LAB BRIEF

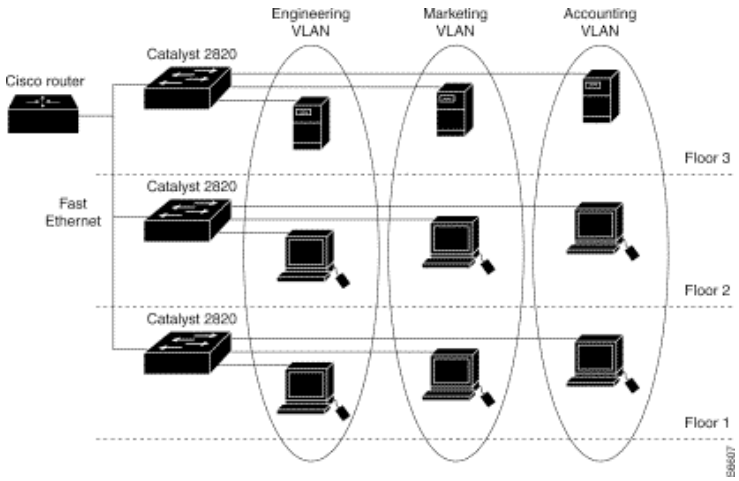
The intention of the lab is to present the concept of the Virtual LAN (VLAN). Students will then implement and configure a static type VLAN in the Packet Tracer program.

Submission

No submission needed.

VLANS

A VLAN is any **broadcast domain** that is partitioned and isolated in a computer network at the *data link layer (OSI layer 2). VLANs are LANs that are constructed over switches where multiple LANs can be hosted over a single switch, or a single LAN can be hosted over multiple switches. A LAN is essentially a single **broadcast domain.** Previously we have separated LANs (broadcast domains) from each other using routers. This can also be achieved with switches. See the diagram below.



The Network contains three separate LANs. The devices on these LANs, however, are spread over multiple floors in an office. The broadcast messages from any device on a LAN should not be ‘heard’ by devices on the other LANs. This creates security and performance increases for the separate LANs.

*Note - A router (layer 3 device) is still needed to pass packets **between** VLANs.

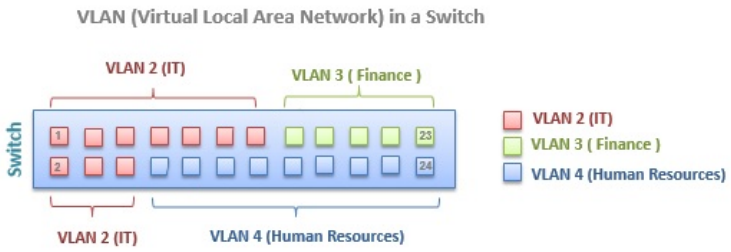
Why not use a separate switch for each LAN?

This usually comes down to cost, cabling logistics and performance. Switches are expensive, if we only have 5 devices on a particular LAN it is not cost effective to have a single switch for that LAN (switches can support up to 24+ devices). It makes sense in this case for this small LAN to share the switch with other devices from another LAN. For cabling, if the devices on a single LAN are geographically distant from each other, it is difficult (or impossible) to cable the devices to the same switch. The shorter the cables, the higher performance can be achieved in the network. Therefore, all devices should be connected to their closest switch.

Configuring VLANs

Configuring the VLANs is done on the switches. We will configure a number of separate LANs over a number of different switches. We can do this in two ways:

Static port assignment:

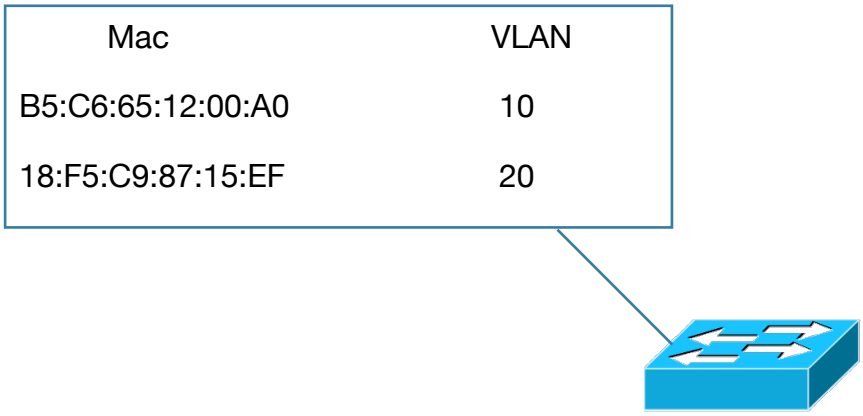


Each port is assigned a VLAN number. Ports with the same VLAN number are considered to be on the same VLAN. Any device connected to this port will automatically become a part of the VLAN that the port is associated with.

This is the method we will use today!

Note - The device at the end of this port still has to have the correct IP address for the LAN to functions correctly!!!

Dynamic port assignment:



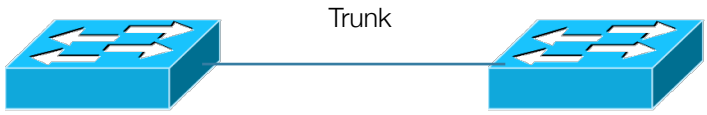
A switch holds a table of MAC (Layer 2) addresses - the table is often held on a remote (management) device. These addresses are associated with a VLAN number. When the device is connected to the switch the MAC address is checked and it is placed on the corresponding LAN.

Note: This solution has proved to be quite cumbersome for large companies. The list of MAC addresses must be maintained which can be a large task.

VLAN Trunks

Trunks are connections between switches or routers that hold traffic for multiple VLANs. The connections between switches or to a router must be configured to carry VLAN traffic.

Encapsulation is needed here!!!!



SWITCH CONFIGURATION

Switches follow the same privilege possibilities as routers:

Mode	Purpose	Prompt	Command to enter	Command to exit
User EXEC	Allow you to connect with remote devices, perform basic tests, temporary change terminal setting, list system information	Switch >	Default mode after booting. Login with password, if configured.	Use *exit* command
Privileged EXEC	Allow you to set operating parameters. It also includes high level testing and list commands like show, copy, debug.	Switch #	Use *enable* command from user exec mode	Use *exit* command
Global Configuration	Contain commands those affect the entire system	Switch (config)#	Use *configure terminal* command from privileged exec mode	Use *exit* command
Interface Configuration	Contain commands those modify the operation of an interface	Switch (config-if)#	Use *interface type number* command from global configuration mode	Use *exit* command to return in global configuration mode
VLAN configuration	Create new VLAN or configure, modify, delete existing VLAN.	Switch(config-vlan)#	Use *vlan vlan-id* from Global configuration mode	Use *exit* to return previous mode. Use *end* command to return in privileged exec mode.
Line configuration	Line configuration commands modify the operation of a terminal line. Line configuration commands always follow a line command, which defines a line number.	Switch(config-line)#	Use *line vty line_number [ending_line_number]* from global configuration mode	Use *exit* command to return in global configuration mode.

Also note the show commands (among others):

```
Switch 2#show vlan brief
Switch 2#show interfaces fastEthernet 0/1 switchport
Switch 2#show interfaces fastEthernet 0/2 switchport
Switch 2#show interfaces fastEthernet 0/3 switchport
Switch 2#show interfaces fastEthernet 0/4 switchport
Switch 2#show interfaces trunk
```

Configure VLANs

First the VLANs must be stated and described on the switch:

```
switch (config)# vlan 10
switch(config-vlan)# name IT_Dept
```

Configure Ports

Next you have to configure the ports so that they are associated with a VLAN.

```
switch (config)# interface fastethernet0/0
switch(config-if)# switchport mode access
switch(config-if)# switchport access vlan 10
```

Configure Trunks

switch (config)# interface fastethernet0/0

- Once in the interface we use the command switchport to make changes on the port.

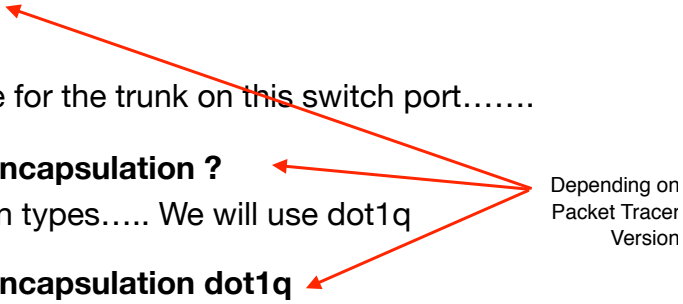
switch (config-if)# switchport trunk ?

- What options does this give?
- We want to set the encapsulation type for the trunk on this switch port…….

switch (config-if)# switchport trunk encapsulation ?

- There are a few different encapsulation types….. We will use dot1q

switch (config-if)# switchport trunk encapsulation dot1q



- Now we can tell the port that it will be a trunk!

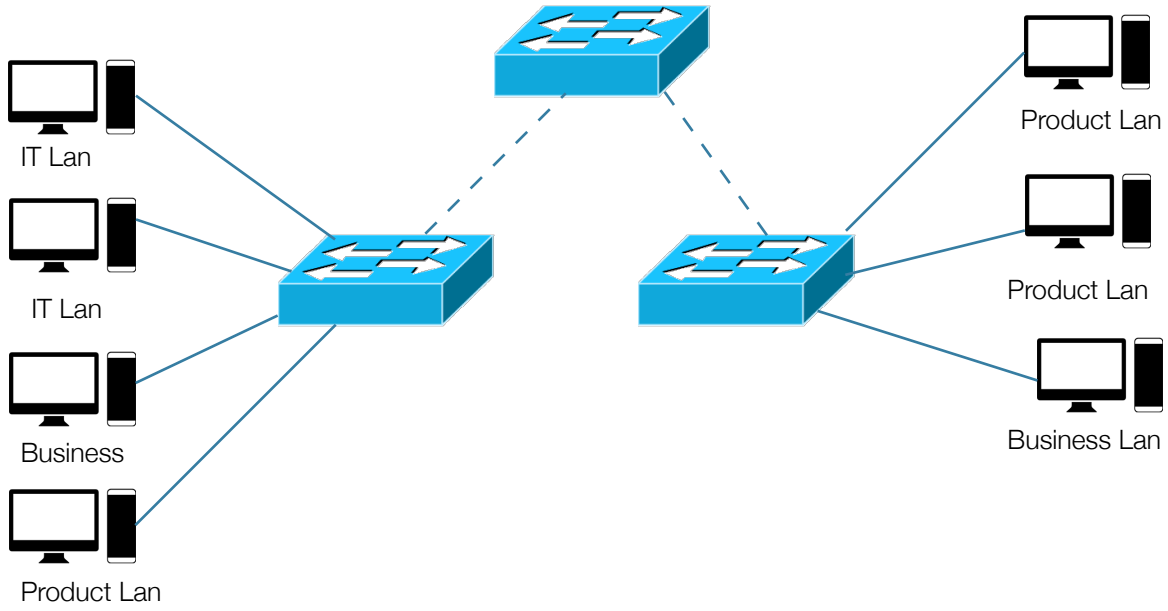
switch(config-if)# switchport mode trunk

- We tell the port what VLANs it will carry.

switch(config-if)# switchport trunk allowed vlan 10,20

NETWORK CONFIGURATION

Design the network to correspond to the diagram below.



Where the IP ranges for the LANs are:

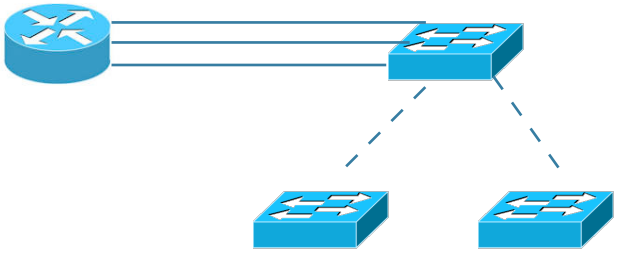
IT Lan - 192.168.10.xxx

Business Lan - 192.168.20.xxx

Product Lan - 192.168.30.xxx

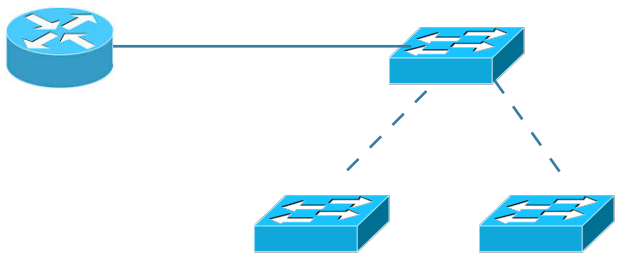
Devices on the same LANs should be able to communicate with each other. Devices on different LANs should not be able to communicate with each other!

Add a router to allow the devices on different VLANs to be able to talk to each other!



“ROUTER ON A STICK”

The previous network leads to a bit of a waste. We need three cables and three interfaces to handle the three VLANs. We can do this more efficiently!



This would be a far more efficient use of resources. However, the interface is now in control of:
3 different networks!
3 different IP ranges!
So it needs to have 3 different IP addresses (Each address within one of the 3 networks).

We can achieve this with sub-interfaces - Give one interface multiple addresses so it acts like multiple interfaces.

router (config)# interface fastethernet0/1
- This is the interface we will split into multiple sub-interfaces

- If we know it will be used for sub interfaces We know it will carry multiple ip addresses. Then we don't want a top level ip address on the interface.

router (config-if)# no ip address

- We can then configure the sub-interfaces on this interface.

router (config-if)# interface fastethernet0/1.1

router (config-subif)#
- Now we are on the sub-interface and we can give it some configuration

We know that each subinterface needs an IP address. One for each of the networks it will handle traffic for.

We also know that the link will carry traffic from multiple VLANS on a trunk……..
How does each sub-interface know what traffic it should handle?
It also uses the special **encapsulation** technique to distinguish between different VLAN traffic.

router (config-subif)# encapsulation dot1q 10

- this tells the router sub-interface that it will use **dot1q** encapsulation type.
- It also tells the sub-interface that it is responsible for taking VLAN **10** traffic

Once you have the correct IP address and the correct encapsulation type and the correct VLAN identifier on each of the sub-interfaces, the network should work correctly.

NOTES for “Router on a Stick”:

The switch interface that is connected to the router should be in trunk mode but should also enable spanning tree

switch (config-if)# spanning-tree portrait trunk

For security and efficiency reasons the sub-interfaces on the router should perform virtual reassembly of packets.

router (config-subif)# ip nat inside

router (config-subif)# ip virtual-reassembly