IT TECHNOLOGY

NETWORKING

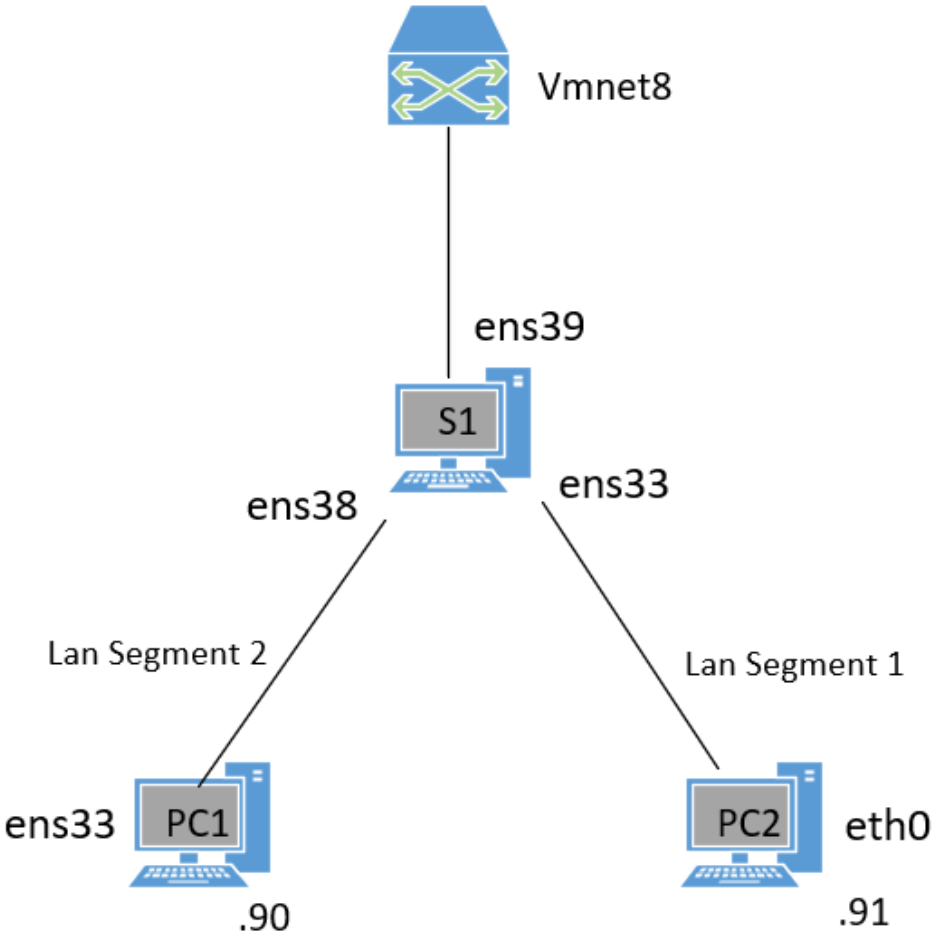
Assignment 4, The Switch and The Hub

University College
Author

Buterchi Bogdan

bdbu37436@edu.ucl.dk

The diagram



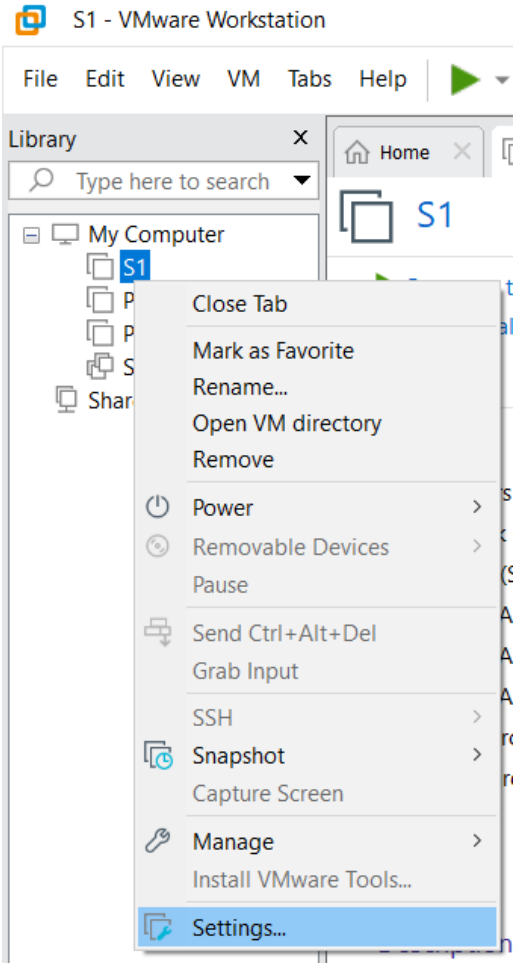
Setting up the switch

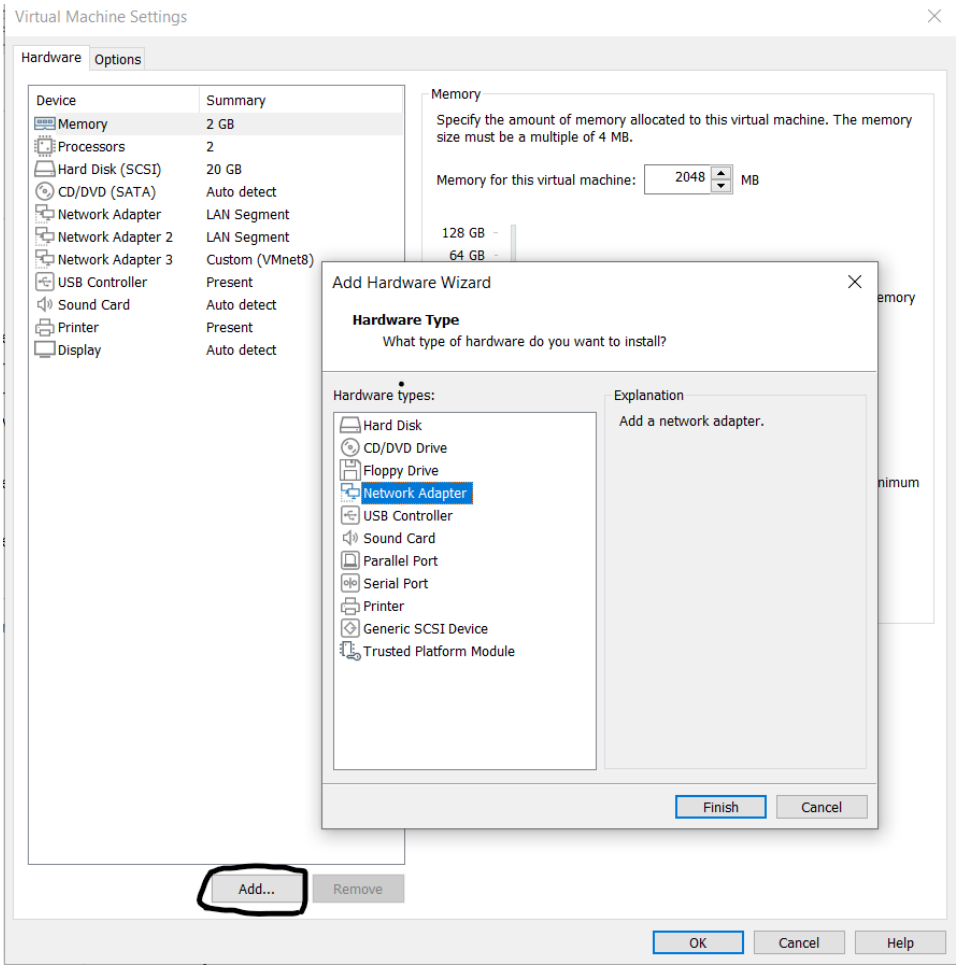
First we will install “bridge-utils” and “ifupdown” on the Switch. To do this use the command “sudo apt-get install bridge-utils ifupdown". But before doing this make sure you have your system up to date with the “sudo apt update” then “sudo apt upgrade”.

We will interconnect all devices together according to the diagram.

The switch will be an Xubuntu virtual machine, along with PC1, while PC2 will be a RaspberryBuster.

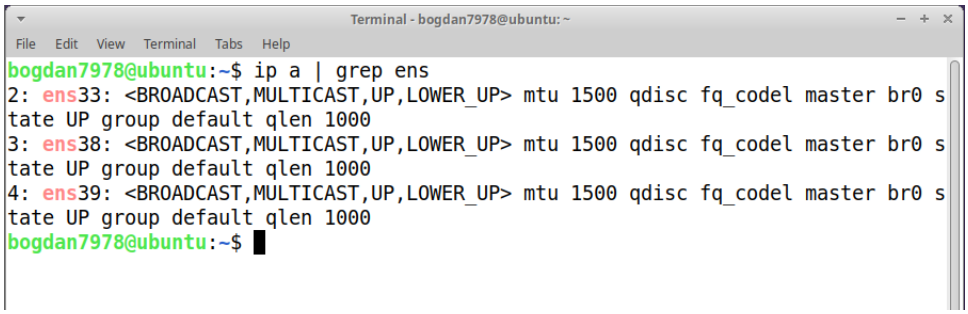
Right-click on the virtual Switch and go to Setting and then Add another “Network Adapter”


S1 - VMware Workstation
File
Edit
View
VM
Tabs
Help
Library
Type here to search
My Computer
S1
Home
S1
Close Tab
Mark as Favorite
Rename...
Open VM directory
Remove
Power
Removable Devices
Pause
Send Ctrl+Alt+Del
Grab Input
SSH
Snapshot
Capture Screen
Manage
Install VMware Tools...
Settings...

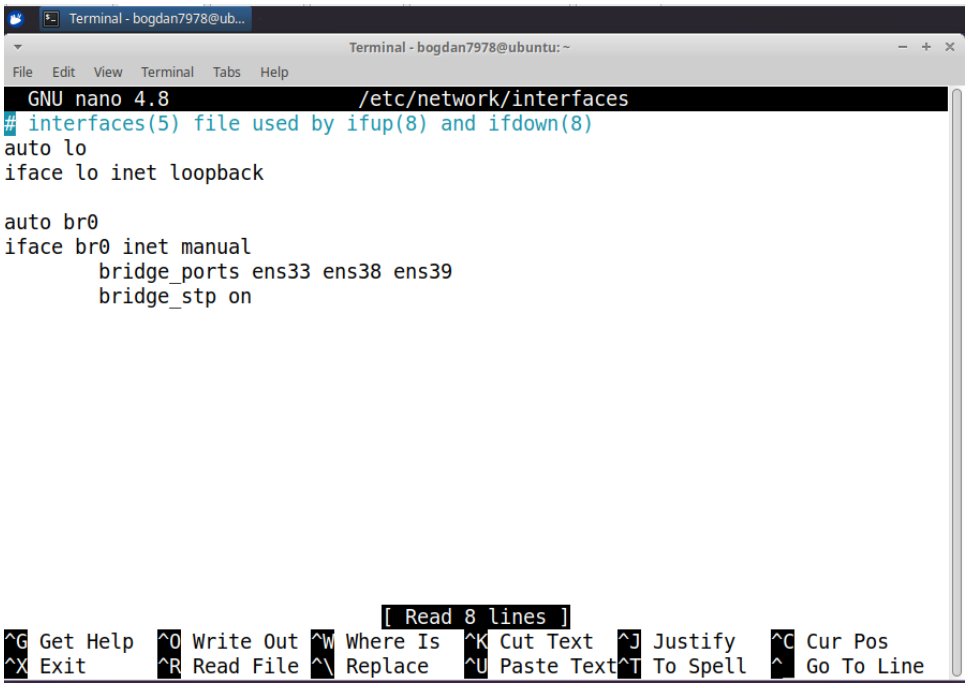


Do this 2 times so you have 3 Network adapters in total.

Now we will check on the Switch, with the “ip a | grep ens” command, to see the network adapters ens connections. They are in the same order as the network adapters.



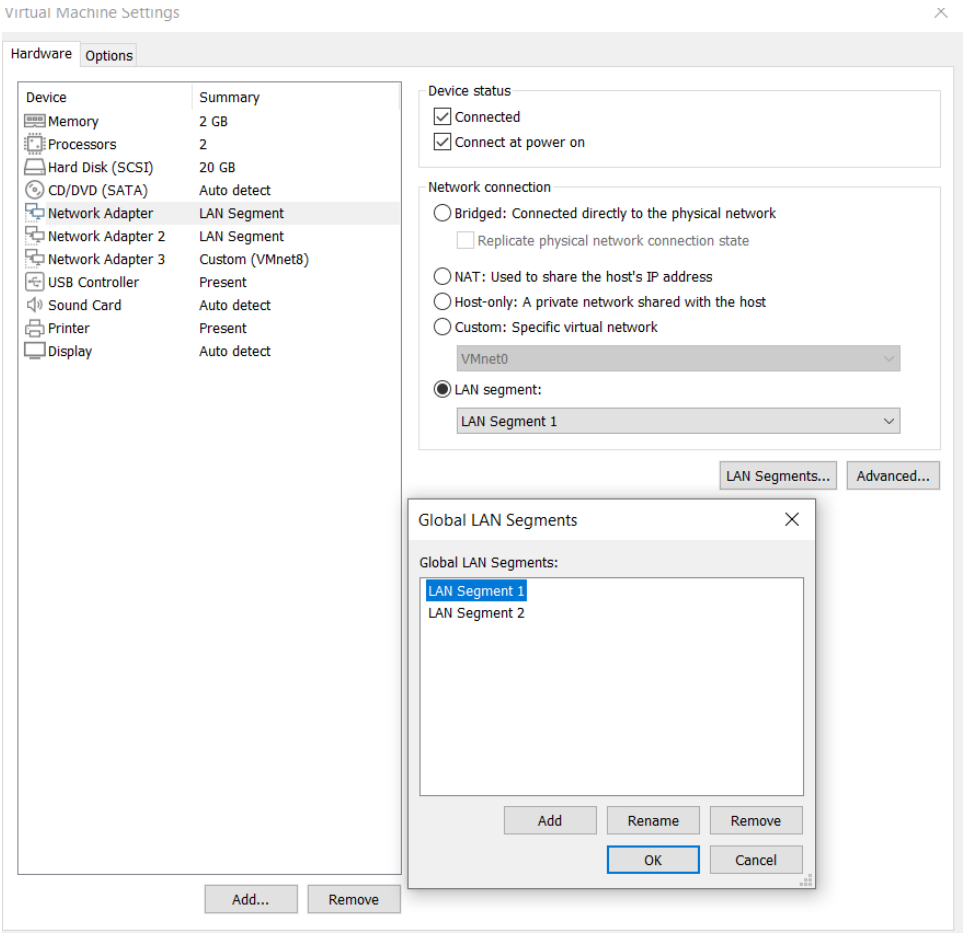
Now we need to set up the switch alias the bridge. To do this you use the command “sudo nano /etc/network/interfaces” in the terminal and enter this ens’s



And now configure the Spanning Tree Protocoll to be off so we don't have STP traffic blurring captures, with the command "sudo brctl stp br0 off".

Setting up the two PC's

Go to the Switch settings and click on Network Adapters. For the first and second one we need to make a "Lan Segment" to connect PC1 and PC2

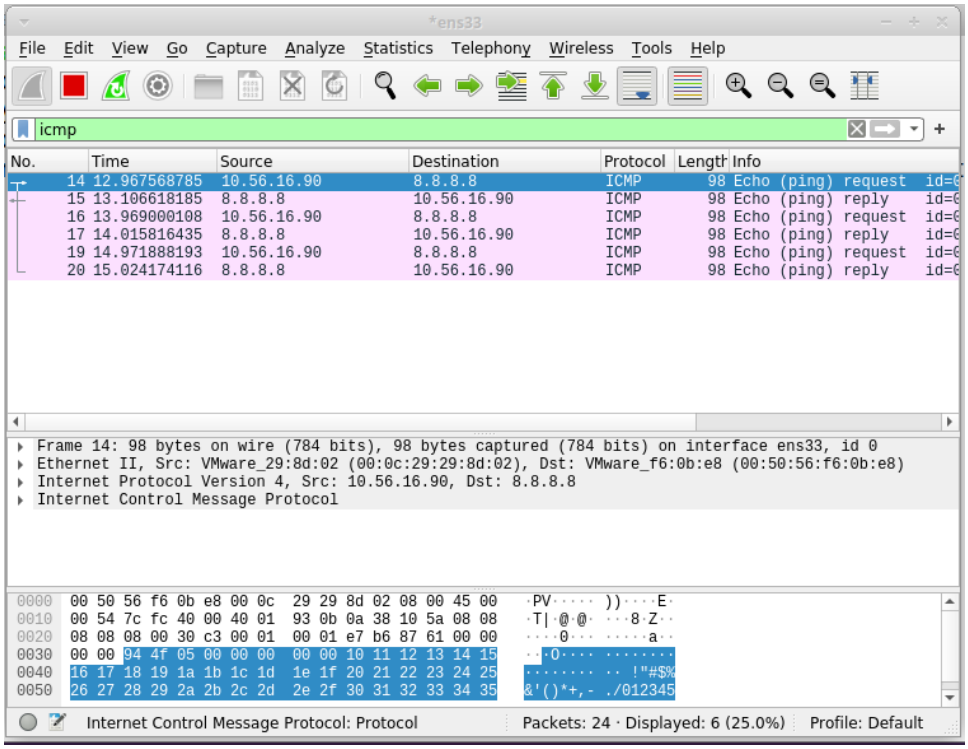


We need two Lan Segments because we connect to PC's. The third network adapter needs to be on "custom, VMnet8" so we also have connection to the internet.

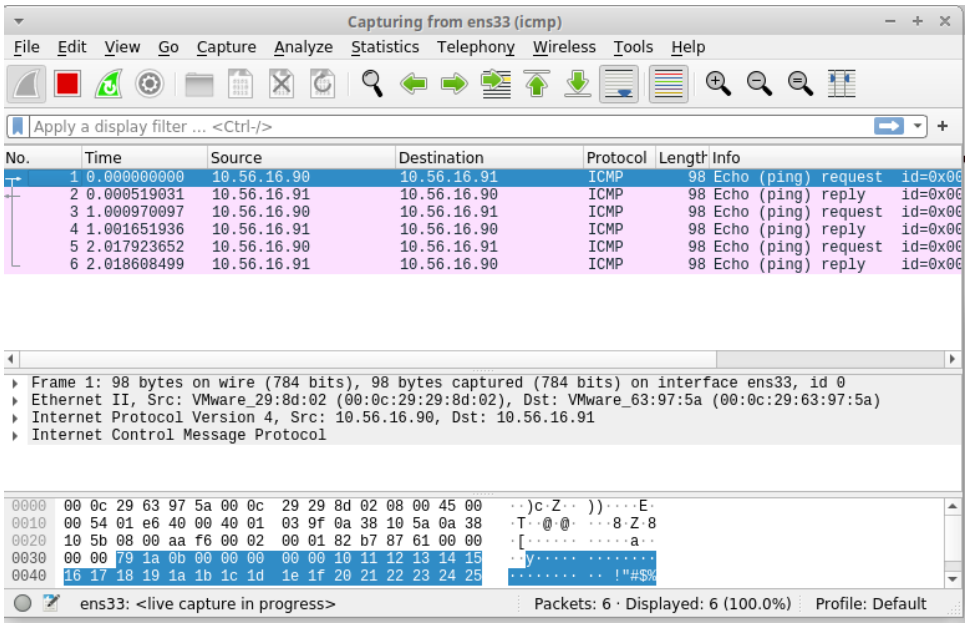
Now do the same for PC1 and PC2. Select for one of them Lan Segment1 and for the other Lan segment2.

Wireshark on PC1 and PC2

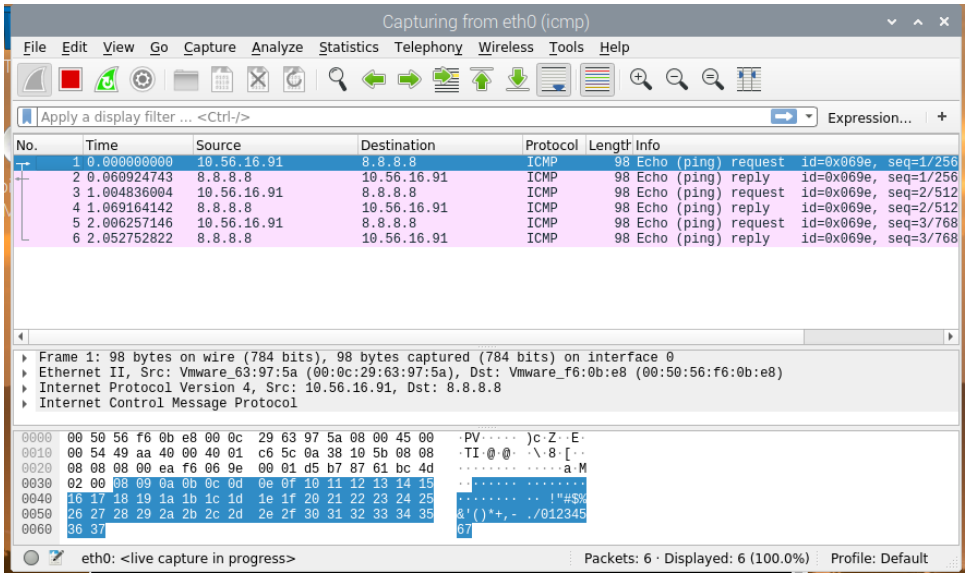
On PC1 open wireshark with "sudo wireshark" and put in the "icmp" filter and try pinging 8.8.8.8 to see if we have acces to the internet

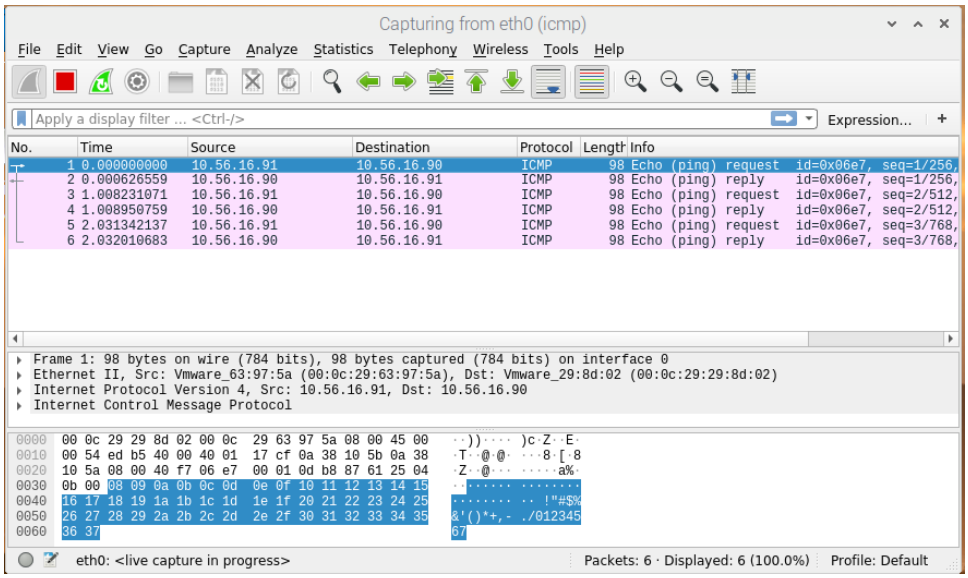


Now let's try to ping from PC1 to PC2 to see if they have connection between each other.



Looks like it is working. Now try for PC2 to 8.8.8.8 and to PC1.





Working as well for PC2.

Mac adresses

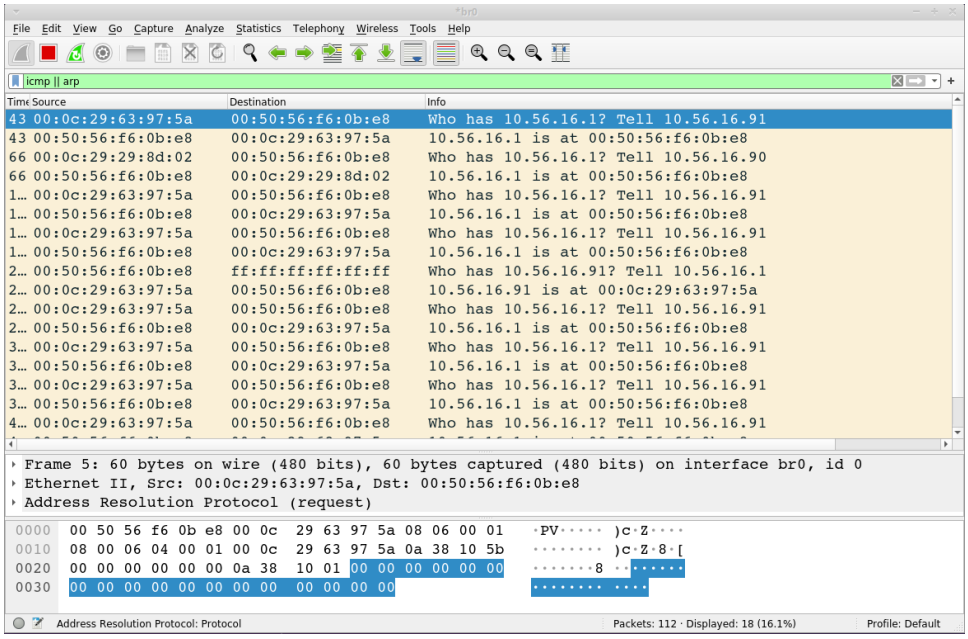
We will check, with the command "sudo brctl showmacs br0" on the Switch, the mac adresses.

```
Terminal - bogdan7978@ubuntu: ~
File  Edit  View  Terminal  Tabs  Help
bogdan7978@ubuntu:~$ sudo brctl showmacs br0
[sudo] password for bogdan7978:
port no mac addr                is local?       ageing timer
  2     00:0c:29:29:8d:02       no                61.18
  1     00:0c:29:63:97:5a       no                 0.19
  1     00:0c:29:93:4b:c3       yes                0.00
  1     00:0c:29:93:4b:c3       yes                0.00
  2     00:0c:29:93:4b:cd       yes                0.00
  2     00:0c:29:93:4b:cd       yes                0.00
  3     00:0c:29:93:4b:d7       yes                0.00
  3     00:0c:29:93:4b:d7       yes                0.00
  3     00:50:56:f6:0b:e8       no                 0.10
bogdan7978@ubuntu:~$
```

These are the MAC adresses learned by our switch so far and the ports assigned to them, the first and last are from our PC1 and PC2. The dynamically learned MAC addresses are deleted after the MAC address age value has expired. This frees unused addresses from the MAC address table for other active subscribers.

Wireshark on the Switch

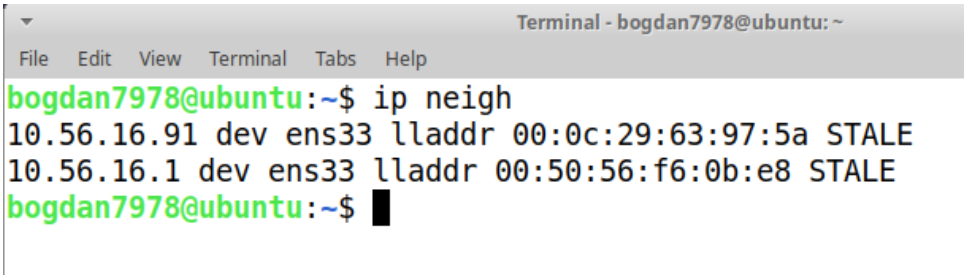
We will run wireshark on the switch to see all the forwarding broadcasted frames.



And this is the result.

ARP table for PC1 and PC2

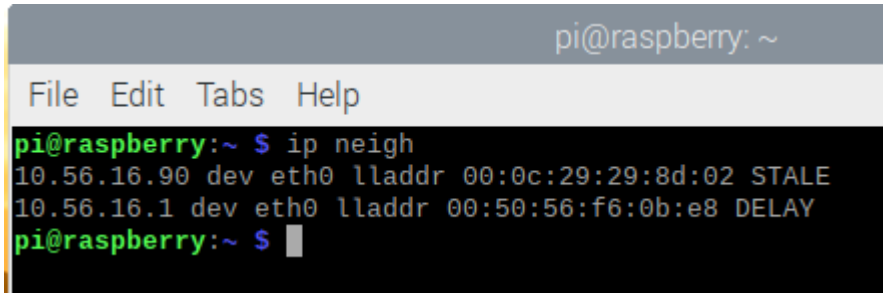
We will run the command “ip neigh” on PC1 and PC2.



For PC1, so the ARP table should look like this

IP ADRESS	MAC ADRESS
10.56.16.91	00:0c:29:63:97:5a
10.56.16.1	00:50:56:f6:0b:e8

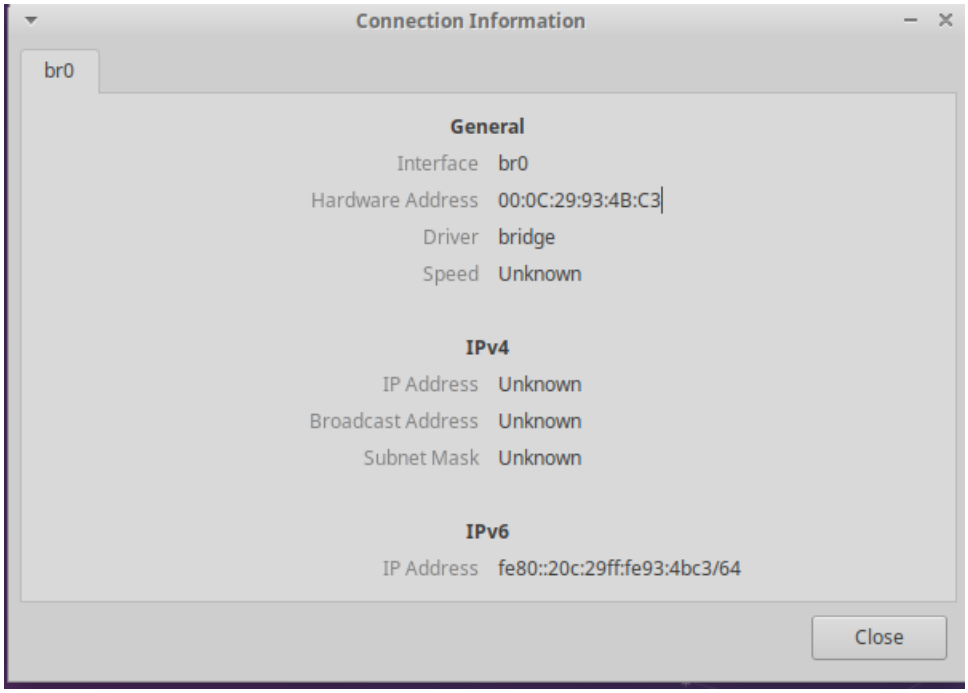
And for PC2



And the ARP table

IP ADRESS	MAC ADRESS
10.56.16.90	00:0c:29:29:8d:02
10.56.16.1	00:50:56:f6:0b:e8

If we go to “connection information” for the Switch we can see the MAC adress

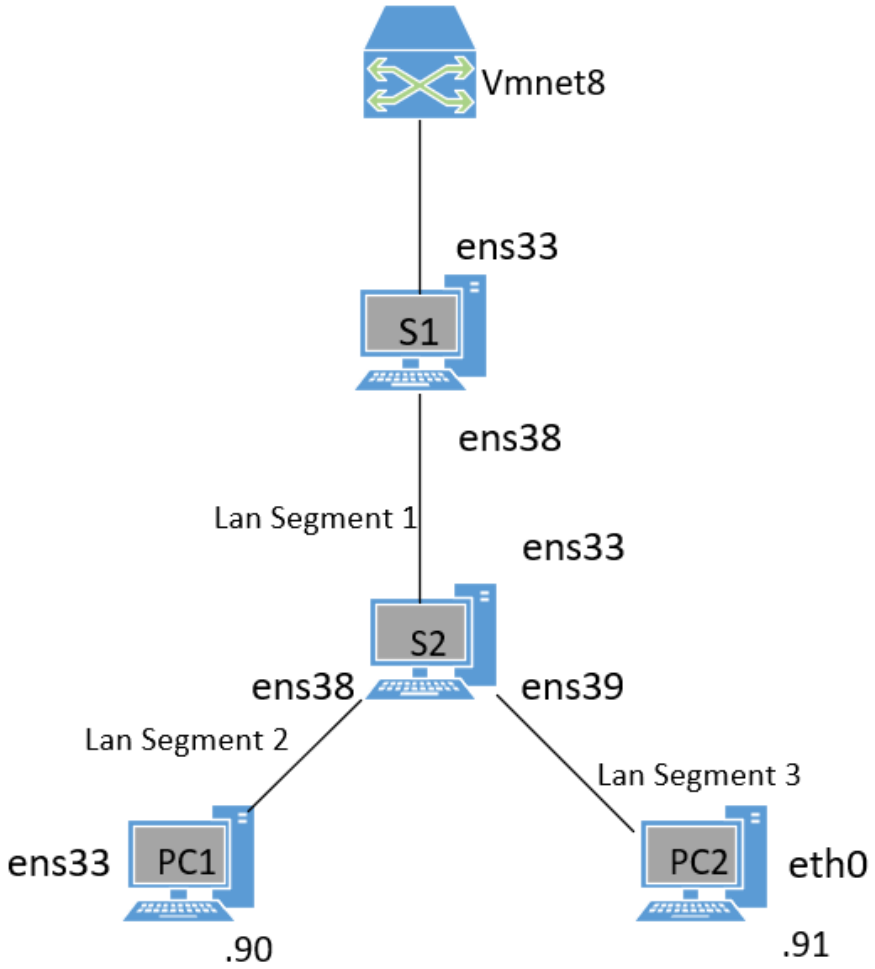


Looks like the switch's MAC adress can be regarded as invisible.

Adding a second SWITCH

We will now add a second switch into the whole mix.

It all should look like this



So i made a clone from the first Xubuntu acting as a switch and interconnected all the PC's

Now let's try to ping the internet from PC1 and PC2, also each other, to see if it works

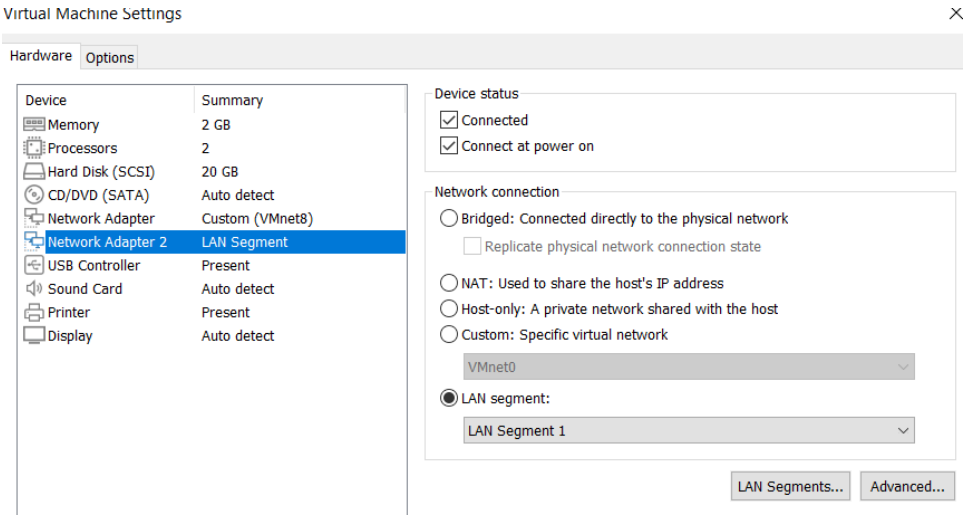
```
Terminal - bogdan7978@ubuntu: ~
File  Edit  View  Terminal  Tabs  Help
bogdan7978@ubuntu:~$ ping 8.8.8.8 -c 3
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=2421 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=1413 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=394 ms

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2039ms
rtt min/avg/max/mdev = 393.891/1409.046/2420.656/827.427 ms, pipe 3
bogdan7978@ubuntu:~$ ip neigh
10.56.16.1 dev ens33 lladdr 00:50:56:f6:0b:e8 REACHABLE
bogdan7978@ubuntu:~$ ping 10.56.16.91 -c 3
PING 10.56.16.91 (10.56.16.91) 56(84) bytes of data.
64 bytes from 10.56.16.91: icmp_seq=1 ttl=64 time=0.743 ms
64 bytes from 10.56.16.91: icmp_seq=2 ttl=64 time=0.757 ms
64 bytes from 10.56.16.91: icmp_seq=3 ttl=64 time=0.764 ms

--- 10.56.16.91 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2037ms
rtt min/avg/max/mdev = 0.743/0.754/0.764/0.008 ms
bogdan7978@ubuntu:~$
```

From PC1 we can see that it works

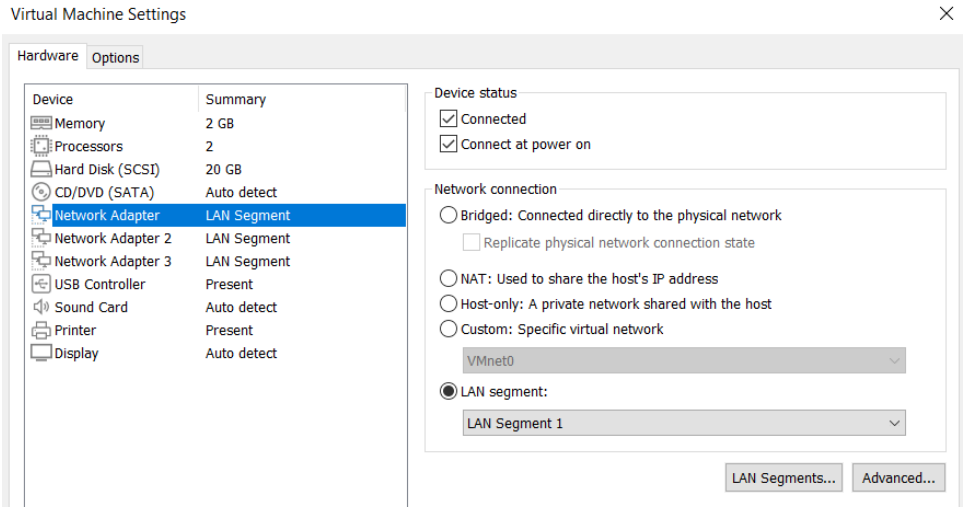
```
pi@raspberry: ~
File  Edit  Tabs  Help
pi@raspberry:~ $ ping 8.8.8.8 -c 3
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=49.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=56.6 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=128 time=43.1 ms

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 9ms
rtt min/avg/max/mdev = 43.080/49.876/56.609/5.526 ms
pi@raspberry:~ $ ip neigh
10.56.16.1 dev eth0 lladdr 00:50:56:f6:0b:e8 REACHABLE
pi@raspberry:~ $ ping 10.56.16.90 -c 3
PING 10.56.16.90 (10.56.16.90) 56(84) bytes of data.
64 bytes from 10.56.16.90: icmp_seq=1 ttl=64 time=0.805 ms
64 bytes from 10.56.16.90: icmp_seq=2 ttl=64 time=0.812 ms
64 bytes from 10.56.16.90: icmp_seq=3 ttl=64 time=0.667 ms

--- 10.56.16.90 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 44ms
rtt min/avg/max/mdev = 0.667/0.761/0.812/0.070 ms
pi@raspberry:~ $ ip neigh
10.56.16.1 dev eth0 lladdr 00:50:56:f6:0b:e8 REACHABLE
10.56.16.90 dev eth0 lladdr 00:0c:29:29:8d:02 REACHABLE
pi@raspberry:~ $
```

Also PC2

Also this is the setup for S1



Also for S2



And PC1 and PC2 are connected to S2 through Lan Segment 2 and 3.

So this was Assignment 4 with the Switch, also the OLA for Networking, now we know how to connect a switch in VMware and how these work.

Thank you!