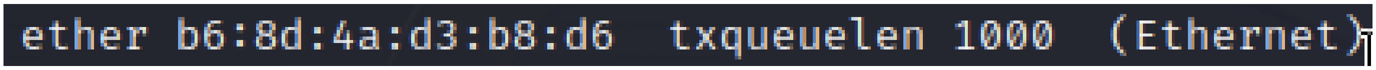
Name: Tianyi Lu

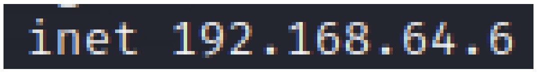
Execution

a. b6:8d:4a:d3:b8:d6



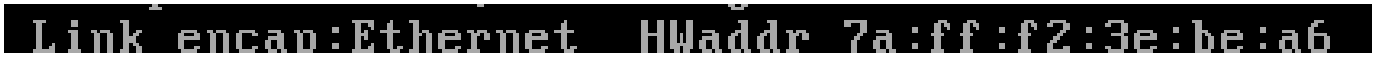
```
ether b6:8d:4a:d3:b8:d6  txqueuelen 1000  (Ethernet)
```

b. 192.168.64.6



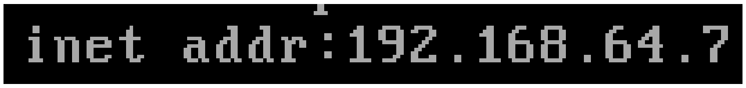
```
inet 192.168.64.6
```

c. 7a:ff:f2:3e:be:a6



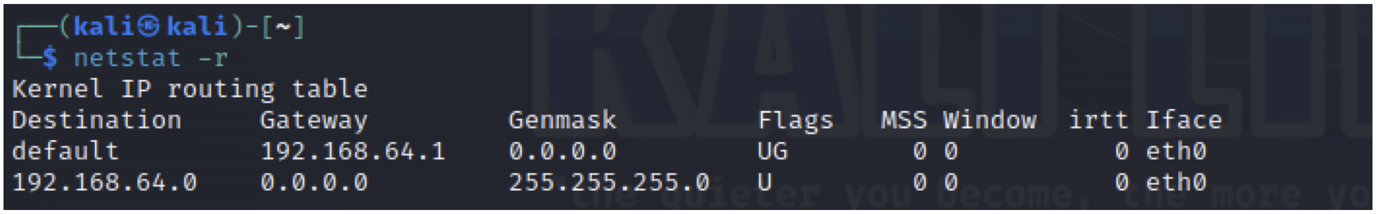
```
Link encap:Ethernet  HWaddr 7a:ff:f2:3e:be:a6
```

d. 192.168.64.7



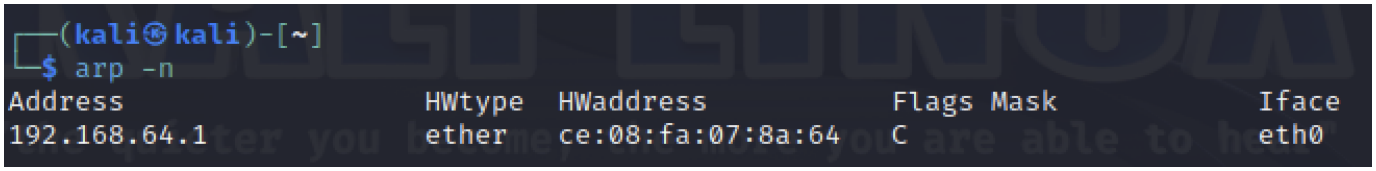
```
inet addr:192.168.64.7
```

e.



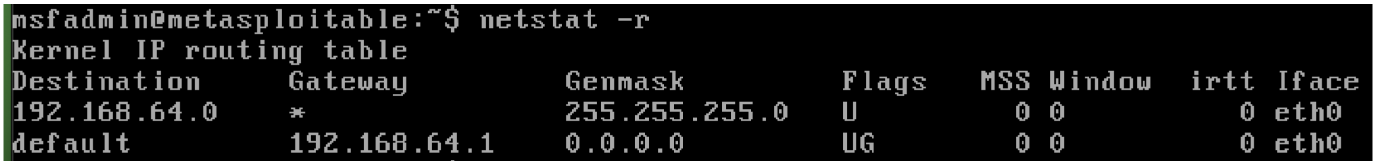
```
┌──(kali㉿kali)-[~]
└─$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         192.168.64.1    0.0.0.0         UG        0 0          0 eth0
192.168.64.0    0.0.0.0         255.255.255.0   U         0 0          0 eth0
```

f.



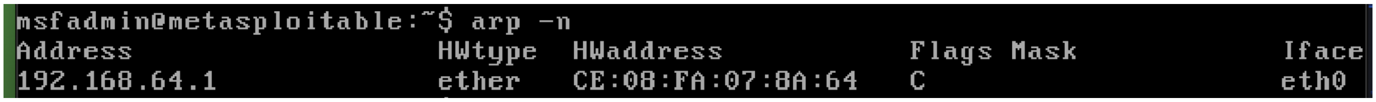
```
┌──(kali㉿kali)-[~]
└─$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.64.1             ether   ce:08:fa:07:8a:64   C                     eth0
```

g.



```
msfadmin@metasploitable:~$ netstat -r
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.



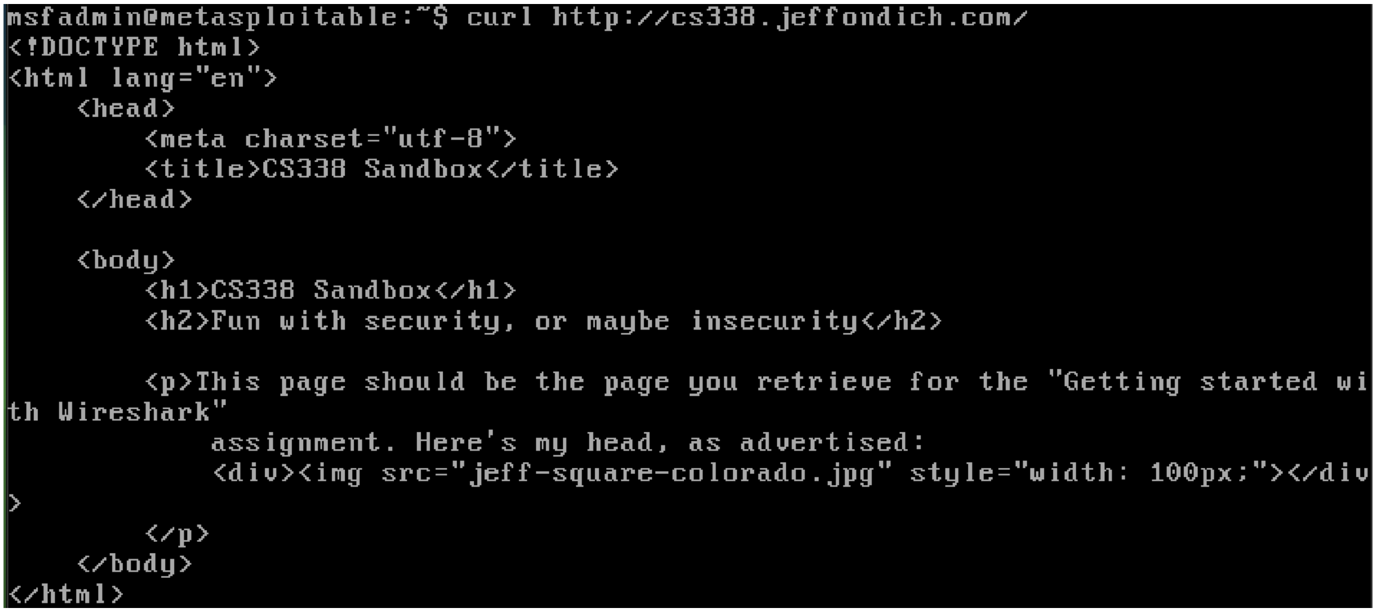
```
msfadmin@metasploitable:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.64.1             ether   CE:08:FA:07:8A:64   C                     eth0
```

i. The `netstat -r` command output shows that the default gateway for this machine is at IP address 192.168.64.1. This is typically the address of the router or next-hop that packets should be sent to when the destination is not on the local subnet.

The `arp -n` command output shows the ARP table. In this case, the IP address 192.168.64.1 is mapped to the MAC address CE:08:FA:07:08:64.

Thus, the TCP SYN packet to start the HTTP query should be sent to the MAC address CE:08:FA:07:08:64.

j. HTTP response on Metasploitable:



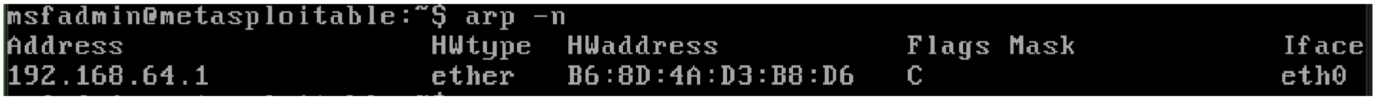
```
msfadmin@metasploitable:~$ curl http://cs338.jeffondich.com/
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>CS338 Sandbox</title>
    </head>

    <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><img src="jeff-square-colorado.jpg" style="width: 100px;"></div
>
        </p>
    </body>
</html>
```

Wireshark didn't capture any packet.

l. The MAC address for the default gateway 192.168.64.1 has changed to Kali's MAC address.

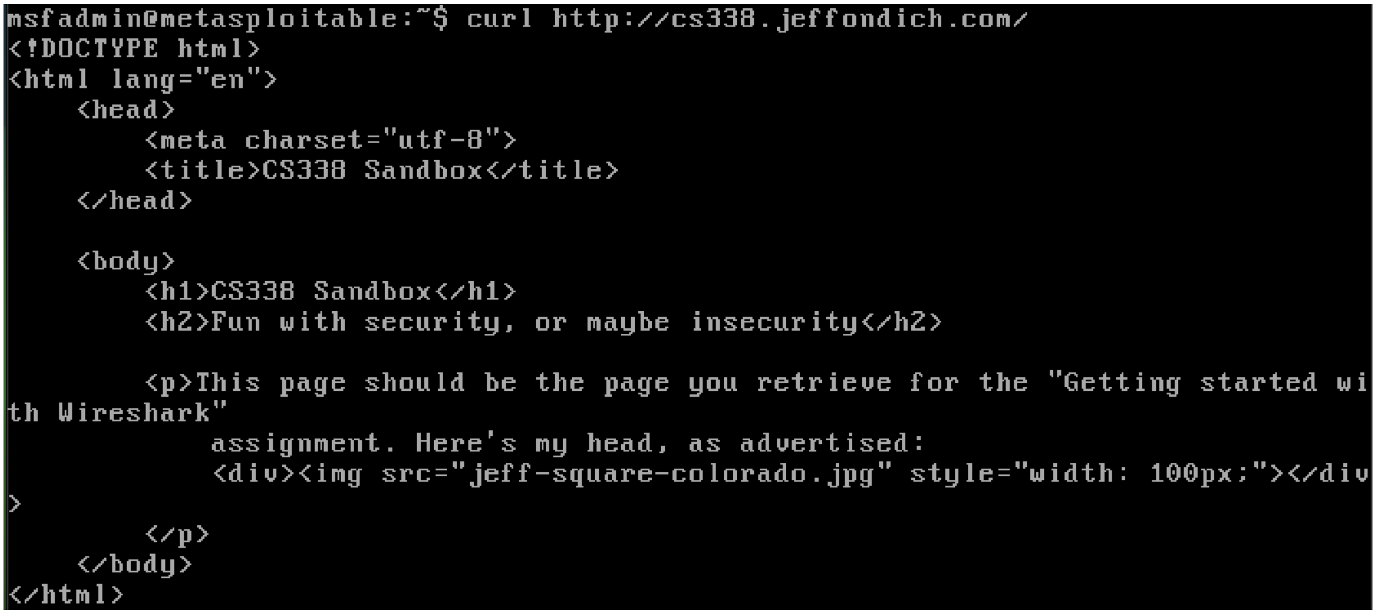


```
msfadmin@metasploitable:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.64.1             ether   B6:8D:4A:D3:B8:D6   C                     eth0
```

m. Without actually doing it yet, predict what will happen if you execute "curl http://cs338.jeffondich.com/" on Metasploitable now. Specifically, to what MAC address will Metasploitable send the TCP SYN packet? Explain why.

Metasploitable will send the TCP SYN packet to Kali's MAC address b6:8d:4a:d3:b8:d6.
The ARP cache tells the Metasploitable machine that the default gateway (at IP address 192.168.64.1) has the MAC address b6:8d:4a:d3:b8:d6. Therefore, the Metasploitable machine will send the TCP SYN packet to this MAC address, which, because of ARP poisoning, belongs to the attacker's machine rather than the actual gateway.

o. HTTP response on Metasploitable:

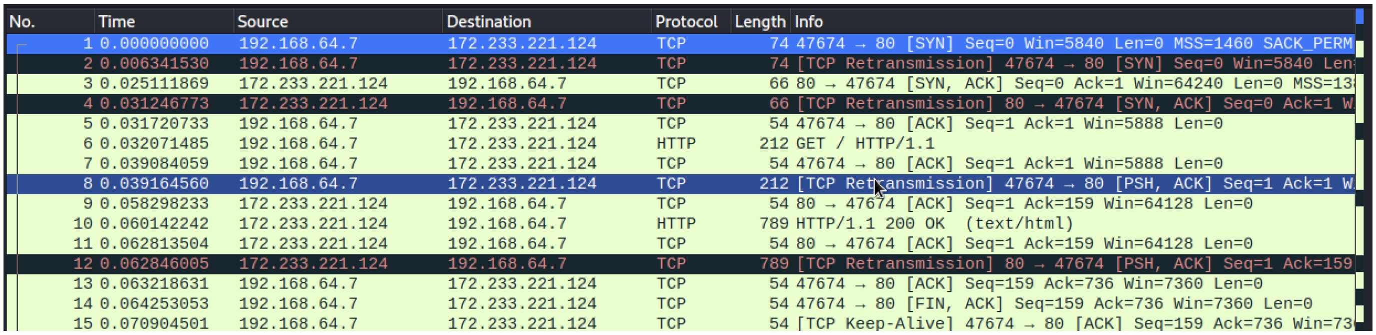


```
msfadmin@metasploitable:~$ curl http://cs338.jeffondich.com/
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>CS338 Sandbox</title>
    </head>

    <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><img src="jeff-square-colorado.jpg" style="width: 100px;"></div
>
        </p>
    </body>
</html>
```

Packets captured in Wireshark:

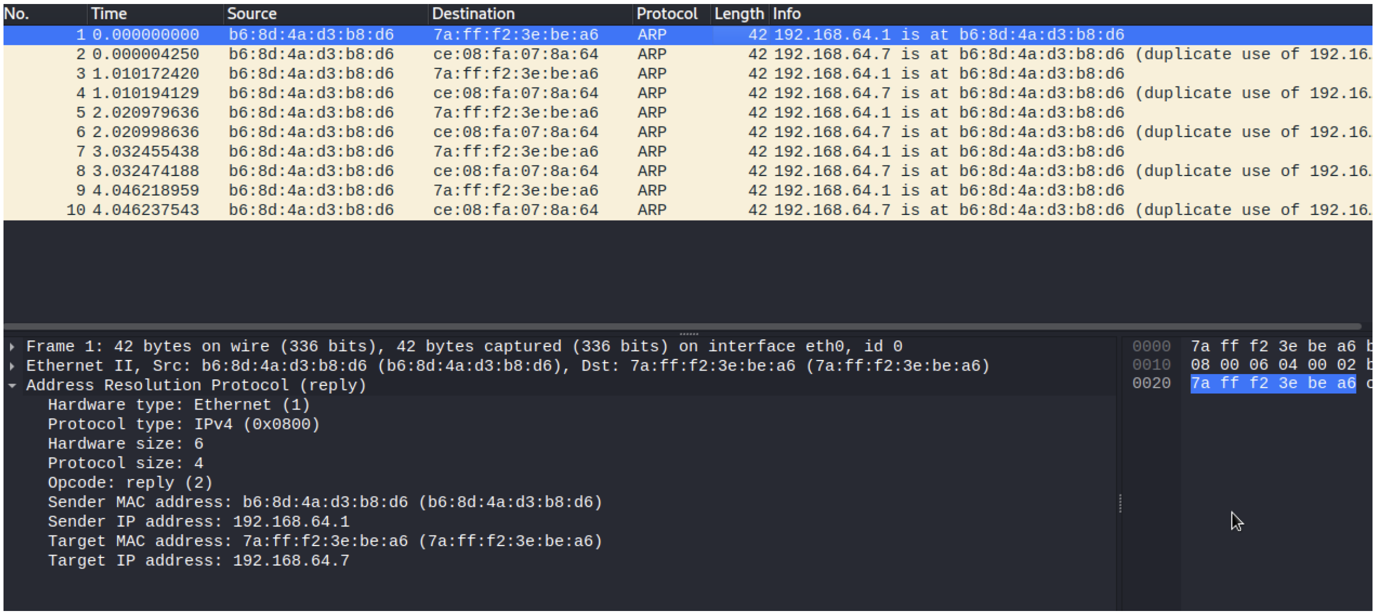


No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.64.7	172.233.221.124	TCP	74	47674 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM
2	0.006341530	192.168.64.7	172.233.221.124	TCP	74	[TCP Retransmission] 47674 → 80 [SYN] Seq=0 Win=5840 Len
3	0.025111869	172.233.221.124	192.168.64.7	TCP	66	80 → 47674 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=13
4	0.031246773	172.233.221.124	192.168.64.7	TCP	66	[TCP Retransmission] 80 → 47674 [SYN, ACK] Seq=0 Ack=1 W
5	0.031720733	192.168.64.7	172.233.221.124	TCP	54	47674 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0
6	0.032071485	192.168.64.7	172.233.221.124	HTTP	212	GET / HTTP/1.1
7	0.039084059	192.168.64.7	172.233.221.124	TCP	54	47674 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0
8	0.039164560	192.168.64.7	172.233.221.124	TCP	212	[TCP Retransmission] 47674 → 80 [PSH, ACK] Seq=1 Ack=1 W
9	0.058298233	172.233.221.124	192.168.64.7	TCP	54	80 → 47674 [ACK] Seq=1 Ack=159 Win=64128 Len=0
10	0.060142242	172.233.221.124	192.168.64.7	HTTP	789	HTTP/1.1 200 OK (text/html)
11	0.062813504	172.233.221.124	192.168.64.7	TCP	54	80 → 47674 [ACK] Seq=1 Ack=159 Win=64128 Len=0
12	0.062846005	172.233.221.124	192.168.64.7	TCP	789	[TCP Retransmission] 80 → 47674 [PSH, ACK] Seq=1 Ack=159
13	0.063218631	192.168.64.7	172.233.221.124	TCP	54	47674 → 80 [ACK] Seq=159 Ack=736 Win=7360 Len=0
14	0.064253053	192.168.64.7	172.233.221.124	TCP	54	47674 → 80 [FIN, ACK] Seq=159 Ack=736 Win=7360 Len=0
15	0.070904501	192.168.64.7	172.233.221.124	TCP	54	[TCP Keep-Alive] 47674 → 80 [ACK] Seq=159 Ack=736 Win=73

From Kali, we can see the TCP handshake packets and HTTP request and response between Metasploitable (192.168.64.7) and cs338.jeffondich.com (172.233.221.124)

p.

After running ARP poisoning on Kali, Kali will repeatedly send out corrupted ARP response to Metasploitable's MAC address. The next time Metasploitable sends ARP request for 192.168.64.1, corrupted ARP response will step in. Then, Metasploitable will receive attacker's MAC address for the default gateway and store this entry in its ARP cache.

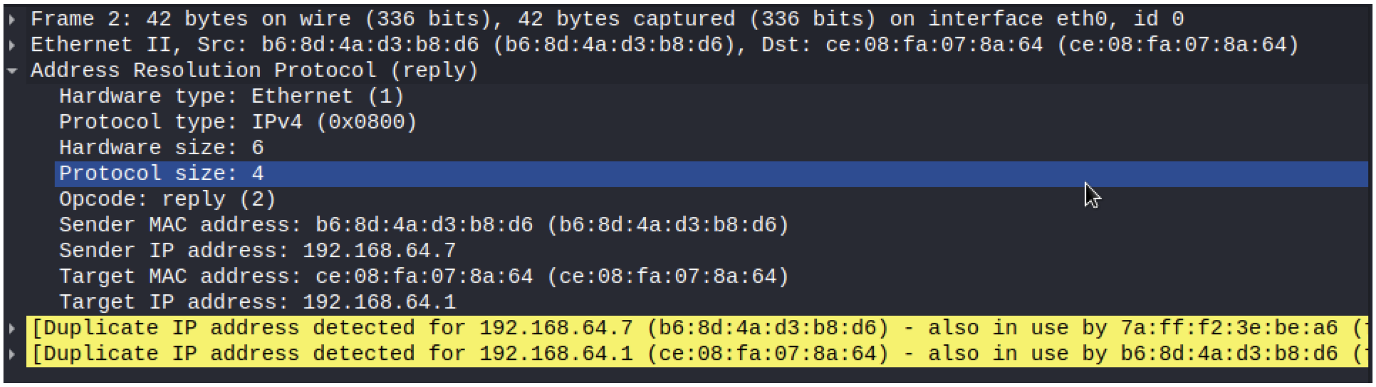


```
No.  Time          Source             Destination        Protocol Length Info
   1 0.000000000   b6:8d:4a:d3:b8:d6  7a:ff:f2:3e:be:a6  ARP          42 192.168.64.1 is at b6:8d:4a:d3:b8:d6
   2 0.000004250   b6:8d:4a:d3:b8:d6  ce:08:fa:07:8a:64  ARP          42 192.168.64.7 is at b6:8d:4a:d3:b8:d6 (duplicate use of 192.16
   3 1.010172420   b6:8d:4a:d3:b8:d6  7a:ff:f2:3e:be:a6  ARP          42 192.168.64.1 is at b6:8d:4a:d3:b8:d6
   4 1.010194129   b6:8d:4a:d3:b8:d6  ce:08:fa:07:8a:64  ARP          42 192.168.64.7 is at b6:8d:4a:d3:b8:d6 (duplicate use of 192.16
   5 2.020979636   b6:8d:4a:d3:b8:d6  7a:ff:f2:3e:be:a6  ARP          42 192.168.64.1 is at b6:8d:4a:d3:b8:d6
   6 2.020998636   b6:8d:4a:d3:b8:d6  ce:08:fa:07:8a:64  ARP          42 192.168.64.7 is at b6:8d:4a:d3:b8:d6 (duplicate use of 192.16
   7 3.032455438   b6:8d:4a:d3:b8:d6  7a:ff:f2:3e:be:a6  ARP          42 192.168.64.1 is at b6:8d:4a:d3:b8:d6
   8 3.032474188   b6:8d:4a:d3:b8:d6  ce:08:fa:07:8a:64  ARP          42 192.168.64.7 is at b6:8d:4a:d3:b8:d6 (duplicate use of 192.16
   9 4.046218959   b6:8d:4a:d3:b8:d6  7a:ff:f2:3e:be:a6  ARP          42 192.168.64.1 is at b6:8d:4a:d3:b8:d6
  10 4.046237543   b6:8d:4a:d3:b8:d6  ce:08:fa:07:8a:64  ARP          42 192.168.64.7 is at b6:8d:4a:d3:b8:d6 (duplicate use of 192.16
```

```
▸ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
▸ Ethernet II, Src: b6:8d:4a:d3:b8:d6 (b6:8d:4a:d3:b8:d6), Dst: 7a:ff:f2:3e:be:a6 (7a:ff:f2:3e:be:a6)
▾ Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: b6:8d:4a:d3:b8:d6 (b6:8d:4a:d3:b8:d6)
    Sender IP address: 192.168.64.1
    Target MAC address: 7a:ff:f2:3e:be:a6 (7a:ff:f2:3e:be:a6)
    Target IP address: 192.168.64.7
```

```
0000  7a ff f2 3e be a6 b
0010  08 00 06 04 00 02 b
0020  7a ff f2 3e be a6 c
```

q. The ARP spoofing detector will detect duplicated MAC addresses for a specific IP address in ARP message histories. ARP spoofing is very likely happening especially when the IP address for the default gateway on the network is mapped to multiple MAC addresses.



```
▸ Frame 2: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
▸ Ethernet II, Src: b6:8d:4a:d3:b8:d6 (b6:8d:4a:d3:b8:d6), Dst: ce:08:fa:07:8a:64 (ce:08:fa:07:8a:64)
▾ Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: b6:8d:4a:d3:b8:d6 (b6:8d:4a:d3:b8:d6)
    Sender IP address: 192.168.64.7
    Target MAC address: ce:08:fa:07:8a:64 (ce:08:fa:07:8a:64)
    Target IP address: 192.168.64.1
▸ [Duplicate IP address detected for 192.168.64.7 (b6:8d:4a:d3:b8:d6) - also in use by 7a:ff:f2:3e:be:a6 (
▸ [Duplicate IP address detected for 192.168.64.1 (ce:08:fa:07:8a:64) - also in use by b6:8d:4a:d3:b8:d6 (
```

Synthesis

a. When Alice attempts to communicate with Bob over a network, her packets are initially sent to the local network's default gateway interface before being forwarded to Bob. To correctly identify this gateway, Alice needs to know both its IP and MAC addresses. She typically obtains the MAC address from her local ARP cache. However, this process can be compromised by Mal. Mal can exploit the ARP cache's functionality, where it updates whenever an IP address is resolved to a MAC address. By repeatedly sending ARP responses to Alice while posing as the default gateway, Mal can deceive Alice into updating her ARP cache with a falsified IP-MAC mapping. Consequently, all of Alice's packets intended for Bob will be rerouted through Mal first, allowing him to view them in their entirety.

b. It is detectable. If a Alice keeps a record of ARP responses, a sudden change in the MAC address associated with an IP address, without a corresponding change in the network could be suspicious. There are tools available that can monitor the ARP traffic on the network and alert administrators to unusual patterns, such as the same IP address being associated with different MAC addresses in a short period of time, or a single MAC address claiming to own multiple IP addresses.

c. Bob generally can't detect the attack. Bob, which is likely remote and not on the same local network, does not see ARP traffic because ARP requests and responses are not routed across the internet.

d. Using HTTPS instead of HTTP would not prevent ARP poisoning itself because ARP poisoning attacks occur at the local network. However, HTTPS can mitigate some of the risks associated with ARP poisoning, particularly the risk of man-in-the-middle attacks that could intercept or modify the data being transmitted.

For Alice:
Alice would still detect the ARP poisoning itself by monitoring local ARP traffic.
HTTPS ensures that the communication between the Alice and Bob is encrypted. Therefore, even if an attacker intercepts the traffic through ARP poisoning, they would not be able to read or modify the HTTPS traffic easily due to the strong encryption.

For Bob:
Bob would still not detect the ARP poisoning, as it does not participate in the ARP process of the local network. However, Bob can detect a MitM attack if the attacker tries to intercept the HTTPS connection and present a false certificate, as browsers and clients check the validity of the server's SSL certificate. If the certificate does not match or is not signed by a trusted certificate authority, the browser or client would alert the user.