Spring 2014
Computer Networks
CMPE323

Laboratory Experiment No. 3: Introduction to VLANs

Aims and Objectives:

- Introduce students to Virtual Local Area Networks (VLANs),
- the general semantics of VLANs within a switch,
- the general semantics of VLANs between switches,
- and the structure of IEEE 802.1Q tags which allows for maintaining inter-switch VLAN separation.

Materials Required:

- Ethernet switches,
- PCs with Ethernet adapters,
- and Straight-through/crossover/rollover cables.

Change Log:

- 3-3-2014: original document – mkhonji.
- 3-3-2014: fixed error about default Cisco trunking behaviour + added missing argument for SPAN monitoring command – mkhonji.

1 Introduction

1.1 Limitations in the previous lab

The following limitations existed in the previous lab[1]:

- Broadcast domain limitations — each switch formed a single broadcast domain where *unknown unicast*[2] and *broadcast*[3] messages were forwarded to all connected ports on the same Ethernet switch. This can cause efficiency[4] and security[5] issues.

- Management limitations — in order to have separate broadcast domains, separate physical switches were used, and in order to relocate a PC from one broadcast domain into another, we had to physically disconnect it from one switch and physically connect it to another switch.

1.2 A solution to the limitations

In this lab, by using Virtual LANs (VLANs), you will learn how to address the two limitations that existed in the previous lab as follows:

- Fine-grained broadcast domains — by assigning groups of switch ports to different VLANs, you essentially create different broadcast domains within a single switch. Although this is virtual (by software configuration), it is practically the same as using separate physical switches from the perspective of the connected nodes (or PCs).

- Easier management — when a network administrator wishes to relocate one PC from a broadcast domain into another, all what is needed is changing the software configuration of the switch port such that its VLAN membership is updated to the desired one. In other words, we no longer need to physical disconnect/connect cables in order to relocate nodes (or PCs) from one broadcast domain into another.

  Additionally, it is possible to maintain the semantics of such virtual broadcast domains across multiple Ethernet switches. Since broadcast domains have identifiers (i.e. VLAN ID), a special connection[6] between two switches allows MAC frames of different VLANs to be tagged accordingly when being transmitted to other switches, so that the switches can maintain broadcast domain isolation based on such VLAN IDs (as the IDs are communicated by the MAC frame IEEE 802.1Q tags).

[1] Lab 2: Introduction to Ethernet Networks

[2] MAC packets with destination MAC addresses that do not exist in the MAC address table of the forwarding switch

[3] MAC packets with destination MAC addresses of `FF:FF:FF:FF:FF:FF`.

[4] Efficiency problems as excessive messages can consume the link bandwidth, specially when the broadcast domain is large.

[5] Some attacks can be performed only when the attacker resides in the same broadcast domain as the victim; e.g. MAC address table spoofing, and ARP spoofing.

[6] A connection between two switches that accepts tagged MAC frames by IEEE 802.1Q tags.

1.3 Specifications of the IEEE 802.1Q tag

The standard IEEE 802.1Q defines an Ethernet tag that facilitates the following:

- A new `Type` field is added. It must have the value of `0x8100` in order to indicate that there are 2 proceeding octets describe Tag Control Information as depicted in Figure 1 — under the scope of this lab.
- A set of bits to convey the VLAN ID. As depicted in Figure 2, there are 12 bits that are assigned to specify the VLAN ID. — under the scope of this lab.
- A set of bits to describe the priority of the MAC frame which helps later in Quality of Service (QoS) tasks — Priority Code Point (PCP) and Drop Eligible Indicator (DEI) are beyond the scope of this lab.

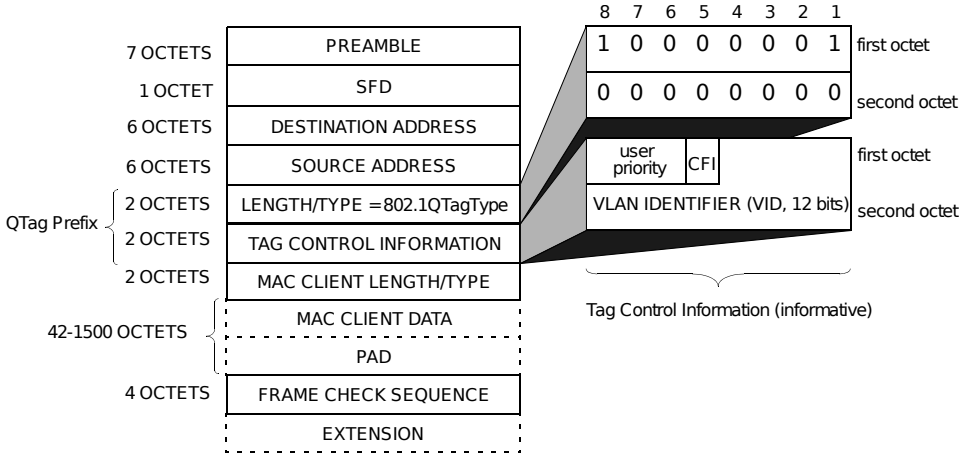

Figure 1: MAC packet example with a IEEE 802.1Q tag — Source: IEEE Std. 802.1Q-2011, Annex G.1.

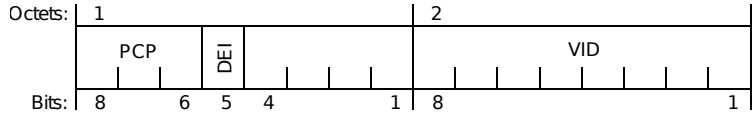

Figure 2: VLAN Tag Control Information (TCI) — Source: IEEE Std. 802.1Q-2011, Section 9.6.

2 Lab Preparation

- Setup the lab topology as depicted in Figure 3.
- Reboot all the lab PCs[7].
- Erase[8] the configuration of the switches, and then reboot[9] them.

[7] Use the `reboot` command in case you are using Linux PCs.

[8] `erase startup-config`.

[9] `reload`.

- Run WireShark on all the involved PCs.

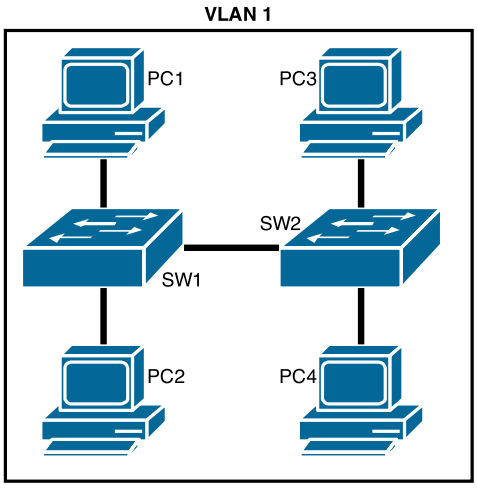

Figure 3: Initial lab network diagram.

3 Lab experiments

3.1 A single broadcast domain

[20 points]

Note: When done, show your work to the lab engineer for grading purposes.

Using `scapy` or the provided C code, broadcast the following MAC packet:

- Source MAC address: `00:00:00:00:00:0X`[10].
- Destination MAC address (broadcast): `FF:FF:FF:FF:FF:FF`.
- Data type: `0x05FF`[11].
- Payload data: `TeamY`[12].

Then answer the following questions:

1. List the PC IDs that received the MAC packet that you have sent earlier.
2. Logically thinking, list problems that can arise if many more Ethernet switches are interconnected to form a large broadcast domain.

[10] `X` is your PC ID (e.g. PC1 will have the MAC address `00:00:00:00:00:01`).

[11] You are free to choose any other valid value for the `Type` field.

[12] Y is your team ID. For example, for team 1 you would send octets `0x54, 0x65, 0x61, 0x6D, 0x31`.

3.2 Virtually separated broadcast domains

[40 points]

Note: When done, show your work to the lab engineer for grading purposes.

As depicted in Figure 4, re-configure both of the switches `SW1` and `SW2` as follows:

1. In both of the switches, create[13] VLANs 100 and 200. `SW2`.
2. In both of the switches, assign[14] PC1 & PC3 to VLAN 100, and PC2 & PC4 to VLAN 200.
3. In both of the switches, configure[15] the interfaces that connects them together (the inter-switch link) as trunk interfaces. Then verify that the trunk interface is up and running by the `show` commands[16].

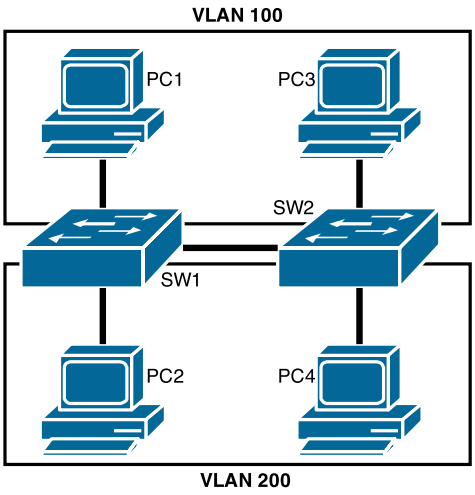

Figure 4: Lab network diagram with VLANs.

Then answer the following questions:

1. Send a broadcast MAC packet from PC1, then list the PCs that received the MAC packet.
2. Send a broadcast MAC packet from PC2, then list the PCs that received the MAC packet.
3. Send a broadcast MAC packet from PC3, then list the PCs that received the MAC packet.
4. Send a broadcast MAC packet from PC4, then list the PCs that received the MAC packet.
5. Based on the tests above, what do you conclude with regards to the scope of the broadcast domain?

[13] `enable`, `configure terminal`, `vlan 100`, and `vlan 200`

[14] `enable`, `configure terminal`, `interface FastEthernet 0/A`, and `switchport access vlan B`, where `A` is the interface that ID, and `B` is the VLAN ID.

[15] `enable`, `configure terminal`, `interface FastEthernet 0/T`, `switchport trunk encapsulation dot1q`, and `switchport mode trunk`.

[16] `show interface trunk`.

3.3 Analyze IEEE 802.1Q tags

[40 points]

Note: When done, show your work to the lab engineer for grading purposes.

Connect your laptop[17] to your team's switch as depicted in Figure 5, run WireShark on your laptop, then configure a monitoring session in your team's switch as follows:

- Configure the trunk port of your team's switch as a monitoring source[18].
- Configure the laptop's port as a monitoring destination[19].

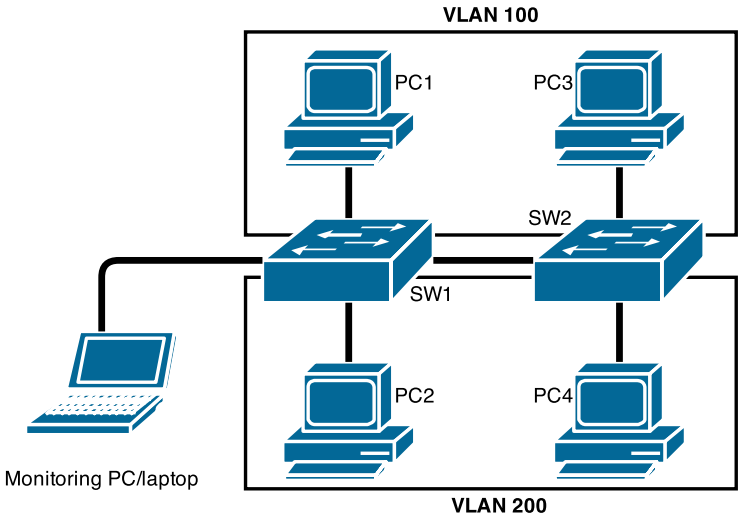

Figure 5: Lab network diagram with VLANs after connecting the monitoring PC/laptop.

Then answer the following questions:

1. Send a broadcast MAC packet from PC1, then view the WireShark instance on the monitoring laptop to analyze the IEEE 802.1Q tag. What is the content of the tag?.
2. Send a broadcast MAC packet from PC2, then view the WireShark instance on the monitoring laptop to analyze the IEEE 802.1Q tag. What is the content of the tag?.
3. Send a broadcast MAC packet from PC3, then view the WireShark instance on the monitoring laptop to analyze the IEEE 802.1Q tag. What is the content of the tag?.

[17] You may use a desktop PC if available.

[18] `enable`, `configure terminal`, `monitor session 1 source interface FastEthernet 0/T`, where `T` is the trunk port ID.

[19] `enable`, `configure terminal`, `monitor session 1 destination interface FastEthernet 0/L encapsulation replicate` , where `L` is the port that connects to the monitoring laptop.

4. Send a broadcast MAC packet from PC4, then view the WireShark instance on the monitoring laptop to analyze the IEEE 802.1Q tag. What is the content of the tag?.

5. Based on the observations earlier, how can inter-connected Ethernet switches maintain VLAN separation of frames?

3.4 Extra questions (not graded)

In this section, you will be able to use your laptop to inject MAC packets into various VLANs.

Note: when done, show your answers to the lab engineer for feedback. If the lab time is not enough, take your time and submit it in a later time. Although not graded, it can strengthen your understanding of the subject.

1. Remove the monitoring session that you have configured earlier in Section 3.3, and instead, configure the port `FastEthernet 0/L` as a trunking port.

2. Then, using your understanding of the format of IEEE 802.1Q tags, craft a message from the laptop such that you are able to send a message to targeted VLAN.

   For example, send a message from your laptop (which is connected to the trunk port) to VLAN 100 and test if PCs in VLAN 100 receive the message. Then, repeat it with VLAN 200 and test if PCs in VLAN 200 can receive the message.

3.4.1 Discussion

In this task, you were able to gain access to all VLANs after configuring the switchport that faces your laptop as a trunking port. Which is not necessarily a security threat as the trunking port configuration was manual after having physical access to the switch, which is usually only available to authorized personnel.

However, since Cisco switches run an insecure setup of VLAN Trunking Protocol (VTP)[20] by default, it is possible to enable such trunking ports automatically from your PC/laptop without the need of configuring the switch.

This is why it is recommended to disable VTP on ports that face end users PCs (access ports), and secure VTP negotiations with passwords when facing other switches.

Note: VTP is a Cisco-specific protocol and other vendors may not have similar protocols.

[20] A protocol that allows switches to negotiate in order to automatically create trunking interfaces.