OLD DOMINION UNIVERSITY

CYSE 301 CYBERSECURITY TECHNIQUES AND OPERATIONS

Assignment #2: Lab 2- Traffic Tracing and Sniffing

Antonio Shields

TASK A: GET STARTED WITH WIRESHARK (5 POINT EACH X 6 QUESTIONS = 30 POINTS)

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

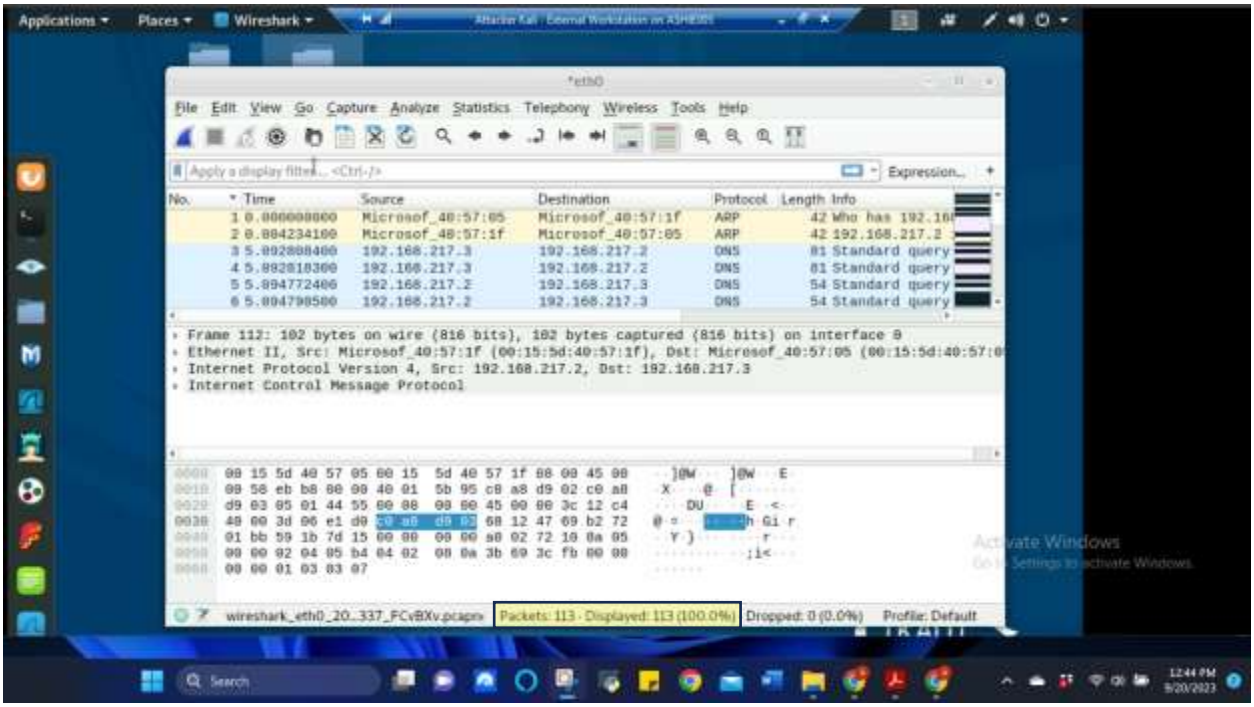

Figure 1 Screenshot of Wireshark in Attacker Linux for Task A.1

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 also 113 packets are currently being displayed (highlighted in yellow at bottom).

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

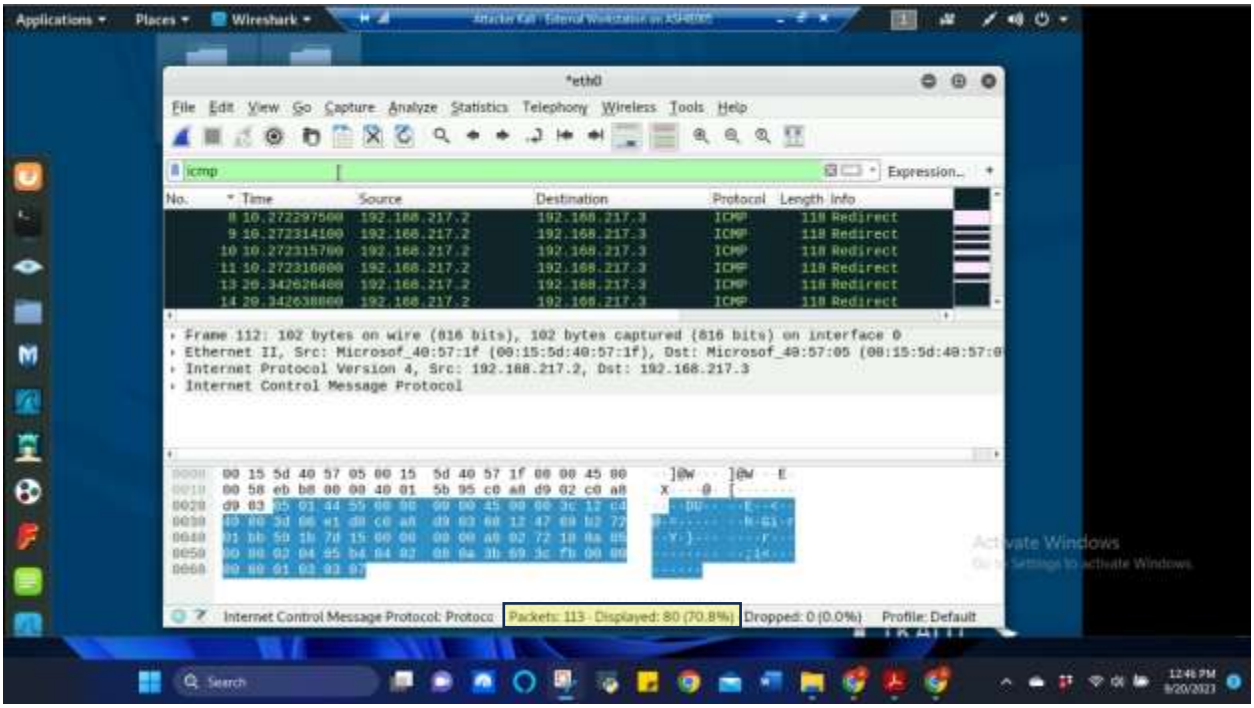

Figure 2 Screenshot of Wireshark with "ICMP" filter in Attacker Linux for Task A.2

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 (highlighted in yellow at bottom).

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?

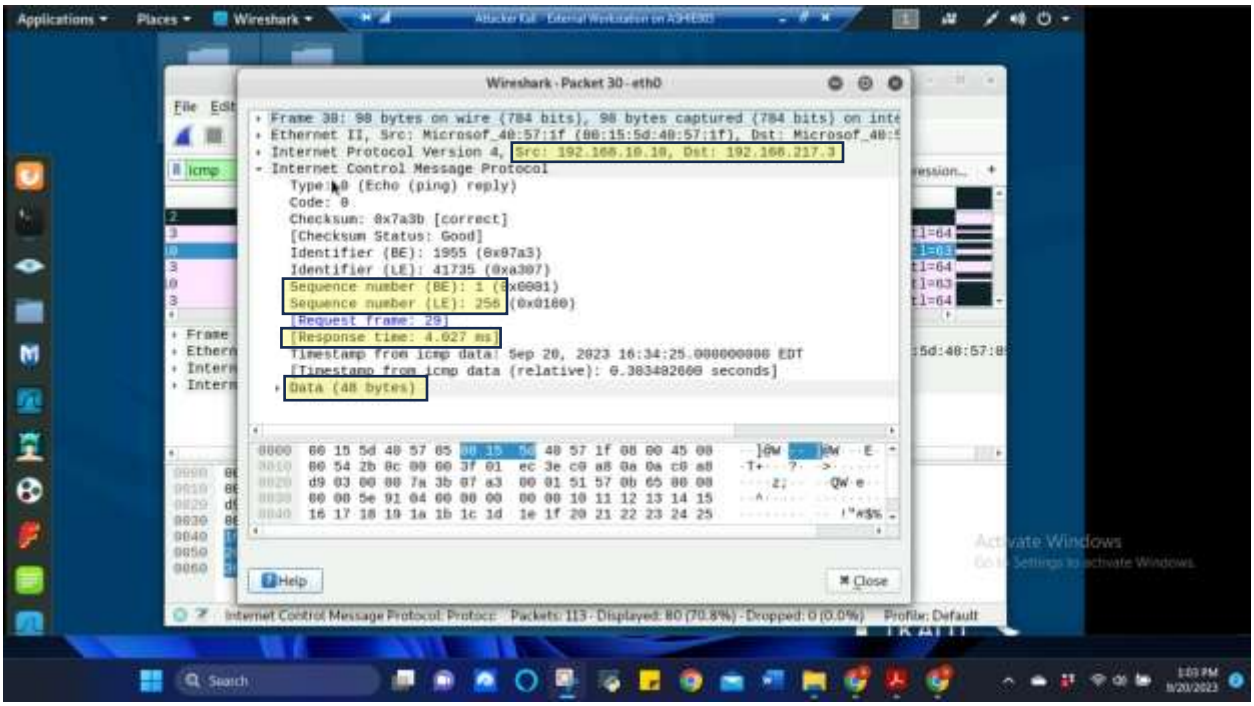

Figure 3 Screenshot of Echo reply message from Packet 30 in Wireshark on Attacker Linux for Task A.3

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?

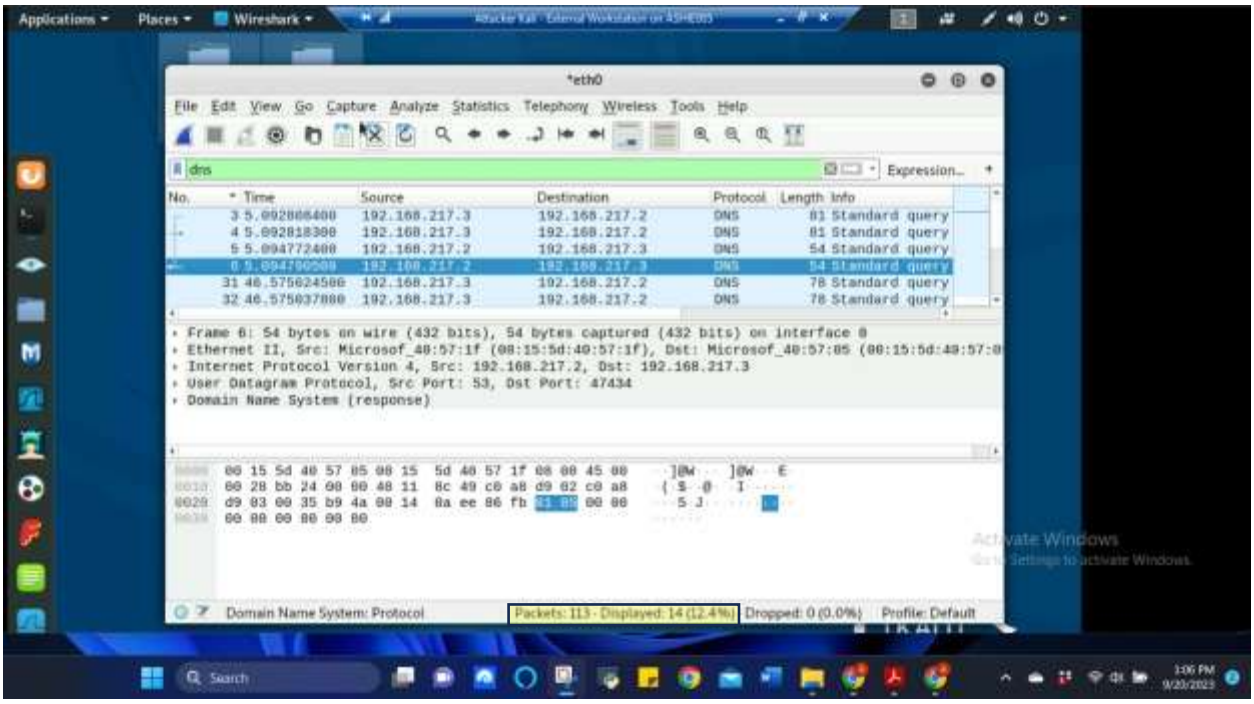

Figure 4 Screenshot of Wireshark with "DNS" filter in Attacker Linux for Task A.4

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**.

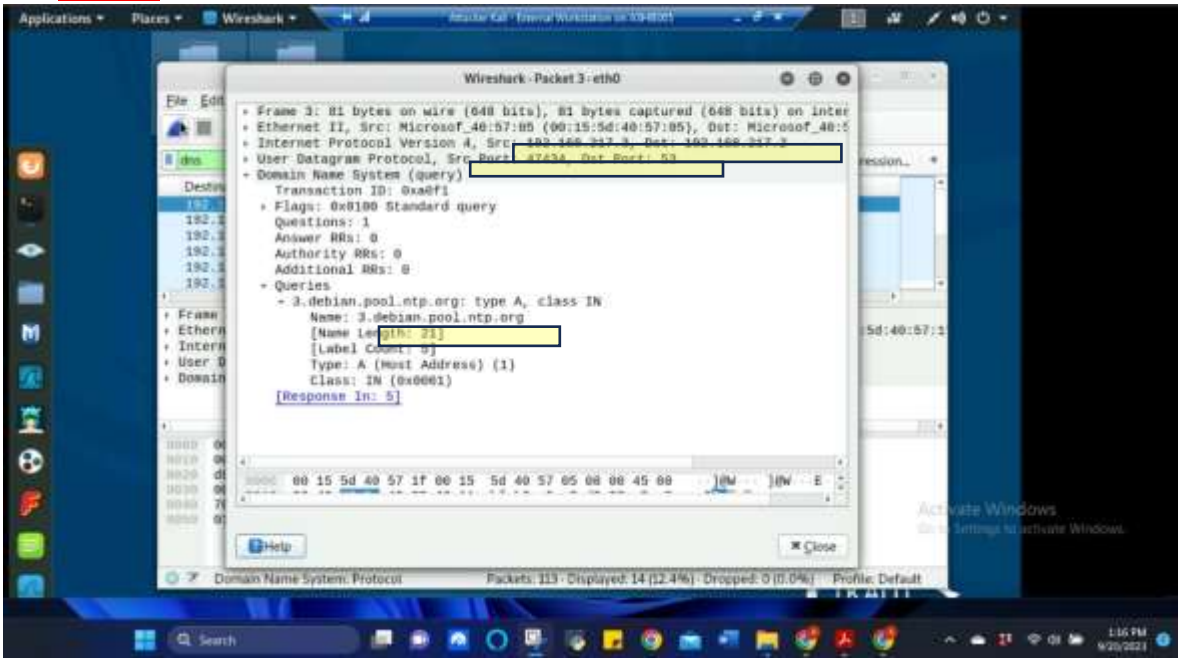

Figure 5 Screenshot of DNS query packet from Packet 3 in Wireshark on Attacker Linux for Task A.5

The above screenshot is from the DNS query Packet 3. 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).

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?

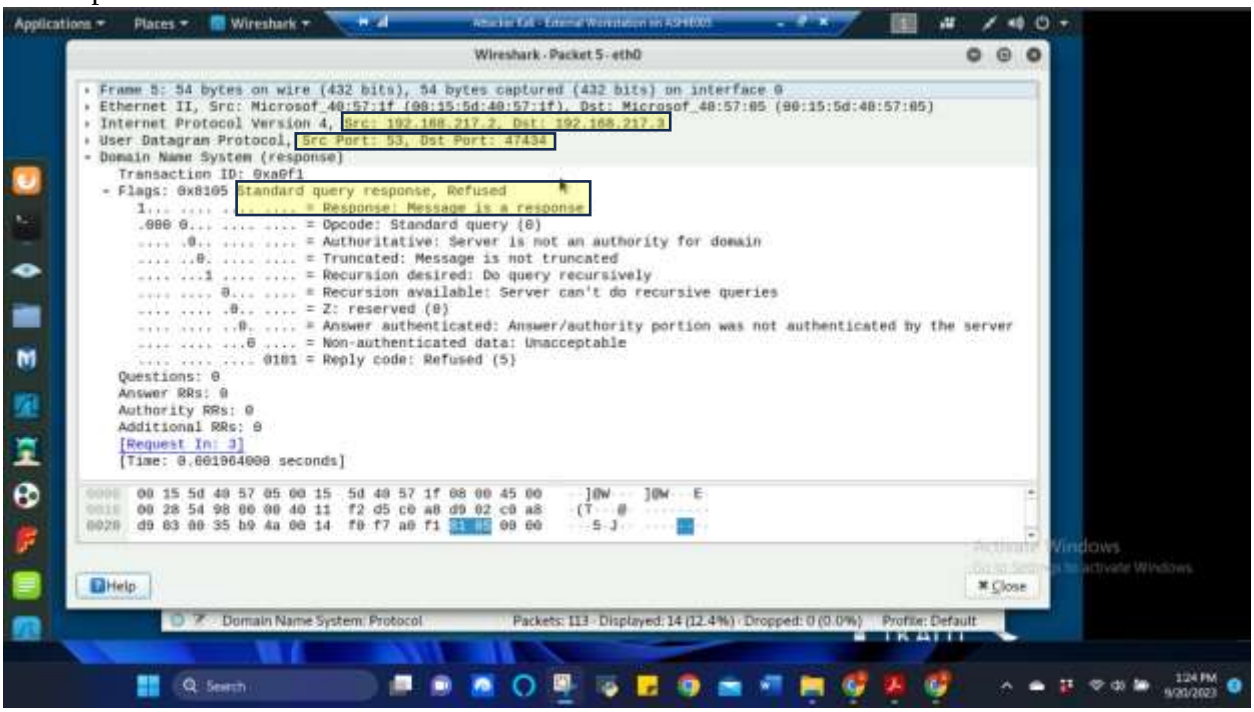

Figure 6 Screenshot of corresponding DNS response from Packet 5 in Wireshark on Attacker Linux for Task A.6

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.

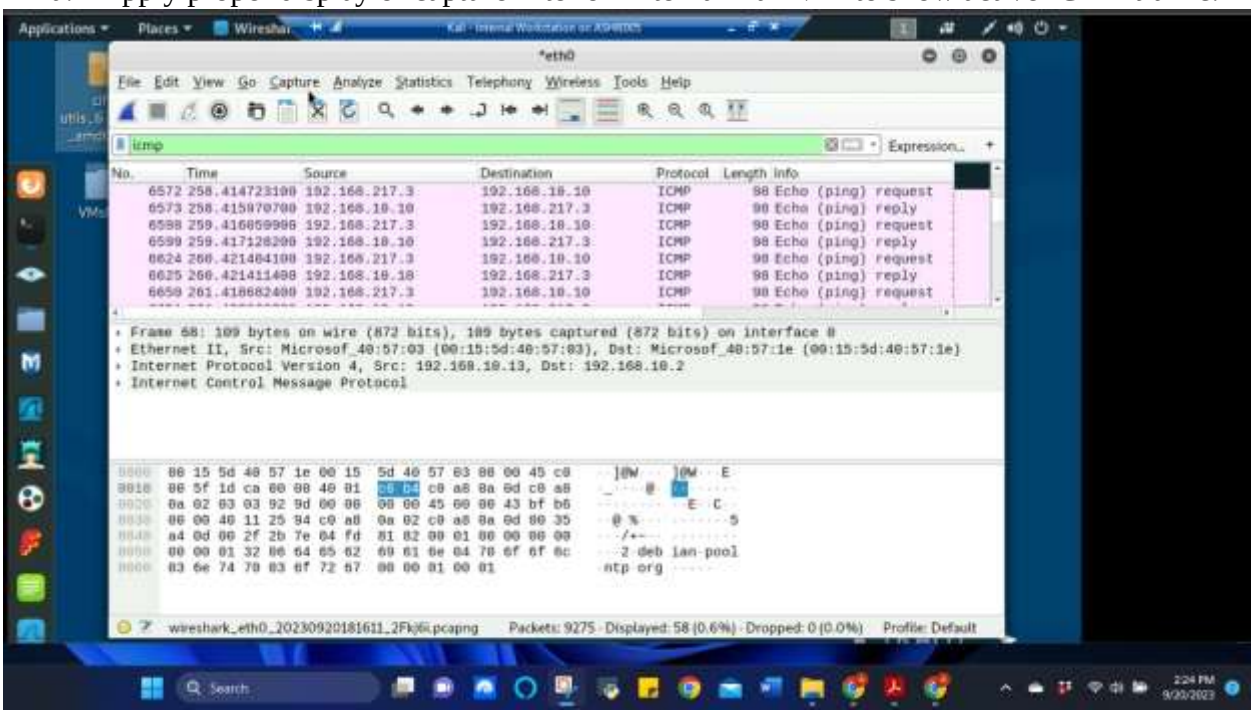

Figure 7 Screenshot of Wireshark on Internal Kali sniffing traffic between External Kali and Ubuntu VM. This show the "ICMP" filter being used for Task B.1a

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.

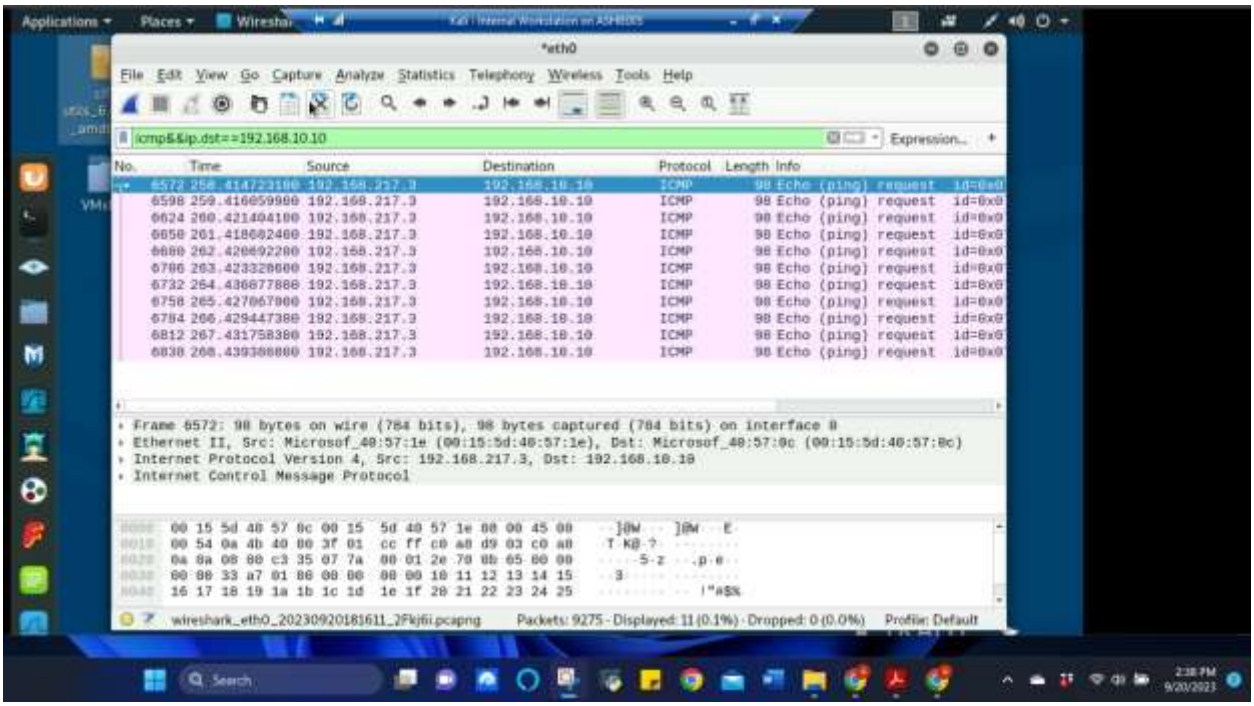

Figure 8 Screenshot of Wireshark on Internal Kali sniffing traffic between External Kali and Ubuntu VM. This show the "ICMP" and "ip destination of ubuntu" filter being used for Task B.1b

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). This was achieved by using the filter icmp&&ip.dst = = 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 cyse301, and the password is password. You can follow the steps below to access the FTP server.

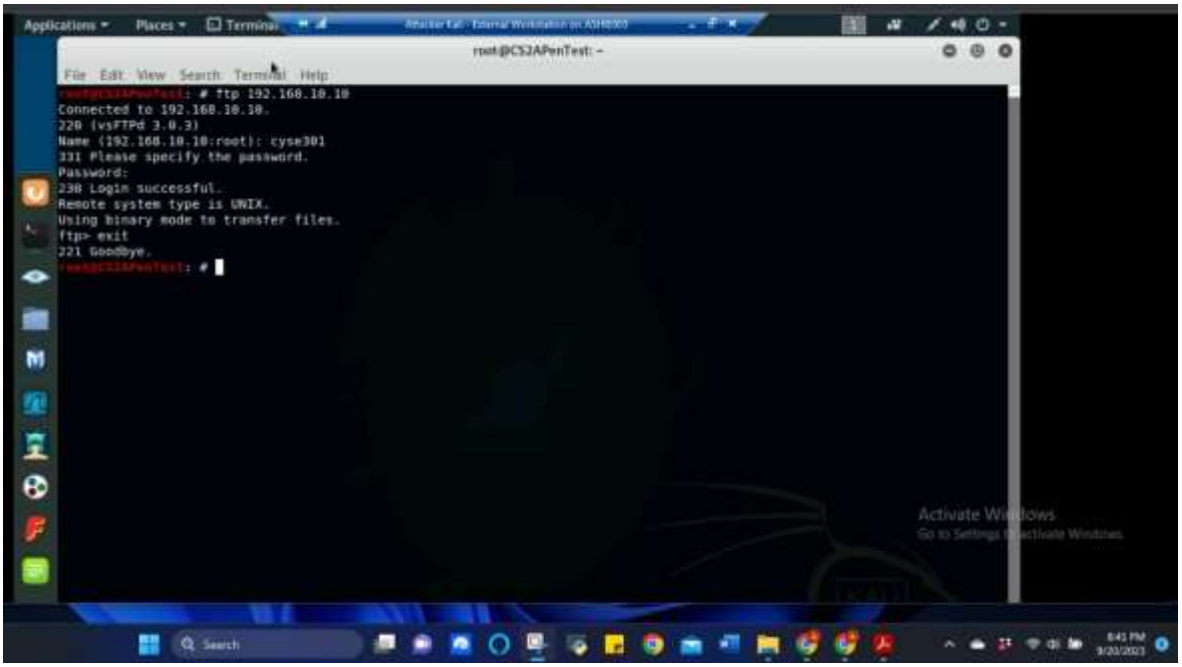

Figure 9 Screenshot of accessing the Ubuntu FTP Server using External Kali. This shows the ftp command in the External Kali terminal to access the FTP server in Ubuntu for Task B.2a

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 (cyse301) and password (password) the login will display as successful and file transfer can commence. Exit command was given because no file transfer at this time.

b. Unfortunately, Internal Kali, the attacker, is also sniffing to the communication. Therefore, all of 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.

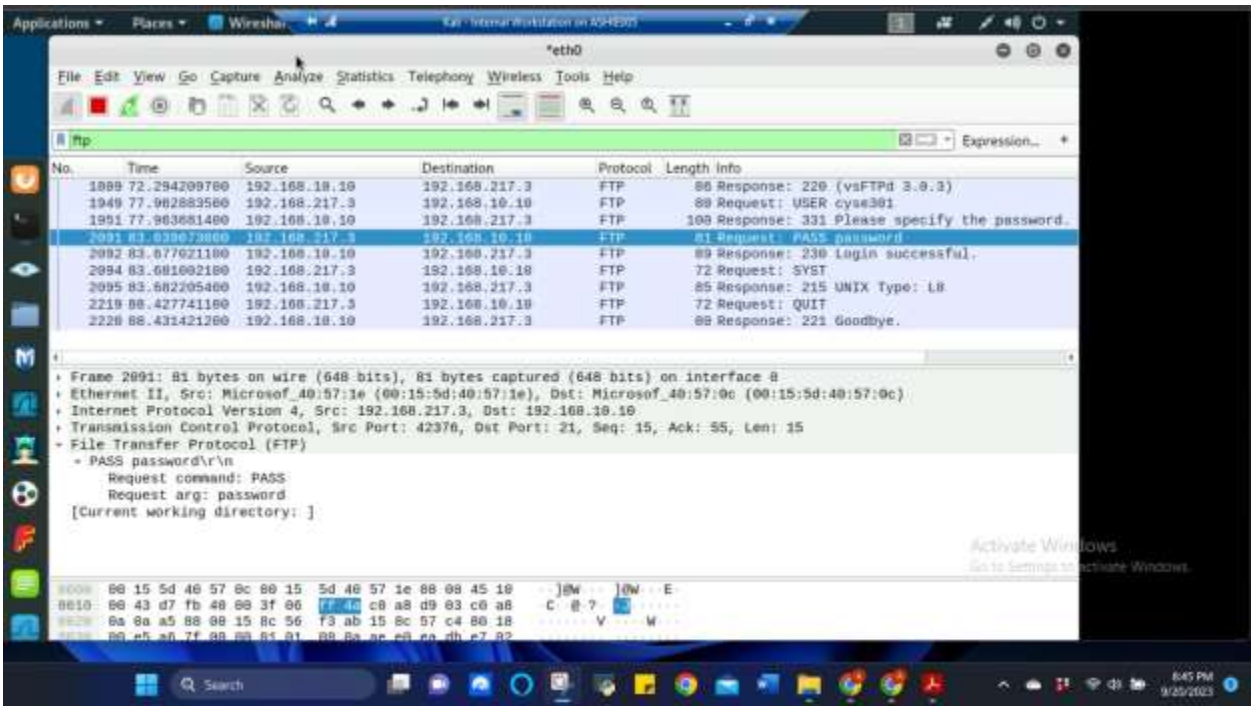

Figure 10 Screenshot of Wireshark from Internal Kali showing the "FTP" traffic by using the "FTP" filter for Task B.2b

The above screenshot shows that when you filter just the FTP traffic in Wireshark on the Internal Kali, you are able to see the entire FTP interaction between External Kali and Ubuntu. This screenshot shows that the username used was "cyse301" 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 reaccess 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.

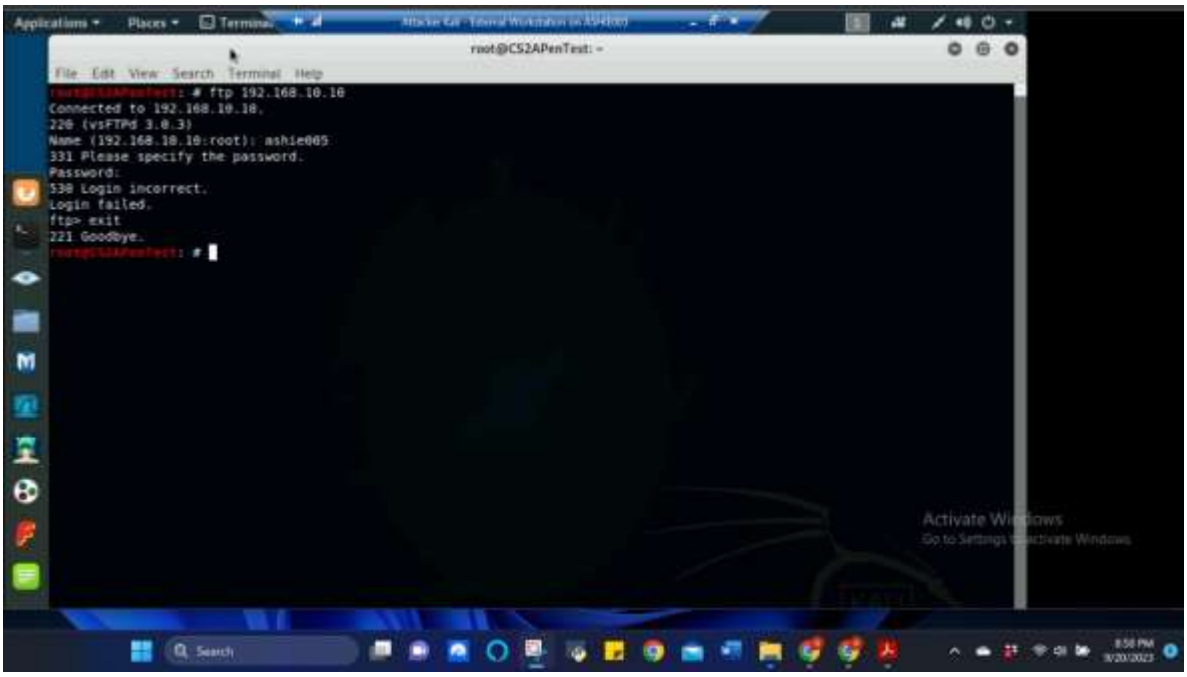

Figure 11 Screenshot of accessing the Ubuntu FTP Server again using External Kali. This shows the ftp command in the External Kali terminal to access the FTP server in Ubuntu for Task B.2c

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.

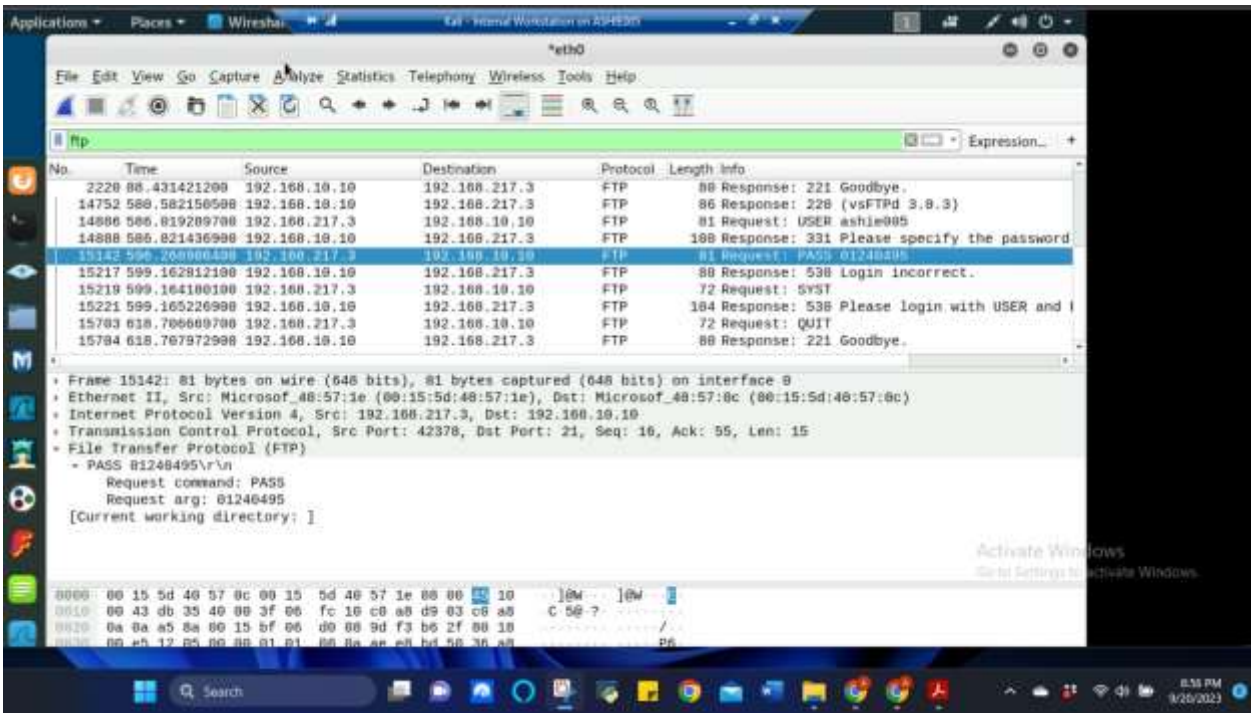

Figure 12 Screenshot of Wireshark from Internal Kali showing the "FTP" traffic by using the "FTP" filter for Task B.2c

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 and my UIN as my username. It also shows that the login was incorrect as well.

TASK C: EXTRA CREDIT: STEAL FILES WITH WIRESHARK (15 POINTS)

Login to Ubuntu VM, and create a file in your home directory, named “YOUR_MIDAS.txt”. Put the current timestamp and your name in the file.

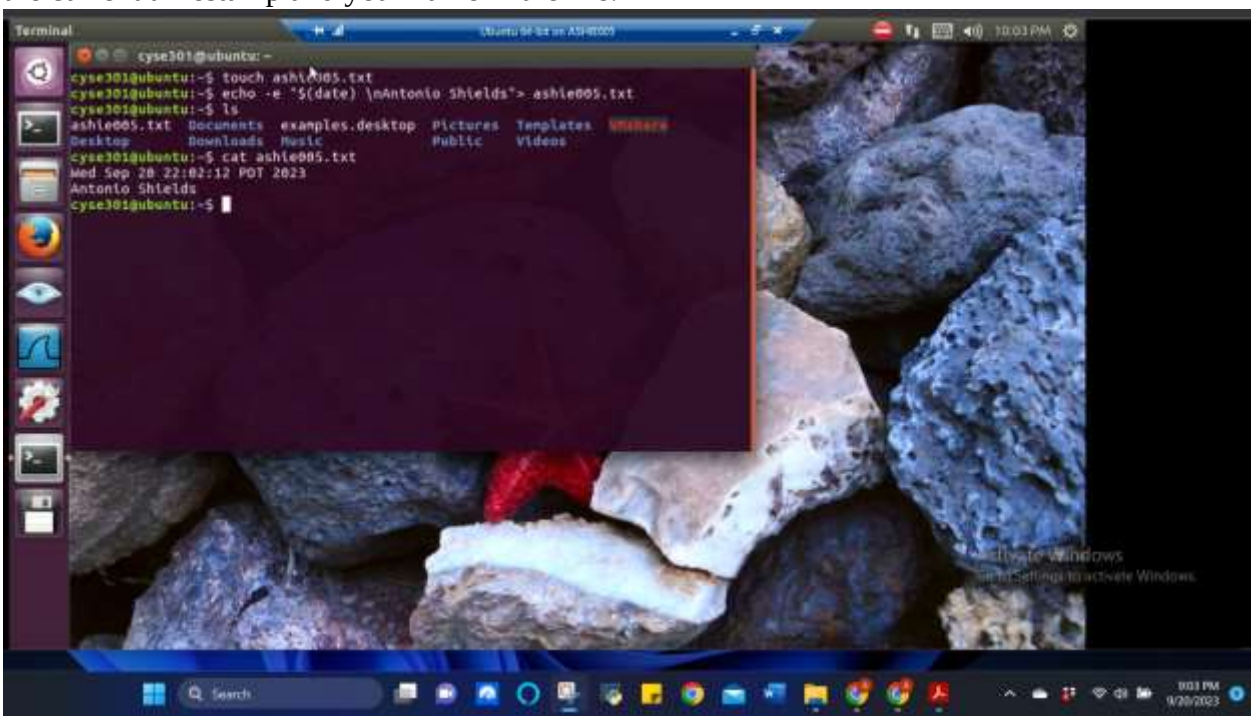

Figure 13 Screenshot of the terminal in Ubuntu VM and creating a file and inputting the required information for Task EC

The above screenshot shows creating a file using the touch command using my MIDAS ID as the title and also entering the current timestamp and name in the file. The ls command was used to ensure the file existed in the directory and the cat command was used to verify the information in the .txt file.

Once you have the file ready in Ubuntu, switch back to External Kali. Get the file you just created with FTP protocol remotely.

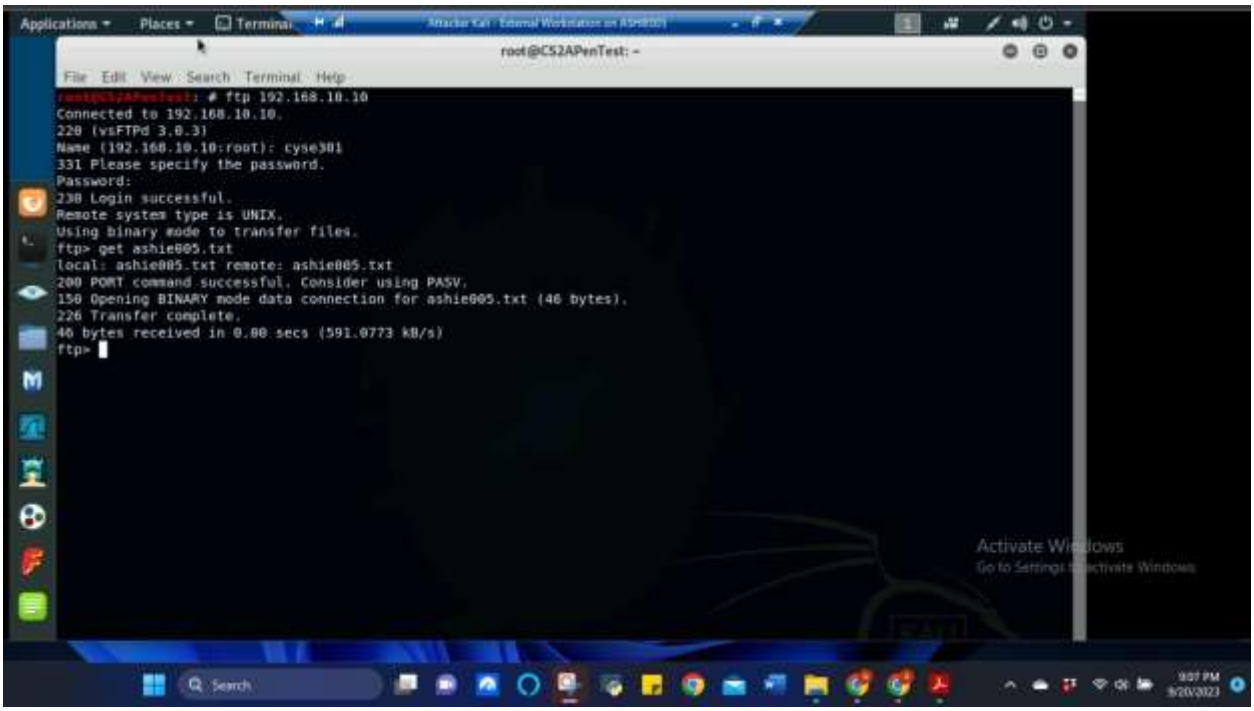

Figure 14 Screenshot of the "FTP" command being used in External Kali to access the FTP server in Ubuntu VM and the created file being transferred to Ubuntu for Task EC.

The above screenshot shows the "FTP" command being used in the External Kali terminal to connect to the FTP server in Ubuntu using the correct credentials. Once successfully logged in, the created file that made in the previous step was transferred and shows that the transfer was completed.

As an attacker, you need to complete the following tasks in Internal Kali:

1. Apply a proper display filter to display the FTP-DATA packets between External Kali and Ubuntu VM.

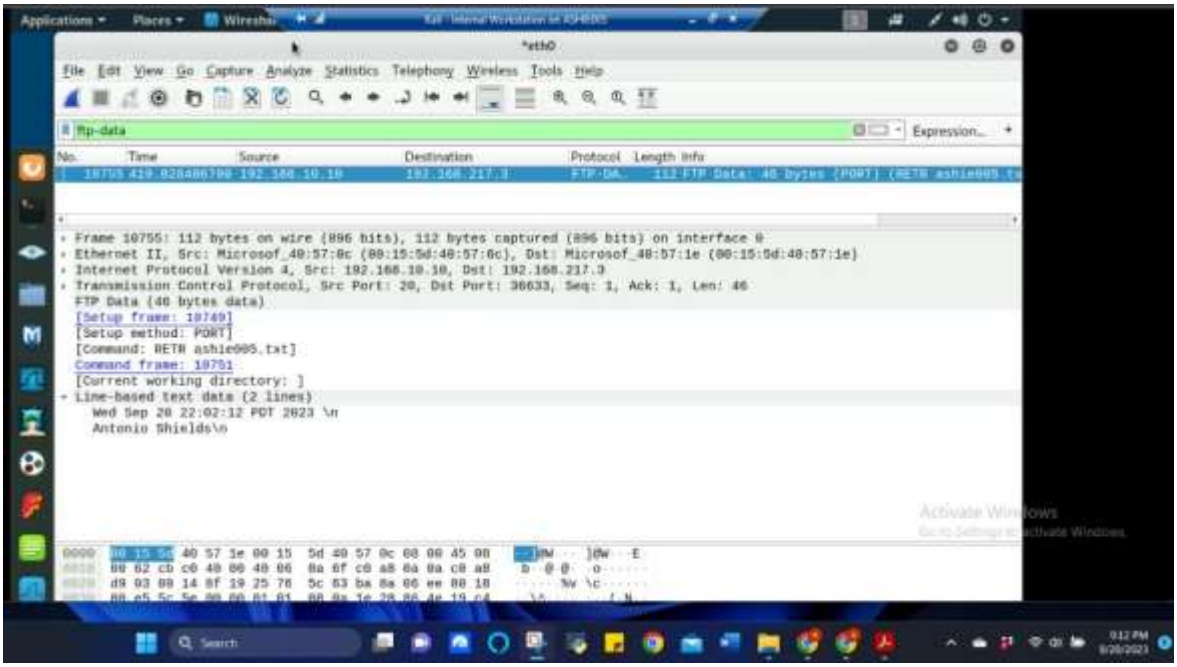

Figure 15 Screenshot of Wireshark from Internal Kali showing the "FTP-data" traffic by using the "FTP-data" filter for Task EC-1

The above screenshot shows the "FTP-DATA" filter being applied to retrieve the packets between External Kali and Ubuntu VM.

2. Follow the tcp steam of the FTP-DATA packet, and view the content of the file just transferred.

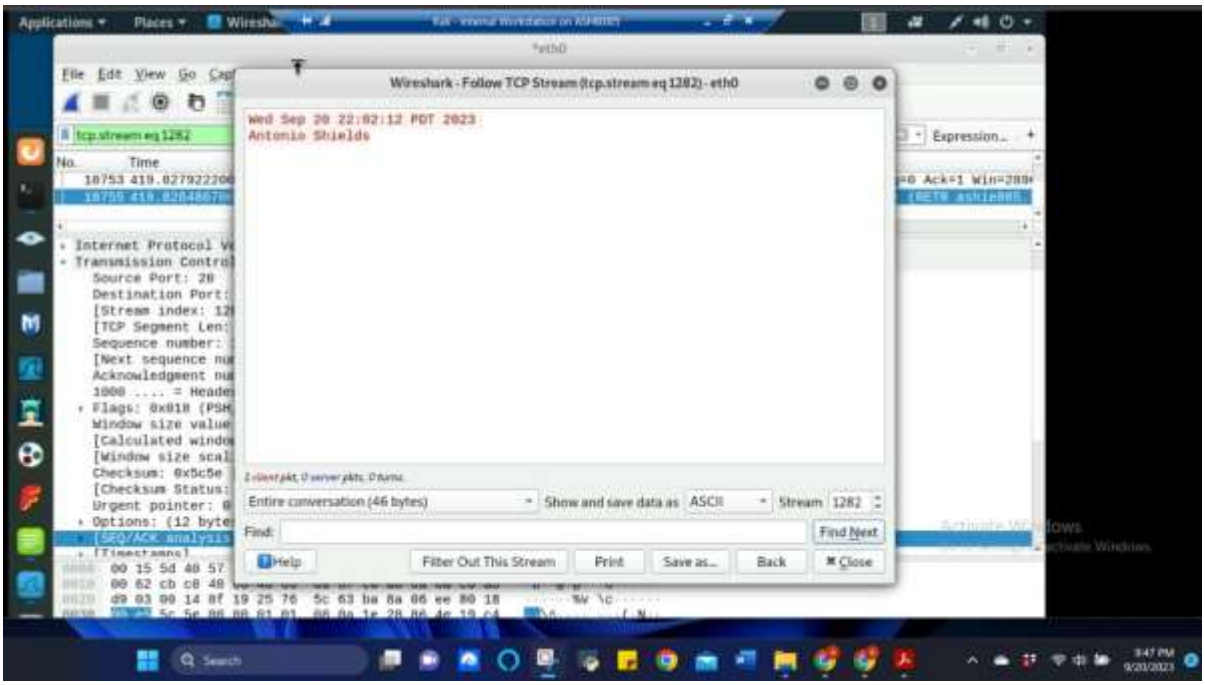

Figure 16 Screenshot of Wireshark from Internal Kali showing the TCP stream of the FTP-data packet and the contents of the file Task EC-2

The above screenshot shows that by going to “Analyze” on the menu bar, then go to “Follow” and then select “TCP Stream” the contents of the file selected will appear as shown. Using the “Save As” button at the bottom will allow you to save the contents as a .txt file in the place of your choice.

3. Export (Save) the transferred file as a text file in Internal Kali, and view the content.

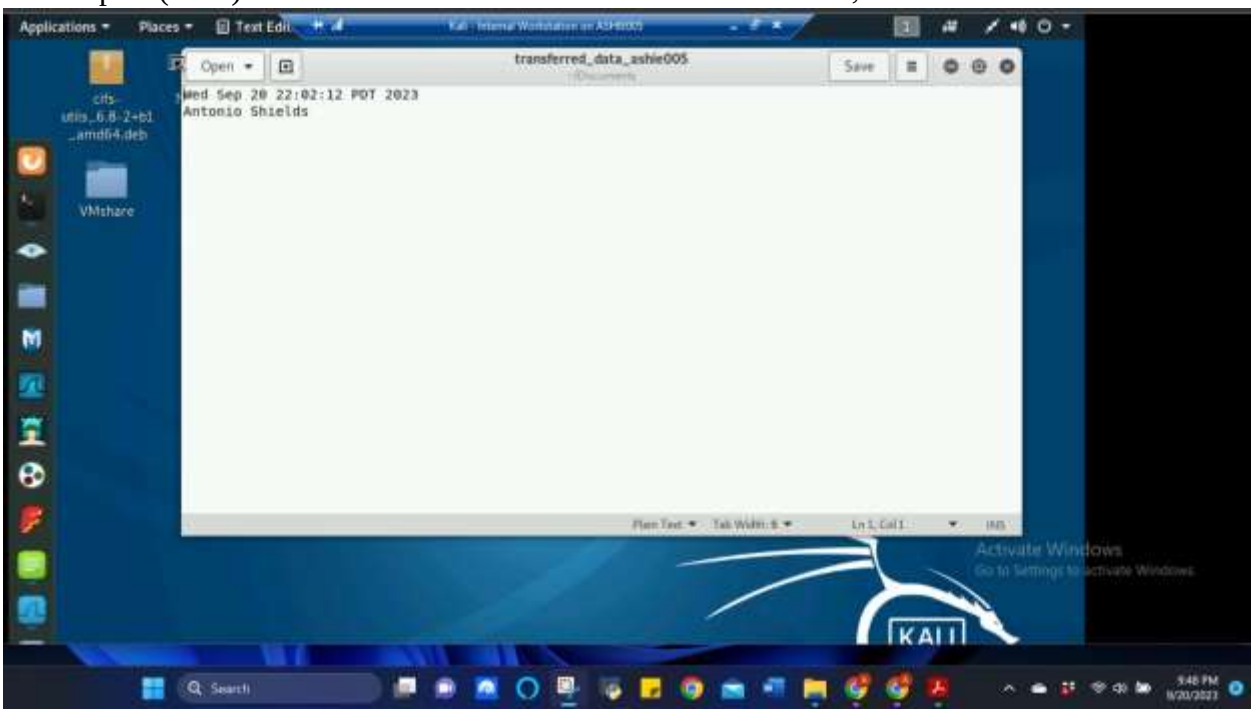

Figure 17 Screenshot of the .txt file opened in text editor to view the content for Task EC-3

The above screenshot shows the results of saving the tcp stream content as a .txt file and opening it up using text editor to verify successful saving and execution.