

Computer Networks – Practical session #2

Ethernet performance & VLAN

M1 MOSIG

March 2022

When booting the computers, choose “FreeBSD”, and then login with:
Username: “root”
Password: “root./”

In this practical session, you will experiment with performance testing of Ethernet networks. The goal will be to understand the behaviour and performance of a communication network in two cases: a shared medium (CSMA/CD using a hub), and a dedicated medium without collision.

Then, the second part will be dedicated to learning about VLANs. You will manipulate VLANs again in the next practical session.

Don't forget to send your report to the teacher at the end of the practical session!

1 Analyzing network performances

In this part of the practical session, you will experiment with collisions in a shared media, and perform experiments to determine the performance of the local network.

1.1 CSMA/CD

As a reminder, the Media Access Control protocol of Ethernet is called CSMA/CD: Carrier Sense Multiple Access/Collision Detection. It is designed to be resilient to collisions at the physical layer, which happen when two computers try to transmit at the same time on the same medium.

Note: In most modern networks based on Ethernet switches, frame collisions are not possible. Indeed, an Ethernet switch isolates the collision domain of each connected computer, by buffering each frame it receives before sending it to the destination. In contrast, an Ethernet hub acts as a simple relay of the electrical signals sent by computers. The difference is depicted in Figure 1. For the purpose of this practical session, we will study Ethernet hubs as a simple model for shared media where collisions cannot be avoided (for instance Wi-Fi).

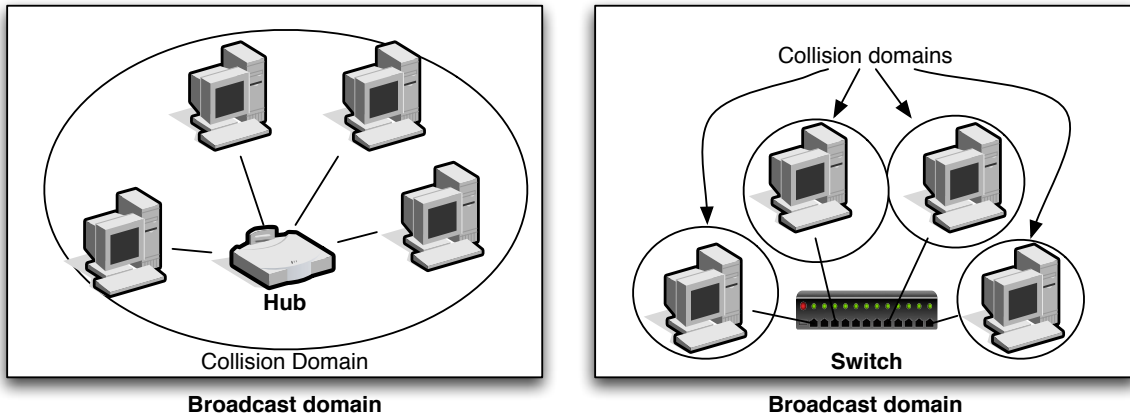


Figure 1: Collision domain: hub vs. switch

1.2 First evidence of collisions

To see collisions, it is therefore necessary to use a hub, or configure an interface in half-duplex

- ☞ Setup the network depicted in Figure 2, using static IP addressing.

To configure a static IP address on your computer, you can use:

```
ifconfig interface XX.XX.XX.XX/YY up
```

- ☞ To configure an interface of the switch in half duplex, connect the serial line to the switch, type `minicom`, hit the return key several times and type `configure` and then `int <port #> speed-duplex 100-half` (to undo this change, type `int <port #> speed-duplex auto`).

Q 1 — Why do you want to set *only* the PC1/switch interface in half duplex?

- ☞ On pc1, run `netstat` to display at regular intervals (let's say 2 seconds) the number of collisions observed on the network interface connected to the hub (here, `igb0`) :

```
# netstat -w 2 igb0
```

- ☞ On pc1 and pc2, run `udptarget -k`. This program will wait for incoming UDP packets, and print statistics on packets it receives.

- ☞ On pc2, use `udpmt` to generate a continuous UDP stream towards pc1.

The syntax for `udpmt` is the following (more options will be discussed later):

```
udpmt [-p udp_port] target_ip
```

Q 2 — Note the number of collisions on pc1.

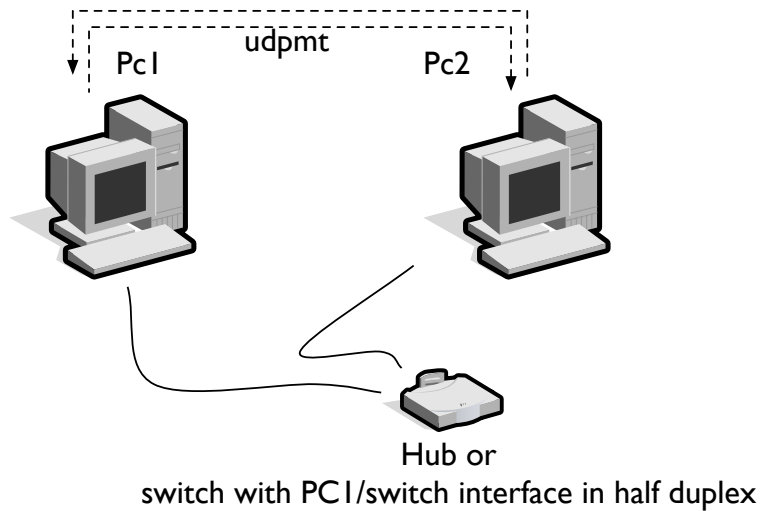


Figure 2: First evidence of collisions

☞ On pc1, use `udpmt` to generate a second UDP stream towards pc2. The first stream from pc2 to pc1 must still be running.

Q 3 — Note the number of collisions on pc1. Explain the change of behavior, discussing the CSMA/CD protocol.

1.3 How to measure effective throughput

In the following sections, you will use the `udpmt` program to measure the effective throughput of the network. Remember that the effective throughput is the throughput that the user actually experiences, as opposed to the theoretical capacity of the network (e.g. 100 Mbit/s). The effective throughput is lower than the capacity for various reasons: other users using the network, collisions...

For the experiments, you may need the following options for `udpmt`:

- `-s <size>`: packet size in bytes
- `-d <secs>`: test duration in seconds (default: transmit indefinitely until stopped)

See `man udpmt` for other options.

1.4 Effective throughput for a single stream

☞ Connect pc3 and pc4 directly with an Ethernet cable (**without the switch**).

☞ On each computer, use `udpmt` to send an UDP stream to the another computer.

Q 4 — What is the throughput reported by `udpmtarget`?

☞ Perform the experiment again with varying packet size (from 60 to 2880 bytes). If `udptarget` displays lots of errors, run it with `udptarget -k -q`.

Q 5 — *What is the effect of packet size on the measured throughput? Write down a table showing how the throughput changes with the packet size. In particular, pay close attention to what happens for small packets and around 1472 bytes. What can you deduce, remembering that the capacity of the Ethernet adapters on your computers is 1000 Mbit/s?*

☞ To understand why the effective throughput is lower than the capacity, perform a packet capture with Wireshark during a few seconds only.

Q 6 — *Analyze the frames captured by Wireshark. What can you deduce about the performance anomaly, in particular when you use a packet size greater than 1472 bytes?*

Here are a few points that will help you to understand:

- `udpmt` uses the UDP protocol to send its messages and measure bandwidth;
- if an IP packet is too large to fit in a single Ethernet frame, it will be fragmented;
- the performance bottleneck can either comes from the network capacity, or from the CPU of the computers (sender and/or receiver);
- the CPU processing time mainly depends on the number of packets that need to be processed: the size of the packet has a limited impact on CPU performance.

1.5 Effective throughput with multiple streams and a switch

☞ If you did configure an interface of the switch in half duplex, undo it.

☞ Connect all computers back to the switch.

☞ Launch two simultaneous UDP streams through the switch, as depicted in Figure 3.

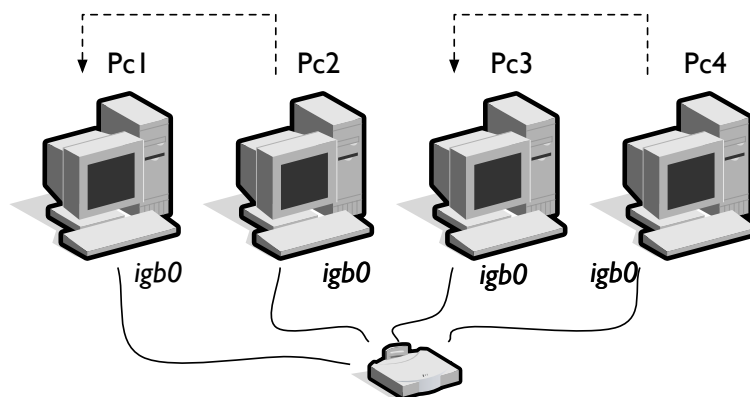


Figure 3: 2 streams and a switch

Q 7 — *What do you notice about the measured throughputs?*

☞ Now run `udpmt` from three stations to a fourth one, as shown in Figure 4. You need to run several `udptarget -k` listening on different ports!

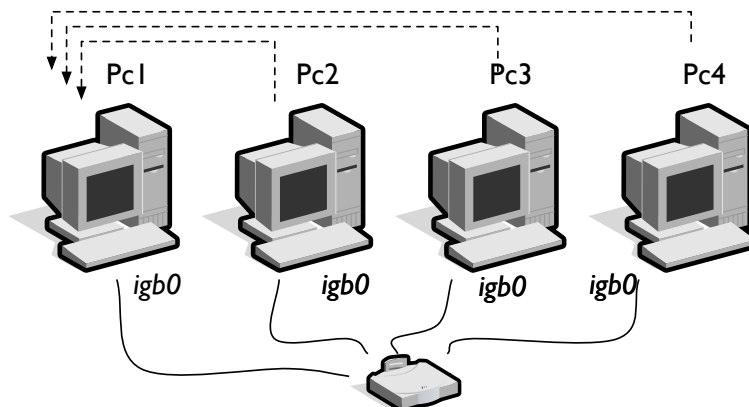


Figure 4: 3 convergent streams

Q 8 — *Compare with the previous results: what throughput do you obtain for each stream? Do you see collisions?*

1.6 Effective throughput with multiple streams and a switch

☞ Perform the same experiments (Figures 3, 4), but replace the switch with a hub.

Q 9 — *For each experiment, compare the results with those obtained with the hub, keeping in mind that the switch works at 1000 Mbit/s while the hub works at 100 Mbit/s. In particular, what can you say about performance? Do you see collisions?*

Q 10 — *Consider 3 converging streams vs. 3 diverging streams on the hub (3 UDP streams as in Figure 4, or 3 streams emanating from PC1 to the others). Which mechanism(s) are responsible for sharing the available capacity between streams? Is it the same mechanism both cases?*

2 Virtual Networks - VLANs

A Local Area Network (LAN) is defined as a single broadcast (or “diffusion”) domain. It means that all hosts in the LAN can be reached directly using broadcast frames.

Some switches (in particular the *HP Procurve 6108*) allows to create several virtual LAN, called **VLANs**. These are LANs that virtually group together different computers: the computers can be grouped according to **logical** (instead of physical) criteria, such as MAC address, port number, protocol, etc. Each virtual LAN behaves like a physical LAN.

Manageable switches allow to configure VLANs. By default, all the ports are attributed to one single VLAN. It is often possible to manage a switch via telnet, a serial line, or a web interface.

2.1 Switch administration how-to

A switch is a layer 2 device, but it implements basic layer 3 functionality, so that it can be managed using IP.


`telnet` will allow a privileged connection: you can take a look on the configuration and pass in configuration mode. The prompt is then:


```
HP ProCurve Switch 6108#
```

Small howto (CLI) :


- There exists a hierarchy in the commands: each level becomes more and more specific. The "?" prints the available commands.
- The tabulation allows an automatic completion ;
- Shortcuts are allowed (if there is no ambiguity):
`sh ru` is equivalent to `show running-configuration`
- `exit` (or `^z`) exits from the current mode
- To cancel or disable a piece of configuration, use `no` followed by the command
- When you use a serial line, you must configure the program with 9600 bauds, 8N1 parity, and deactivate the hardware flow control

2.2 Basic switch configuration

 Connect one PC to the switch, and configure a static IP address `192.168.0.X/24` on the PC.

 Access the switch using `telnet` towards `192.168.0.254`.

If it does not work, reinitialize the switch as described above.

 Verify the current configuration with `show running-config`. In particular, make sure all ports belong to VLAN 1. It should look like this (if not, reset the switch):

```
vlan 1
  name "DEFAULT_VLAN"
  untagged 1-8
  ip address 192.168.0.254 255.255.255.0
  exit
```

To enter in configuration mode, enter `configure terminal` and the prompt will become :

```
HP ProCurve Switch 6108(config)#
```

2.3 VLAN configuration

We will focus on layer 1 VLAN (defined by port) and *VLANs 802.1Q* to tag frames according to their VLAN number.

The reference configuration is described in Figure 5.

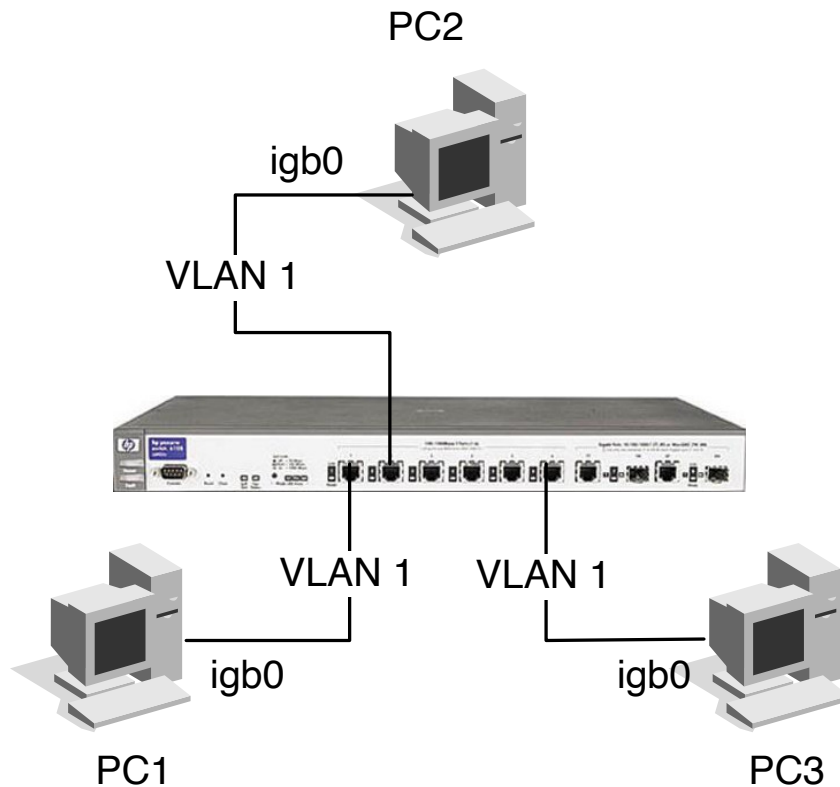


Figure 5: Reference configuration



Configure the interfaces of PC1, PC2 and PC3 in the same subnetwork. Check that they can all communicate together. Make sure at least one PC can still communicate with the switch to configure it (or use the serial line, as before).

2.3.1 Layer 1 VLANs / ports VLANs

In this case, a port of the switch is associated to a single VLAN. Any computer connected to this port will belong to the specified VLAN.

You must first define a new VLAN, and associate a given port to this new VLAN. To create a *VLAN X* and associate *port Y* to this new VLAN:

```
HP ProCurve Switch 6108(config)#vlan X
HP ProCurve Switch 6108(vlan-X)#untagged Y
```

The commands `show vlans` and `show vlan X list` respectively the existing VLANs and their associated ports.

☞ Associate the port plugged to PC3 to a new VLAN (i.e. not the default VLAN).

Q 11 — *Try to ping PC2 from PC3. What happens? Why?*

☞ Assign to PC3 a different IP address (not in the original subnetwork)

☞ Using a USB/Ethernet adapter plugged in PC2, configure a new interface to allow PC2 and PC3 to communicate together, as shown in Figure 6.

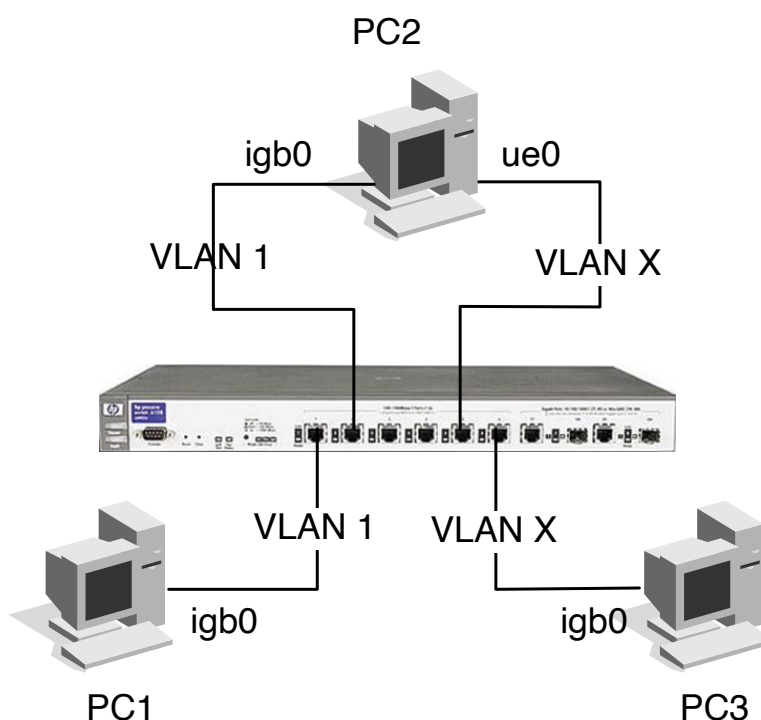


Figure 6: Layer 1 VLANs

Q 12 — *Check that the diffusion domains of the two VLANs are isolated. How do you check that?*

With this approach, a port can only be assigned to a single VLAN. We will now use 802.1Q to tag the frames.

2.4 Tagged VLANs

We will now introduce the notion of virtual interface. We can associate one or several virtual interfaces to one single physical interface, connected to different VLANs.

We will use two virtual interfaces over `igb0` on PC2 to receive tagged frames. You must create and configure a virtual interface (named `igb0vlan42` for instance) by assigning IP parameters, the `vlandev`, the physical interface it will use, the `vlan` (the tag id) (cf. `man ifconfig`). For instance, using vlan id 42:

```
ifconfig vlan42 create
ifconfig vlan42 10.0.0.2/24 vlan 42 vlandev igb0
```

Naturally, the switch configuration must correspond to this configuration. For instance, the port of the switch connected to PC2 must be tagged and own both VLANs. If you want to configure *port Y* of the switch to accept frames from *VLAN X*, you must enter the command:

```
HP ProCurve Switch 6108(vlan-X)#tagged Y
```



Set up the topology described in figure 7. Note that the ports plugged to PC1 and PC3 must not be tagged.

You should disable VLAN tag offloading on the physical interface, otherwise you won't see anything interesting in the next question:

```
ifconfig igb0 -vlanhwtag
```

Q 13 — *Verify that PC2 can ping PC1 and PC3. Using Wireshark, take a look at the frames on PC1, PC2, and PC3. What do you observe? What is the VLAN field in the frame?*

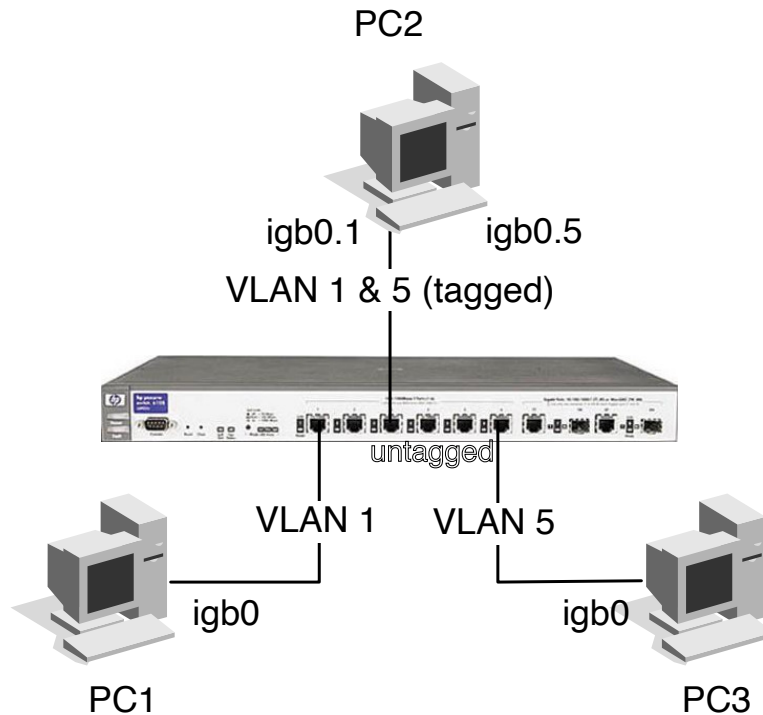


Figure 7: 802.1Q VLANs

Q 14 — *Bonus question: what is the purpose and effect of the `-vlanhwtag` option used above?*

Before leaving, make sure to put all cables and hardware back to their storage location.

To send your report at the end of the practical session, you may need to connect the computer back to the school network. This can be done using `ifconfig em0 delete down` followed by `dhclient em0`.