OLD DOMINION UNIVERSITY

CYSE 301 CYBERSECURITY TECHNIQUES AND OPERATIONS

Assignment: Lab 2– Traffic Tracing and Sniffing

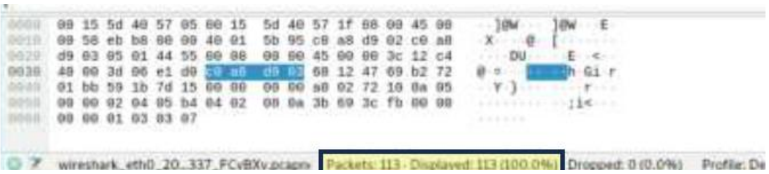
Cora Wilson

01303228

TASK A: GET STARTED WITH WIRESHARK (40 POINTS)

1. How many packets are captured in total? How many packets are displayed?

113 packets were captured and displayed.



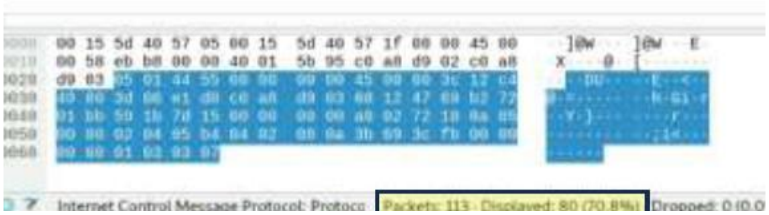
```
0000  00 15 5d 40 57 05 00 15  5d 40 57 1f 08 00 45 00   ..]@W...]@W...E.
0010  00 58 eb b8 00 00 40 01  5b 95 c0 a8 d9 02 c0 a8   .X....@.[.......
0020  d9 03 05 01 44 55 00 00  00 00 45 00 00 3c 12 c4   ....DU....E..<..
0030  40 00 3d 06 e1 d0 c0 a8  d9 03 68 12 47 69 b2 72   @.=.......h.Gi.r
0040  01 bb 59 1b 7d 15 00 00  00 00 a0 02 72 10 0a 05   ..Y.}.......r...
0050  00 00 02 04 05 b4 04 02  08 0a 3b 69 3c fb 00 00   ..........;i<...
0060  00 00 01 03 03 07                                  ......
```

wireshark_eth0_20_337_FCvBXv.pcapn   Packets: 113 · Displayed: 113 (100.0%)   Dropped: 0 (0.0%)   Profile: De

The above screenshot is the Wireshark results from pinging Ubuntu VM for 5-10 seconds. In the screenshot, it shows at the bottom that 113 packets were captured during listening on eth0 and 113 packets are being displayed.

2. Apply "ICMP" as a display filter in Wireshark. Then repeat the previous question (Q1).

113 packets were captured, only 80 were displayed.



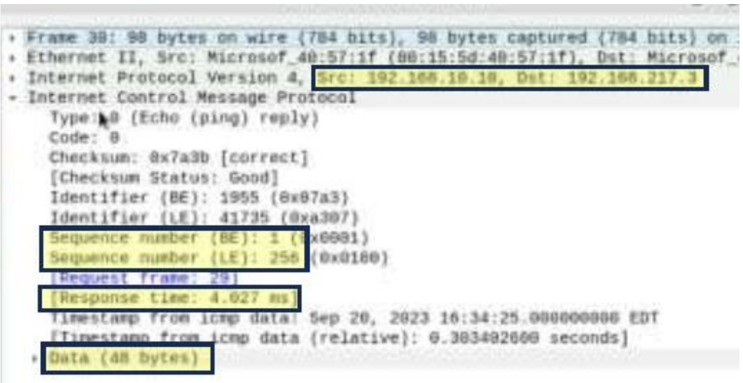
```
0000  00 15 5d 40 57 05 00 15  5d 40 57 1f 08 00 45 00   ..]@W...]@W...E.
0010  00 58 eb b8 00 00 40 01  5b 95 c0 a8 d9 02 c0 a8   .X....@.[.......
0020  d9 03 05 01 44 55 00 00  00 00 45 00 00 3c 12 c4   ....DU....E..<..
0030  40 00 3d 06 e1 d0 c0 a8  d9 03 68 12 47 69 b2 72   @.=.......h.Gi.r
0040  01 bb 59 1b 7d 15 00 00  00 00 a0 02 72 10 0a 05   ..Y.}.......r...
0050  00 00 02 04 05 b4 04 02  08 0a 3b 69 3c fb 00 00   ..........;i<...
0060  00 00 01 03 03 07                                  ......
```

Internet Control Message Protocol: Protoco   Packets: 113 · Displayed: 80 (70.8%)   Dropped: 0 (0.0

Figure 2 Screenshot of Wireshark with "ICMP" filter in Attacker Linux for Task A.2

The above screenshot is the Wireshark results after applying the "ICMP" filter from the previously captured information. In the screenshot, it shows at the bottom that 113 packets were captured during listening on eth0 and only 80 packets are currently being displayed that have "ICMP" protocol.

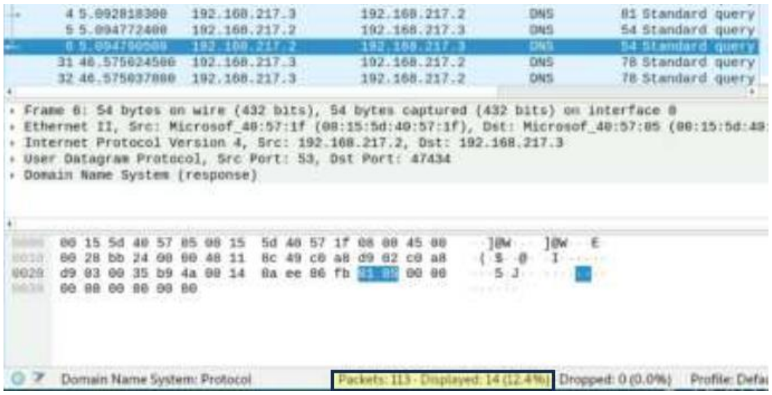
3. Select an Echo (replay) message from the list. What are the source and destination IPs of this packet? What are the sequence number and the size of the data? What is the response time?



```
▸ Frame 30: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on
▸ Ethernet II, Src: Microsof_40:57:1f (00:15:5d:40:57:1f), Dst: Microsof_
▸ Internet Protocol Version 4, Src: 192.168.10.10, Dst: 192.168.217.3
▾ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x7a3b [correct]
    [Checksum Status: Good]
    Identifier (BE): 1955 (0x07a3)
    Identifier (LE): 41735 (0xa307)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Request frame: 29]
    [Response time: 4.027 ms]
    Timestamp from icmp data: Sep 20, 2023 16:34:25.000000000 EDT
    [Timestamp from icmp data (relative): 0.303492600 seconds]
  ▸ Data (48 bytes)
```

Figure 3 Screenshot of Echo reply message from Packet 30 in Wireshark on Attacker Linux for Task A.3 The above screenshot is from Echo reply Packet 30. The source IP is 192.168.10.10 and the destination IP is 192.168.217.3. The sequence numbers are 1 for BE and 256 for LE. The data size is 48 bytes, and the response time is 4.027 ms. (Information highlighted in yellow to bring attention to it).

4. Apply "DNS" as a display filter in Wireshark. How many packets are displayed?



```
     4 5.092818300   192.168.217.3     192.168.217.2     DNS     81 Standard query
     5 5.094772400   192.168.217.2     192.168.217.3     DNS     54 Standard query
     6 5.094790500   192.168.217.2     192.168.217.3     DNS     54 Standard query
    31 40.575624500  192.168.217.3     192.168.217.2     DNS     78 Standard query
    32 40.575037000  192.168.217.3     192.168.217.2     DNS     78 Standard query

▸ Frame 6: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
▸ Ethernet II, Src: Microsof_40:57:1f (00:15:5d:40:57:1f), Dst: Microsof_40:57:05 (00:15:5d:40:
▸ Internet Protocol Version 4, Src: 192.168.217.2, Dst: 192.168.217.3
▸ User Datagram Protocol, Src Port: 53, Dst Port: 47434
▸ Domain Name System (response)

0000  00 15 5d 40 57 05 00 15  5d 40 57 1f 08 00 45 00    ..]@W..]@W..E.
0010  00 28 bb 24 00 00 40 11  8c 49 c0 a8 d9 02 c0 a8    .(.$..@..I......
0020  d9 03 00 35 b9 4a 00 14  8a ee 86 fb 81 82 00 00    ...5.J..........
0030  00 00 00 00 00 00                                   ......

Domain Name System: Protocol    Packets: 113 · Displayed: 14 (12.4%)    Dropped: 0 (0.0%)    Profile: Defa
```

Figure 4 Screenshot of Wireshark with "DNS" filter in Attacker Linux for Task A.4

The above screenshot shows the "DNS" filter being applied. Out of the 113 packets available, only 14 packets are being displayed under the DNS filter (highlighted in yellow).

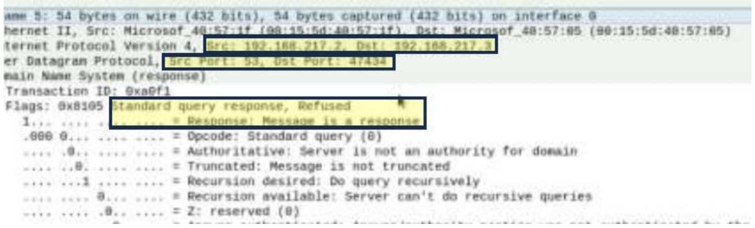
5. Find a DNS query packet. What is the domain name this host is trying to resolve? What is the source IP and port number, destination IP, and port number? Please express in the format: IP:port.

The domain name that the host is trying to resolve is 3.debian.pool.ntp.org.

The source IP and port number are 192.168.217.3:47434 and the destination IP and port number are 192.168.217.2:53 (Information highlighted in yellow to bring attention to it).

IP:port=192.168.217.3:47434

6. Find the corresponding DNS response to the query you selected at the previous step, and what is the source IP and port number, destination IP, and port number? What is the message replied from the DNS server?



```
ame 5: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
hernet II, Src: Microsof_40:57:1f (00:15:5d:40:57:1f), Dst: Microsof_40:57:05 (00:15:5d:40:57:05)
ternet Protocol Version 4, Src: 192.168.217.2, Dst: 192.168.217.3
er Datagram Protocol, Src Port: 53, Dst Port: 47434
main Name System (response)
Transaction ID: 0xa0f1
Flags: 0x8105 Standard query response, Refused
  1... .... .... .... = Response: Message is a response
  .000 0... .... .... = Opcode: Standard query (0)
  .... .0.. .... .... = Authoritative: Server is not an authority for domain
  .... ..0. .... .... = Truncated: Message is not truncated
  .... ...1 .... .... = Recursion desired: Do query recursively
  .... .... 0... .... = Recursion available: Server can't do recursive queries
  .... .... .0.. .... = Z: reserved (0)
```

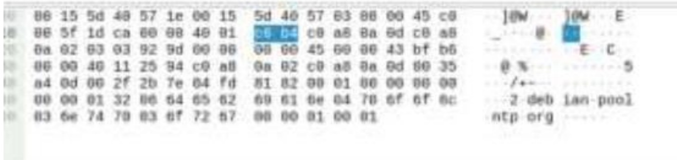
Figure 6 Screenshot of corresponding DNS response from Packet 5 in Wireshark on Attacker Linux for Task A.6 The above screenshot is from the corresponding DNS response from Question 5 as Packet 5. The source IP and port number are 192.168.217.2:53 and the destination IP and port number are 192.168.217.3:47434. The standard query response was Refused and the response says “Message is a response.” (Information highlighted in yellow to bring attention to it).

TASK B. SNIFF LAN TRAFFIC

1. Sniff ICMP traffic (10 + 10 = 20 points)

Open two terminals on External Kali VM. Use one ping Ubuntu VM, and use the other ping Internal Kali.

a. Apply proper display or capture filter on Internal Kali VM to show active ICMP traffic.



```
00 15 5d 40 57 1e 00 15  5d 40 57 03 08 00 45 c0   ..]@W...]@W..E.
00 5f 1d ca 00 00 40 01  08 04 c0 a8 0a 0d c0 a8   ._....@.........
0a 02 03 03 92 9d 00 00  00 00 45 00 00 43 bf b6   ..........E..C..
00 00 40 11 25 94 c0 a8  0a 02 c0 a8 0a 0d 00 35   ..@.%..........5
a4 0d 00 2f 2b 7e 04 fd  81 82 00 01 00 00 00 00   .../+~..........
00 00 01 32 00 64 65 62  69 61 6e 04 70 6f 6f 6c   ...2.deb ian.pool
03 6e 74 70 03 6f 72 67  00 00 01 00 01            .ntp.org .....
```

Figure 7 Screenshot of Wireshark on Internal Kali sniffing traffic between External Kali and Ubuntu VM. This show

the “ICMP” filter being used for Task B.1a

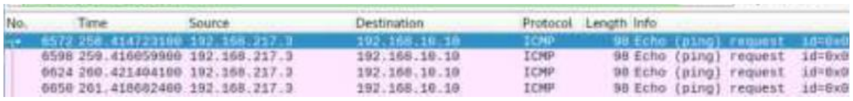
The above screenshot shows the “ICMP” filter being applied and showing the active ICMP

traffic.

b. Apply proper display or capture filter on Internal Kali VM that ONLY displays ICMP

request originated from External Kali VM and goes to Ubuntu 64-bit VM.Display your

current directory in a terminal.



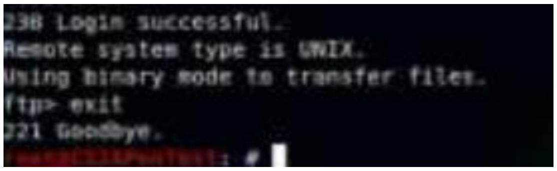
```
No.     Time            Source           Destination      Protocol  Length Info
  6572 258.414723100 192.168.217.3    192.168.10.10    ICMP        98 Echo (ping) request  id=0x0
  6598 259.416059900 192.168.217.3    192.168.10.10    ICMP        98 Echo (ping) request  id=0x0
  6624 260.421404100 192.168.217.3    192.168.10.10    ICMP        98 Echo (ping) request  id=0x0
  6650 261.418602400 192.168.217.3    192.168.10.10    ICMP        98 Echo (ping) request  id=0x0
```

Figure 8 Screenshot of Wireshark on Internal Kali sniffing traffic between External Kali and Ubuntu VM. This show the "ICMP" and "Ip destination of ubuntu" filter being used for Task B.1b

The above screenshot shows the filter being used to show only ICMP request that originated from External Kali (192.168.217.3) to Ubuntu VM (192.168.10.10).

2. Sniff FTP traffic (10 + 15 + 15 = 40 pts points)

a. Ubuntu VM is also serving as an FTP server inside the LAN network. Now, you need to use External Kali to access this FTP server by using the command: ftp [ip_addr of ubuntu VM]. The username for the FTP server is student, and the password is password. You can follow the steps below to access the FTP server.

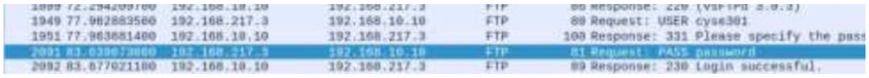


```
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> exit
221 Goodbye.
[illegible]: #
```

Figure 9 Screenshot of accessing the Ubuntu FTP Server using External Kali. This shows the ftp command in the External Kali terminal to access the FTP server in Ubuntu for Task B.2a

The above screenshot shows External Kali accessing the FTP server on Ubuntu VM using the command: "ftp 192.168.10.10" (Ip address of Ubuntu VM). After entering the proper username (student) and password (password) the login will display as successful and file transfer can commence. Exit command was given because there was no file transfer.

b. Unfortunately, Internal Kali, the attacker, is also sniffing to the communication. Therefore, all your communication is exposed to the attacker. Now, you need to find out the password used by External Kali to access the FTP server from the intercepted traffic on Internal Kali. You need to screenshot and explain how you find the password.

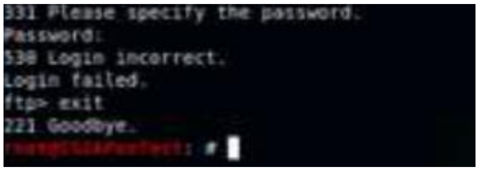


```
  1889 72.294209700  192.168.10.10    192.168.217.3    FTP       86 Response: 220 (vsFTPd 3.0.3)
  1949 77.982883500  192.168.217.3    192.168.10.10    FTP       89 Request: USER cyse301
  1951 77.983881400  192.168.10.10    192.168.217.3    FTP      100 Response: 331 Please specify the pass
  2091 83.039073800  192.168.217.3    192.168.10.10    FTP       81 Request: PASS password
  2092 83.677021100  192.168.10.10    192.168.217.3    FTP       89 Response: 230 Login successful.
```

Figure 10 Screenshot of Wireshark from Internal Kali showing the "FTP" traffic by using the "FTP" filter for Task B.2b

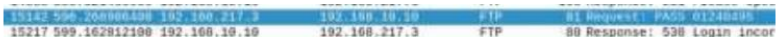
The above screenshot shows that when you filter just the FTP traffic in Wireshark on the Internal Kali, you can see the entire FTP interaction between External Kali and Ubuntu. This screenshot shows that the username used was "Student" and the password that was used was "password" followed by the response of login successful, so that you know it works.

c. After you successfully find the username & password from the FTP traffic, repeat the previous step (2.a), and use your MIDAS ID as the username and UIN as the password to re-access the FTP server from External Kali. Although External Kali may not access the FTP server, you need to intercept the packets containing these “secrets” from the attacker VM, which is Internal Kali.



```
331 Please specify the password.
Password:
530 Login incorrect.
Login failed.
ftp> exit
221 Goodbye.
[illegible] # 
```

Figure 11 Screenshot of accessing the Ubuntu FTP Server again using External Kali. This shows the ftp command in the External Kali terminal to access the FTP server in Ubuntu for Task B.2c The above screenshot shows the FTP Server on Ubuntu being access via External Kali this time using my MIDAS ID as the username and my UIN as the password. This time the login was incorrect and failed, but the traffic was captured, and the “exit” command was entered.



```
15142 596.268000498 192.168.217.3     192.168.10.10     FTP       81 Request: PASS 01240495
15217 599.162812100 192.168.10.10     192.168.217.3     FTP       80 Response: 530 Login incor
```

Figure 12 Screenshot of Wireshark from Internal Kali showing the “FTP” traffic by using the “FTP” filter for Task B.2c

The above screenshot shows the Wireshark information from Internal Kali using the “FTP” filter again. This time it shows the username as my MIDAS ID (Cwils083) and my UIN as my Password. It also shows that the login was incorrect as well.