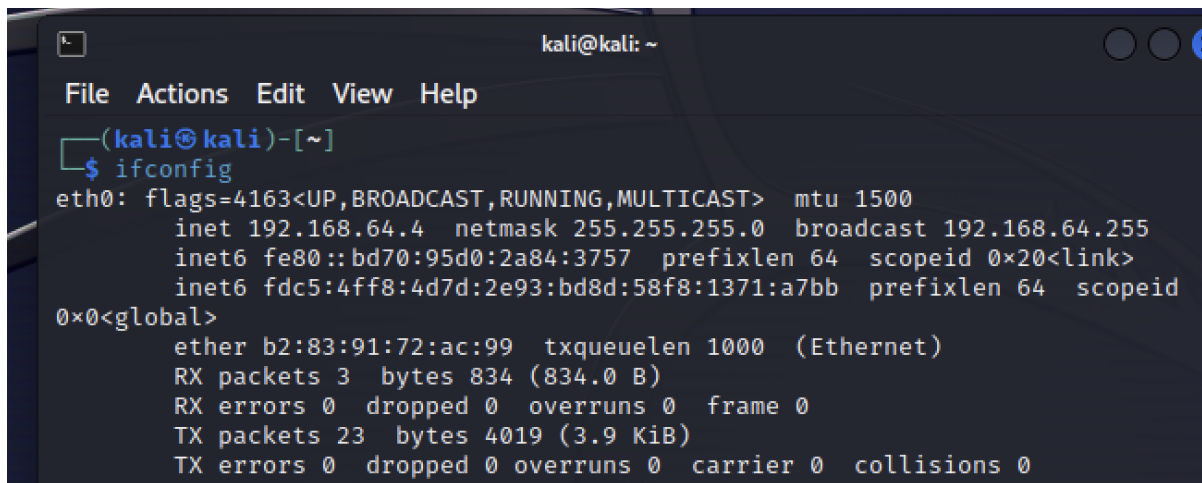


## Person-In-The-Middle VIA Arp Spoofing

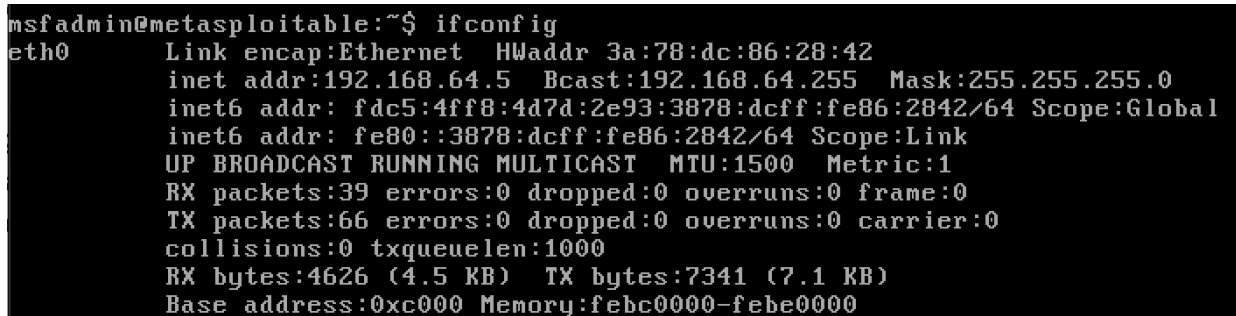
### Execution:

- a. Kali's main interface's MAC address is: (b2:83:91:72:ac:99).



```
kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.64.4 netmask 255.255.255.0 broadcast 192.168.64.255
    inet6 fe80::bd70:95d0:2a84:3757 prefixlen 64 scopeid 0<link>
    inet6 fdc5:4ff8:4d7d:2e93:bd8d:58f8:1371:a7bb prefixlen 64 scopeid
    0<global>
    ether b2:83:91:72:ac:99 txqueuelen 1000 (Ethernet)
    RX packets 3 bytes 834 (834.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 23 bytes 4019 (3.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- b. Kali's main interface's IP address is 192.168.64.4
- c. Metasploitable's main interface's MAC address is: 3a:78:dc:86:28:42



```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 3a:78:dc:86:28:42
          inet addr:192.168.64.5 Bcast:192.168.64.255 Mask:255.255.255.0
          inet6 addr: fdc5:4ff8:4d7d:2e93:3878:dcff:fe86:2842/64 Scope:Global
          inet6 addr: fe80::3878:dcff:fe86:2842/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:39 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4626 (4.5 KB) TX bytes:7341 (7.1 KB)
          Base address:0xc000 Memory:febc0000-febe0000
```

- d. Metasploitable's main interface's IP address is: 3a: 192.168.64.5

- e. Kali's routing table:

```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ netstat -rn  
Kernel IP routing table  
Destination      Gateway          Genmask          Flags   MSS Window  irtt Iface  
0.0.0.0          192.168.64.1    0.0.0.0          UG      0 0        0 eth0  
192.168.64.0    0.0.0.0         255.255.255.0    U        0 0        0 eth0
```

- f. Kali's ARP cache:

```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ arp  
Address            HWtype  HWaddress          Flags Mask          Iface  
192.168.64.1       ether   9e:3e:53:29:03:64  C                  eth0
```

- g. Metasploitable's routing table:

```
Kernel IP routing table  
Destination      Gateway          Genmask          Flags   MSS Window  irtt Iface  
192.168.64.0    *                255.255.255.0    U        0 0        0 eth0  
default         192.168.64.1    0.0.0.0          UG      0 0        0 eth0
```

- h. Metasploitable's ARP cache:

```
msfadmin@metasploitable:~$ arp  
Address            HWtype  HWaddress          Flags Mask          Iface  
192.168.64.1       ether   9E:3E:53:29:03:64  C                  eth0
```

- i. Metasploitable does not have immediate access to the IP address for cs338.jeffondich.com so it must use the MAC address of the default destination IP address (9E:3E:53:29:03:64).

- j. I saw an HTTP response on Metasploitable, but Kali did not capture any packets.

```
msfadmin@metasploitable:~$ curl http://cs338.jeffondich.com/
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>CS338 Sandbox</title>
  </head>

  <body>
    <h1>CS338 Sandbox</h1>
    <h2>Fun with security, or maybe insecurity</h2>

    <p>This page should be the page you retrieve for the "Getting started with Wireshark" assignment. Here's my head, as advertised:
    <div></div>
  </p>
  </body>
</html>
```

- k. Began ARP poisoning Metasploitable.

- l. There is now a second iteration of the row previously shown on the arp table. The MAC address is now different, corresponding to Kali's main interface's MAC address.

```
msfadmin@metasploitable:~$ arp
Address          HWtype  HWaddress      Flags Mask    Iface
192.168.64.1     ether   B2:83:91:72:AC:99  C             eth0
192.168.64.1     ether   B2:83:91:72:AC:99  C             eth0
```

- m. I predict that Metasploitable will send its TCP Syn packet to Kali's MAC address, which is now the mac address corresponding to the outgoing destination IP address on the ARP table..
- n. Started capturing.

- o. Metasploitable receives an HTTP response:

```
msfadmin@metasploitable:~$ curl http://cs338.jeffondich.com/
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>CS338 Sandbox</title>
  </head>

  <body>
    <h1>CS338 Sandbox</h1>
    <h2>Fun with security, or maybe insecurity</h2>

    <p>This page should be the page you retrieve for the "Getting started wi
th Wireshark"
    assignment. Here's my head, as advertised:
    <div></div>
  </p>
  </body>
</html>
```

And Wireshark captures the following information:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.64.5	45.79.89.123	TCP	74	35908 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM TSva
2	0.005343854	192.168.64.5	45.79.89.123	TCP	74	[TCP Retransmission] 35908 → 80 [SYN] Seq=0 Win=5840 Len=0 MS
3	0.059078697	45.79.89.123	192.168.64.5	TCP	66	80 → 35908 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1382 SA
4	0.065379062	45.79.89.123	192.168.64.5	TCP	66	[TCP Retransmission] 80 → 35908 [SYN, ACK] Seq=0 Ack=1 Win=64
5	0.065836068	192.168.64.5	45.79.89.123	TCP	54	35908 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0
6	0.067571047	192.168.64.5	45.79.89.123	HTTP	212	GET / HTTP/1.1
7	0.073343073	192.168.64.5	45.79.89.123	TCP	54	35908 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0
8	0.073400907	192.168.64.5	45.79.89.123	TCP	212	[TCP Retransmission] 35908 → 80 [PSH, ACK] Seq=1 Ack=1 Win=58
9	0.127547004	45.79.89.123	192.168.64.5	TCP	54	80 → 35908 [ACK] Seq=1 Ack=159 Win=64128 Len=0
10	0.128815227	45.79.89.123	192.168.64.5	HTTP	785	HTTP/1.1 200 OK (text/html)
11	0.129277358	45.79.89.123	192.168.64.5	TCP	54	80 → 35908 [ACK] Seq=1 Ack=159 Win=64128 Len=0
12	0.129302400	45.79.89.123	192.168.64.5	TCP	785	[TCP Retransmission] 80 → 35908 [PSH, ACK] Seq=1 Ack=159 Win=
13	0.130113993	192.168.64.5	45.79.89.123	TCP	54	35908 → 80 [ACK] Seq=159 Ack=732 Win=7360 Len=0
14	0.135789185	192.168.64.5	45.79.89.123	TCP	54	35908 → 80 [FIN, ACK] Seq=159 Ack=732 Win=7360 Len=0
15	0.137325953	192.168.64.5	45.79.89.123	TCP	54	[TCP Keep-Alive] 35908 → 80 [ACK] Seq=159 Ack=732 Win=7360 Le
16	0.137340745	192.168.64.5	45.79.89.123	TCP	54	[TCP Retransmission] 35908 → 80 [FIN, ACK] Seq=159 Ack=732 Wi
17	0.190736249	45.79.89.123	192.168.64.5	TCP	54	80 → 35908 [FIN, ACK] Seq=732 Ack=160 Win=64128 Len=0
18	0.193301530	45.79.89.123	192.168.64.5	TCP	54	[TCP Retransmission] 80 → 35908 [FIN, ACK] Seq=732 Ack=160 Wi
19	0.193914328	192.168.64.5	45.79.89.123	TCP	54	35908 → 80 [ACK] Seq=160 Ack=733 Win=7360 Len=0
20	0.201291790	192.168.64.5	45.79.89.123	TCP	54	[TCP Dup ACK 19#1] 35908 → 80 [ACK] Seq=160 Ack=733 Win=7360

Observing the captured packets, it seems like Metasploitable and cs338.jeffondich.com engaged in a TCP handshake, Metasploitable requested the page's html, cs338.jeffondich.com sent the html file, then they closed the connection.

- p. It seems that Kali repeatedly sends an ARP packet to Metasploitable telling it that the MAC address of the default IP address in its ARP table is Kali's MAC address, while also sending an ARP packet to the default IP that the MAC address for Metasploitable is Kali's MAC address. By continually sending these ARP packets, Metasploitable is convinced to change the MAC address of its default IP address and changes its ARP table.
- q. To design an ARP spoofing detector we could set up a system that determines if a request for an ARP packet was sent before accepting its information. This could fail when a request is sent out and an ARP spoof is received before an authentic packet. We could also keep track of which IP addresses match to which MAC addresses, this could

generate false positives when this information is changed for a new device under the same IP.

**Synthesis:**

- a. To intercept traffic between Alice and Bob. Mal utilizes ARP packets: a tool used for informing individuals the address they should send an item to make it to their destination. Mal will contact Alice and tell her that Bob's IP address corresponds to Mal's MAC address, then contact Bob and tell him Alice's IP address corresponds to Mal's MAC address. Thus, both Alice and Bob will be convinced that, in order to send information to each other, they must deliver it to Mal's address. This allows Mal to view the traffic between Alice and Bob and control its flow by being an intermediate.
- b. From Alice's perspective, this attack is undetectable because it comes in the same form of information that she must accept in order to determine where to send information. To detect the attack, she could use one of the solutions described in part q.
- c. From Bob's perspective, this attack is undetectable. Bob is built to receive ARP packets the same as Alice and would expect to be sent an ARP packet without requesting in order to send it back to a user. Mal's packet would have the same style of ethernet header and be indistinguishable from one sent by Alice in order to open communication.
- d. The use of HTTPS would encrypt messages. This would mean Mal would be unable to view the content of packets or alter them without the change being detected. As a result, Alice and Bob could confirm each others' MAC addresses with encrypted exchanges and determine that Mal existed as an intermediate.