Redes de Computadores (RCOMP) – 2017/2018

Laboratory Class Script - PL04

• LAN and virtual LAN (VLAN). • IPv4 addressing. • ARP tables. • IPv4 static routeing. • Simplifying routeing tables.	• Cisco Packet Tracer

1. LAN and virtual LAN (VLAN)

A LAN (Local Area Network) is a set of end nodes connected to the same shared layer two transmission technology. Within a LAN, nodes are able to directly transmit frames to each other with no restrictions. The LAN concept is also directly related to the **broadcast domain** concept. When a node sends a frame to the broadcast address, the frame will reach every other node within the LAN but will never leave the LAN.

Layer two communication between nodes belonging to different LANs is not possible, nodes belonging to different LANs are required to use a layer three protocol to be able to communicate. Under the point of view of layer three protocols, each LAN is a different network, routers operate at layer three and have the mission of transferring layer three packets between different LANs.

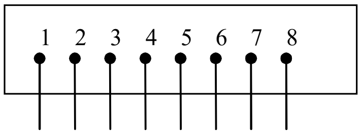
Nevertheless, as we have already seen, more than one layer three network can be defined over a single LAN, though a single layer three network cannot be split over more than one LAN.

A virtual LAN (VLAN) is a subset of a LAN we want to be operating as being a separate LAN, in other words, we can split a LAN into several VLANs in such a way each VLAN is equivalent to a separate LAN. Therefore, each VLAN is a broadcast domain and nodes belonging to different VLANs cannot communicate directly using layer two technologies.

VLAN operation is based on layer two devices, notably switches, but also layer two end node network interfaces, including routers and servers, may be VLAN aware.

Port-based VLANs can be set up on switches by assigning switch ports to different VLANs, as result, the switch will then forward frames only between ports belonging to the same VLAN, including broadcast frames. Under operations point of view, such a switch becomes equivalent to several independent and unconnected switches (one for each defined VLAN).

Let's take as an example a 8 port switch:



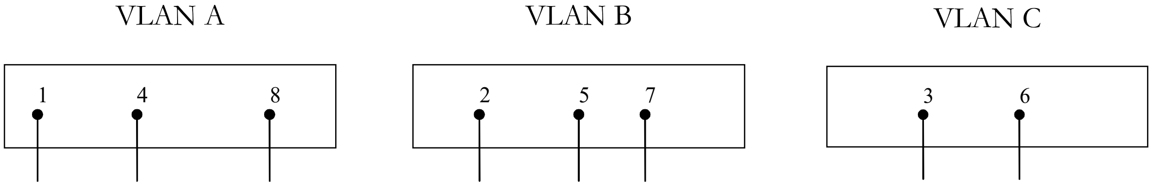
We can, for instance, define the following VLANs in this switch:

VLAN A: Port 1; Port 4; Port 8

VLAN B: Port 2; Port 5; Port 7

VLAN C: Port 3; Port 6

The original switch will be now equivalent to three separate unconnected switches:



For instance, a frame received in port 1 will only be retransmitted on ports 4 or 8, even if the destination address is FF:FF:FF:FF:FF:FF (layer two broadcast address).

1.1. VLAN frame tagging

The major operation principle for a VLAN-aware switch is: never mix frames belonging to different VLANs. Yet, for the sake of hardware optimization, it's possible to assign several VLANs to the same switch port. To be able to do so, each frame must carry a label or tag identifying to which VLAN it belongs to, this will allow the switch to avoid mixing frames of different VLANs.

The IEEE 802.1q standard describes how to place 12-bits identifiers, called VLANID, in ethernet frames. By using IEEE 802.1q assigning several VLANs to the same switch port is possible, in Cisco devices this is called **Trunk-mode**, in opposition to **Access-mode** where only one VLAN is assigned to the port.

Frames sent and received through a Trunk-mode port are not standard ethernet frames, they carry IEEE 802.1q VLANIDs. Ports on both ends of a cable connection must be configured the same mode using the same VLANIDs.

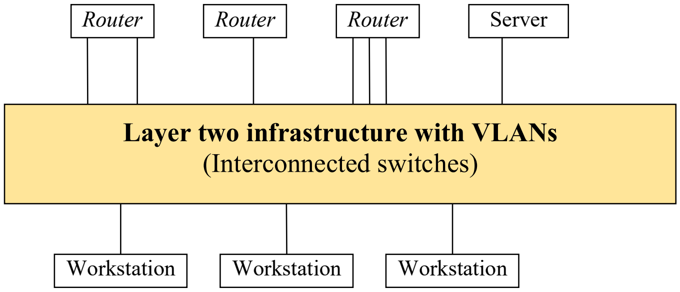
Trunk-mode ports can save a lot of hardware, imagine we have two switches, each with four VLANs: A, B, C and D, we usually want VLANs to be the same on both switches, so VLAN A on one switch must be connected to VLAN A on the other switch, and so on. With access mode ports only, this would require eight ports and four cable connections (one for each VLAN). By using trunk mode, the problem can be solved with a single cable connection and two trunk-mode ports.

Switches identify VLANs by VLANID (12-bits number), however, network administrators can also assign them names for easier management. VLAN names are local to each switch, in what concerns frame transaction between switches what really matters is the VLANID. VLANIDs must be set accordingly in all switches.

1.2. Layer two infrastructures with VLANs

One advantage of VLANs is they can be remotely managed, the network administrator can remotely access each switch and therefore change VLANs assigned to each port.

If we want to achieve a flexible layer two platform capable of being adapted to any layer three networks requirements we must build a **continuous layer two network**, based on interconnected VLAN-capable switches. Over this **continuous layer two network**, we can then enforce and manage whatever VLANs are required to meet the current needs.


Router
Router
Router
Server
Layer two infrastructure with VLANs
(Interconnected switches)
Workstation
Workstation
Workstation

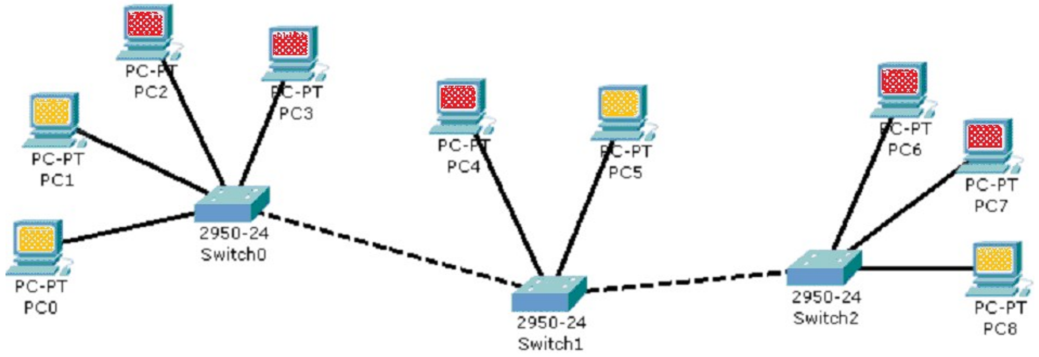
For instance, to change the network (VLAN) to which a node is connected it's just a matter of remotely accessing the switch to which the node is connected and change the VLAN assigned to the port. Without VLANs, accomplishing the same purpose would require going the appropriate cross-connect and physically change a patch cord connection at the patch panel.

Between different VLANs there is no layer two connectivity, this is provided by routers at layer three that are connected to several VLANs. A single physical connection of a router to the infrastructure can be used for several VLANs connections in trunk mode.

Nevertheless, it must be said VLANs are not entirely equivalent to physically independent networks, this is because the hardware is shared between different VLANs. For instance, a traffic overload in one VLAN will affect the shared infrastructure and will, therefore, disturb other VLANs.

2. Use the Packet Tracer tool to create the following layout

End node colours represent the VLAN each is connected to. Colours are for the sake of this diagram clarity and not to be set on Packet Tracer.



Although all nodes share the same layer two infrastructure we aim at achieving two distinct and independent networks.

Yellow nodes network: PC0; PC1; PC5 and PC8

Red nodes network: PC2; PC3; PC4; PC6 and PC7

2.1. Configure the three switches to achieve the desired purpose (using VLANs).

- For the yellow nodes network use VLANID=5

- For the red nodes network use VLANID=6

- These end nodes are not VLAN-aware, so they don't recognise IEEE 802.1q frames, therefore, the switch ports they are connected to should be in access mode, assigned to the single desired VLAN.

- We want both VLANs to be global to all switches, thus ports used to interconnect switches should be set to trunk-mode, with both VLANs assigned to them.

2.2. Test each VLAN wide by broadcasting ICMP echo requests

To be able to use ICMP we must first set some nodes IPv4 addresses:

- Assign to PC1 the 192.168.20.1/24 IPv4 private address.

- Assign to PC4 the 192.168.30.1/24 IPv4 private address.

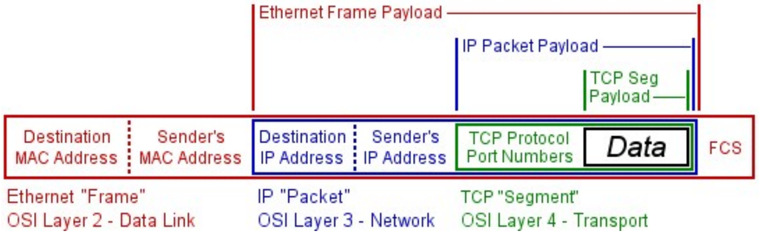
Now use the Add Complex PDU tool to send an ICMP echo request to the IPv4 broadcast address (255.255.255.255) every 5 seconds, from PC1 and then from PC4.

- Check that each VLAN is equivalent to an independent LAN (broadcast domain). Traffic, even when sent to the broadcast address, in never propagated from one VLAN to the other VLAN.

3. ARP – *Address Resolution Protocol*

When IPv4 (network layer three protocol) operates over ethernet (a layer two technology), each IPv4 packet is transported inside an ethernet frame, we say IPv4 packets are **encapsulated** in ethernet frames, in other words the IPv4 packet is the payload of the ethernet frame.

In the image we can see an IPv4 packet (blue) encapsulated in an ethernet frame (red), we can also see that, in turn, the IPv4 packet will be used by other upper-level protocols like ICMP or TCP in the image.



IPv4 addressing is independent of ethernet addressing (as it's supposed to be), this presents a challenge. IPv4 32-bits addresses mean nothing to ethernet, ethernet handles 48-bits addresses. The point is, when the IPv4 layer wants to send an IPv4 packet it must encapsulate the packet in an ethernet frame, to do so successfully it must also set the correct **Destination MAC Address** for the frame, however, it only knows the **Destination IP Address**.

Somehow, IPv4 must learn to which **Destination MAC Address** does the **Destination IP Address** corresponds to. This means: what is the **MAC Address** of the network node that is using that **IP Address?**

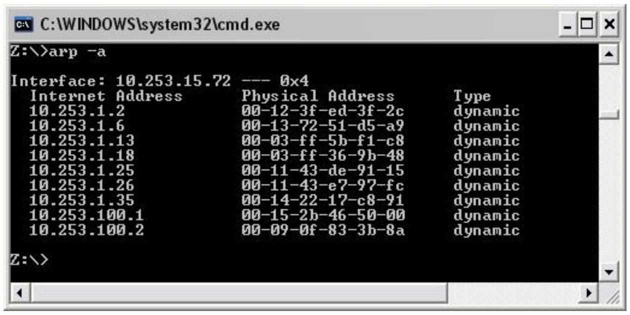
ARP (Address Resolution Protocol) was designed to solve this issue, IPv4 cannot operate without the help of ARP. We can see ARP as a function like this:

MAC node address = ARP (IPv4 node address)

ARP layer works side by side with IP and manages the so-called ARP table (not to be confused with switches MAC table)

The ARP table holds equivalences between IPv4 addresses and layer two MAC addresses. This is a dynamically managed table, while there are no communications, it is empty. Entries are created as needed, they have a short time to live and are removed if not refreshed.

When IPv4 asks ARP for a MAC address, it may be already at the table and in that case it's immediately returned.



```
Z:\>arp -a

Interface: 10.253.15.72 --- 0x4
  Internet Address      Physical Address      Type
  10.253.1.2            00-12-3f-ed-3f-2c     dynamic
  10.253.1.6            00-13-72-51-d5-a9     dynamic
  10.253.1.13           00-03-ff-5b-f1-c8     dynamic
  10.253.1.18           00-03-ff-36-9b-48     dynamic
  10.253.1.25           00-11-43-de-91-15     dynamic
  10.253.1.26           00-11-43-e7-97-fc     dynamic
  10.253.1.35           00-14-22-17-c8-91     dynamic
  10.253.100.1          00-15-2b-46-50-00     dynamic
  10.253.100.2          00-09-0f-83-3b-8a     dynamic

Z:\>
```

If the required MAC address is missing from the table, then the ARP layer uses the ARP protocol itself: an ARP request with the desired IPv4 address is sends in broadcast. Every IPv4 node has an ARP layer listening for ARP requests, they check if the requested IPv4 address is their own local IP address, if so, they reply with their own MAC address. Once the requesting node receives the ARP reply it will add it to the ARP table.

Because ARP operates by using broadcast it will only work within a broadcast domain, in other words nodes are required to be on the same LAN or VLAN.

4. IPv4 packets routeing

Two IPv4 nodes can communicate directly (without routers) only if two preconditions are meet:

A – Both are connected to the same LAN/VLAN (same broadcast domain)

B – Both nodes IPv4 addresses belong to the same IPv4 network.

We can see precondition **A** arises straight from ARP.

An IPv4 sending node must somehow know if a given destination IP address is reachable directly or not. If reachable directly it just needs to use ARP and then encapsulate it inside a layer two frame, otherwise the packet must be sent to a router.

The way a sending node decides this is by matching the given destination IP address with the local IPv4 network if the network prefix is the same we assume direct communication is possible. From here comes condition B.

4.1. Routers

Routers (aka gateways) are layer three intermediate nodes, they forward layer three packets and not layer two frame like switches do. The mission of an IP router is receiving IP packets from one LAN/VLAN and retransmitting then on another LAN/VLAN.

4.2. Using routers

When an IP sending nodes checks the IP destination address of the packet does not belong to the local IP network, it knows a router must be used, so the **router IP address** must be known.

One IP node may be aware of several routers around it, but end nodes are usually aware of only one router they can use, this is an additional required configuration parameter usually called **default-gateway** (aka default-router). If a node is not aware of any available router it will never be able to communicate with nodes beyond the local IP network.

If the destination IP address does not belong to the local IPv4 network, the IPv4 packet must be sent to the default-.gateway instead. Sending to the default-gateway works the same as before, the IPv4 packet must be encapsulated into a layer two frame, and ARP must be used to set the appropriate **Destination MAC Address**. The only difference is that now, we will be requesting ARP for the default-gateway MAC address, and not the destination IP node MAC address.

Because communications with the router use layer two encapsulation and ARP, one condition must be met for a router address to be valid:

A router address is valid only if it belongs to the local IPv4 network.

4.3. Routeing tables (or routing tables)

The difference between a router and an end node is a router is supposed to retransmit IP packets, so it must be connected to more than one IP network. Thus the router mission is more complex, while an end node has only two options (local destination or nonlocal destination) the router has more options.

A typical end node only need to know the local IP network, if the destination address does not belong to the local network then the packet is sent to the default-gateway. This means, all other networks are reachable through the default-gateway.

A typical router is connected to several networks and has several routers available around it to be used. Each router around it will provide access to some IP networks, it needs to know which networks access is provided by each of its neighbour routers. This is the role of the routeing table.

The routeing table is a list of IP networks (IP and prefix), and for each, the IP address of the neighbour router that should be used, the router to be used is called **next-hop**.

Take for instance a router connected to networks 192.168.10.0/24 and 192.168.20.0/20, the routeing table can be something like:

Destination	Next-hop
192.168.5.0/24	192.168.10.7
192.168.8.0/24	192.168.10.7
192.168.34.0/24	192.168.20.170
192.168.38.0/24	192.168.20.200

When this router receives an IPv4 packet for forwarding it will look at its destination IPv4 address to see which network it belongs to, options are:

1º Belongs to network 192.168.10.0/24 => direct sending (layer two)

2º Belongs to network 192.168.20.0/24 => direct sending (layer two)

3º Belongs to network 192.168.5.0/20 => send to 192.168.10.7 router

4º Belongs to network 192.168.8.0/24 => send to 192.168.10.7 router

5º Belongs to network 192.168.34.0/24 => send to 192.168.20.170 router

6º Belongs to network 192.168.38.0/24 => send to 192.168.20.200 router

Once a match is found the packet is sent and processing ends. If no match is found the packet is discarded, if this happens, it means the router does not known the destination network.

Checking if an address belongs to a network is achieved by applying the network mask (bit-to-bit and) to the address and see if the network address is obtained.

Routeing tables can be manually set, this is called **static routeing**. There are also routeing protocols that are able to build routeing tables and keep them updated, this is called **dynamic routeing**.

4.4. Default route

We cannot store in single routeing table all networks being used around the internet, however, if a network is unknown to a router it will be unreachable.

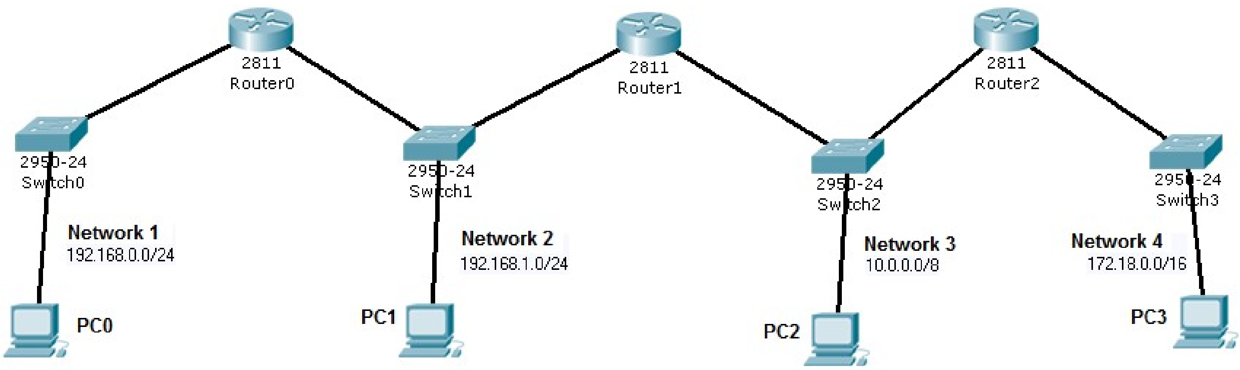
The workaround is defining the special network 0.0.0.0/0 in each routeing table. Because the mask is zero, it will always match any IP address, so it's always be the last entry at the routeing table.

This entry is called the **default-route**, and the corresponding next-hop is called the **default-gateway**. Because is matches every IP address and is placed at the end of the table, the result will be that any packet addressed to an unknown network will be forwarded to the default-gateway.

Usually the default-route allows routeing table simplifications, generally speaking, any routeing table entry with a next-hop equal to the default-gateway can be removed. This is true because in its absence the table processing will continue until reaching the default-route and the result will be the same, the packet is sent to the same router.

5. Use the Packet Tracer tool to create the following layout

There are four IPv4 networks interconnected by three routers



5.1. Define all layer three nodes IPv4 addresses (routers and end nodes)

Used IPv4 addresses must belong to the represented IPv4 networks.

Check that, for now, ARP tables are empty (use the Inspect tool – Magnifying Glass).

5.2. Check IPv4 connectivity

Use the Add PDU tool to send ICMP echo requests between nodes.

Check that tests within each LAN are successful (from routers to local end nodes), but tests between different networks fail.

Also check that ARP tables are not empty any more.

5.3. Define end nodes' (PCs) default gateways

PC1 and PC2 have two alternative routers, in both cases use Router1 as default-gateway.

5.4. Test again communications, now only between end nodes (PCs)

Check that between PC1 and PC2 everything works fine, but not with other nodes.

Try again in simulation mode to try understanding what is happening.

5.5. Define each router routeing table to solve the issue

For each router, check the remote networks it is not aware of. For each, add a static routeing entry to inform to where the packets should be forwarded to reach that network.

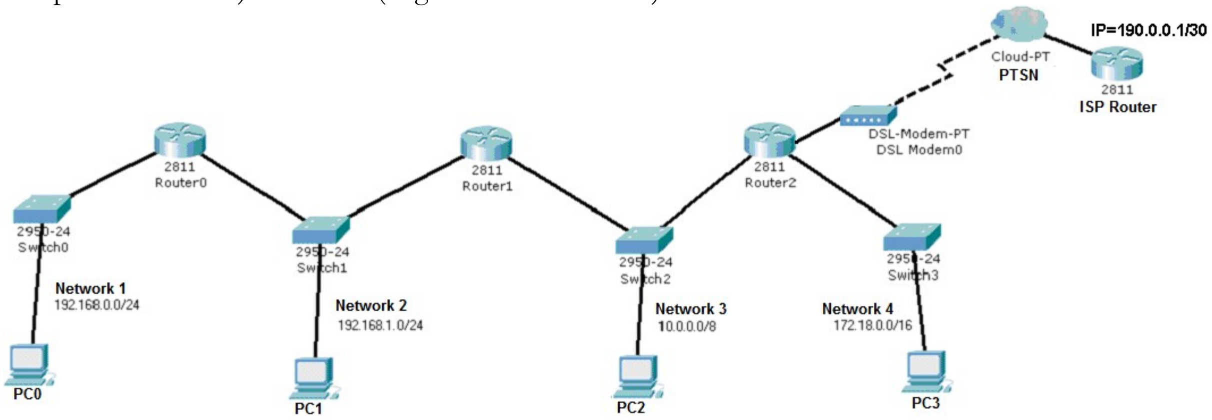
5.6. Test again IPv4 connectivity between all end nodes

Check that now every node can communicate with every other node.

Check that ARP tables include only local addresses.

6. Keep the previous layout and add an internet connection to Router 2

Router 2 will be connected to an ISP (Internet Service Provider) using the PSTN (Public Switched Telephone Network) with DSL (Digital Subscriber Line).



To establish the layer two connection, configure the cloud associating the DSL connection to the Ethernet connection

Set appropriate IPv4 addresses for the new router and for the new Router 2 interface. Because the internet connection mask has 30 bits, there are only two valid node addresses, if the ISP Router is using 190.0.0.1/30, then the only available valid node address is 190.0.0.2/30.

Before advancing, check if there is IPv4 connectivity between Router2 and the ISP Router.

6.1. Change the routers routeing tables to represent the new reality

Now there is an internet connection, therefore you must add a default-route to each routeing table, whenever the destination address of a packet is locally unknown, it should be routed to the ISP Router.

Add the necessary default-route entries in each local router.

In simulation mode use the Add Complex PDU tool to create an ICMP echo request addressed no a locally unknown address, say **10.10.10.10**. Send the request from PC0.

Check that all routers are forwarding unknown address packets in such a way they reach the ISP Router.

6.2. Simplify routeing tables

Check that some entries in Router0 and Router1 are now useless because their next-hop is the default-gateway. Remove them.

Check that all works as before.