Procedure of CNC Experiment-8

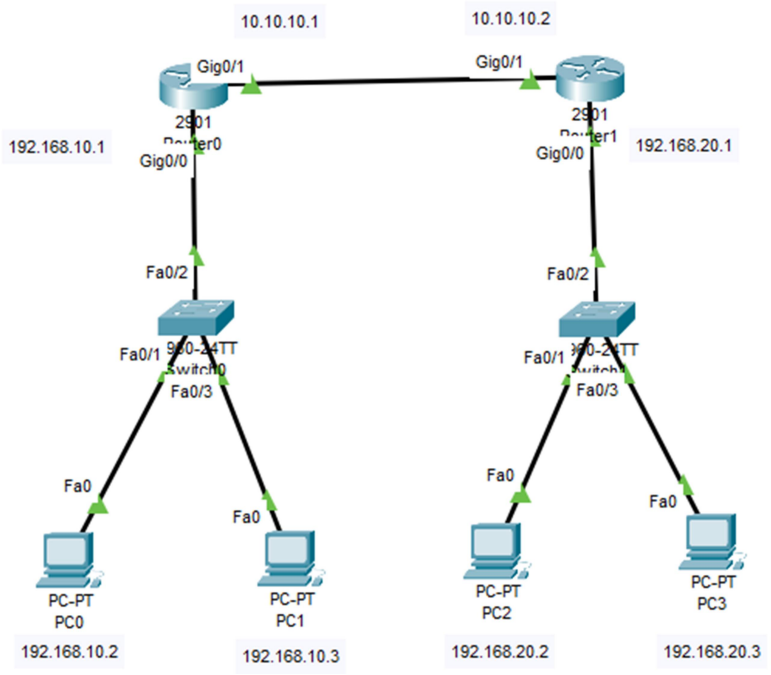
****

Aim 1:

To verify connectivity between two different networks by implementing dynamic routing using the **Routing Information Protocol** (RIP) in Cisco Packet Tracer**.**

Procedure

Step 1: Create Network Topology and do the following basic **c**onfigure PCs as shown in the diagram. (Include Default Gateway)



Step 2: Configure the Router (R0 and R1) interfaces as shown in the diagram.

On Router0 (R0):

```
Router>en
Router#config t
Router(config)#int gig0/0
Router(config-if)#ip address 192.168.10.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#int gig0/1
Router(config-if)#ip address 192.168.20.1 255.255.255.0
Router(config-if)#ip address 10.10.10.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

On Router1 (R1):

```
Router>en
Router#config t
Router(config)#int gig0/0
Router(config-if)#ip address 192.168.20.1 255.255.255.0
Router(config-if)#no shutdown
```

Router(config-if)#exit
Router(config)#int gig0/1
Router(config-if)#ip address 10.10.10.2 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit

Step 3: Configure RIP Dynamic Routing

The descriptions of the commands used are:

router rip: Enables the RIP routing protocol on the router.

version 2: Enables RIP version 2**.**

network <network address>: Specifies the networks that will participate in RIP. RIP advertises these networks to neighboring routers.

no auto-summary: Disables automatic classful route summarization.

On Router0 (R0)

Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 192.168.10.0
Router(config-router)#network 10.10.10.0
Router(config-router)#no auto-summary
Router(config-router)#exit

On Router1 (R1)

Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 192.168.20.0
Router(config-router)#network 10.10.10.0
Router(config-router)#no auto-summary
Router(config-router)#exit

Step 4: Verify Routing Tables

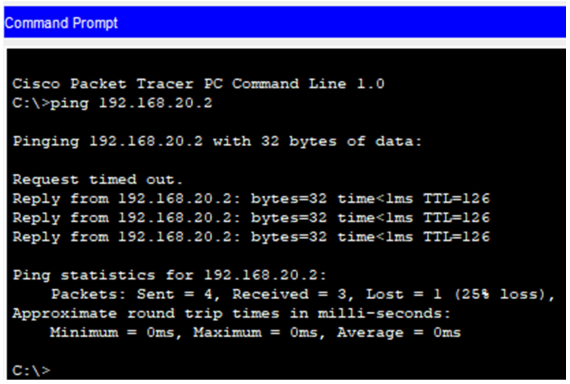
On both routers: show ip route

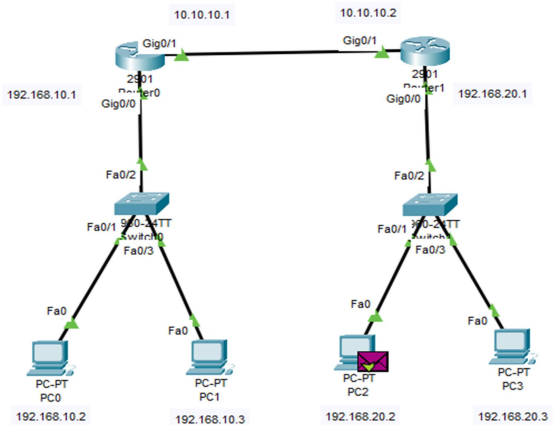
Routes learned via RIP are marked with **R**.

Step 5: Test Connectivity

1. From **PC1**, ping **PC2**: ping 192.168.20.2
2. From **PC2**, ping **PC1**: ping 192.168.10.2

Successful replies confirm connectivity.





Aim 2:

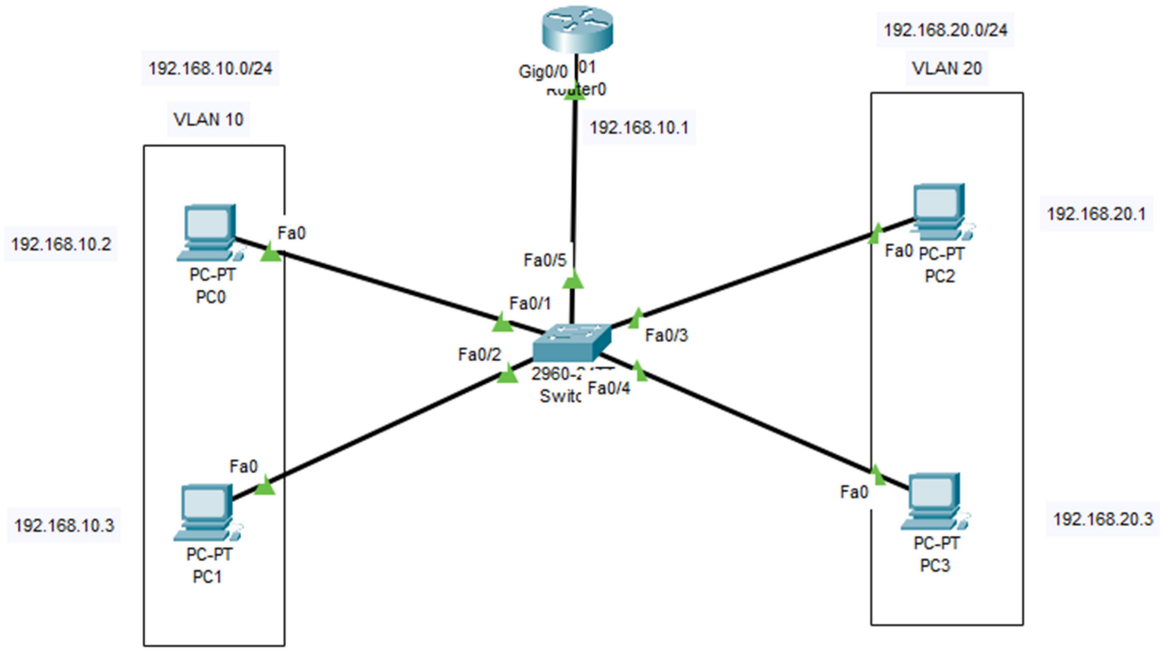
Configuring VLANs using layer 2 switch and verifying the connectivity between hosts using inter VLAN routing.

- Create multiple VLANs on a **Layer-2 switch**
- Assign switch ports to VLANs
- Configure **Inter-VLAN Routing using a router**
- Verify connectivity **within VLANs** and **between VLANs**

Why VLAN + Inter-VLAN Routing?

- A **Layer-2 switch** can **create VLANs** but **cannot route between them**
- Hosts in **different VLANs cannot communicate directly**
- To make VLANs talk to each other, we use **Inter-VLAN Routing**
- In Packet Tracer, this is done using: **Router-on-a-Stick method**

Network Topology



Step-by-Step Implementation (No Skipping)

STEP 1: Place Devices in Packet Tracer

1. Open **Cisco Packet Tracer**
   - Add: **1 Router, 1 Switch** (2960 – Layer-2) and **4 PCs**
2. Connect:
   - PCs → Switch (straight-through)
   - Switch Fa0/5 → Router Gig0/0

STEP 2: Assign IP Address to PCs

VLAN 10

- **PC0**
  - IP: 192.168.10.2
  - Subnet: 255.255.255.0
  - Gateway: 192.168.10.1
- **PC1**
  - IP: 192.168.10.3
  - Gateway: 192.168.10.1

VLAN 20

- **PC2**
  - IP: 192.168.20.2
  - Gateway: 192.168.20.1
- **PC3**
  - IP: 192.168.20.3
  - Gateway: 192.168.20.1

At this point nothing will ping (this is expected)

STEP 3: Create VLANs on Layer-2 Switch

Switch CLI

```
Switch>en
Switch#config t
Switch(config)#vlan 10
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#exit
Switch#show vlan brief
```

STEP 4: Assign Switch Ports to VLANs

Here: PC0 → Fa0/1, PC1 → Fa0/2, PC2 → Fa0/3, PC3 → Fa0/4

```
Switch#config t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#interface range fa0/1 - 2
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
Switch(config-if-range)#exit

Switch(config)#interface range fa0/3 - 4
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 20
Switch(config-if-range)#exit
```

Now:

- **PC0 ↔ PC1 can ping**
- **PC2 ↔ PC3 can ping**
- **Between VLANs → NO communication**

STEP 5: Configure Trunk Port (MOST IMPORTANT)

Switch port connected to router must carry **both VLANs**

```
Switch(config)#interface fa0/5
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
```

STEP 6: Configure Inter-VLAN Routing on Router

This is the **key concept**

Router CLI

```
Router>en
Router#config t
Router(config)#interface gig0/0
Router(config-if)#no shutdown
Router(config-if)#exit

Router(config)#interface gig0/0.10
Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip address 192.168.10.1 255.255.255.0
Router(config-subif)#exit

Router(config)#interface gig0/0.20
Router(config-subif)#encapsulation dot1Q 20
Router(config-subif)#ip address 192.168.20.1 255.255.255.0
Router(config-subif)#exit
```

Router now acts as gateway for both VLANs

STEP 7: Verification (This proves inter-VLAN routing)

From PC0: ping 192.168.20.2

From PC2: ping 192.168.10.2

✔ Ping succeeds
✔ Inter-VLAN routing working