



## Architecture & Protocols

### Lab 3

### Report Practical Work

3<sup>ème</sup> année - RTS

PETIT Alexandre  
Arthur LAUMY

## TP3 REPORT

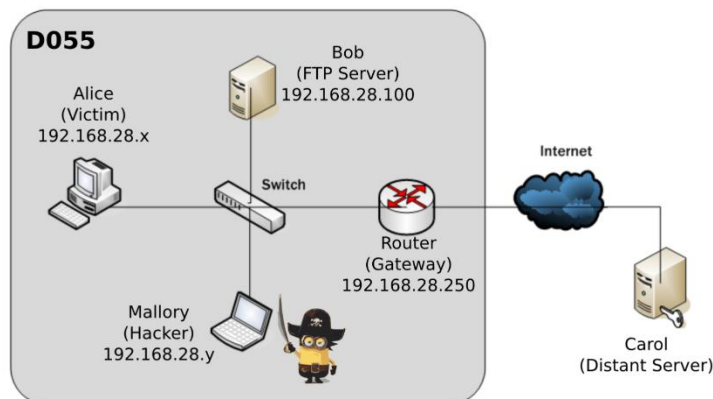


FIG. 1 : Operating diagram

## I – Setting up the Layout

```
tpreseau@d055-pc8:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s31f6: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 54:bf:64:64:a6:88 brd ff:ff:ff:ff:ff:ff
    inet 192.168.28.8/24 brd 192.168.28.255 scope global enp0s31f6
        valid_lft forever preferred_lft forever
    inet6 fe80::56bf:64ff:fe64:a688/64 scope link
        valid_lft forever preferred_lft forever
3: enp1s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether b4:96:91:2b:69:25 brd ff:ff:ff:ff:ff:ff
4: wlp2s0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether f8:34:41:80:3d:ce brd ff:ff:ff:ff:ff:ff
5: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:3c:05:db brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
        valid_lft forever preferred_lft forever
6: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel master virbr0 state DOWN group default qlen 1000
    link/ether 52:54:00:3c:05:db brd ff:ff:ff:ff:ff:ff
7: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:75:2f:b7:8f brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
```

We use the `ip addr` command and find an IP of 192.168.28.8 with a mask of 24. We used the following addresses for our attack:

- Hackeur : 192.168.28.8
- Target 1: 192.168.28.7
- Target 2: 192.168.28.100

## II – NMAP

```

tpreseau@d055-pc8:~$ nmap 192.168.28.7

Starting Nmap 7.60 ( https://nmap.org ) at 2024-10-09 13:44 CEST
Nmap scan report for 192.168.28.7
Host is up (0.00015s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds

```

The Nmap command is used to map the open and available ports of an IP address. In our case we are looking at two TCP ports: SSH and HTTP, two potential vulnerabilities.

```

tpreseau@d055-pc8:~$ nmap -sP 192.168.28.7

Starting Nmap 7.60 ( https://nmap.org ) at 2024-10-09 13:45 CEST
Nmap scan report for 192.168.28.7
Host is up (0.00049s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.00 seconds
tpreseau@d055-pc8:~$ nmap -sP 192.168.28.11

Starting Nmap 7.60 ( https://nmap.org ) at 2024-10-09 13:46 CEST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.01 seconds

```

The `-sP` option is used to check that an IP address is working (if the address exists or if the computer is switched on).

### III – ARP Poisoning

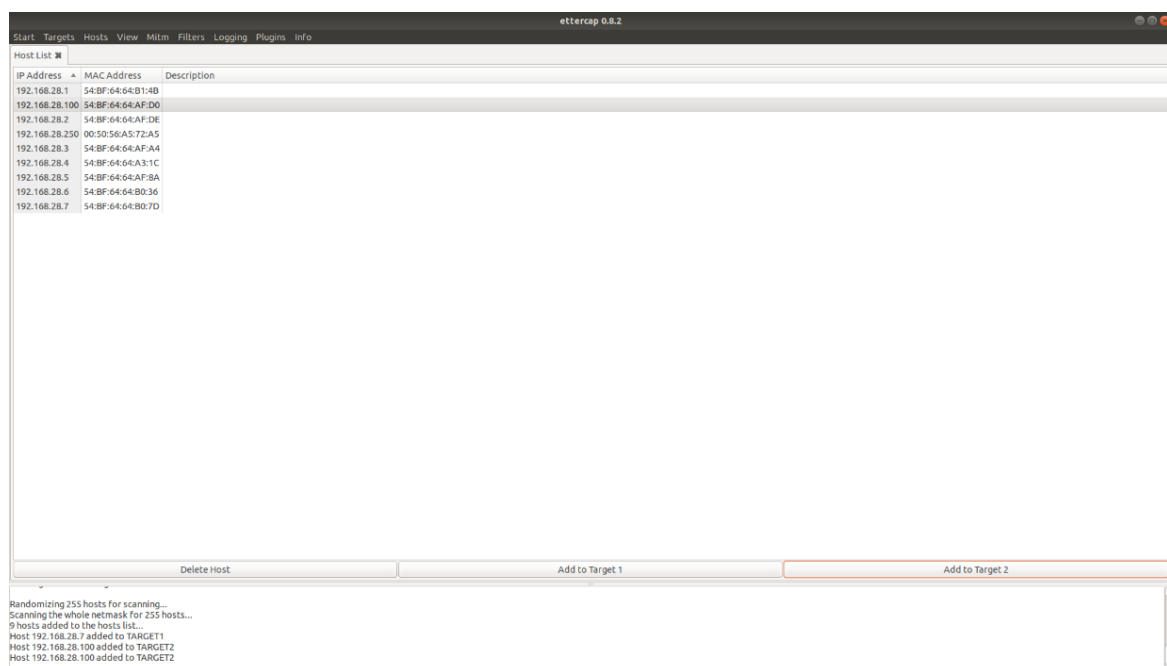


Figure 1 – Ettercap Window

```

tpreseau@d055-pc100: ~
File Edit View Search Terminal Help
https://ubuntu.com/18-04
New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
Last login: Wed Oct  9 13:57:41 2024 from 192.168.28.7
tpreseau@d055-pc100:~$ arp

```

Adresse	TypeMap	AdresseMat	Indicateurs	Iface
_gateway	ether	00:50:56:a5:72:a5	C	enp0s
192.168.28.5	ether	54:bf:64:64:af:8a	C	enp0s
192.168.28.4	ether	54:bf:64:64:a3:1c	C	enp0s
192.168.28.6	ether	54:bf:64:64:b0:36	C	enp0s
192.168.28.7	ether	54:bf:64:64:b0:7d	C	enp0s
192.168.28.1	ether	54:bf:64:64:b1:4b	C	enp0s
192.168.28.3	ether	54:bf:64:64:af:a4	C	enp0s
192.168.28.2	ether	54:bf:64:64:af:de	C	enp0s
192.168.28.8	ether	54:bf:64:64:a6:88	C	enp0s

```

tpreseau@d055-pc100:~$ arp
tpreseau@d055-pc100:~$ arp

```

Adresse	TypeMap	AdresseMat	Indicateurs	Iface
_gateway	ether	00:50:56:a5:72:a5	C	enp0s
192.168.28.5	ether	54:bf:64:64:b0:36	C	enp0s
192.168.28.4	ether	54:bf:64:64:a3:1c	C	enp0s
192.168.28.6	ether	54:bf:64:64:b0:36	C	enp0s
192.168.28.7	ether	54:bf:64:64:a6:88	C	enp0s
192.168.28.1	ether	54:bf:64:64:af:de	C	enp0s
192.168.28.3	ether	54:bf:64:64:a3:1c	C	enp0s
192.168.28.2	ether	54:bf:64:64:af:de	C	enp0s
192.168.28.8	ether	54:bf:64:64:a6:88	C	enp0s

Looking at the ARP table from 192.168.28.100 (retrieved via SSH connection), we can see that the MAC address of our computer (.8) and that of our destination (.7) are the same during ARP poisoning. The station with the IP address 192.168.28.100 therefore thinks it is addressing .7 when it talks to us.

```

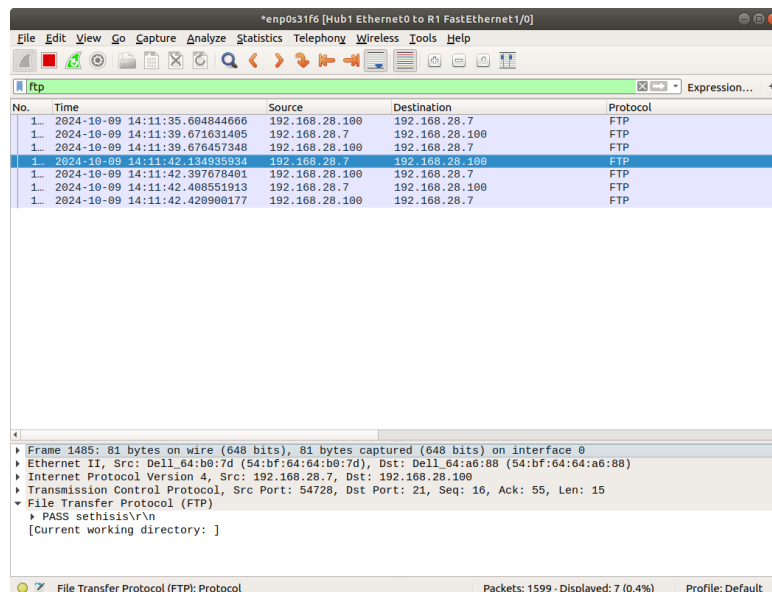
tpreseau@d055-PC7:~$ arp

```

Address	Hwtype	HwAddress	Flags Mask	Iface
169.254.121.12		(Incomplete)		enp0s31f6
192.168.28.3	ether	54:bf:64:64:af:a4	C	enp0s31f6
_gateway	ether	00:50:56:a5:72:a5	C	enp0s31f6
169.254.29.127		(Incomplete)		enp0s31f6
169.254.169.11		(Incomplete)		enp0s31f6
192.168.28.4	ether	54:bf:64:64:a3:1c	C	enp0s31f6
10.10.26.252	ether	58:cd:c9:4e:4c:ed	C	wlp2s0
192.168.28.100	ether	54:bf:64:64:a6:88	C	enp0s31f6
169.254.57.109		(Incomplete)		enp0s31f6
169.254.140.153		(Incomplete)		enp0s31f6
192.168.28.5	ether	54:bf:64:64:af:8a	C	enp0s31f6
169.254.14.193		(Incomplete)		enp0s31f6
169.254.14.193		(Incomplete)		enp0s31f6
192.168.28.8	ether	54:bf:64:64:a6:88	C	enp0s31f6
169.254.225.180		(Incomplete)		enp0s31f6
10.10.27.251	ether	00:50:56:b4:3a:f6	C	wlp2s0
192.168.28.1	ether	54:bf:64:64:b1:4b	C	enp0s31f6
169.254.218.241		(Incomplete)		enp0s31f6
169.254.251.115		(Incomplete)		enp0s31f6
169.254.180.8		(Incomplete)		enp0s31f6
169.254.215.237		(Incomplete)		enp0s31f6
169.254.115.63		(Incomplete)		enp0s31f6
10.10.26.117	ether	d0:39:57:68:c4:e5	C	wlp2s0
169.254.76.144		(Incomplete)		enp0s31f6
169.254.82.182		(Incomplete)		enp0s31f6
169.254.130.233		(Incomplete)		enp0s31f6
169.254.179.142		(Incomplete)		enp0s31f6
192.168.28.6	ether	54:bf:64:64:b0:36	C	enp0s31f6
169.254.179.79		(Incomplete)		enp0s31f6
10.10.27.187	ether	a4:cf:99:4f:03:5f	C	wlp2s0
10.10.26.221	ether	14:7d:da:08:e1:6a	C	wlp2s0
_gateway	ether	70:4c:a5:7f:3a:3e	C	wlp2s0
169.254.156.85		(Incomplete)		enp0s31f6
192.168.28.2	ether	54:bf:64:64:af:de	C	enp0s31f6
169.254.180.174		(Incomplete)		enp0s31f6
169.254.182.151		(Incomplete)		enp0s31f6
169.254.212.131		(Incomplete)		enp0s31f6
169.254.50.113		(Incomplete)		enp0s31f6
169.254.0.124		(Incomplete)		enp0s31f6
169.254.255.255		(Incomplete)		enp0s31f6
169.254.120.93		(Incomplete)		enp0s31f6
169.254.169.25		(Incomplete)		enp0s31f6

Figure 2 - ARP Table of the First target

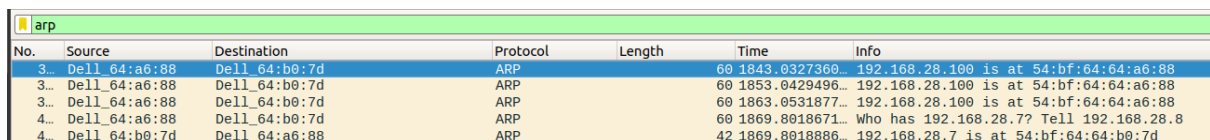
On the 192.168.28.7 side, we can see the same phenomenon in the ARP table between our machine (192.168.28.8) and the other target (192.168.28.100). This allows us to verify the man in the middle attack.



The image shows a Wireshark capture of FTP traffic on the interface 'enp0s31f6'. The packet list shows several FTP packets. The packet details pane for packet 1 (Frame 1485) shows the Ethernet II, Internet Protocol Version 4, and File Transfer Protocol (FTP) layers. The FTP layer shows the command 'PASS sethisiss\r\n' and the current working directory as '/'. The packet status bar at the bottom indicates 'File Transfer Protocol (FTP): Protocol' and 'Packets: 1599 - Displayed: 7 (0.4%)'.

No.	Time	Source	Destination	Protocol
1.	2024-10-09 14:11:35.604844666	192.168.28.100	192.168.28.7	FTP
1.	2024-10-09 14:11:39.671631405	192.168.28.7	192.168.28.100	FTP
1.	2024-10-09 14:11:39.676457348	192.168.28.100	192.168.28.7	FTP
1.	2024-10-09 14:11:42.134935934	192.168.28.7	192.168.28.100	FTP
1.	2024-10-09 14:11:42.397679401	192.168.28.100	192.168.28.7	FTP
1.	2024-10-09 14:11:42.406551913	192.168.28.7	192.168.28.100	FTP
1.	2024-10-09 14:11:42.428980177	192.168.28.100	192.168.28.7	FTP

Figure 3 - Wireshark Hacker



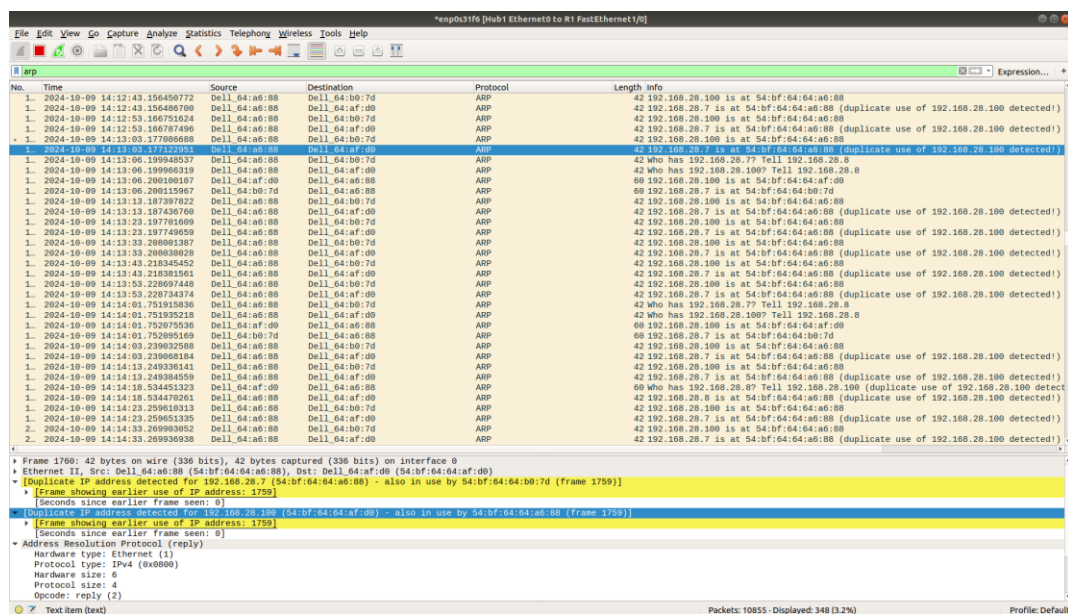
The image shows a Wireshark capture of ARP traffic. The packet list shows several ARP packets. The packet details pane for packet 1 (Frame 1485) shows the Ethernet II, Internet Protocol Version 4, and File Transfer Protocol (FTP) layers. The FTP layer shows the command 'PASS sethisiss\r\n' and the current working directory as '/'. The packet status bar at the bottom indicates 'File Transfer Protocol (FTP): Protocol' and 'Packets: 1599 - Displayed: 7 (0.4%)'.

No.	Source	Destination	Protocol	Length	Time	Info
3...	Dell 64:a6:88	Dell 64:b0:7d	ARP	60	1843.0327360...	192.168.28.100 is at 54:bf:64:64:a6:88
3...	Dell 64:a6:88	Dell 64:b0:7d	ARP	60	1853.0429496...	192.168.28.100 is at 54:bf:64:64:a6:88
3...	Dell 64:a6:88	Dell 64:b0:7d	ARP	60	1863.0531877...	192.168.28.100 is at 54:bf:64:64:a6:88
4...	Dell 64:a6:88	Dell 64:b0:7d	ARP	60	1869.8018671...	Who has 192.168.28.7? Tell 192.168.28.8
4...	Dell 64:b0:7d	Dell 64:a6:88	ARP	42	1869.8018886...	192.168.28.7 is at 54:bf:64:64:b0:7d

Figure 4 - ARP Traffic

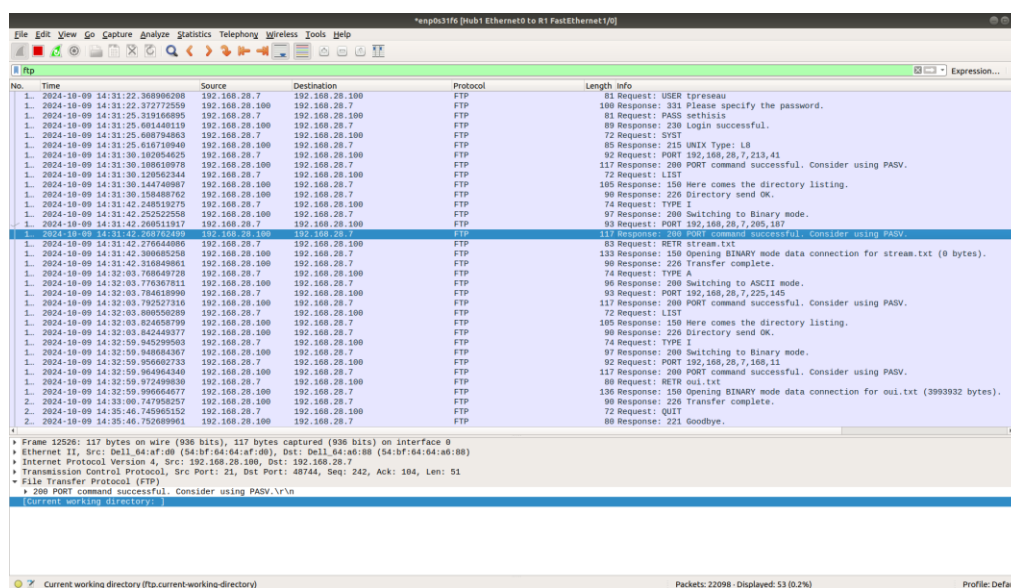
We can see that with ARP poisoning we have managed to recover the FTP network frames during the request and therefore the password. For the ARP traffic, we can see that the MAC addresses have changed as shown above.

Also, the hacker's Wireshark shows that the duplication of the 192.168.28.100 address has been detected.



The image shows a Wireshark capture of ARP traffic. The packet list shows several ARP packets. The packet details pane for packet 1 (Frame 1485) shows the Ethernet II, Internet Protocol Version 4, and File Transfer Protocol (FTP) layers. The FTP layer shows the command 'PASS sethisiss\r\n' and the current working directory as '/'. The packet status bar at the bottom indicates 'File Transfer Protocol (FTP): Protocol' and 'Packets: 10855 - Displayed: 348 (3.2%)'.

No.	Time	Source	Destination	Protocol	Length	Info
1.	2024-10-09 14:12:43.156450772	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:12:43.156486700	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:12:53.166751624	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:12:53.166787496	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:03.177886688	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:03.177886688	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:06.199485537	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	Who has 192.168.28.7? Tell 192.168.28.8
1.	2024-10-09 14:13:06.199485537	Dell 64:a6:88	Dell 64:af:00	ARP	42	Who has 192.168.28.100? Tell 192.168.28.8
1.	2024-10-09 14:13:06.200160107	Dell 64:af:00	Dell 64:a6:88	ARP	60	192.168.28.100 is at 54:bf:64:64:af:00
1.	2024-10-09 14:13:06.200159967	Dell 64:b0:7d	Dell 64:a6:88	ARP	60	192.168.28.7 is at 54:bf:64:64:b0:7d
1.	2024-10-09 14:13:13.187391822	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:13.187436760	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:23.197740959	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:23.197740959	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:33.208001387	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:33.208001387	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:33.208038028	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:43.218345452	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:43.218345452	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:43.218381561	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:43.218381561	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:53.226097448	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:13:53.226097448	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:13:53.229734374	Dell 64:a6:88	Dell 64:af:00	ARP	42	Who has 192.168.28.7? Tell 192.168.28.8
1.	2024-10-09 14:14:01.751919386	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	Who has 192.168.28.100? Tell 192.168.28.8
1.	2024-10-09 14:14:01.751919386	Dell 64:a6:88	Dell 64:af:00	ARP	60	192.168.28.100 is at 54:bf:64:64:af:00
1.	2024-10-09 14:14:01.751935218	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:14:01.751935218	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:14:01.752095169	Dell 64:b0:7d	Dell 64:a6:88	ARP	60	192.168.28.7 is at 54:bf:64:64:b0:7d
1.	2024-10-09 14:14:03.239032588	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:14:03.239032588	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:14:13.249336141	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:14:13.249336141	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:14:18.534453223	Dell 64:af:00	Dell 64:a6:88	ARP	60	Who has 192.168.28.7? Tell 192.168.28.100 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:14:18.534470261	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.8 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:14:23.259610313	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:14:23.259610313	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)
1.	2024-10-09 14:14:23.259651335	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88
1.	2024-10-09 14:14:23.259651335	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
2.	2024-10-09 14:14:33.269903052	Dell 64:a6:88	Dell 64:b0:7d	ARP	42	192.168.28.100 is at 54:bf:64:64:a6:88
2.	2024-10-09 14:14:33.269903052	Dell 64:a6:88	Dell 64:af:00	ARP	42	192.168.28.7 is at 54:bf:64:64:a6:88 (duplicate use of 192.168.28.100 detected!)



There are many other things to look out for, such as ls commands and file recovery.

```
-rw-r--r-- 1 1001 1001 1441 Oct 03 2023 tomat.gif
-rw-r--r-- 1 1001 1001 1317 Apr 20 2019 toto.asc
-rw-r--r-- 1 1001 1001 926 Apr 20 2019 topo.gpg
-rw-r--r-- 1 1001 1001 1399883 Mar 21 2024 tp_reseau_sujet.pdf
drwxr-xr-x 20 0 0 4096 Jan 13 2020 v8
226 Directory send OK.
ftp> get oui.txt
local: oui.txt remote: oui.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for oui.txt (3993932 bytes).
226 Transfer complete.
3993932 bytes received in 0.73 secs (5.2248 MB/s)
ftp> exit
221 Goodbye.
tpreseau@0055-PC7:~$
```

Figure 5 - FTP of Target 1

## IV – Use of an EtterCap Filter

```
tpreseau@0055-PC7:~$ ftp 192.168.28.100
Connected to 192.168.28.100.
220 (RTS's are the best)
Name (192.168.28.100:tpreseau): tpreseau
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Figure 6 - Effect of an Ettercap Filter

By using the filter specified in the statement, we can modify the user's request, as here where we have added a message.

**Remark:** Be careful when using compiling, it is very case and space sensitive!



## V – Spying with DriftNet

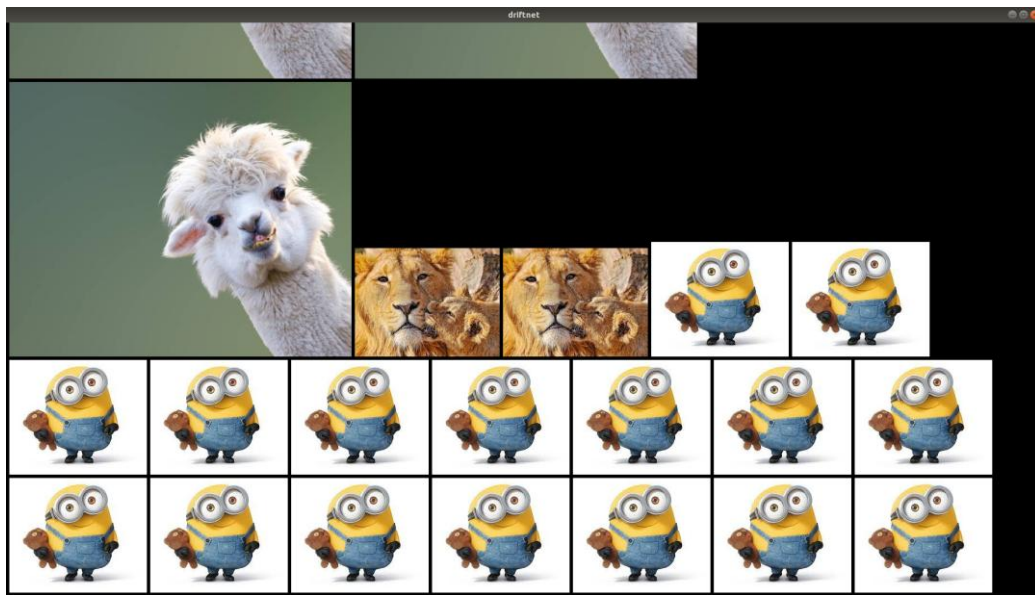


Figure 7 - DriftNet of several pictures get during an FTP Protocol

We can recover a picture using Driftnet while the picture is downloaded from an FTP Protocol.

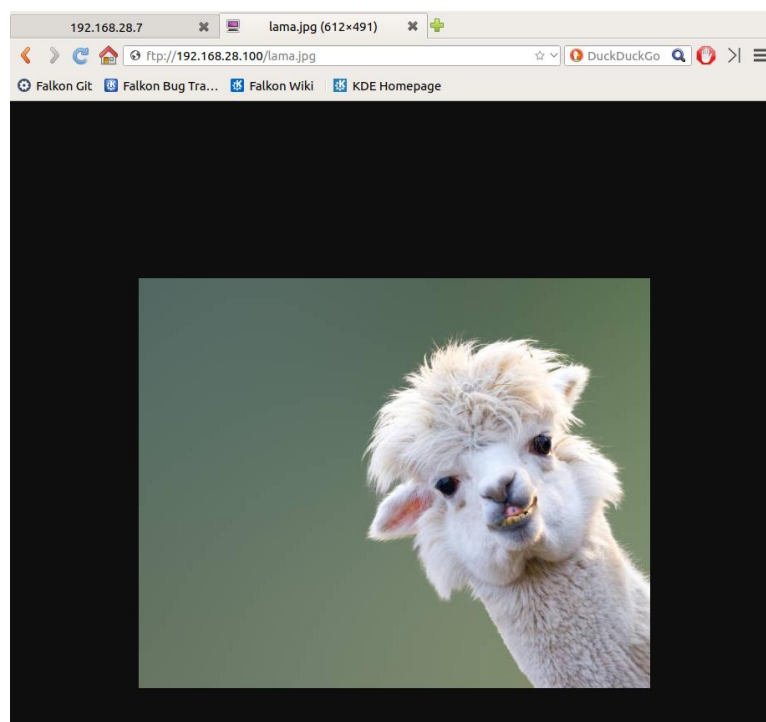


Figure 8 - Standard Web Access



Figure 9 - Driftnet during standard web access

Similarly, with a web access standard, it is possible to retrieve the image using Driftnet.

**Remark:** jpg appears once, jpeg appears twice on the hacker's screen (even if we open the picture on the internet). With Driftnet the hacker can see the file that the victim retrieved from her ftp connection with Bob.

## VI – DNS Poisoning

```

tpreseau@d055-pcB: ~/Downloads
File Edit View Search Terminal Tabs Help
tpreseau@d055-pcB: ~
#####
#####
# microsoft sucks ;)
# redirect it to www.linux.org
#
microsoft.com A 198.182.196.56
*.microsoft.com A 198.182.196.56
www.microsoft.com PTR 198.182.196.56 # Wildcards in PTR are not allowed
www.perdu.com A 193.51.47.207
*.perdu.com A 193.51.47.207
www.perdu.com PTR 193.51.47.207
cyu.fr A 10.10.17.5
*.cyu.fr A 10.10.17.5
www.cyu.fr PTR 10.10.17.5
frontal-web.cyu.fr PTR 10.10.17.5
#####
# no one out there can have our domains...
#
www.alor.org A 127.0.0.1
www.naga.org A 127.0.0.1
www.naga.org AAAA 2001:db8::2
#####
# dual stack enabled hosts does not make life easy
# force them back to single stack
www.ietf.org A 127.0.0.1
www.ietf.org AAAA ::
www.example.org A 0.0.0.0
/etc/ettercap/etter.dns" 118L, 5496C
61,30 57%

```

Figure 10 - Edit of /etc/ettercap/etter\_dns

We modify the /etc/ettercap/etter\_dns file to redirect the address to www.ensea.fr (10.10.17.5).



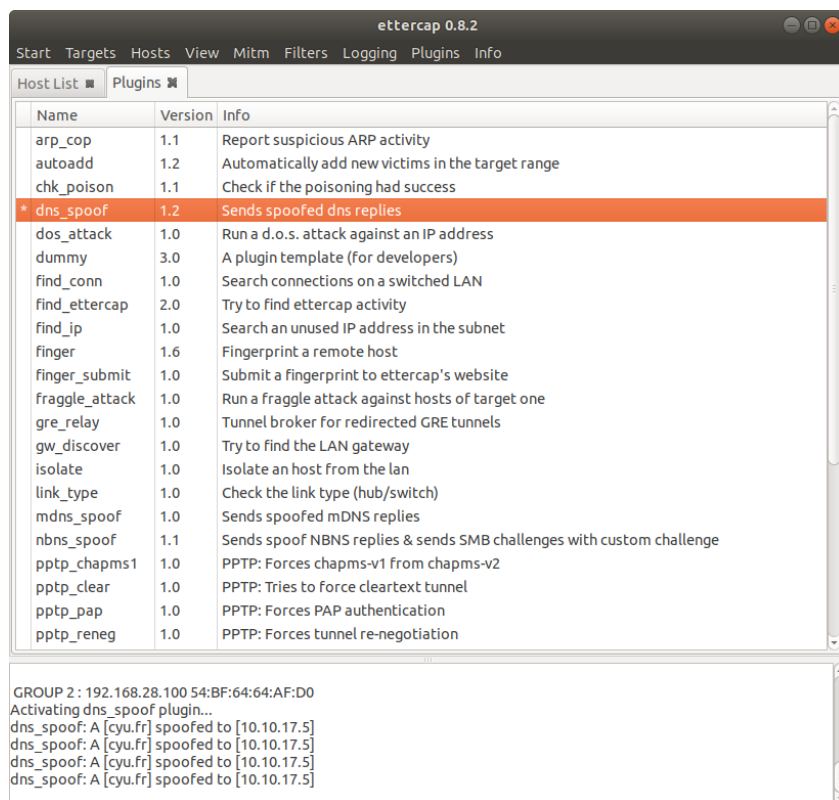


Figure 11 - DNS Spoofing

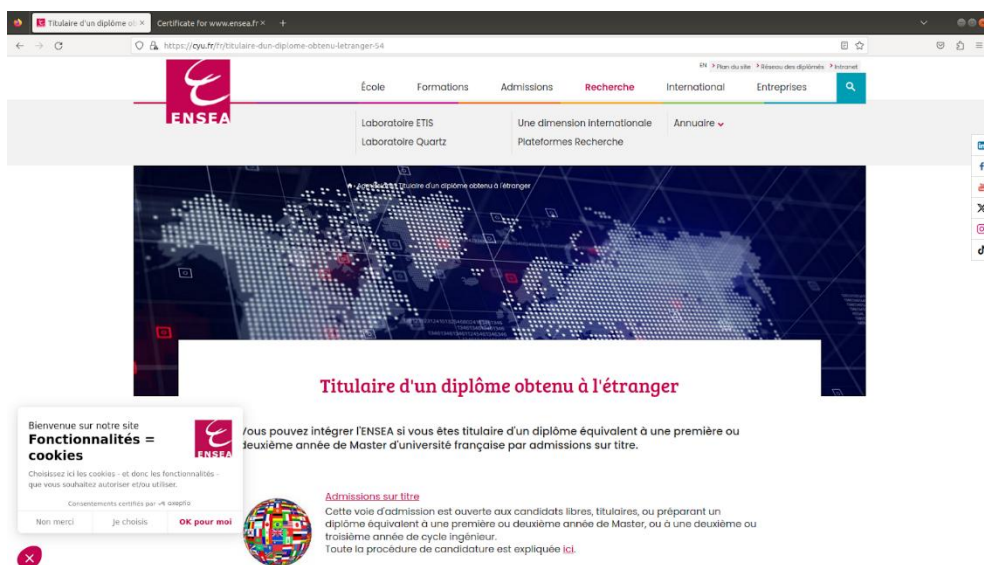


Figure 11 -12 Result of the web page

For the cyu.fr address, we therefore get the home page of www.ensea.fr with its associated pages, which clearly shows another vulnerability.