Namal Institute Mianwali

Computer Networks Laboratory Manual #12

Virtual Local Area Networks (VLAN)

Course Title	Computer Networks	**Course Number**	CS – 331 L
Instructor	Shahzad Arif shahzad.arif@namal.edu.pk	**Lab Engineer**	Asad Majeed asad.majeed@namal.edu.pk

Name	**Muhammad Abrar Hussain**
Roll No	NIM-BSCS-2020-62
Assignment No.	12

Virtual LANS

In simple terms, a VLAN is a set of workstations within a LAN that can communicate with each other as though they were on a single, isolated LAN.

VLAN is a logical grouping of layer two devices sharing same broadcast domain. VLAN can span over the multiple physical locations. In this article I will explain VLAN in detail with examples. Switch and bridge both are capable to create and manage VLAN. Switch is the upgraded version of bridge. In this article I will use Switch for demonstration purpose.

VLAN is a logical grouping of networking devices. When we create VLAN, we actually break large broadcast domain in smaller broadcast domains. Consider VLAN as a subnet. Same as two different subnets cannot communicate with each other without router, different VLANs also requires router to communicate.

What does it mean to say that they “communicate with each other as though they were on a single, isolated LAN”? Among other things, it means that:

- Broadcast packets sent by one of the workstations will reach all the others in the VLAN
- Broadcasts sent by one of the workstations in the VLAN will not reach any workstations that are not in the VLAN
- Broadcasts sent by workstations that are not in the VLAN will never reach workstations that are in the VLAN
- The workstations can all communicate with each other without needing to go through a gateway. For example, IP connections would be established by ARPing for the destination IP and sending packets directly to the destination workstation—there would be no need to send packets to the IP gateway to be forwarded on.
- The workstations can communicate with each other using non-routable protocols.

The purpose of VLANs

The basic reason for splitting a network into VLANs is to reduce congestion on a large LAN. To understand this problem, we need to look briefly at how LANs have developed over the years.

Initially LANs were very flat—all the workstations were connected to a single piece of coaxial cable, or to sets of chained hubs. In a flat LAN, every packet that any device puts onto the wire gets sent to every other device on the LAN.

As the number of workstations on the typical LAN grew, they started to become hopelessly congested; there were just too many collisions, because most of the time when a workstation tried to send a packet, it would find that the wire was already occupied by a packet sent by some other device.

This section describes the three solutions for this congestion that were developed:

- Using routers to segment LANs
- Using switches to segment LANs
- Using VLANs to segment LANs

Using VLANs to segment LANs : As LANs became larger, data rates became faster, and users desired greater flexibility, the routers in a network started to become a bottleneck. This is because:

- Routers typically forward data in software, and so are not as fast as switches
- Splitting up a LAN using routers meant that a LAN typically corresponded to a particular physical location. This became limiting when many users had laptops, and wanted to be able to move between buildings, but still have the same network environment wherever they plugged in.

So, switch vendors started implementing methods for defining "virtual LANs"—sets of switch ports, usually distributed across multiple switches that somehow interacted as though they were in a single isolated LAN. This way, workstations could be separated off into separate LANs without being physically divided up by routers.

At about the same time, hubs became less popular and have been largely replaced by L2 switches. This has made the whole concept of a collision domain somewhat historical. In modern networks, a "collision domain" mostly consists of a single device attached to an L2 switch port, or possibly a PC with something like an IP phone attached to it.

Advantages of using VLANs

1. **Solve broadcast problem**

When we connect devices into the switch ports, switch creates separate collision domain for each port and single broadcast domain for all ports. Switch forwards a broadcast frame from all possible ports. In a large network having hundreds of computers, it could create performance issue. Of course we could use routers to solve broadcast problem, but that would be costly solution since each broadcast domain requires its own port on router. Switch has a unique solution to broadcast issue known as VLAN. In practical environment we use VLAN to solve broadcast issue instead of router.

Each VLAN has a separate broadcast domain. Logically VLANs are also subnets. Each VLAN requires a unique network number known as VLAN ID. Devices with same VLAN ID are the members of same broadcast domain and receive all broadcasts. These broadcasts are filtered from all ports on a switch that aren't members of the same VLAN.

2. **Reduce the size of broadcast domains**

VLAN increase the numbers of broadcast domain while reducing their size. For example we have a network of 100 devices. Without any VLAN implementation we have single broadcast domain that contain 100 devices. We create 2 VLANs and assign 50 devices in each VLAN. Now we have two broadcast domains with fifty devices in each. Thus more VLAN means more broadcast domain with less devices.

3. **Allow us to add additional layer of security**

VLANs enhance the network security. In a typical layer 2 network, all users can see all devices by default. Any user can see network broadcast and responds to it. Users can access any network resources located on that specific network. Users could join a workgroup by just attaching their system in existing switch. This could create real trouble on security platform. Properly configured VLANs gives us total control over each port and users. With VLANs, you can control the users from gaining unwanted access over the resources. We can put the group of users that need high level security into their own VLAN so that users outside from VLAN can't communicate with them.

4. **Make device management easier**

Device management is easier with VLANs. Since VLANs are a logical approach, a device can be located anywhere in the switched network and still belong to the same broadcast domain. We can move a user from one switch to another switch in same network while keeping his original VLAN. For example our company has a five story building and a single layer two network. In this scenario, VLAN allows us to move the users from one floor to another floor while keeping his original VLAN ID. The only limitation we have is that device when moved, must still be connected to the same layer 2 network.

5. **Allow us to implement the logical grouping of devices**

Allow us to implement the logical grouping of devices by function instead of location

VLANs allow us to group the users by their function instead of their geographic locations. Switches maintain the integrity of your VLANs. Users will see only what they are supposed to see regardless what their physical locations are.

6. **Performance**.

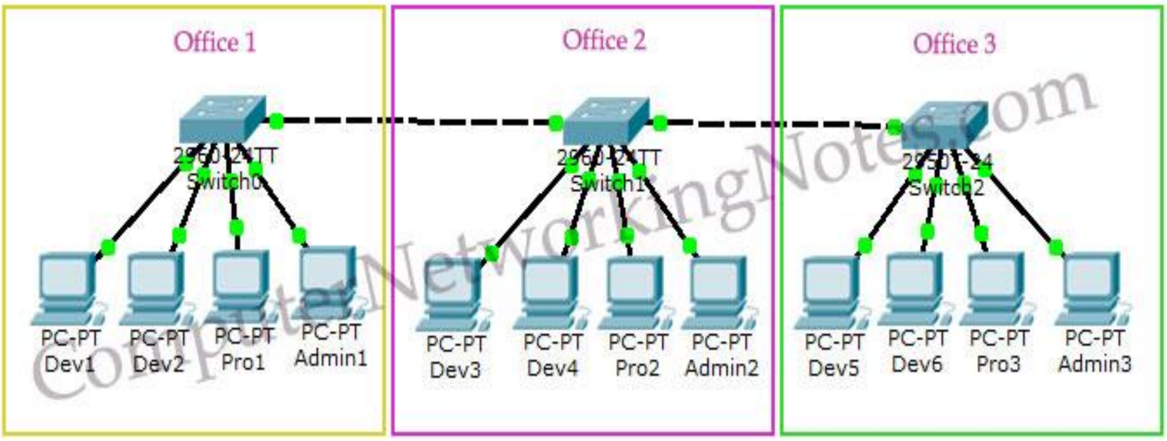
As mentioned above, routers that forward data in software become a bottleneck as LAN data rates increase. Doing away with the routers removes this bottleneck. Formation of virtual workgroups. Because workstations can be moved from one VLAN to another just by changing the configuration on switches, it is relatively easy to put all the people working together on a particular project all into a single VLAN. They can then more easily share files and resources with each other. To be honest, though, virtual workgroups sound like a good idea in theory, but often do not work well in practice. It turns out that users are usually more interested in accessing company-wide resources (file servers, printers, etc.) than files on each others' PCs.

7. **Greater flexibility**.
   If users move their desks, or just move around the place with their laptops, then, if the VLANs are set up the right way, they can plug their PC in at the new location, and still be within the same VLAN. This is much harder when a network is physically divided up by routers.

8. **Ease of partitioning off resources**.
   If there are servers or other equipment to which the network administrator wishes to limit access, then they can be put off into their own VLAN. Then users in other VLANs can be given access selectively

VLAN Examples

To understand VLAN more clearly let's take an example.



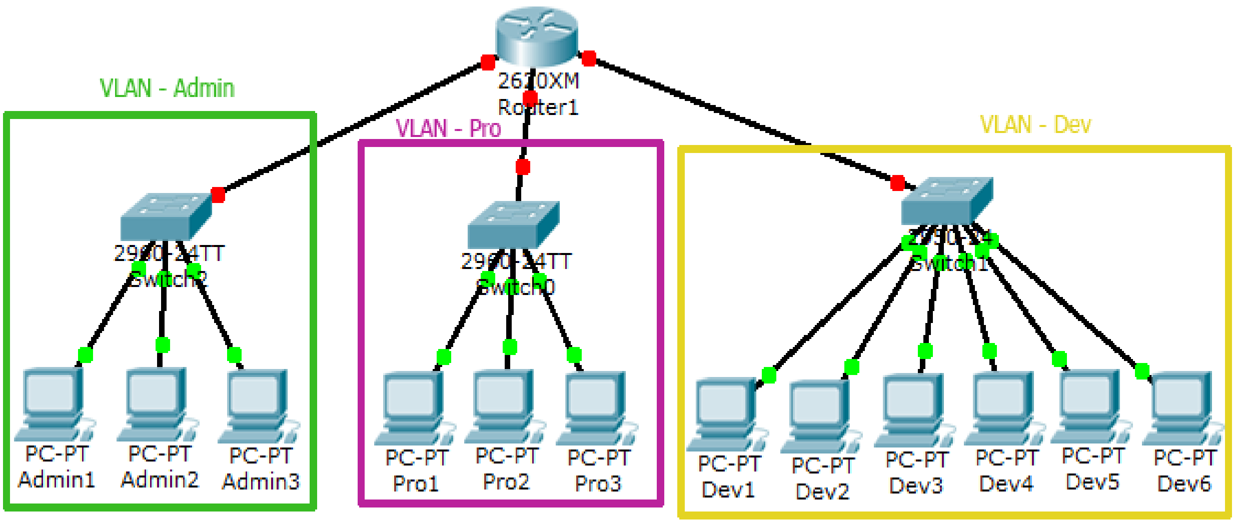
❖ Our company has three offices.

- All offices are connected with back links.
- Company has three departments Development, Production and Administration.
- Development department has six computers.
- Production department has three computers.
- Administration department also has three computers.
- Each office has two PCs from development department and one from both production and administration department.
- Administration and production department have sensitive information and need to be separate from development department.

With default configuration, all computers share same broadcast domain. Development department can access the administration or production department resources.
With VLAN we could create logical boundaries over the physical network. Assume that we created three VLANs for our network and assigned them to the related computers.

- ✓ VLAN **Admin** for Administration department
- ✓ VLAN **Dev** for Development department
- ✓ VLAN **Pro** for Production department

Physically we changed nothing but logically we grouped devices according to their function. These groups [VLANs] need router to communicate with each other. Logically our network look likes following diagram.



With the help of VLAN, we have separated our single network in three small networks. These networks do not share broadcast with each other improving network performance. VLAN also enhances the security. Now Development department cannot access the Administration and Production department directly. Different VLAN can communicate only via Router where we can configure wild range of security options.

VLAN Membership

VLAN membership can be assigned to a device by one of two methods

1. Static
2. Dynamic

These methods decide how a switch will associate its ports with VLANs.

Static

Assigning VLANs statically is the most common and secure method. It is pretty easy to set up and supervise. In this method we manually assign VLAN to switch port. VLANs configured in this way are usually known as port-based VLANs.

Static method is the most secure method also. As any switch port that we have assigned a VLAN will keep this association always unless we manually change it. It works really well in a networking environment where any user movement within the network needs to be controlled.

Dynamic

In dynamic method, VLANs are assigned to port automatically depending on the connected device. In this method we have configure one switch from network as a server. Server contains device specific information like MAC address, IP address etc. This information is mapped with VLAN. Switch acting as server is known as VMPS (VLAN Membership Policy Server). Only high end switch can configured as VMPS. Low end switch works as client and retrieve VLAN information from VMPS.

Dynamic VLANs supports plug and play movability. For example if we move a PC from one port to another port, new switch port will automatically be configured to the VLAN which the user belongs. In static method we have to do this process manually.
VLAN Connections
During the configuration of VLAN on port, we need to know what type of connection it has.
Switch supports two types of VLAN connection

1. Access link
2. Trunk link

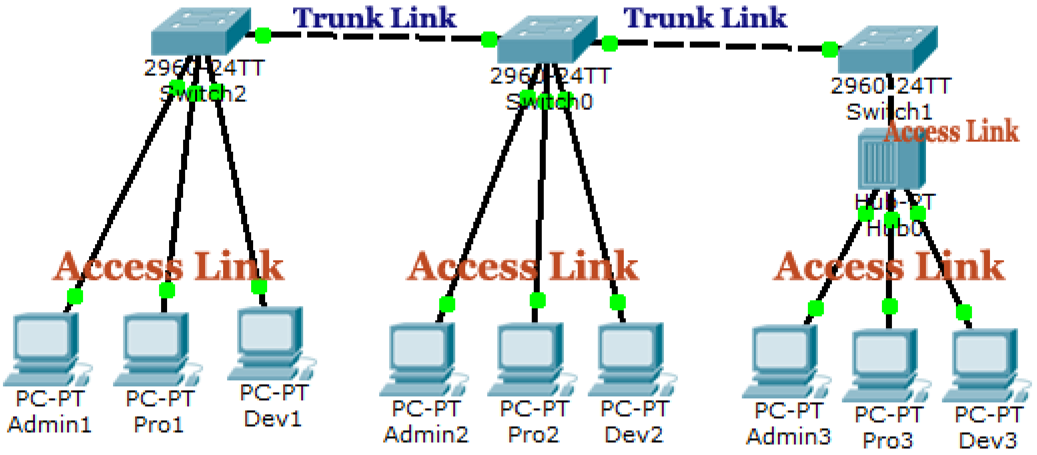
Access link

Access link connection is the connection where switch port is connected with a device that has a standardized Ethernet NIC. Standard NIC only understand IEEE 802.3 or Ethernet II frames. Access link connection can only be assigned with single VLAN. That means all devices connected to this port will be in same broadcast domain.

For example twenty users are connected to a hub, and we connect that hub with an access link port on switch, then all of these users belong to same VLAN. If we want to keep ten users in another VLAN, then we have to purchase another hub. We need to plug in those ten users in that hub and then connect it with another access link port on switch.

Trunk link

Trunk link connection is the connection where switch port is connected with a device that is capable to understand multiple VLANs. Usually trunk link connection is used to connect two switches or switch to router. Remember earlier in this article I said that VLAN can span anywhere in network, that is happen due to trunk link connection. Trunking allows us to send or receive VLAN information across the network. To support trunking, original Ethernet frame is modified to carry VLAN information.



Trunk Tagging

In trunking a separate logical connection is created for each VLAN instead of a single physical connection. In tagging switch adds the source port's VLAN identifier to the frame so that other end device can understands what VLAN originated this frame. Based on this information destination switch can make intelligent forwarding decisions on not just the destination MAC address, but also the source VLAN identifier.

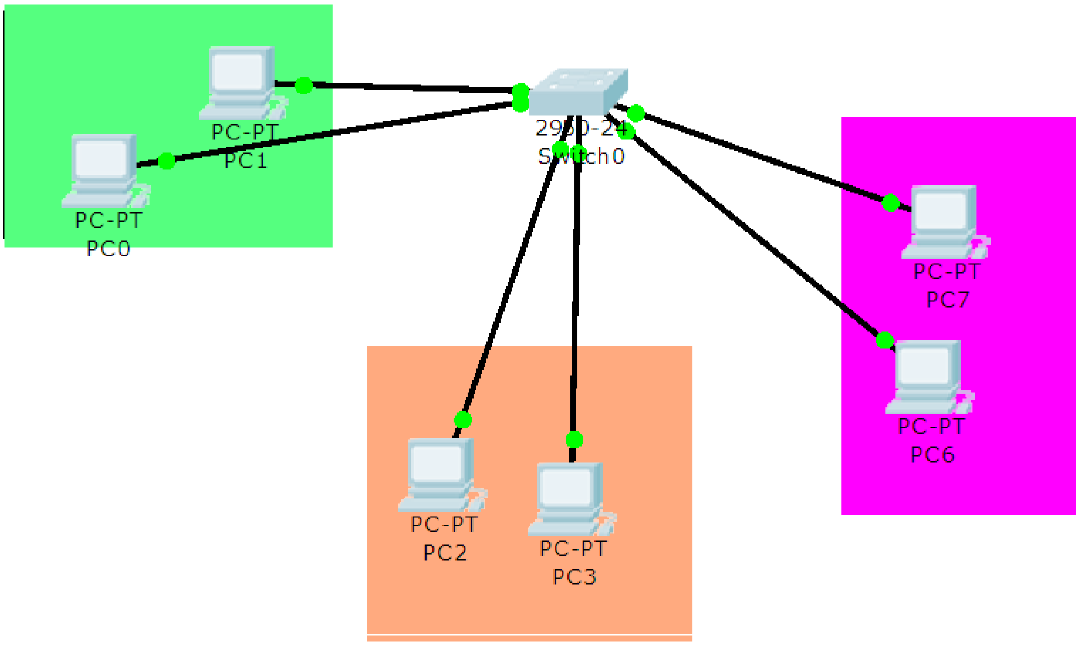
Since original Ethernet frame is modified ensure that when we set up a trunk connection on a switch's port, the device at the other end also supports the same trunking protocol and has it configured. If the device at the other end doesn't understand these modified frames it will drop them. The modification of these frames, commonly called tagging.

Tagging is done in hardware by application-specific integrated circuits (ASICs).to add information, standard NICs will not understand this information and will typically drop the frame. Therefore, we need to

Switch supports two types of Ethernet trunking methods:

1. ISL [Inter Switch Link, Cisco's proprietary protocol for Ethernet]
2. Dot1q [IEEE's 802.1Q, protocol for Ethernet]

How to create VLAN in Packet tracer



Step 01

Design above network in Packet Tracer

Go to configuration mode of switch

a. Change the hostname of switch

```
Switch(config)# hostname NAMAL
```

b. Create VLAN(s)

```
Switch(config)# vlan vlan- ID

Switch(config)# name VLAN-NAME
```

```
Switch(config)# vlan 10

Switch(config)# name admin
Switch(config)# vlan 20

Switch(config)# name ComputerScience


Switch(config)# vlan 30

Switch(config)# name Engineering
```

c. ***Assigning interface to the VLANs***

For VLAN 10

```
Switch(config)# interface range fastethernet 0/1 – 8

Switch(config)# switchport mode access

Switch(config)# switchport access vlan 10
```

For VLAN 20

```
Switch(config)# interface range fastethernet 0/9 – 15

Switch(config)# switchport mode access

Switch(config)# switchport access vlan 20
```

For VLAN 30

```
Switch(config)# interface range fastethernet 0/16 – 24

Switch(config)# switchport mode access

Switch(config)# switchport access vlan 30


Switch# copy running-config startup-config
```

d. ***Check the connectivity of the users in the same vlan and different vlans***

e. ***Create a trunk port if you want to connect to a Router or a switch***

```
Switch(config)#interface fastEthernet 0/10
Switch(config-if)#switchport mode trunk
```

f. Router configuration for inter vlan routing

Inter vlan communication be done using router-on-a-stick method. The router-on-a-stick method requires you to create a subinterface for each VLAN to be routed.

A sub interface is created using the interface interface_id.subinterface_id global configuration mode command. The sub interface syntax is the physical interface followed by a period and a subinterface number. Although not required, it is customary to match the subinterface number with the VLAN number.

Each subinterface is then configured with the following two commands:

- encapsulation dot1q vlan_id [native]: This command configures the subinterface to respond to 802.1Q encapsulated traffic from the specified vlan-id. The native keyword option is only appended to set the native VLAN to something other than VLAN 1.

- ip address ip-address subnet-mask: This command configures the IPv4 address of the subinterface. This address typically serves as the default gateway for the identified VLAN.

Repeat the process for each VLAN to be routed. Each router subinterface must be assigned an IP address on a unique subnet for routing to occur.

When all subinterfaces have been created, enable the physical interface using the no shutdown interface configuration command. If the physical interface is disabled, all subinterfaces are disabled.

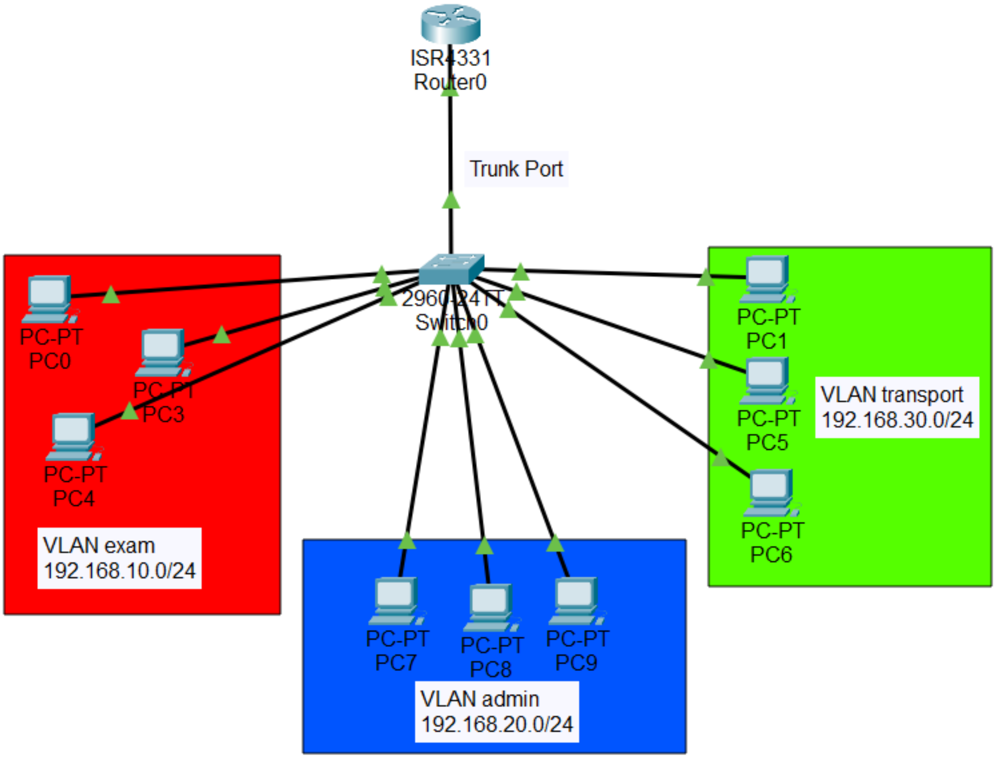
Sample CLI commands are shown below

```
R1(config)# interface G0/0/1.10

R1(config-subif)# description Default Gateway for VLAN 10

R1(config-subif)# encapsulation dot1Q 10

R1(config-subif)# ip add 192.168.10.1 255.255.255.0

R1(config-subif)# exit

R1(config)#
```

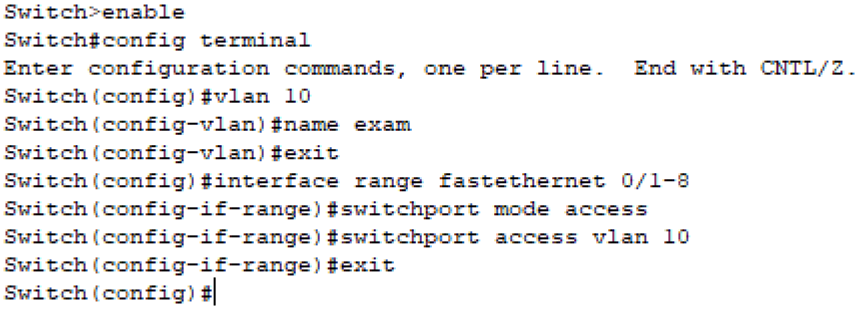
```
R1(config)# interface G0/0/1.20
R1(config-subif)# description Default Gateway for VLAN 20
R1(config-subif)# encapsulation dot1Q 20
R1(config-subif)# ip add 192.168.20.1 255.255.255.0
R1(config-subif)# exit
R1(config)# interface G0/0/1
R1(config-if)# description Trunk link to S1
R1(config-if)# no shut
R1(config-if)# end
```

Exercise

Perform the following Task and provide the solution.

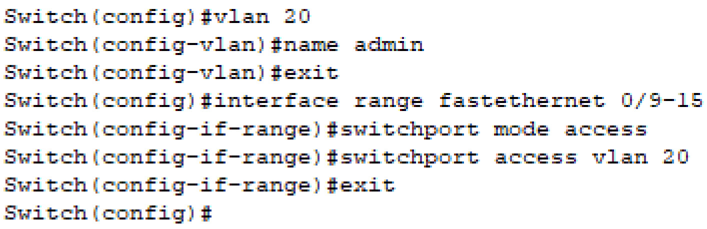


- Snapshot of switch CLI for configuration of vlan exam and its access ports



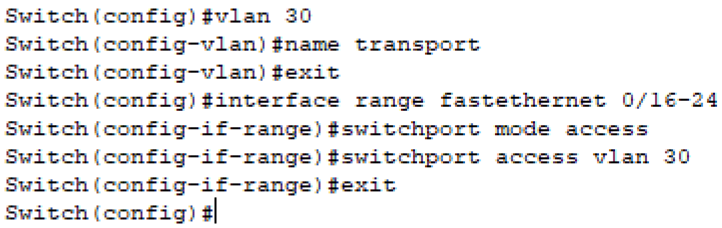
```
Switch>enable
Switch#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name exam
Switch(config-vlan)#exit
Switch(config)#interface range fastethernet 0/1-8
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
Switch(config-if-range)#exit
Switch(config)#
```

- Snapshot of switch CLI for configuration of vlan admin and its access ports



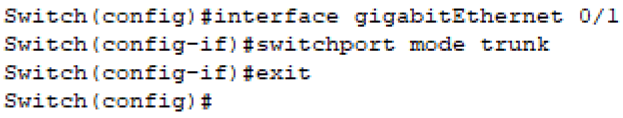
```
Switch(config)#vlan 20
Switch(config-vlan)#name admin
Switch(config-vlan)#exit
Switch(config)#interface range fastethernet 0/9-15
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 20
Switch(config-if-range)#exit
Switch(config)#
```

- Snapshot of switch CLI for configuration of vlan transport and its access ports



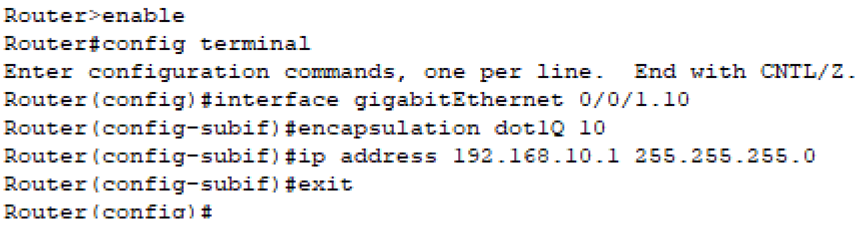
```
Switch(config)#vlan 30
Switch(config-vlan)#name transport
Switch(config-vlan)#exit
Switch(config)#interface range fastethernet 0/16-24
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 30
Switch(config-if-range)#exit
Switch(config)#
```

- Snapshot of switch CLI for configuration of trunk port



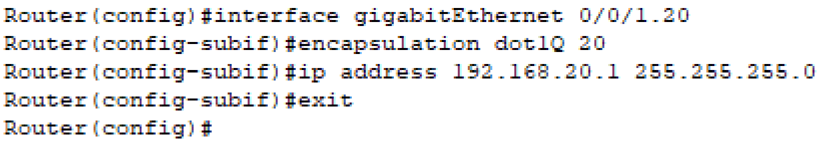
```
Switch(config)#interface gigabitEthernet 0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
Switch(config)#
```

- Snapshot of the router CLI for configuration vlan exam



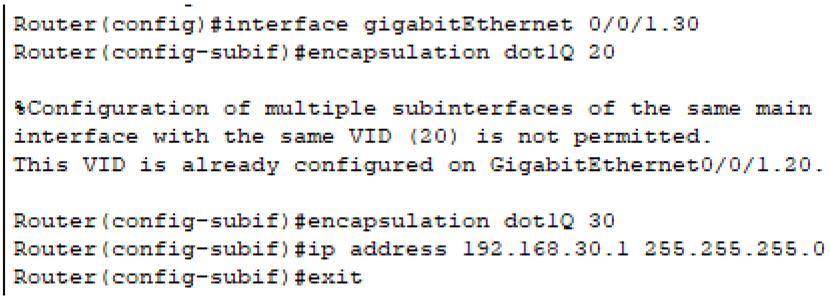
```
Router>enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface gigabitEthernet 0/0/1.10
Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip address 192.168.10.1 255.255.255.0
Router(config-subif)#exit
Router(config)#
```

- Snapshot of the router CLI for configuration of vlan admin



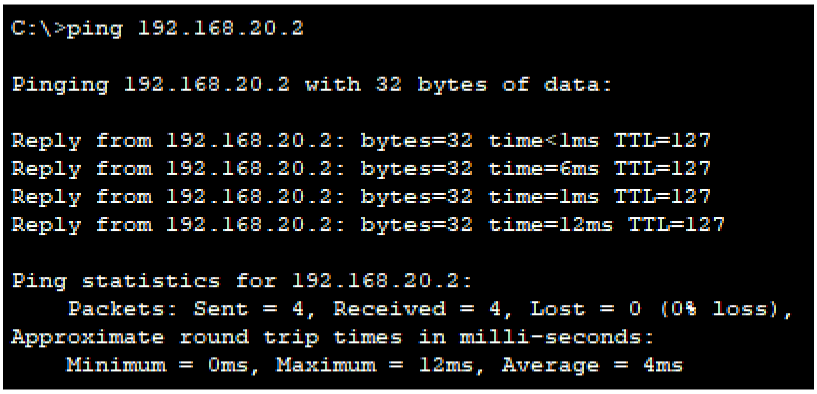
```
Router(config)#interface gigabitEthernet 0/0/1.20
Router(config-subif)#encapsulation dot1Q 20
Router(config-subif)#ip address 192.168.20.1 255.255.255.0
Router(config-subif)#exit
Router(config)#
```

- Snapshot of the router CLI for configuration of vlan transport

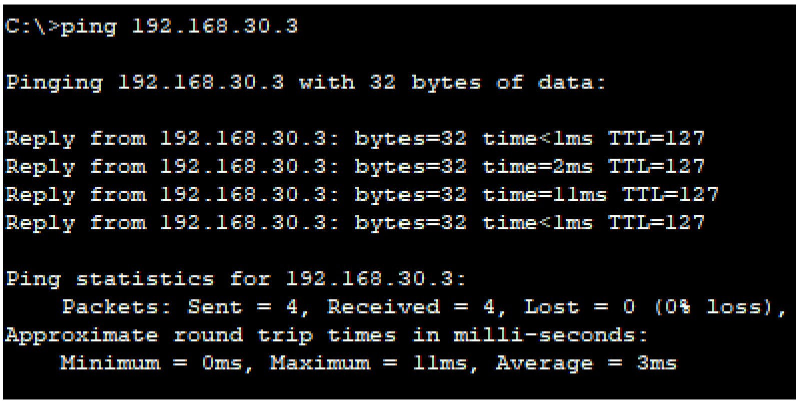


```
Router(config)#interface gigabitEthernet 0/0/1.30
Router(config-subif)#encapsulation dot1Q 20

%Configuration of multiple subinterfaces of the same main
interface with the same VID (20) is not permitted.
This VID is already configured on GigabitEthernet0/0/1.20.

Router(config-subif)#encapsulation dot1Q 30
Router(config-subif)#ip address 192.168.30.1 255.255.255.0
Router(config-subif)#exit
```

- Snapshot of the ping from Exam vlan to Admin and Transport

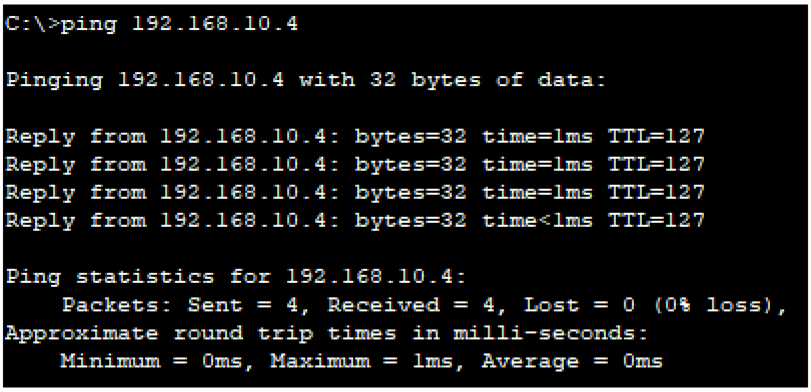


```
C:\>ping 192.168.20.2

Pinging 192.168.20.2 with 32 bytes of data:

Reply from 192.168.20.2: bytes=32 time<1ms TTL=127
Reply from 192.168.20.2: bytes=32 time=6ms TTL=127
Reply from 192.168.20.2: bytes=32 time=1ms TTL=127
Reply from 192.168.20.2: bytes=32 time=12ms TTL=127

Ping statistics for 192.168.20.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 12ms, Average = 4ms
```

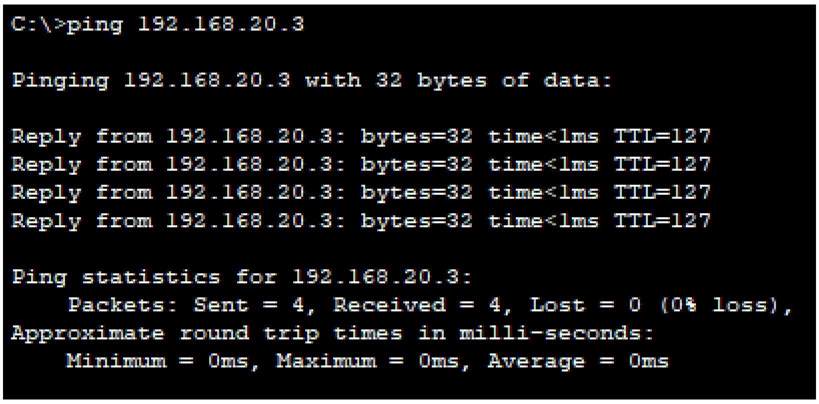


```
C:\>ping 192.168.30.3

Pinging 192.168.30.3 with 32 bytes of data:

Reply from 192.168.30.3: bytes=32 time<1ms TTL=127
Reply from 192.168.30.3: bytes=32 time=2ms TTL=127
Reply from 192.168.30.3: bytes=32 time=11ms TTL=127
Reply from 192.168.30.3: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.30.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 3ms
```

- Snapshot of the ping from Transport vlan to Admin and Exam

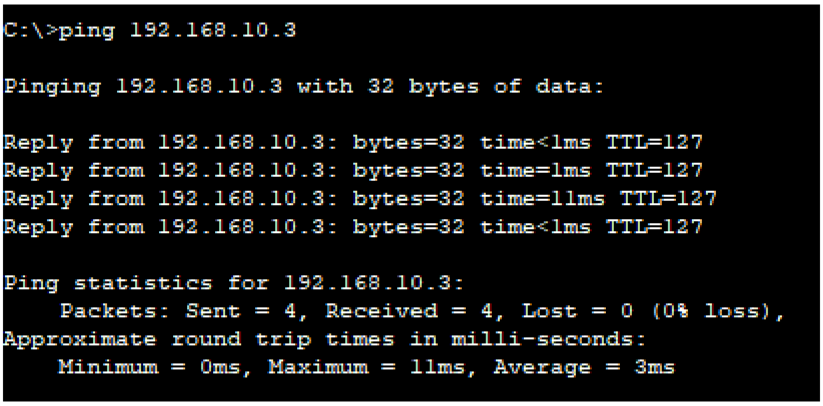


```
C:\>ping 192.168.10.4

Pinging 192.168.10.4 with 32 bytes of data:

Reply from 192.168.10.4: bytes=32 time=1ms TTL=127
Reply from 192.168.10.4: bytes=32 time=1ms TTL=127
Reply from 192.168.10.4: bytes=32 time=1ms TTL=127
Reply from 192.168.10.4: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.10.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

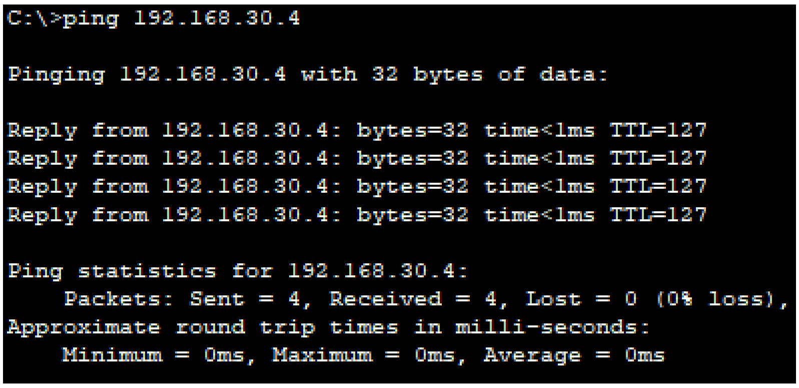


```
C:\>ping 192.168.20.3

Pinging 192.168.20.3 with 32 bytes of data:

Reply from 192.168.20.3: bytes=32 time<1ms TTL=127
Reply from 192.168.20.3: bytes=32 time<1ms TTL=127
Reply from 192.168.20.3: bytes=32 time<1ms TTL=127
Reply from 192.168.20.3: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.20.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- Snapshot of the ping from Admin vlan to Exam and Transport



```
C:\>ping 192.168.10.3

Pinging 192.168.10.3 with 32 bytes of data:

Reply from 192.168.10.3: bytes=32 time<1ms TTL=127
Reply from 192.168.10.3: bytes=32 time=1ms TTL=127
Reply from 192.168.10.3: bytes=32 time=11ms TTL=127
Reply from 192.168.10.3: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.10.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 3ms
```



```
C:\>ping 192.168.30.4

Pinging 192.168.30.4 with 32 bytes of data:

Reply from 192.168.30.4: bytes=32 time<1ms TTL=127
Reply from 192.168.30.4: bytes=32 time<1ms TTL=127
Reply from 192.168.30.4: bytes=32 time<1ms TTL=127
Reply from 192.168.30.4: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.30.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

- **Upload the lab manual and .pkt file to QoBE.**

Rubrics Sheet

Lab Exercise	Marks
VLAN configuration of given departments	5
Routers on a stick configuration	5