Computer Networks Lab 11

Course: Computer Networks (CL3001) **Semester:** Spring 2024
Instructor: Sameer Faisal **T.A:** N/A

Note:

- Maintain discipline during the lab.
- Listen and follow the instructions as they are given.
- Just raise hand if you have any problem.
- Completing all tasks of each lab is compulsory.
- Get your lab checked at the end of the session.

Lab Objective

- Introduction to Virtual Area Networks - VLANS.
- Types of Connections in VLAN.
- Introduction to InterVLAN Routing.
- Configuration of VLAN.
- Configuration InterVLAN Routing.

Virtual Area Networks

1. Introduction to Virtual Area Network

A traditional LAN comprising of workstations connected to each other by means of a hub or a repeater form a single collision and broadcast domains. Due to this, these devices propagate any incoming data throughout the network. To prevent collisions from traveling through all the workstations in the network, a bridge or a switch can be used. These devices will not forward collisions, but still will allow broadcasts and multicasts to pass through.
A router, therefore, may be used to prevent broadcasts and multicasts from traveling through different networks. To stop broadcasts in a same LAN segment, VLAN's allow a network manager to logically segment a LAN into different broadcast domains so that packets are only switched between ports that are designated for the same VLAN.

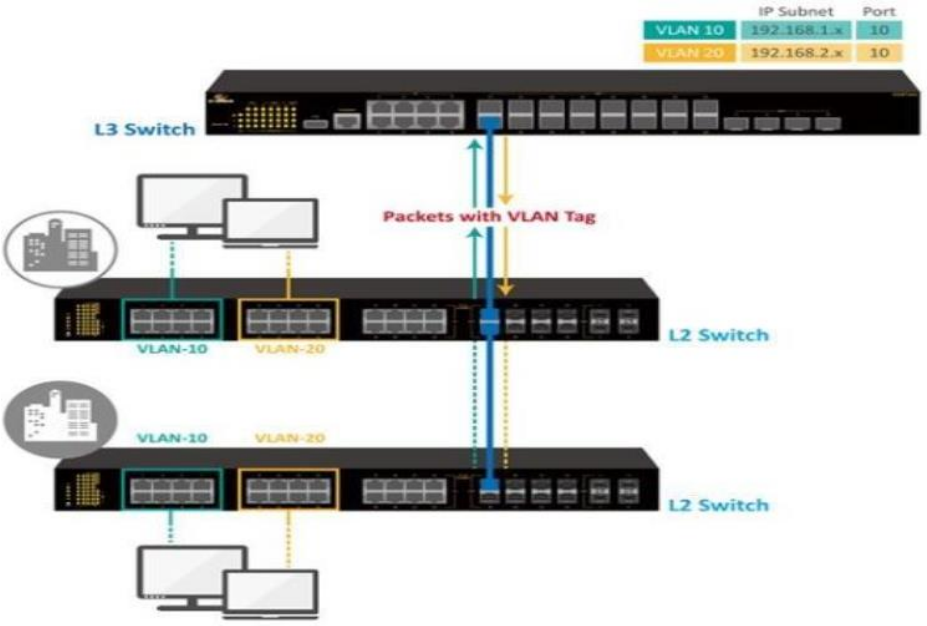
A Virtual Local Area Network can be defined as a group of networking devices in the same broadcast domain, logically.
Since this is a logical segmentation and not a physical one, it means that the devices in the same VLAN may be widely separated in the network; both by geography and location, workstations do not have to be physically located together.

VLAN helps you group users together according to their function rather than their physical location. This means Users on different floors of the same building, or even in different buildings can now belong to the same LAN. This makes the management much simpler.

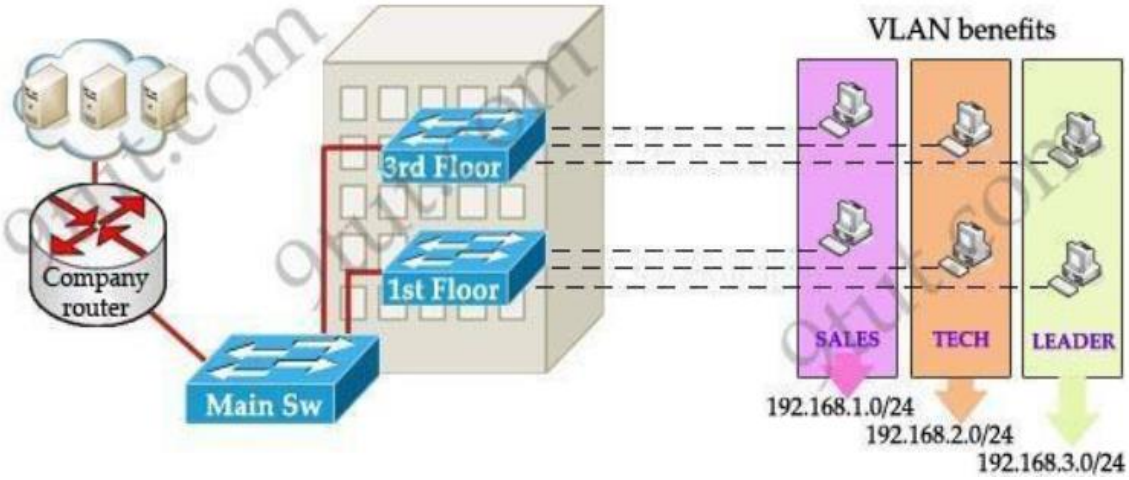
Some of the benefits of VLANs are:

1. They improve network performance by reducing the size of broadcast domains. In a broadcast domain, every device can send packets to every other device, and every packet must be received and processed. When a broadcast domain becomes very large, this can degrade the performance of switches on the network due to the high volumes of broadcast data.
2. VLANs allow for the adding of additional layers of security. For example, a specific VLAN can be created for users with specific security clearances.
3. VLANs make device management easier. If a user moves to a new physical location, the physical workstation of that user does not need to be reconfigured. Also, if a user stays in the same location but changes jobs, only the VLAN membership of the workstation needs to be changed.

For multiple VLANs to communicate with each other, a router is required. Routers between VLANs filter broadcast traffic, enhance network security, perform address summarization, and mitigate network congestion.



Real world scenario:



Take a real-world example as shown in the above figure. As VLANs break up broadcast domains, so now if a computer in Sales broadcasts, only computers in Sales will receive that frame.

It is important to point out that you don't have to configure a VLAN until your network gets so large and has so much traffic that you need one. Many times, people are simply using VLAN's because the network they are working on was already using them.

You need to consider using VLAN's in any of the following situations:
1. You have more than 200 devices on your LAN You have a lot of broadcast traffic on your LAN.
2. Groups of users need more security or are being slowed down by too many broadcasts? Groups of users need to be on the same broadcast domain because they are running the same applications.
3. An example would be a company that has VoIP phones. The users using the phone could be on a different VLAN, not with the regular users. Or, just to make a single switch into multiple virtual switches.

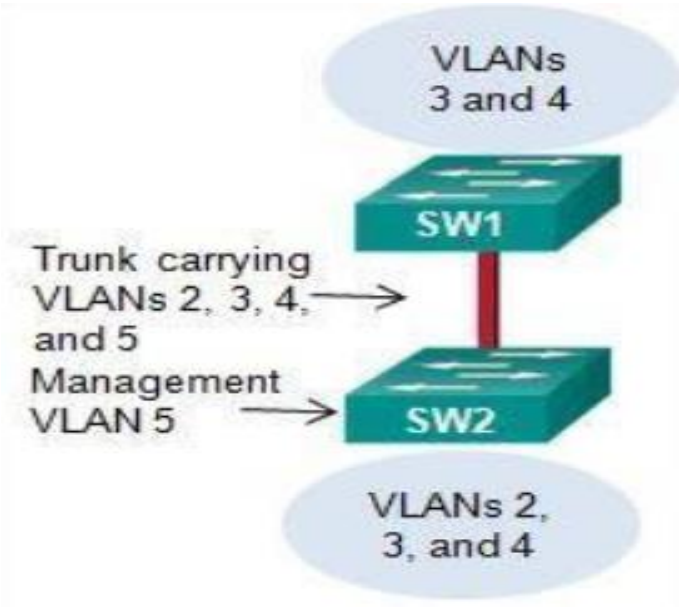
Another important fact is that, on a Cisco switch, VLAN's are enabled by default and ALL devices are already in a VLAN. The VLAN that all devices are already in is VLAN 1. So, by default, you can just use all the ports on a switch and all devices will be able to talk to one another.

2. Types of Connection in VLAN

Devices on a VLAN can be connected in three ways based on whether the connected devices are VLAN aware or VLAN-unaware. Recall that a VLAN-aware device is one which understands VLAN memberships (i.e. which users belong to a VLAN) and VLAN formats.
Below are the types of connection in VLAN. They are:

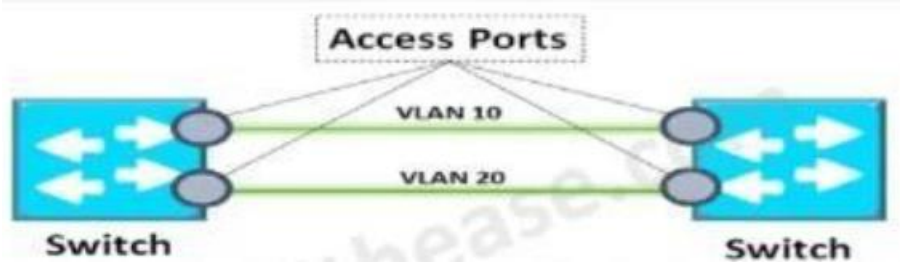
1. Trunk Link

All the devices connected to a trunk link, including workstations, must be VLAN-aware. All frames on a trunk link must have a special header attached. These special frames are called tagged frames as shown in the below figure.

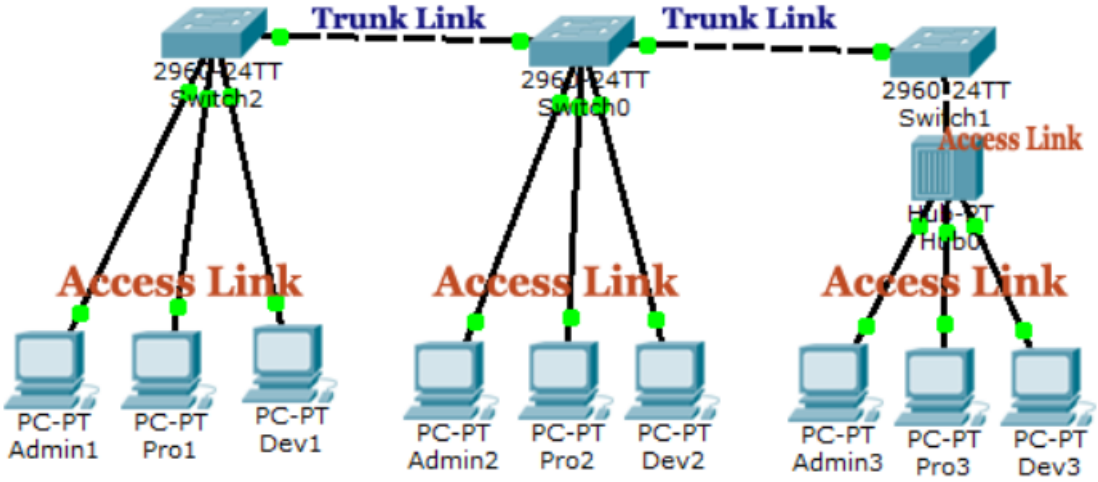


2. Access Link

An access link connects a VLAN-unaware device to the port of a VLAN-aware bridge. All frames on access links must be implicitly tagged (untagged). The VLAN-unaware device can be a LAN segment with VLAN-unaware workstations or it can be a number of LAN segments containing VLAN-unaware devices (legacy LAN).



The combine pictorial view of Access and Trunk link is given in the below figure:

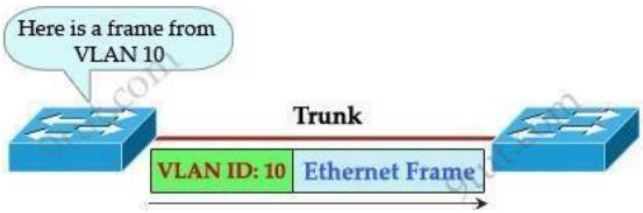


Communication in VLAN

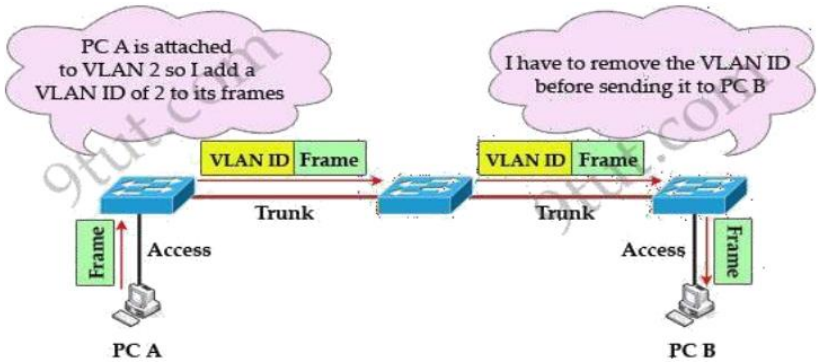
Hosts in the same VLAN can communicate normally even they are connecting to 2 or more different switches. When using multiple VLANs in networks that have multiple interconnected switches, we need to use VLAN Trunking between the switches. With VLAN trunking, the switches tag each frame sent between switches so that the receiving switch knows which VLAN the frame belongs to.
This tag is known as a VLAN ID. A VLAN ID is a number which is used to identify a VLAN.

3. Introduction to InterVLAN Routing

To enable different VLANs to communicate with each other need a router. Without a router, the computers within each VLAN can communicate with each other but not with any other computers in another VLAN. For example, we need a router to transfer file from LEADER to TECH.
This is called “inter-VLAN routing”.



The tag is only added and removed by the switches when frames are sent out on the trunk links. Hosts don't know about this tag because it is added on the first switch and removed on the last switch. The figure below describes the process of a frame sent from PC A to PC B.



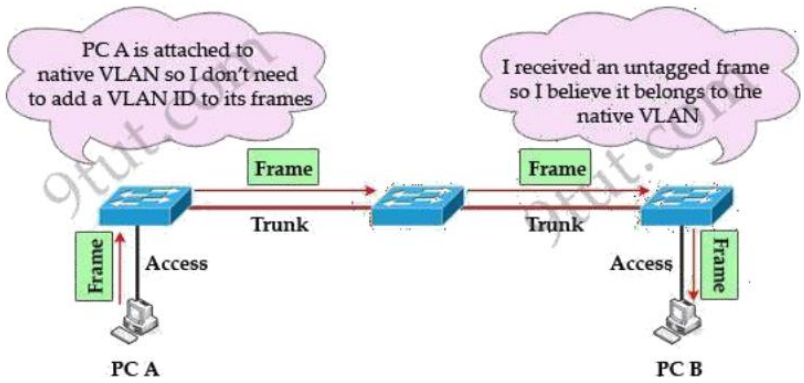
Note: Trunk link does not belong to a specific VLAN; rather it is a conduit for VLANs between switches and routers.

To allow inter-VLAN routing you need to configure trunking on the link between router and switch. Therefore, in our example we need to configure 3 links as "trunk".

Cisco switches support two different trunking protocols, Inter-Switch Link (ISL) and IEEE 802.1q. Cisco created ISL before the IEEE standardized trunking protocol. Because ISL is Cisco proprietary, it can be used only between two Cisco switches. 802.1q is usually used in practical.

In 802.1q encapsulation, there is a concept called native VLAN that was created for backward compatibility with old devices that don't support VLANs. Native VLAN works as follows:

1. Frame belonging to the native VLAN is not tagged when sent out on the trunk links.
2. Frame received untagged on the trunk link is set to the native VLAN.



So if an old switch doesn't support VLAN it can still "understand" that frame and continue sending it (without dropping it).

Every port belongs to at least one VLAN. If a switch receives untagged frames on a trunkport, they are assumed to be part of the native VLAN. By default, VLAN 1 is the default and native VLAN but this can be changed on a per port basis by configuration.

4. Configuration of VLAN

Creating a VLAN:

Step 1: Enter privileged EXEC mode

Switch>enable

Step 2: Enter global configuration mode.

Switch#config terminal

Step 3: Create VLAN

Switch(config)#vlan X (X can be any natural number)

Step 4: Give name to VLAN

Switch(config-vlan)#name XYZ (Name of VLAN)

Notice that we don't need to exit out of "vlan mode" to create another VLAN.

Setting VLAN Membership:

Assign VLAN to each port:

Step 5: Enter interface configuration mode.

Switch(config)#interface type port(int fa0/1)

Step 6: Set the mode of port as trunk or access

Switch(config-if) #switchport mode access/trunk (access when pc-switch else trunk)

Step 7: If port is in access mode, assign a VLAN to the port.

Switch(config-if) #switchport access vlan-number

Notice that for port connecting to host we must configure it as access port

5. Configuration of InterVLAN Routing

Step 8: Enter interface configuration mode.

Router(config)#interface type port

Step 9: Enter sub-interface configuration mode.

Router(config-if)#interface type port.subport

Step 10: Set the ip address of the subinterface.

Router(config-subif)#ip address X.X.X.X Y.Y.Y.Y

Step 11: Set the encapsulation type and vlan allowed on sub-interface.

Router(config-subif)# encapsulation dot1qvlan number

6. Example Topology and Its Configuartion

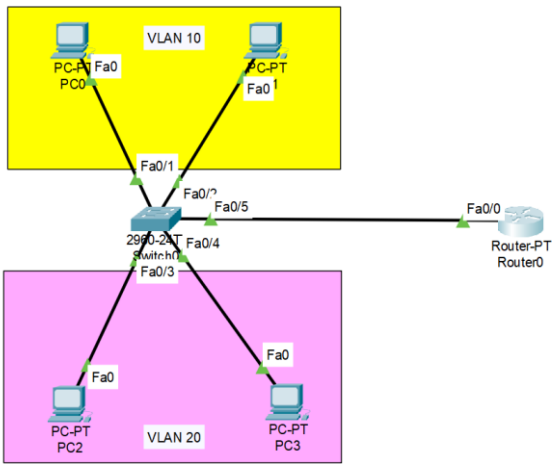
A Virtual LAN (VLAN) is simply a logical LAN, just as its name suggests. VLANs have similar characteristics with those of physical LANs, only that with VLANs, you can logically group hosts even if they are physically located on separate LAN segments.

We treat each VLAN as a separate subnet or broadcast domain. For this reason, to move packets from one VLAN to another, we have to use a router or a layer 3 switch.

VLANs are configured on switches by placing some interfaces into one broadcast domain and some interfaces into another. For this tutorial, we'll configure 2 VLANs on a switch. We'll then proceed and configure a router to enable communication between the two VLANs.

Step 1:

In Cisco Packet Tracer, create the network topology as shown below:



Step 2:
Create 2 VLANs on the switch: VLAN 10 and VLAN 20. You can give them custom names:

```
Switch>
Switch>en
Switch#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name SALES
Switch(config-vlan)#vlan 20
Switch(config-vlan)#name IT
Switch(config-vlan)#
```

Step 3:
Assign switch ports to the VLANs. Remember each VLAN is viewed as separate broadcast domain:

And just before you configure, have in mind that switch ports could be either **access** or **trunk**:

- An **access port** is assigned to a single VLAN. These ports are configured for switch ports that connect to devices with a normal network card, for example a PC in a network.
- An access port is assigned to a single VLAN. These ports are configured for switch ports that connect to devices with a normal network card, for example a PC in a network.

So in our case, we'll configure switch interfaces fa 0/1 through fa 0/4 as access ports to connect to our PCs. Here, interfaces fa 0/1 and fa 0/2 are assigned to VLAN 10 while interfaces fa 0/3 and fa 0/4 are assigned to VLAN 20.

Switch Interface fa0/5 will be configured as trunk port, as it will be used to carry traffic between the two VLANs via the router.

```
Switch(config)#interface fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#interface fa0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#interface fa0/3
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
Switch(config-if)#interface fa0/4
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
Switch(config-if)#
```

Worth noting: We could have configured all the above interfaces as access ports using interface range command as shown below:

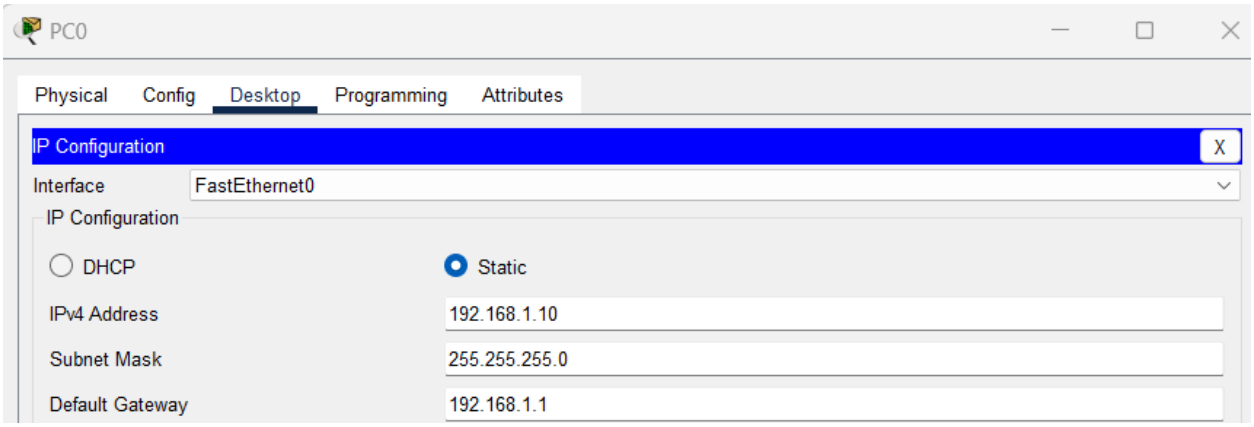
```
Switch(config-if)#int range fa0/1-4
Switch(config-if-range)#switchport mode access
```

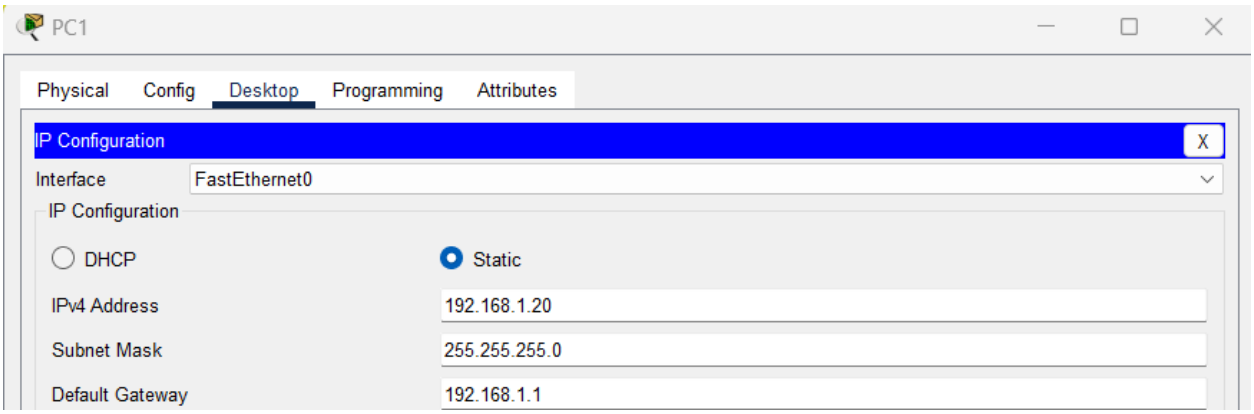
In the above commands, we have specified an interface range and then proceeded to configure all the ports specified as access ports.

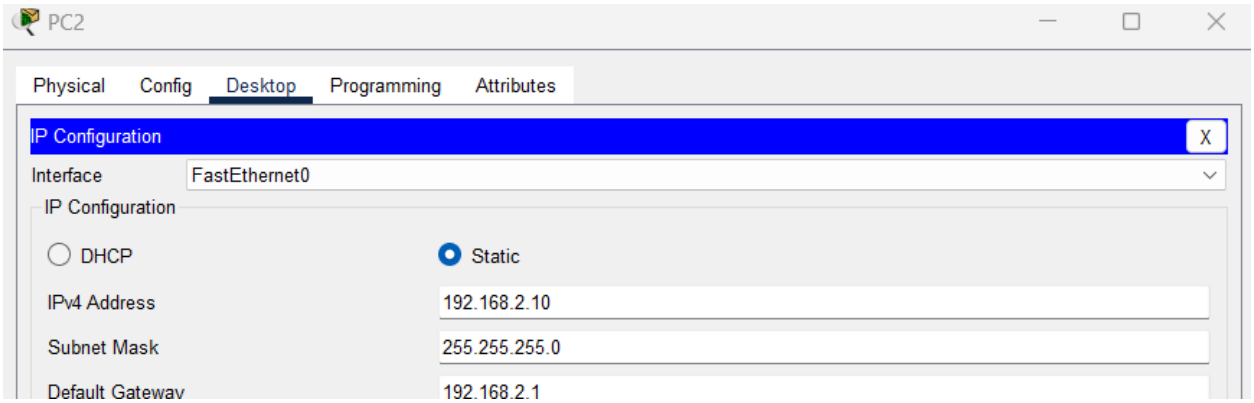
Interface fa0/5 is configured as trunk and will be used to for inter-VLAN communication:

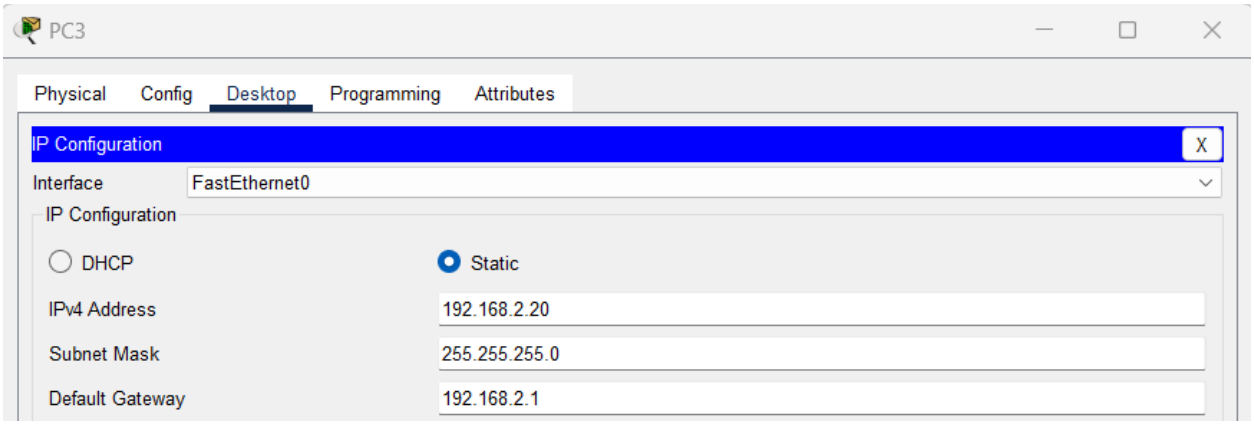
```
Switch(config)#interface fa0/5
Switch(config-if)#switchport mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/5, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/5, changed state to up

Switch(config-if)#
```

The next thing is to assign static IP addresses to the four PCs which are located in the separate VLANs. PC0 and PC1 fall in VLAN 10 while PC2 and PC3 fall in VLAN 20.







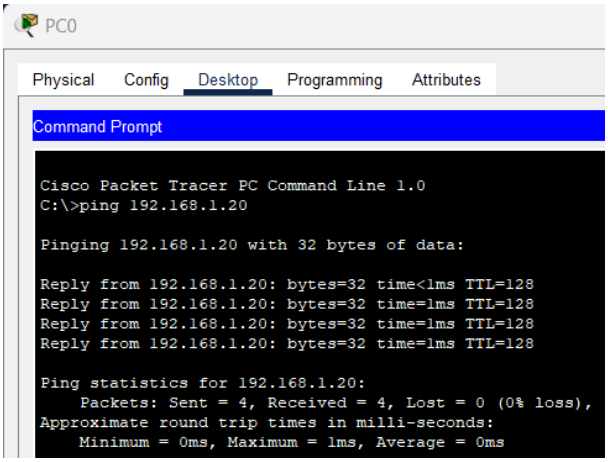


And now it's very clear that we treat a VLAN just like a physical LAN when assigning IP addresses.

At this point let's try to test connectivity within VLANs and between VLANs.

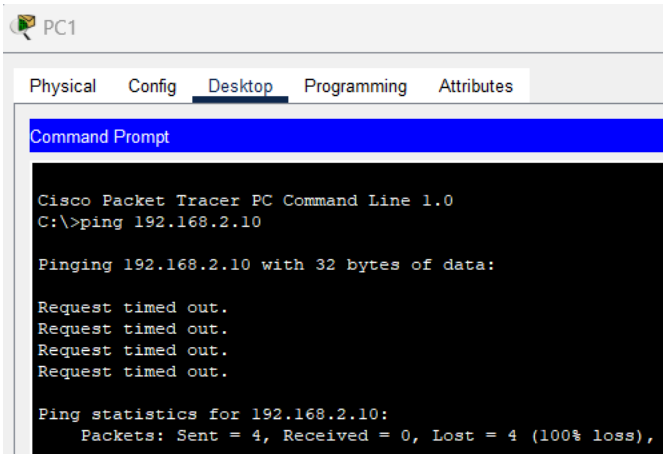
To test communication between hosts in the same VLAN:

Ping PC1 from PC0 both in VLAN 10. Ping test should be successful.



To test connectivity between hosts in different VLANs:

Ping PC2 in VLAN 20 from PC1 in VLAN 10.



Here it will definitely fail. Because **inter-VLAN routing** is not yet enabled. We've used VLANs to place the hosts into two logical networks which can be viewed as separate broadcast domains.

Now, in order to allow the hosts in the two VLANs to communicate, we need to do something extra. And you can guess what. We'll configure the router to permit inter-VLAN communication. Let's do that right away.

We'll configure the router so that it will enable communication between the two vlans via a single physical interface.

We'll divide the single physical interface on the router into logical interfaces (sub interfaces). Each sub-interface will then serve as a default gateway for each of the VLANs. This scenario is called **router on a stick (R.O.A.S)** and will allow the VLANs to communicate through the single physical interface.

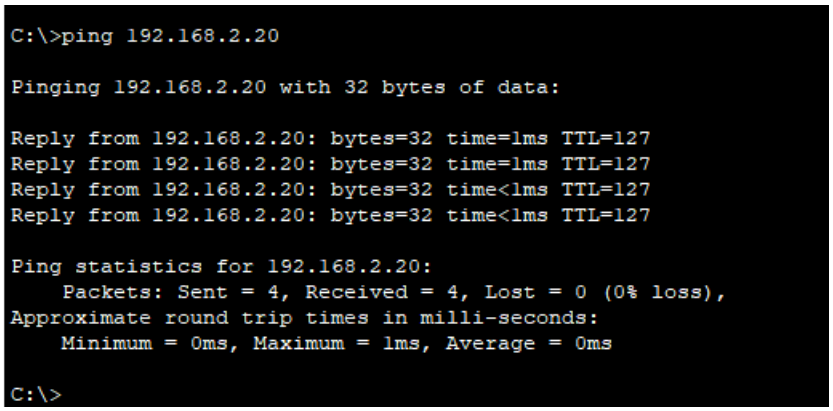
Worth noting: We can't assign an IP address to the router's physical interface that we have subdivided into logical sub-interfaces. We'll instead assign IP addresses to the sub interfaces.

```
Router>en
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int fa0/0
Router(config-if)#no shutdown
Router(config-if)#int fa0/0.10
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.10, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.10, changed state to up

Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip add 192.168.1.1 255.255.255.0
Router(config-subif)#
Router(config-subif)#
Router(config-subif)#int fa0/0.20
Router(config-subif)#
%LINK-5-CHANGED: Interface FastEthernet0/0.20, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0.20, changed state to up

Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip add 192.168.2.1 255.255.255.0
Router(config-subif)#
```

As you can notice from above, the routers physical interface fa0/0 was subdivided into two sub-interfaces (fa0/0.10 and fa0/0.20), which are then configured as trunk interfaces and given IP addresses.

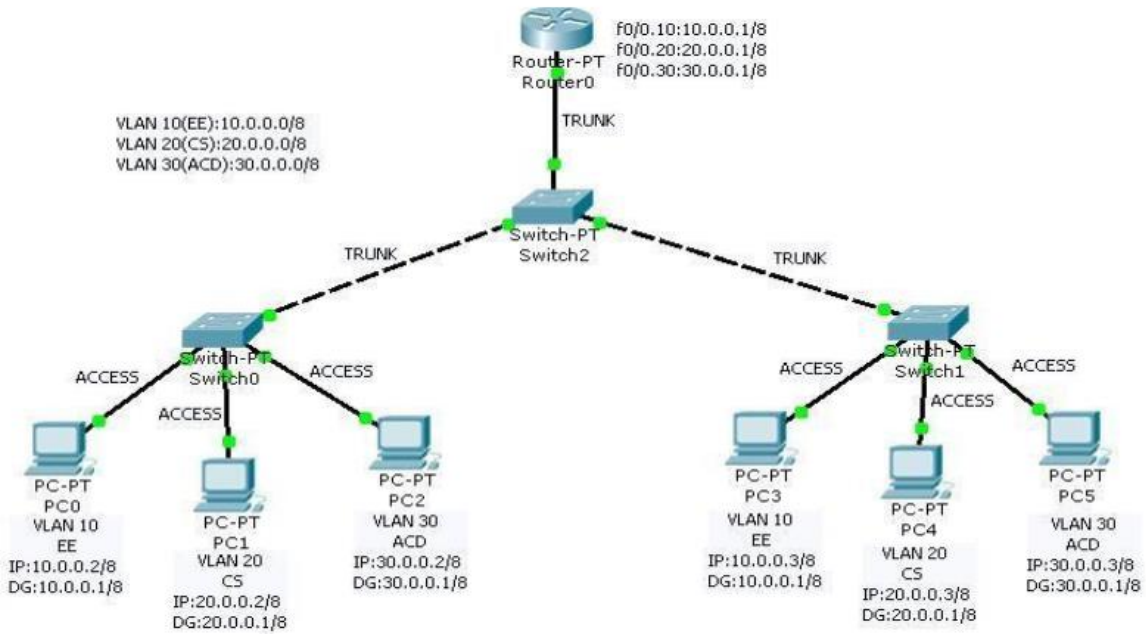
Now we'll test connectivity between computers in different VLANs. Don't forget that it's the router that enables inter-VLAN routing.

Ping PC3 in VLAN 20 from PC1 in VLAN 10. If everything is well configured, then ping should work perfectly.



Lab Exercises

1. Implement the given topology given in the below figure on cisco packet tracer. Perform the following tasks:
   a) Create different VLAN members on the given switches.
   b) Create Trunk and Access link connection.
   c) Create the inter-VLAN routing on the given router.



2. Implement the given topology given in the below figure on cisco packet tracer. Perform the following task
   a) Do perform VLANs and Inter-VLAN Routing.
   b) Dynamic IP addresses should be assigned to all the end devices.
   c) The default gateways should be like XX.XX.1.1, XX.XX.2.1 and so on where XX.XX will be your roll number like 3879 and it will be 38.79.1.1, 38.79.2.1 and so on.