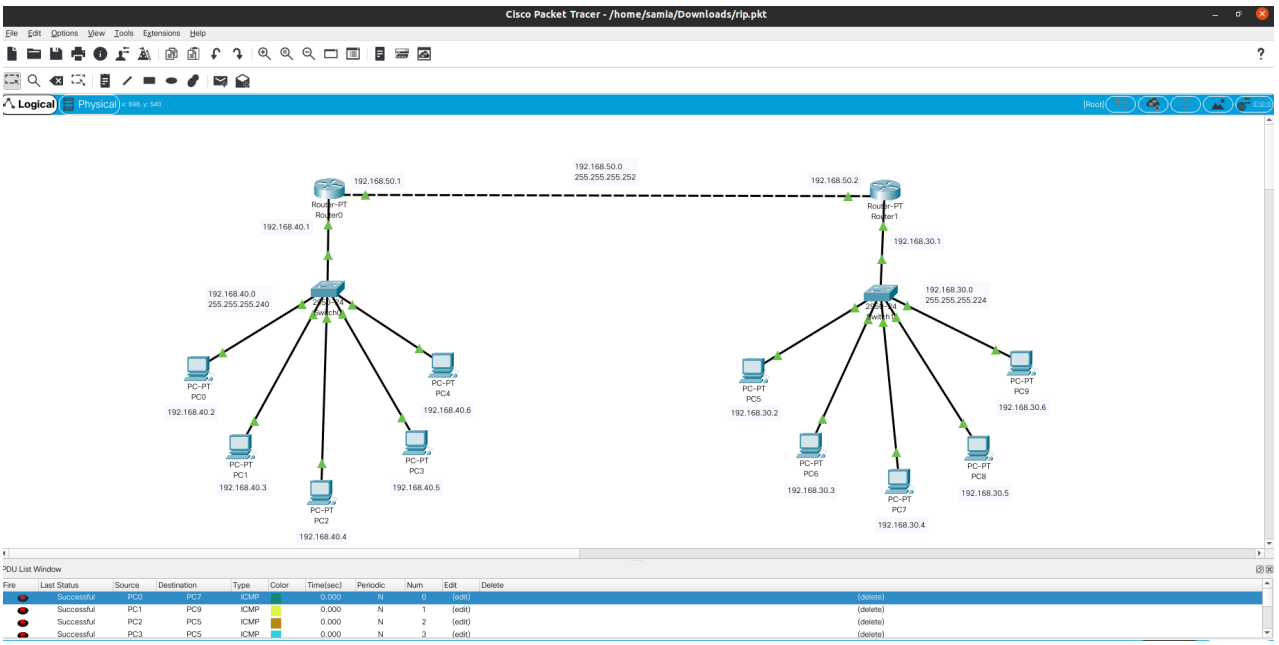
RIP Network Design Screenshot





Result

In this Routing Information Protocol (RIP) network any one of the user or pc can send messages to any other user or pc successfully .



PDU List Window

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC0	PC7	ICMP		0.000	N	0	(edit)	
	Successful	PC1	PC9	ICMP		0.000	N	1	(edit)	
	Successful	PC2	PC5	ICMP		0.000	N	2	(edit)	
	Successful	PC3	PC5	ICMP		0.000	N	3	(edit)	
	Successful	PC4	PC6	ICMP		0.000	N	4	(edit)	
	Successful	PC1	PC8	ICMP		0.000	N	5	(edit)	
	Successful	PC3	PC5	ICMP		0.000	N	6	(edit)	
	Successful	PC3	PC2	ICMP		0.000	N	7	(edit)	
	Successful	PC3	PC4	ICMP		0.000	N	8	(edit)	
	Successful	PC7	PC8	ICMP		0.000	N	9	(edit)	
	Successful	PC6	PC9	ICMP		0.000	N	10	(edit)	
	Successful	PC5	PC8	ICMP		0.000	N	11	(edit)	
	Successful	PC0	PC7	ICMP		0.000	N	12	(edit)	
	Successful	PC2	PC7	ICMP		0.000	N	13	(edit)	

The Command I have used for designing RIP Network

“Router 0”

Router>en **[Router 0 as R1]**
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname R1

R1(config)#interface fastEthernet 0/0
R1(config-if)#ip address 192.168.40.1 255.255.255.240
R1(config-if)#no shut

R1(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

R1(config-if)#exit

R1(config)#interface fastEthernet 1/0
R1(config-if)#ip address 192.168.50.1 255.255.255.252
R1(config-if)#no shut

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet1/0, changed state to up

R1(config-if)#exit

R1(config-if)#router rip
R1(config-router)#network 192.168.40.0
R1(config-router)#network 192.168.50.0
R1(config-router)#exit

“Router 1”

Router>en **[Router 1 as R2]**
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname R2

R2(config)#interface fastEthernet 0/0
R2(config-if)#ip address 192.168.30.1 255.255.255.224
R2(config-if)#no shut

R2(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

R2(config-if)#exit

R2(config)#interface fastEthernet 1/0
R2(config-if)#ip address 192.168.50.2 255.255.255.252
R2(config-if)#no shut

R2(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

```
R2(config-if)#exit

R2(config-if)#router rip
R2(config-router)#network 192.168.30.0
R2(config-router)#network 192.168.50.0
R2(config-router)#exit
```