**LAB 4: VIRTUAL LAN**

**THEORY**

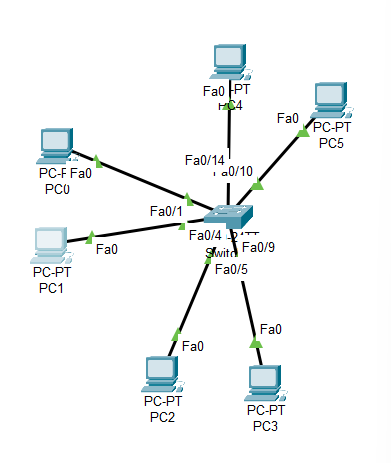
A Virtual LAN (VLAN) is a way to group devices on a network logically, not based on where they are physically located. It allows different devices on different switches to behave as if they are on the same local network. VLANs help improve network performance, security, and management.

**Purpose of VLAN**

* To divide a large network into smaller, easier-to-manage segments.
* To increase security by separating groups of users (like separating HR from IT).
* To reduce network traffic by limiting broadcast domains.

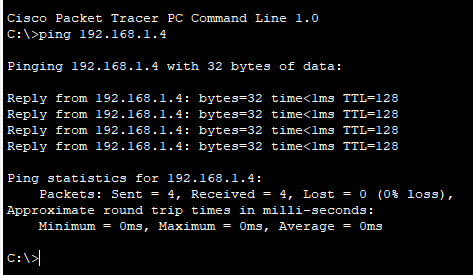
**Lab Setup**

1. Drag and place one switch into the workspace (e.g., Switch0).
2. Add 6 PCs (end devices) and connect them to Switch0 using copper straight-through cables.
3. Make sure each PC is connected to a different port on the switch.



1. Assign the following IP addresses to the 6 PCs:

|  |  |  |
| --- | --- | --- |
| PC Name | IP Address | Subnet Mask |
| PC0 | 192.168.1.1 | 255.255.255.0 |
| PC1 | 192.168.1.2 | 255.255.255.0 |
| PC2 | 192.168.1.3 | 255.255.255.0 |
| PC3 | 192.168.1.4 | 255.255.255.0 |
| PC4 | 192.168.1.5 | 255.255.255.0 |
| PC5 | 192.168.1.6 | 255.255.255.0 |

1. Open the Command Prompt on one and ping to the other device

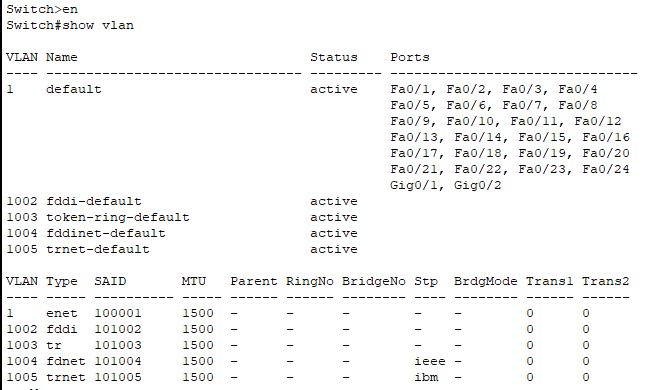
**Configure VLANs on Switch 0**

1. Enters privileged EXEC mode to allow configuration commands.

Switch> en

1. Shows the existing VLANs on the switch.

Switch# show vlan



1. Enters global configuration mode to begin VLAN setup.

Switch# config t

1. Starts configuration for VLAN 10.

Switch(config)# vlan 10

1. Names VLAN 10 as A.

Switch(config-vlan)# name A

1. Starts configuration for VLAN 20.

Switch(config)# vlan 20

1. Names VLAN 20 as B.

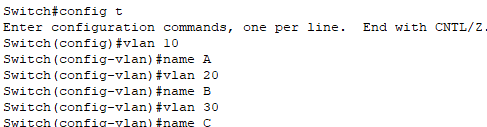
Switch(config-vlan)# name B

1. Starts configuration for VLAN 30.

Switch(config)# vlan 30

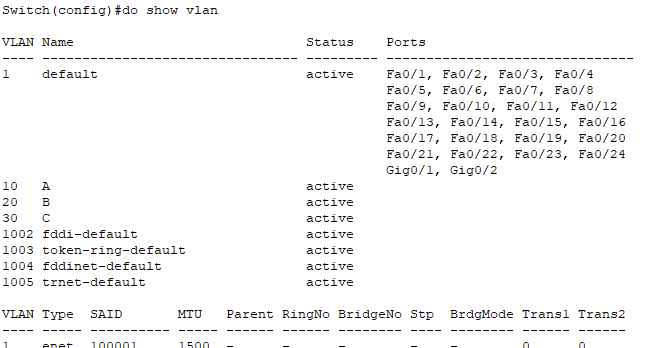
1. Names VLAN 30 as C.

Switch(config-vlan)# name C



1. Displays the updated list of VLANs including A, B, and C.

Switch(config)# do show vlan



**Assigning Ports to VLANs on Switch 0**

After creating the VLANs (10, 20, 30), we assigned specific ports to each VLAN. This helps group end devices logically even if they are connected to different ports. We did this using interface range commands and switchport access mode.

1. **Assigning Ports fa0/1 to fa0/4 to VLAN 10 (Category: 0/1–0/4)**

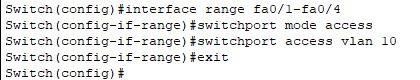
Switch(config)# interface range fa0/1 - fa0/4

Switch(config-if-range)# switchport mode access

Switch(config-if-range)# switchport access vlan 10

Switch(config-if-range)# exit

This assigns ports fa0/1 to fa0/4 to VLAN 10 (A), and sets them as access ports.



1. **Assigning Ports fa0/5 to fa0/9 to VLAN 20 (Category: 0/5–0/9)**

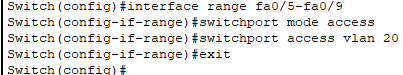
Switch(config)# interface range fa0/5 - fa0/9

Switch(config-if-range)# switchport mode access

Switch(config-if-range)# switchport access vlan 20

Switch(config-if-range)# exit

This assigns ports fa0/5 to fa0/9 to VLAN 20 (B).



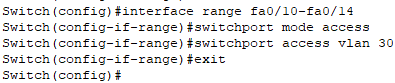
1. **Assigning Ports fa0/10 to fa0/14 to VLAN 30 (Category: 0/10–0/14)**

Switch(config)# interface range fa0/10 - fa0/14

Switch(config-if-range)# switchport mode access

Switch(config-if-range)# switchport access vlan 30

Switch(config-if-range)# exit

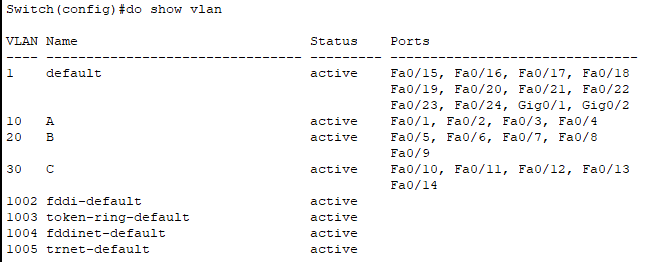


This assigns ports fa0/10 to fa0/14 to VLAN 30 (C).

1. **Verifying VLAN Configuration**

Switch(config)# do show vlan

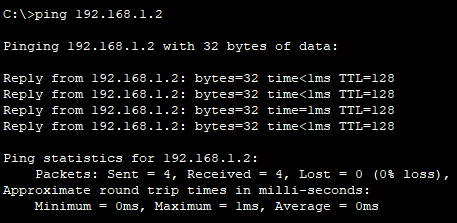
This displays the updated VLAN table, showing which ports are assigned to VLANs A, B, and C.



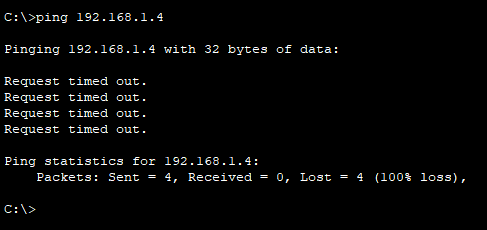
**Ping Test Results**

After configuring VLANs and assigning ports, we performed a ping test to check communication between the devices.

Ping between devices in the same VLAN was successful. This shows that devices assigned to the same VLAN can communicate with each other without any issues.



Ping between devices in different VLANs was unsuccessful. This is expected because devices in separate VLANs are logically isolated from each other, and cannot communicate unless a router or Layer 3 switch is used to route between VLANs.



**Updating IP Addresses for Different VLANs**

Since VLAN A, B, and C are logically separate networks, we assigned different IP address ranges to each VLAN:

* VLAN A (Name: A)
  + IP Range: 192.168.1.x
  + Example: 192.168.1.1, 192.168.1.2 (for two devices)
* VLAN B (Name: B)
  + IP Range: 172.16.0.x
  + Example: 172.16.0.1, 172.16.0.2
* VLAN C (Name: C)
  + IP Range: 10.0.0.x
  + Example: 10.0.0.1, 10.0.0.2

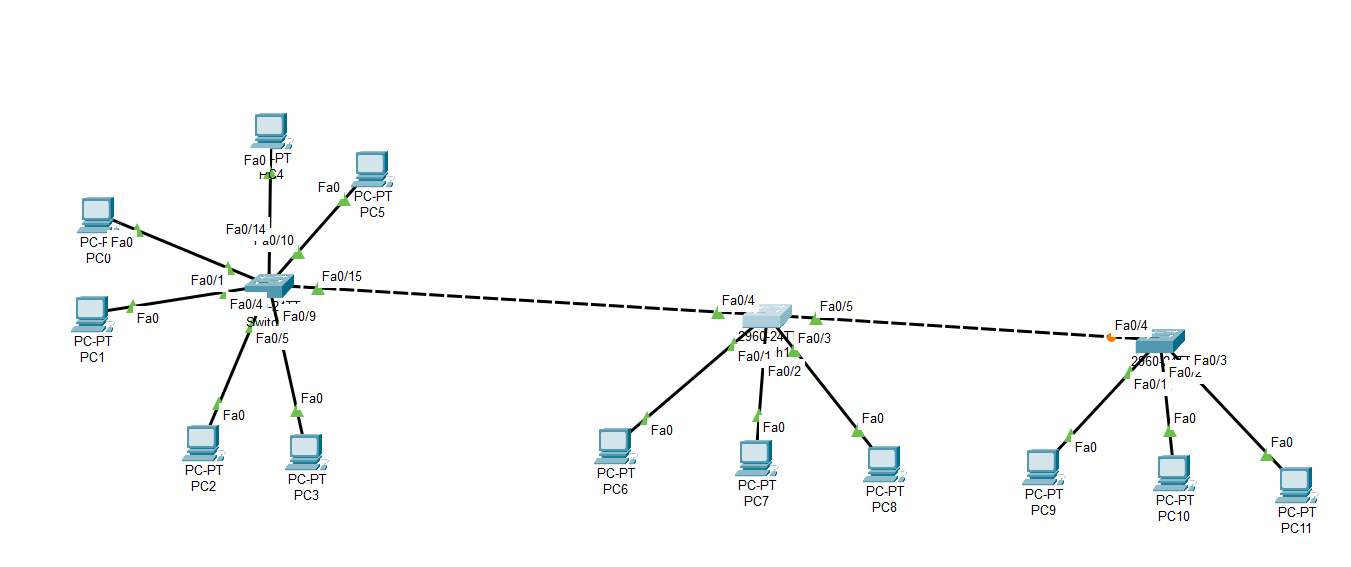
This ensures that each VLAN is on a separate IP network and cannot communicate directly with other VLANs.

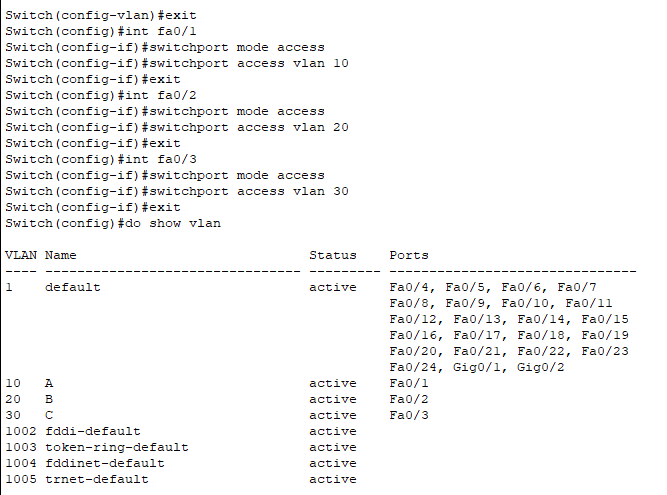
**Adding Two More Switches**

We added two additional switches (Switch 1 and Switch 2) to the network. Each switch had 3 end devices, and each device was assigned to VLAN A, B, or C, with one device per VLAN.

IP Addressing for New Devices:

* Devices in VLAN A:
  + IPs like 192.168.1.3, 192.168.1.4
* Devices in VLAN B:
  + IPs like 172.16.0.3, 172.16.0.4
* Devices in VLAN C:
  + IPs like 10.0.0.3, 10.0.0.4





Each device was connected to the correct VLAN and assigned an IP from the matching subnet.

**Trunking Between Switches**

To allow communication between the same VLANs across different switches (e.g., VLAN A devices on Switch 0, 1, and 2), we configured trunk ports. Trunking allows multiple VLANs to pass through a single switch port.

Connecting the below:

* Switch 0 to Switch 1
* Switch 1 to Switch 2

On each switch, we used FastEthernet port fa0/15 for trunking.

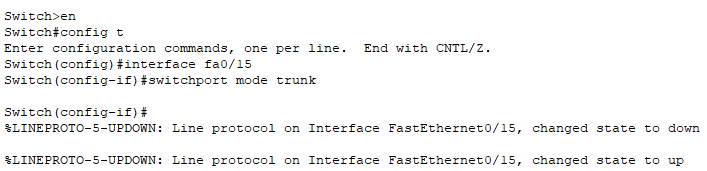
**Trunking Configuration (for switch0)**

Switch(config)# interface fa0/15

Switch(config-if)# switchport mode trunk

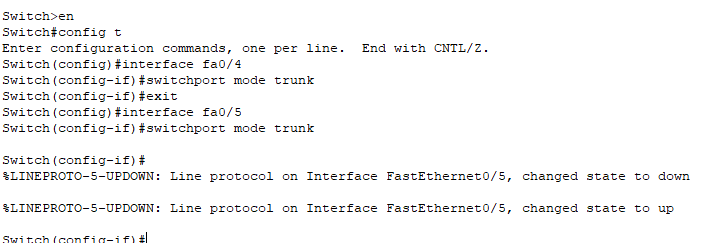
Switch(config-if)# exit

This sets fa0/15 as a trunk port, allowing VLAN traffic to pass between switches.

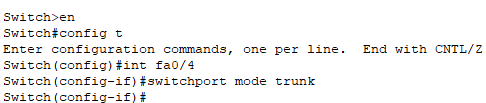


Similarly, truck mode was set for other switches too.

**For switch1,**



**For switch2,**



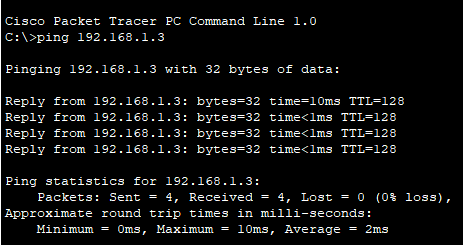
**Testing Trunking with Ping**

After configuring trunking between the switches, we performed a ping test to verify if devices in the same VLAN across different switches could communicate.

**Ping Test Result:**

* From PC1 (in VLAN A on Switch 0), we pinged a device in VLAN A on Switch 1.
* The ping was successful, which confirmed that:
* Trunking was properly configured.
* VLAN traffic was correctly passing between switches.
* Devices in the same VLAN but on different switches can now communicate.

This confirmed that trunking and VLAN setup were working as expected.



**Adding a Router for Inter-VLAN Communication (Router-on-a-Stick)**

To enable communication between different VLANs (A, B, and C), we added a router to Switch 1 using a method called Router-on-a-Stick. This allows the router to handle traffic for multiple VLANs using sub-interfaces on a single physical interface.

**Router Configuration Steps**

We used interface GigabitEthernet0/0 on the router and created sub-interfaces for each VLAN.

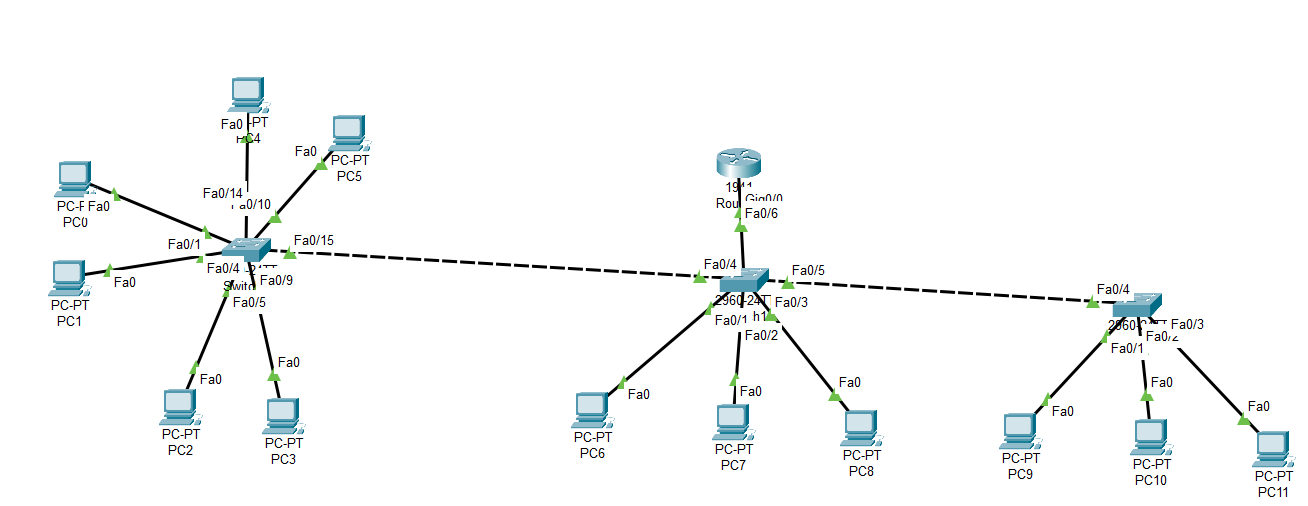
**Step 1: Enable the main interface**

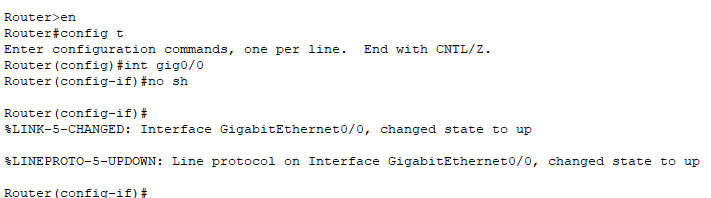
Router(config)# interface gig0/0

Router(config-if)# no sh

Router(config-if)# exit

This turns on the physical interface.





**Step 2: Create Sub-interface for VLAN 10 (A)**

Router(config)# interface gig0/0.10

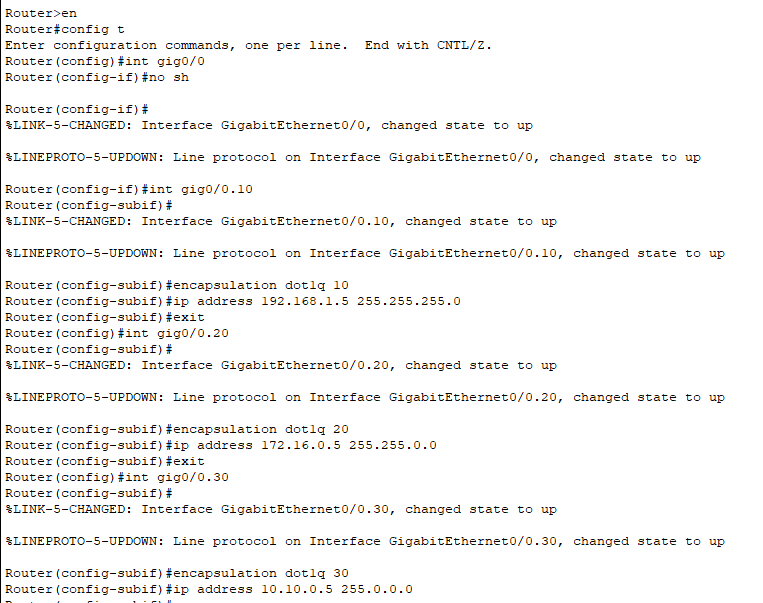
Router(config-subif)# encapsulation dot1Q 10

Router(config-subif)# ip address 192.168.1.5 255.255.255.0

Router(config-subif)# exit

This handles traffic for VLAN 10 using IP 192.168.1.5 as the gateway for that VLAN.

We now repeat similar steps for VLAN 20 and VLAN 30 using sub-interfaces gig0/0.20 and gig0/0.30 with their respective VLAN IDs and IP addresses.



**Configuring the Switch Port Connected to the Router for Trunking**

Before testing communication between VLANs, we configured the switch port connecting Switch 1 to the router as a trunk port. This allows multiple VLAN traffic to pass between the switch and the router.

**Commands on Switch 1**

Switch> enable

Switch# configure terminal

Switch(config)# interface fa0/6

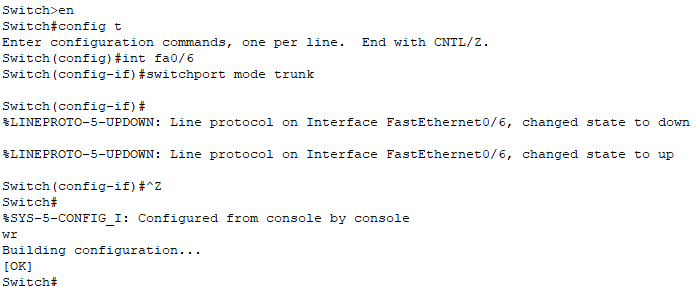
Switch(config-if)# switchport mode trunk

Switch(config-if)# exit

Switch(config)# end

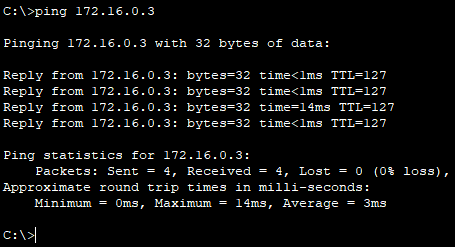
Switch# wr

switchport mode trunk makes the port carry traffic for all VLANs. wr saves the configuration.



**Inter-VLAN Ping Test**

We tested communication between different VLANs using the ping command. From the PC with IP 192.168.1.3 (in VLAN 10), we pinged the PC with IP 172.16.0.3 (in VLAN 20). The ping was successful, confirming that the router-on-a-stick configuration allowed communication between VLANs.



**DISCUSSION**

In this lab, we learned how to create and manage VLANs to separate different groups of devices on the same network. We configured VLANs on switches and assigned ports to those VLANs, which helped us logically divide the network even if devices were physically connected to the same switch. Then, by setting up trunking between switches, we allowed VLAN information to pass through a single link so devices in the same VLAN but on different switches could communicate. Finally, we connected a router using the router-on-a-stick method to enable communication between VLANs. The successful ping tests showed that devices within the same VLAN and across different VLANs (through the router) could communicate properly. This experiment helped us understand how VLANs improve network organization and security.

**CONCLUSION**

This lab demonstrated the process of setting up VLANs, trunking, and inter-VLAN routing in a simple network. By dividing the network into VLANs, we separated traffic logically to reduce congestion and improve security. Trunk ports allowed VLAN traffic to flow between switches, and the router-on-a-stick setup enabled communication between VLANs. The successful ping tests proved that the network was working as expected. Overall, this experiment gave us practical experience in managing VLANs and routing, which are important skills for building efficient and secure networks.