# Module 08 Sniffing

## Lab 1: Perform Active Sniffing

**Lab Scenario**

As a professional ethical hacker or pen tester, the first step is to perform active sniffing on the target network using various active sniffing techniques such as MAC flooding, DHCP starvation, ARP poisoning, or MITM. In active sniffing, the switched Ethernet does not transmit information to all systems connected through the LAN as it does in a hub-based network.

In active sniffing, ARP traffic is actively injected into a LAN to sniff around a switched network and capture its traffic. A packet sniffer can obtain all the information visible on the network and records it for future review. A pen tester can see all the information in the packet, including data that should remain hidden.

An ethical hacker or pen tester needs to ensure that the organization's network is secure from various active sniffing attacks by analyzing incoming and outgoing packets for any attacks.

**Lab Objectives**

- Perform MAC flooding using macof
- Perform a DHCP starvation attack using Yersinia

**Overview of Active Sniffing**

Active sniffing involves sending out multiple network probes to identify access points. The following is the list of different active sniffing techniques:

- **MAC Flooding**: Involves flooding the CAM table with fake MAC address and IP pairs until it is full
- **DNS Poisoning**: Involves tricking a DNS server into believing that it has received authentic information when, in reality, it has not
- **ARP Poisoning**: Involves constructing a large number of forged ARP request and reply packets to overload a switch
- **DHCP Attacks**: Involves performing a DHCP starvation attack and a rogue DHCP server attack
- **Switch port stealing**: Involves flooding the switch with forged gratuitous ARP packets with the target MAC address as the source
- **Spoofing Attack**: Involves performing MAC spoofing, VLAN hopping, and STP attacks to steal sensitive information

# Task 1: Perform MAC Flooding using macof

MAC flooding is a technique used to compromise the security of network switches that connect network segments or network devices. Attackers use the MAC flooding technique to force a switch to act as a hub, so they can easily sniff the traffic.

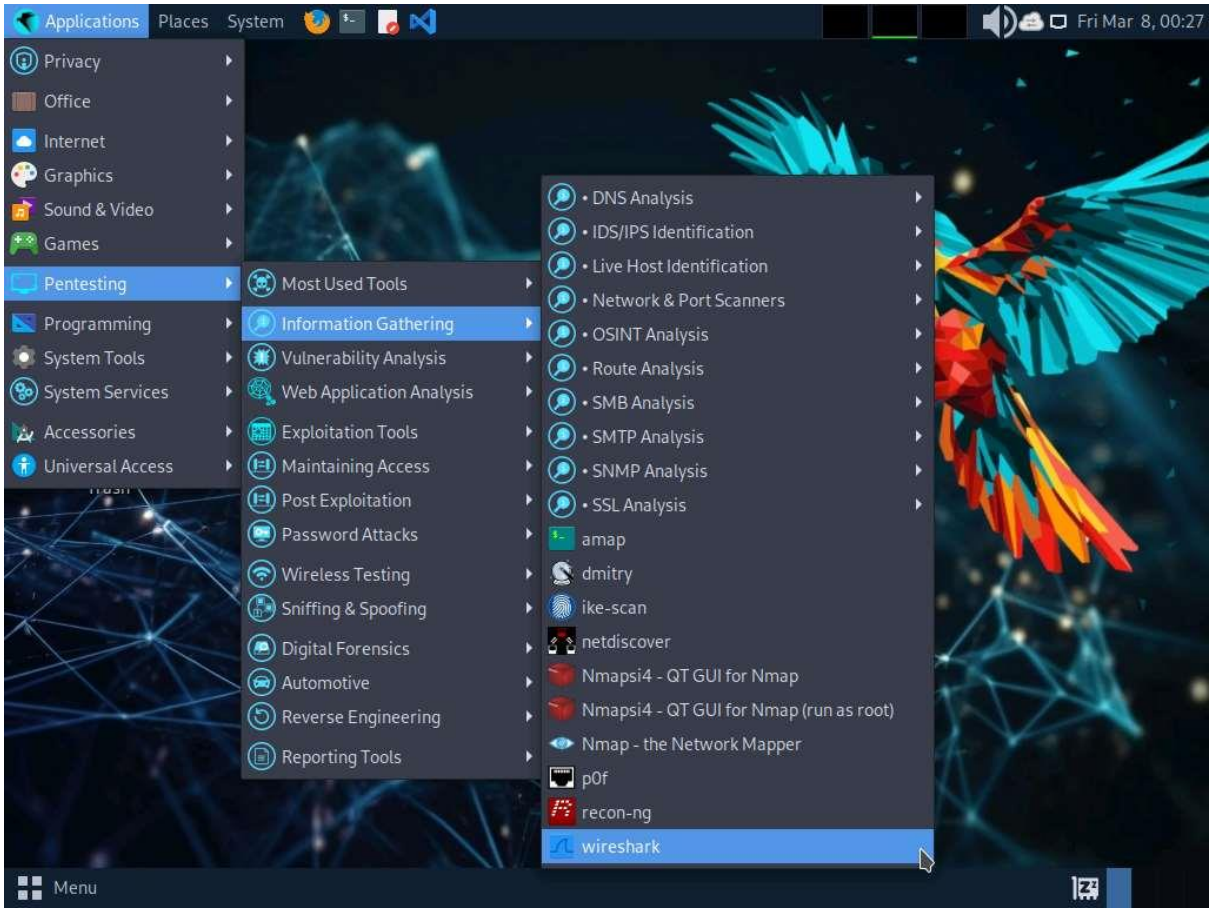
macof is a Unix and Linux tool that is a part of the dsniff collection. It floods the local network with random MAC addresses and IP addresses, causing some switches to fail and open in repeating mode, thereby facilitating sniffing. This tool floods the switch's CAM tables (131,000 per minute) by sending forged MAC entries. When the MAC table fills up, the switch converts to a hub-like operation where an attacker can monitor the data being broadcast.

Here, we will use the macof tool to perform MAC flooding.

1. By default Windows 11 machine selected, to launch **Parrot Security** machine, click Parrot Security and login with **attacker/toor**.

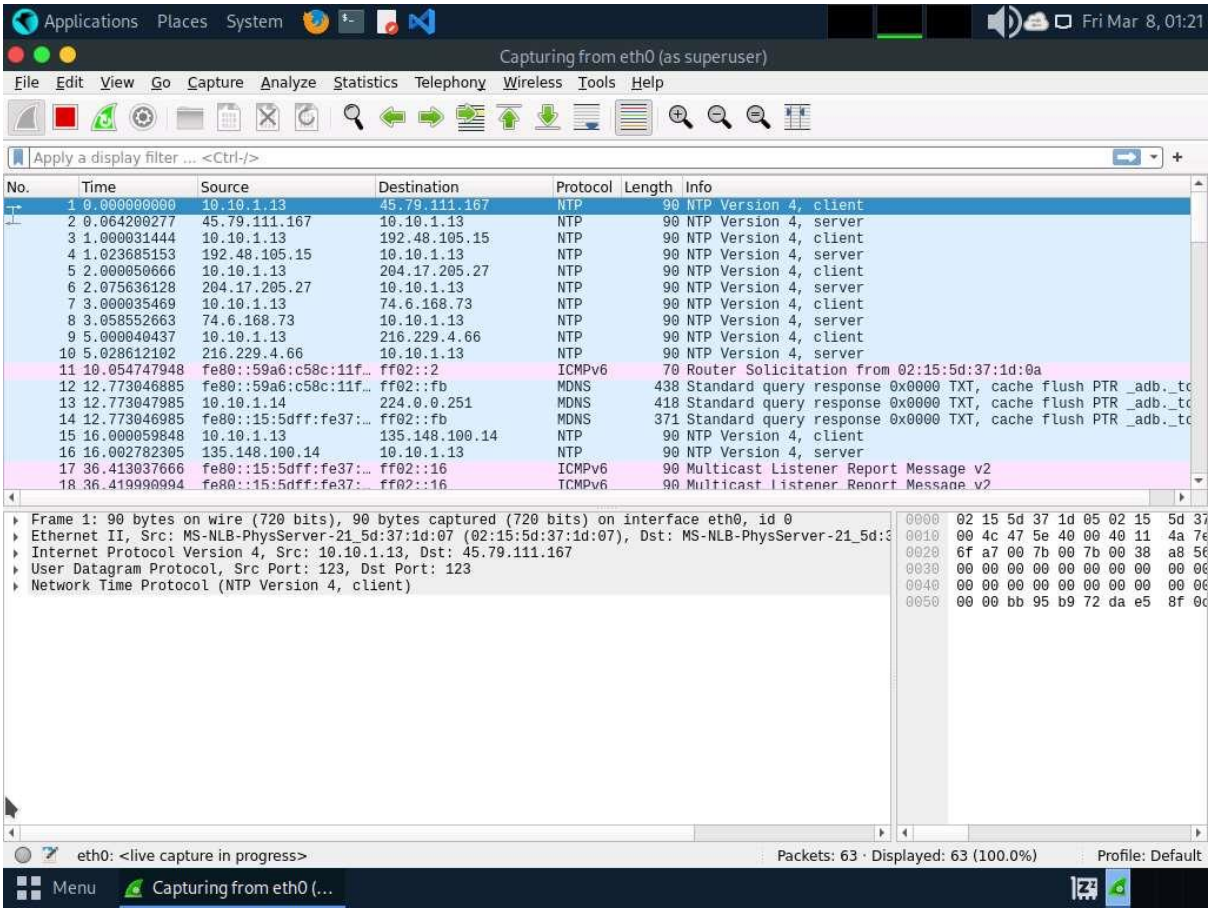
   If a **Parrot Updater** pop-up appears at the top-right corner of **Desktop**, ignore and close it.
   If a **Question** pop-up window appears asking you to update the machine, click **No** to close the window.

2. Click **Applications** in the top-left corner of **Desktop** and navigate to **Pentesting** --> **Information Gathering** --> **wireshark**.




3. A security pop-up appears, authenticate by providing **toor** as a password.

4. **Wireshark Network Analyzer** window appears, start capturing the network traffic on the primary network interface (here, **eth0**).

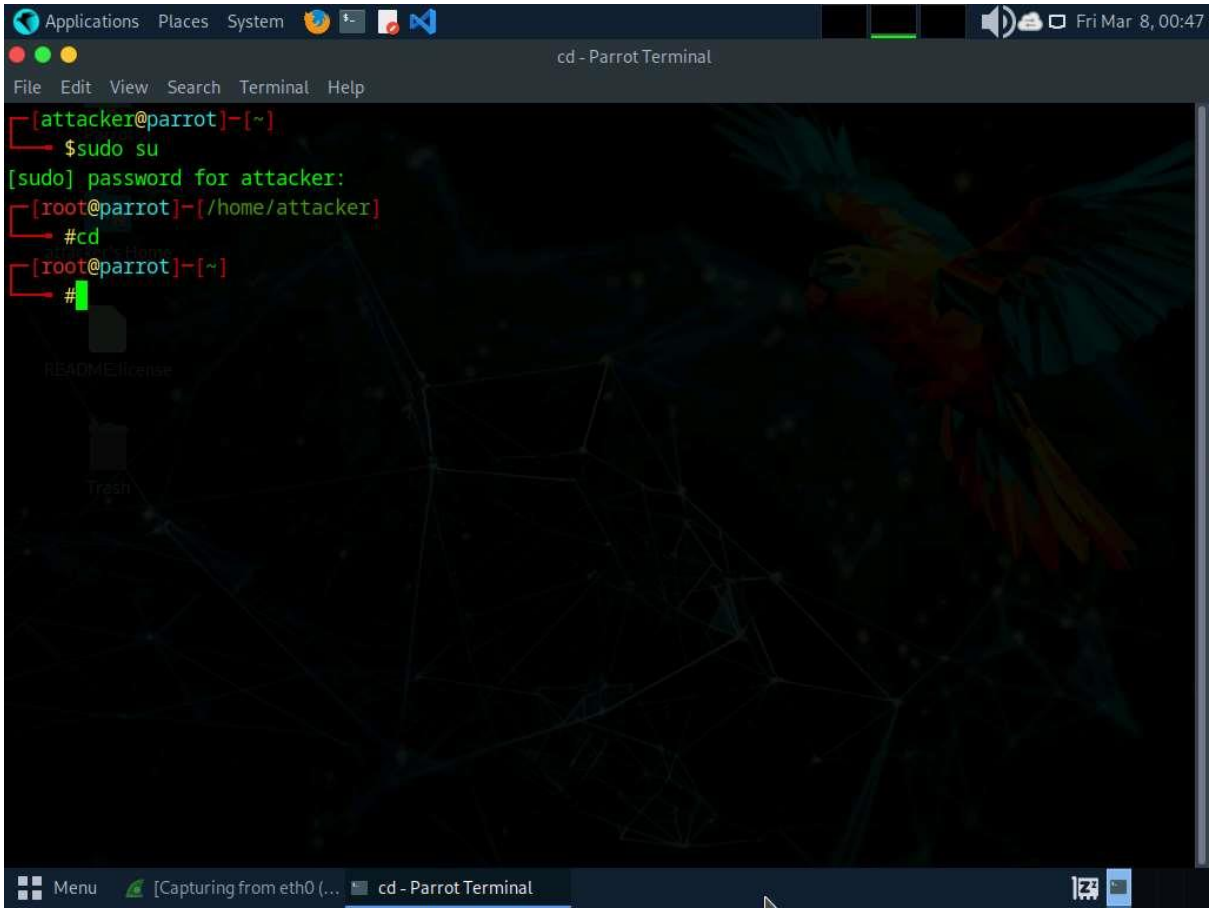



5. Leave the **Wireshark** application running.

6. Open a **Terminal** window and execute **sudo su** to run the programs as a root user (When prompted, enter the password **toor**).

   The password that you type will not be visible.

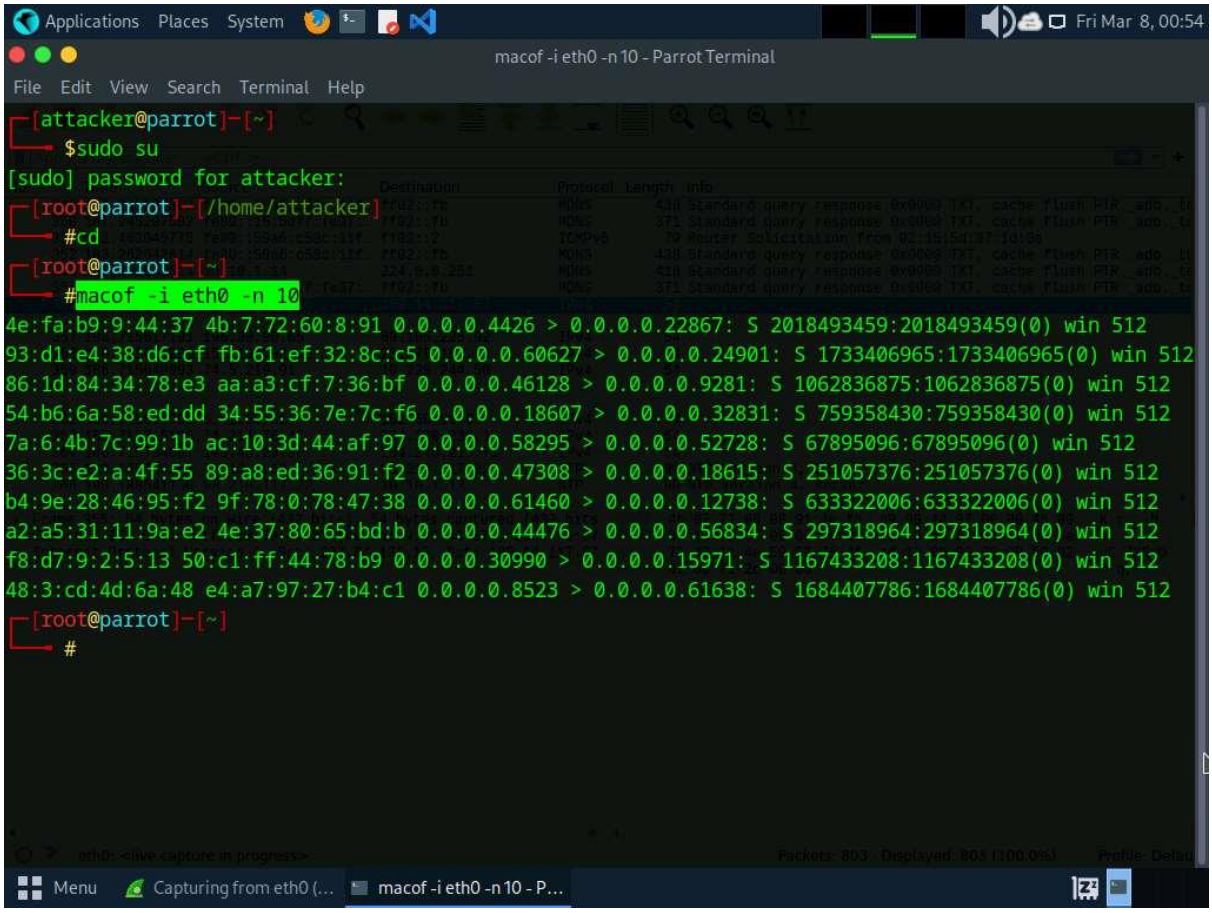
7. Now, run **cd** command to jump to the root directory.




8. Execute **macof -i eth0 -n 10** in the root directory.

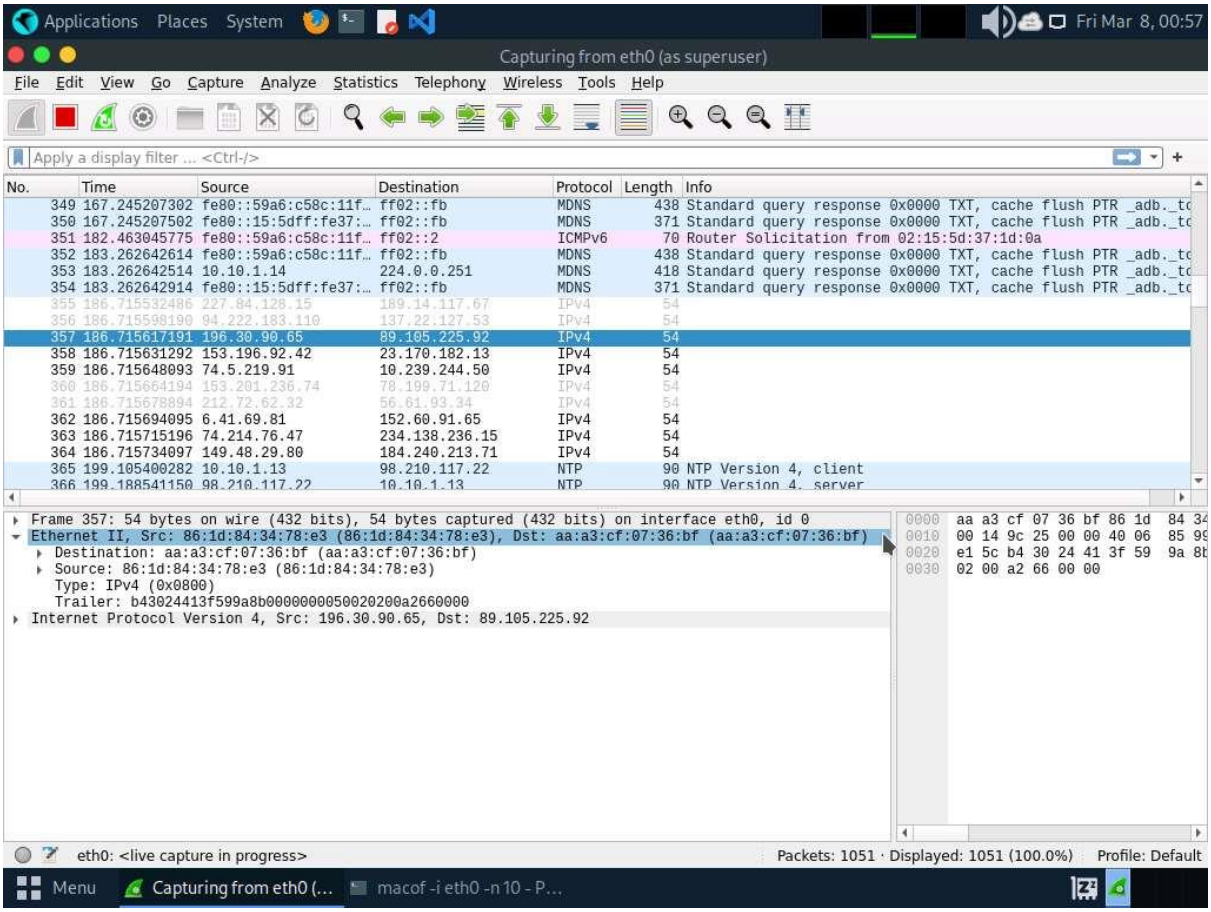
   **-i**: specifies the interface and **-n**: specifies the number of packets to be sent (here, **10**).
   You can also target a single system by issuing the command **macof -i eth0 -d [Target IP Address]** (**-d**: Specifies the destination IP address).

9. This command will start flooding the CAM table with random MAC addresses, as shown in the screenshot.




10. Switch to the **Wireshark** window and observe the **IPv4** packets from random IP addresses.

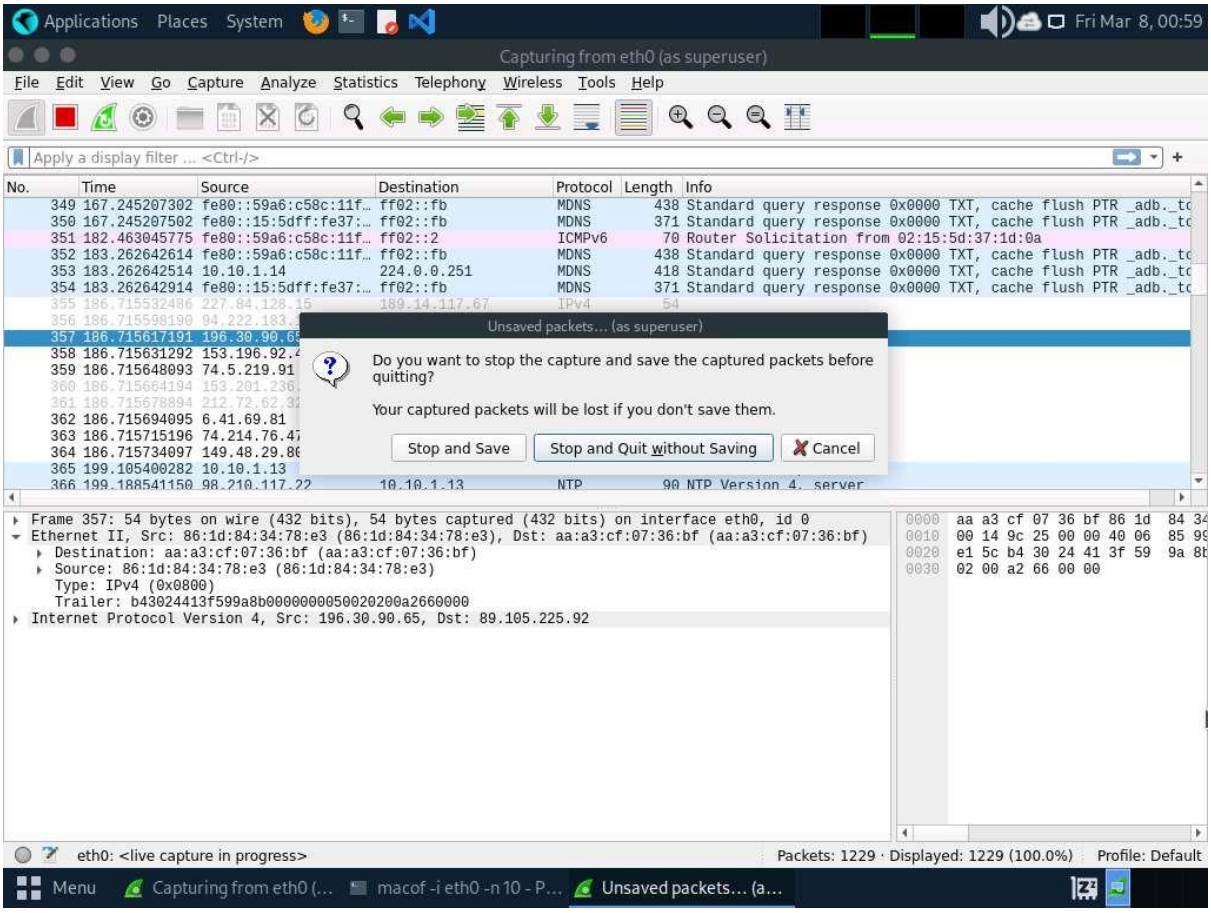
11. Click on any captured **IPv4** packet and expand the **Ethernet II** node in the packet details section. Information regarding the source and destination MAC addresses is displayed, as shown in the screenshot.




12. Similarly, you can switch to a different machine to see the same packets that were captured by Wireshark in the **Parrot Security** machine.

13. Macof sends the packets with random MAC and IP addresses to all active machines in the local network. If you are using multiple targets, you will observe the same packets on all target machines.

14. Close the **Wireshark** window. If an **Unsaved packets**… pop-up appears, click **Stop and Quit without Saving** to close the Wireshark application.




15. This concludes the demonstration of how to perform MAC flooding using macof.

16. Close all open windows and document all the acquired information.

**Question 8.1.1.1**

Use macof on the Parrot Security machine to perform MAC flooding on the Windows 11 target machine. What is the default size of the IP packets that macof uses to flood the CAM table with random MAC addresses?

## Task 2: Perform a DHCP Starvation Attack using Yersinia

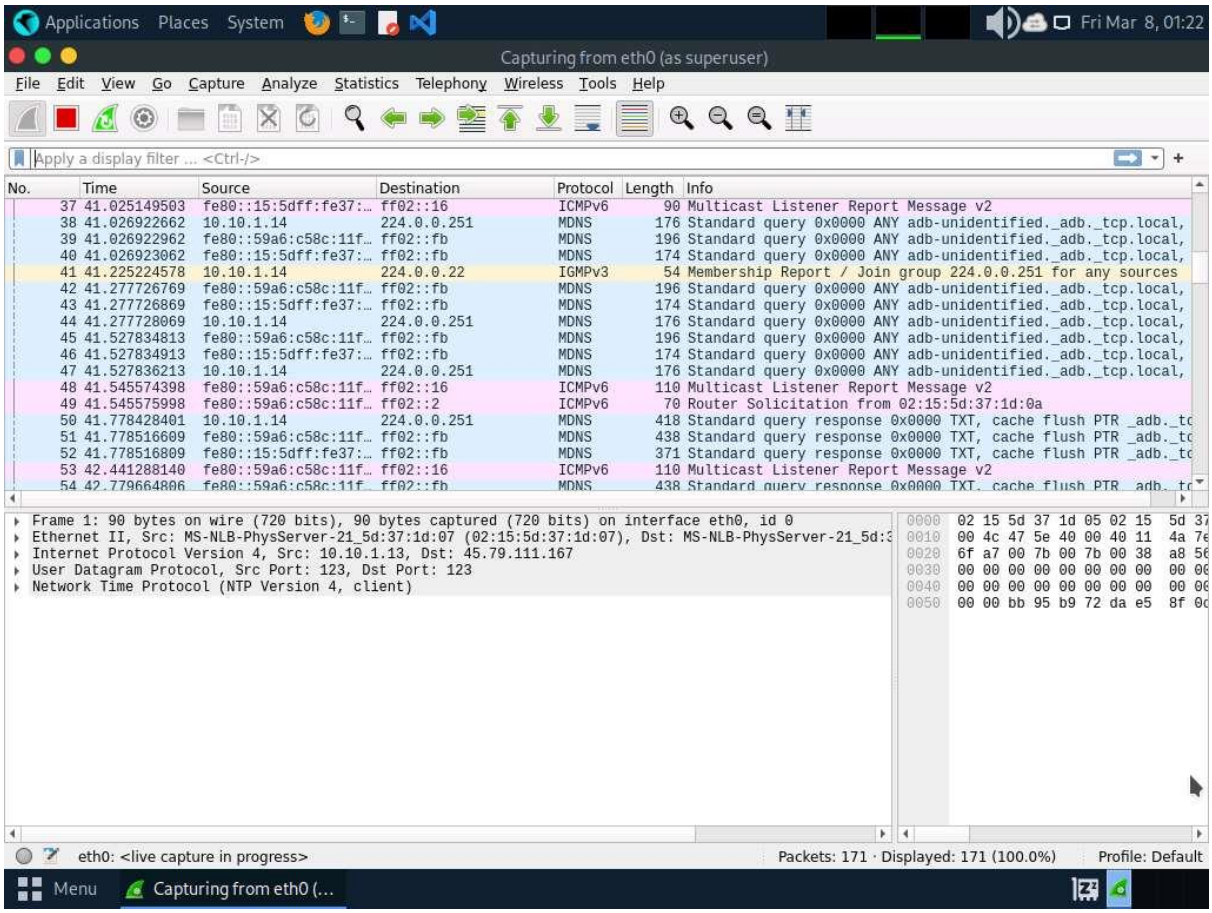
In a DHCP starvation attack, an attacker floods the DHCP server by sending a large number of DHCP requests and uses all available IP addresses that the DHCP server can issue. As a result, the server cannot issue any more IP addresses, leading to a Denial-of-Service (DoS) attack. Because of this issue, valid users cannot obtain or renew their IP addresses, and thus

fail to access their network. This attack can be performed by using various tools such as Yersinia and Hyenae.

Yersinia is a network tool designed to take advantage of weaknesses in different network protocols such as DHCP. It pretends to be a solid framework for analyzing and testing the deployed networks and systems.

Here, we will use the Yersinia tool to perform a DHCP starvation attack on the target system.

1. In **Parrot Security** machine, launch **Wireshark** and start packet capturing on available ethernet or interface (here,**eth0**).



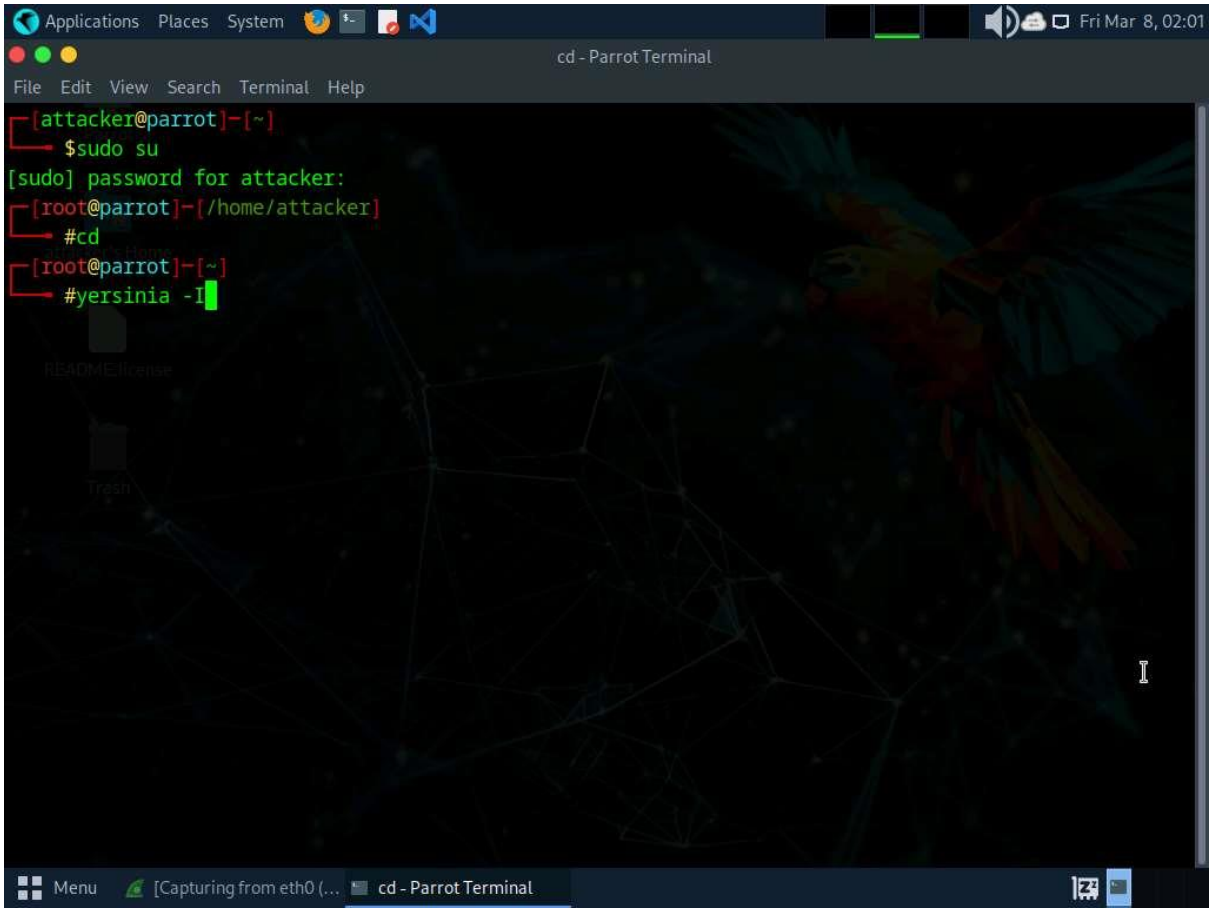

2. Leave the **Wireshark** application running.

3. Open a **Terminal** window and execute **sudo su** to run the programs as a root user (When prompted, enter the password **toor**). Run **cd** to navigate to the root directory.

   Click the **Maximize Window** icon to maximize the terminal window.
   The interactive mode of the Yersinia application only works in a maximized terminal window.

4. Run **yersinia -I** to open Yersinia in interactive mode.

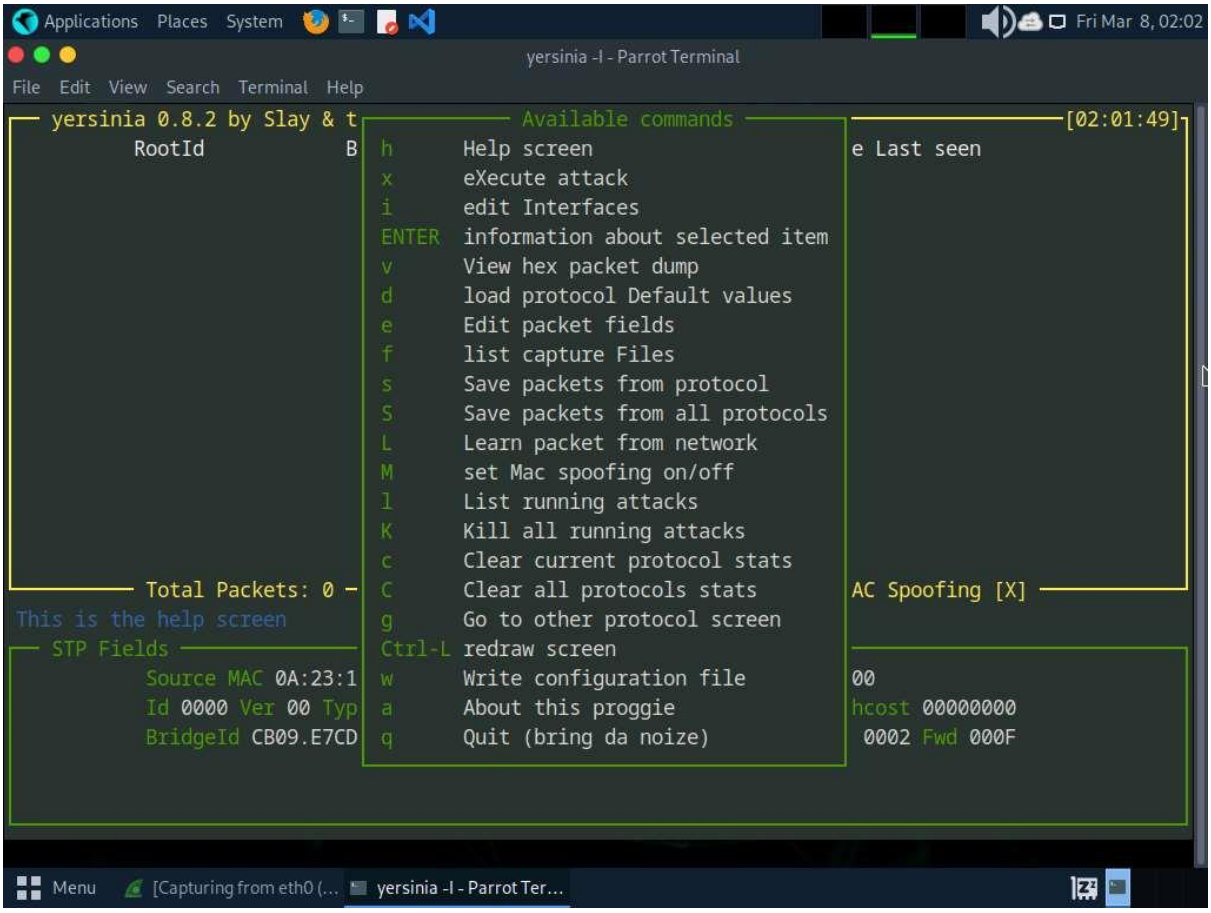
**-I**: Starts an interactive session.




5. Yersinia interactive mode appears in the terminal window.

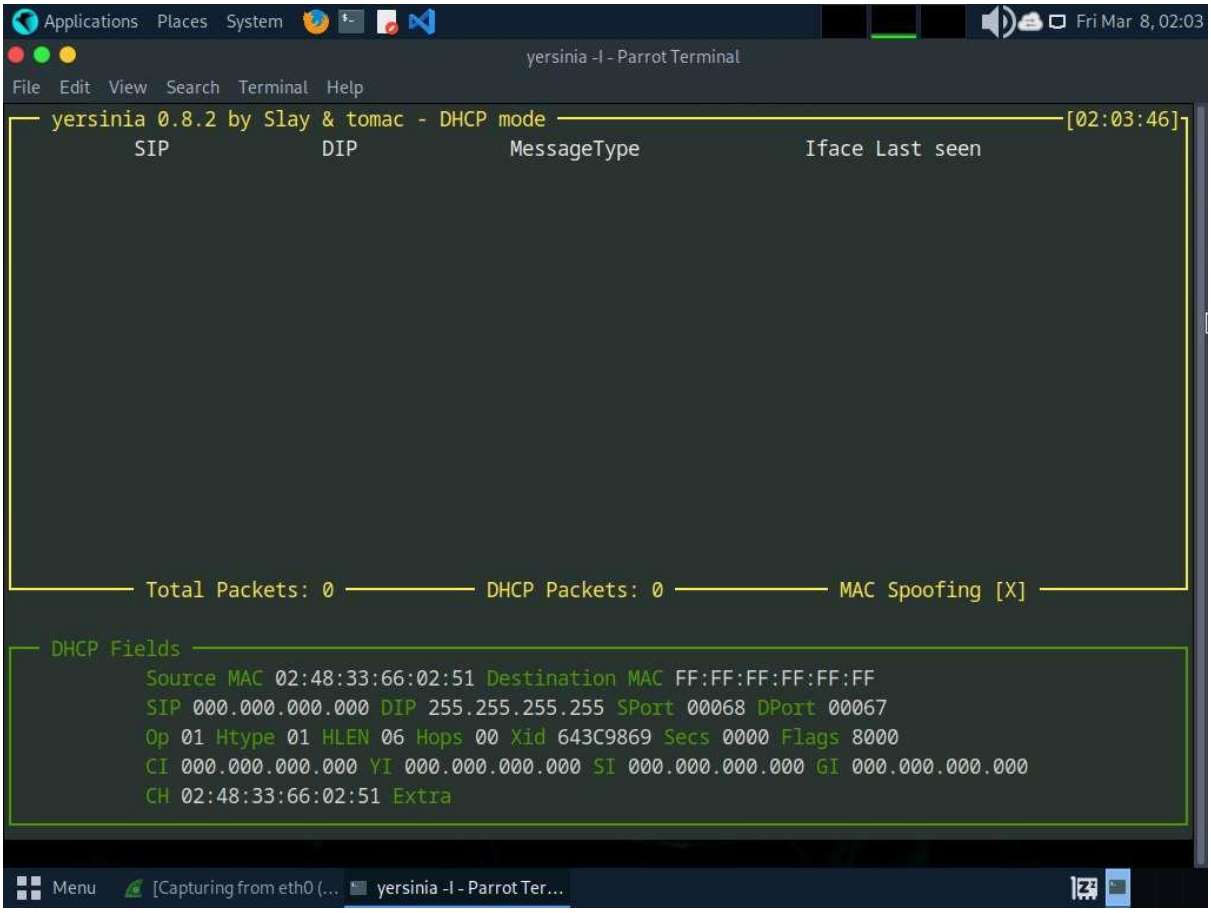
6. To remove the **Notification window**, press any key, and then press **h** for help.

7. The **Available commands** option appears, as shown in the screenshot.



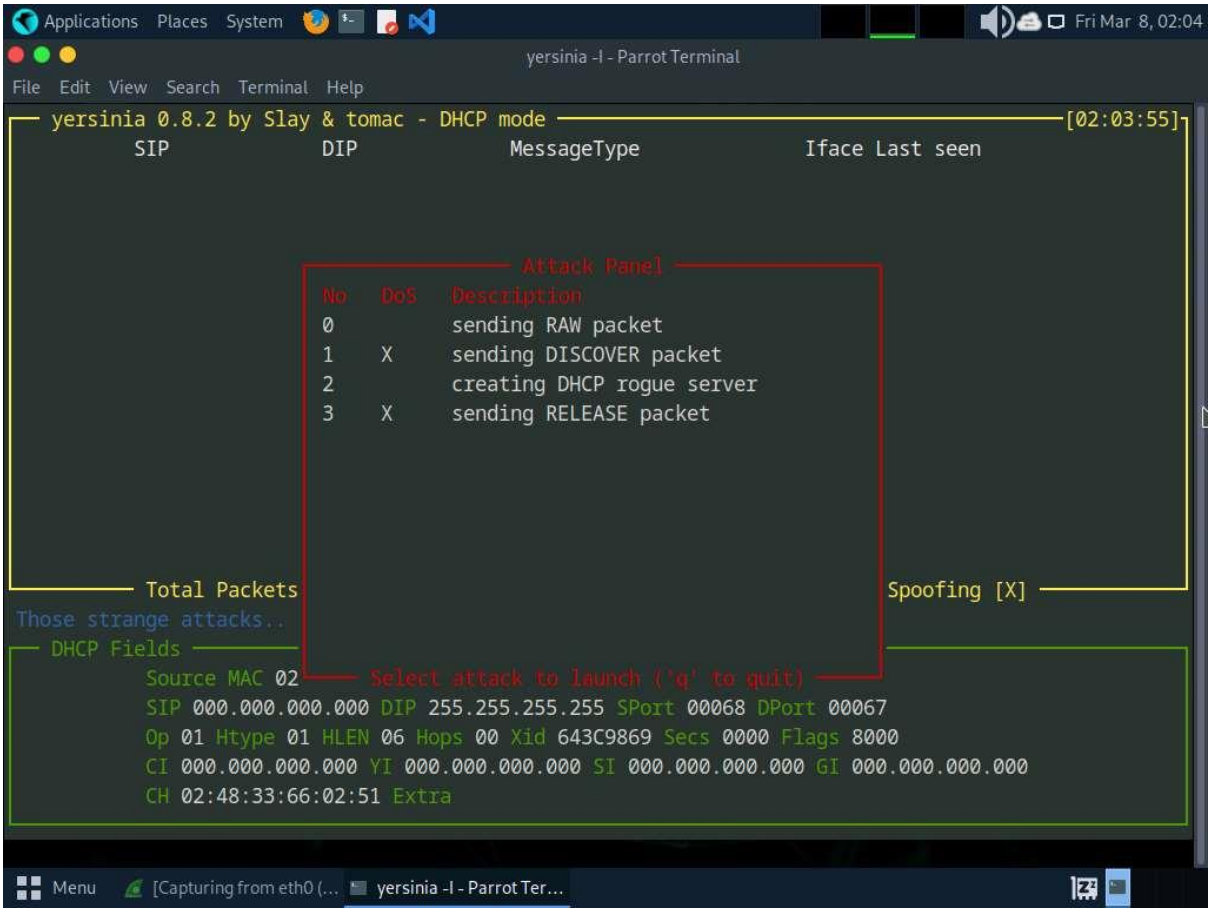

8. Press **q** to exit the help options.

9. Press **F2** to select DHCP mode. In DHCP mode, **STP Fields** in the lower section of the window change to **DHCP Fields**, as shown in the screenshot.

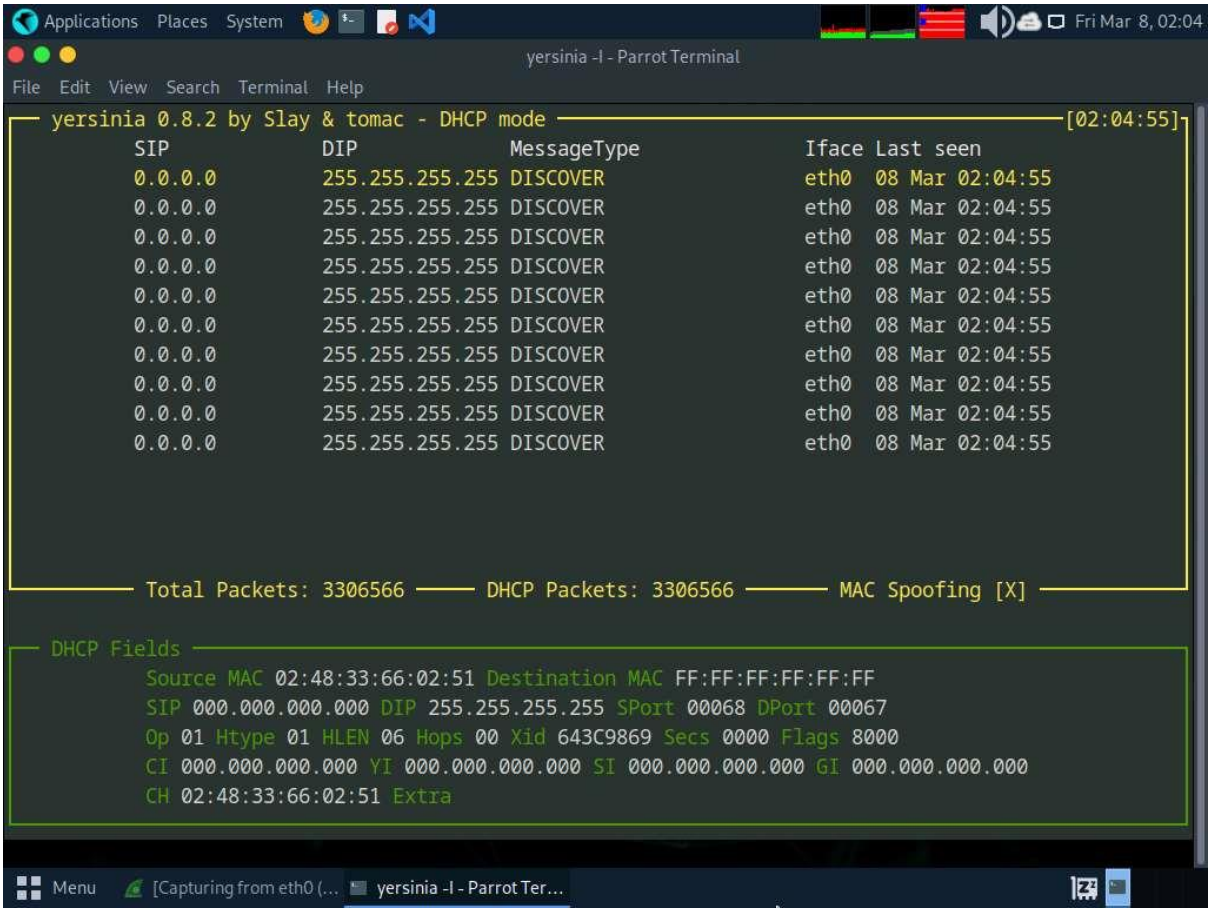



10. Press **x** to list available attack options.

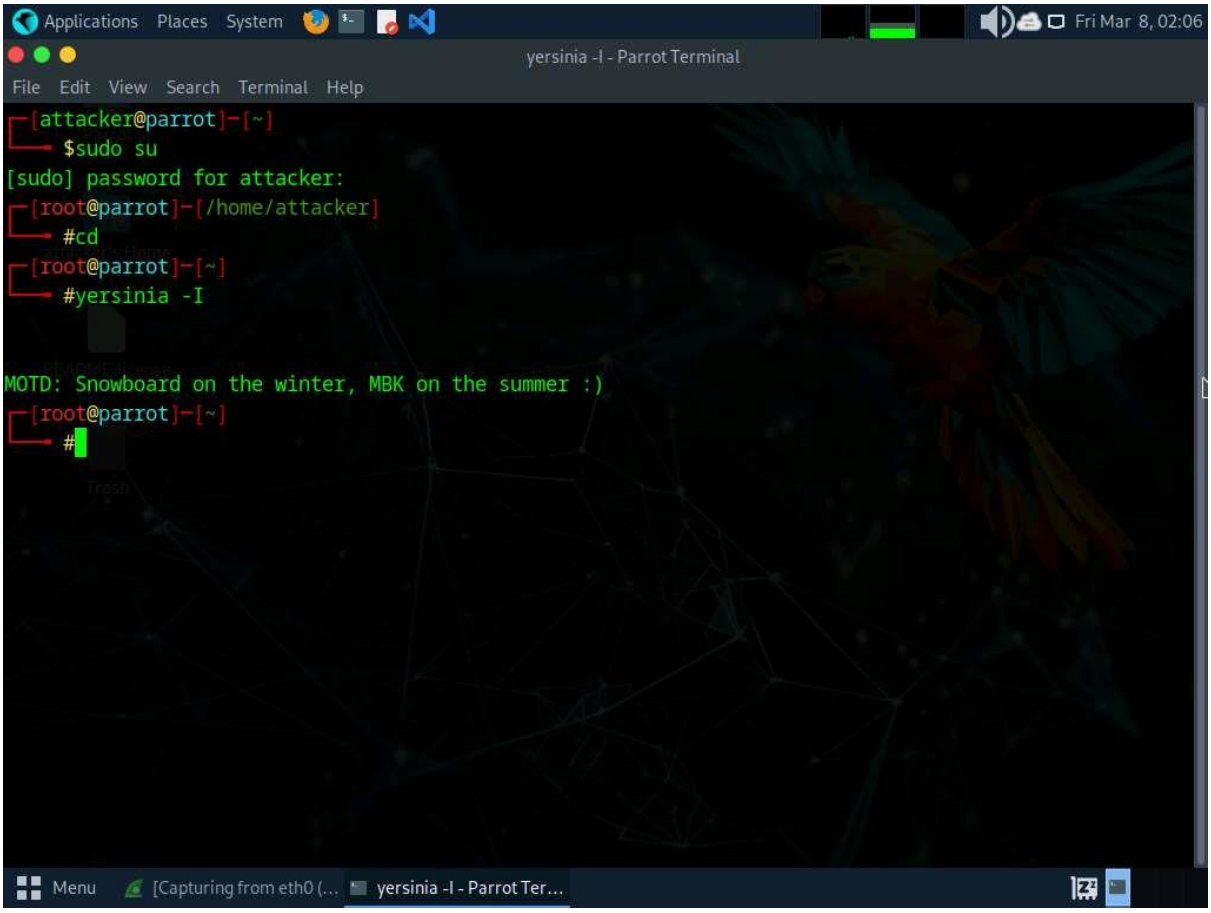
11. The **Attack Panel** window appears; press **1** to start a DHCP starvation attack.



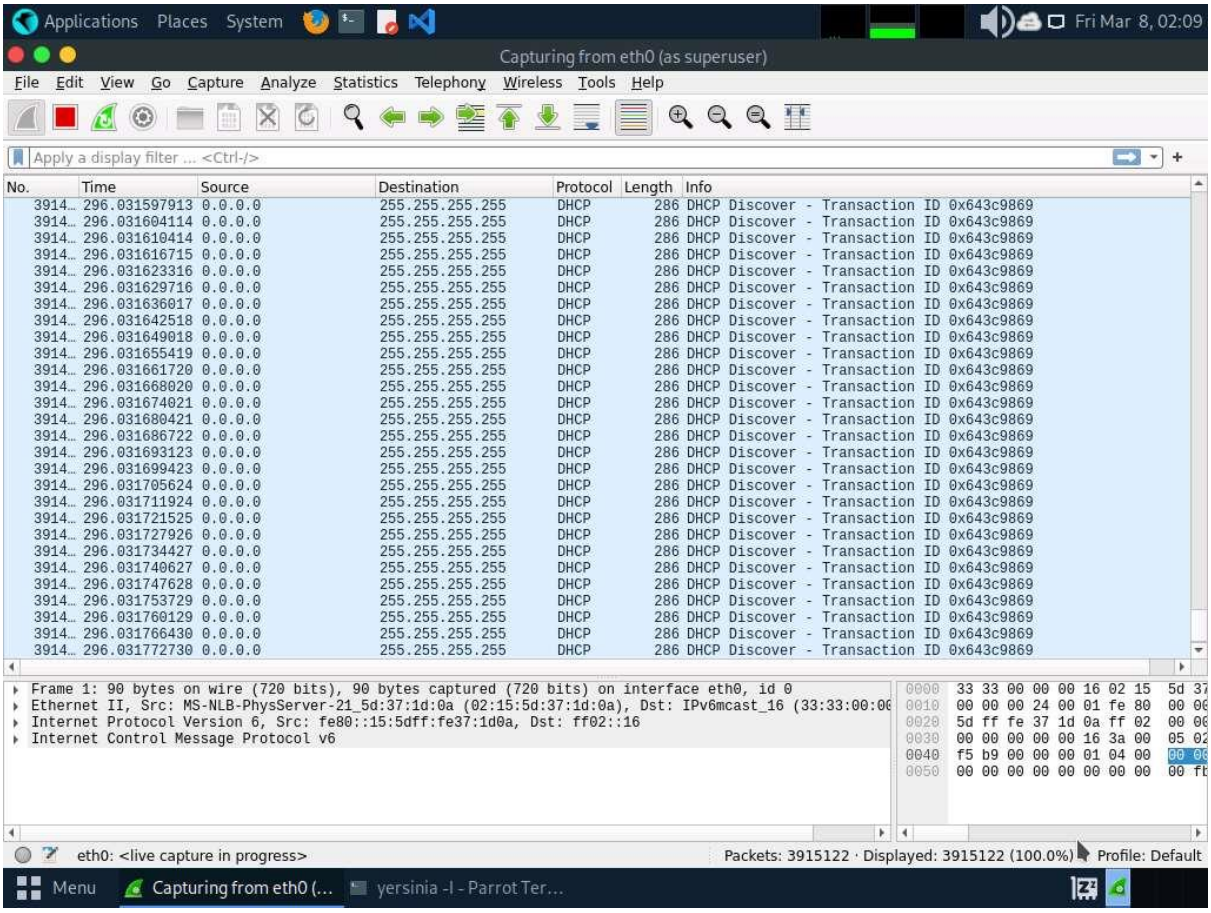

12. **Yersinia** starts sending DHCP packets to the network interface as shown in the screenshot.



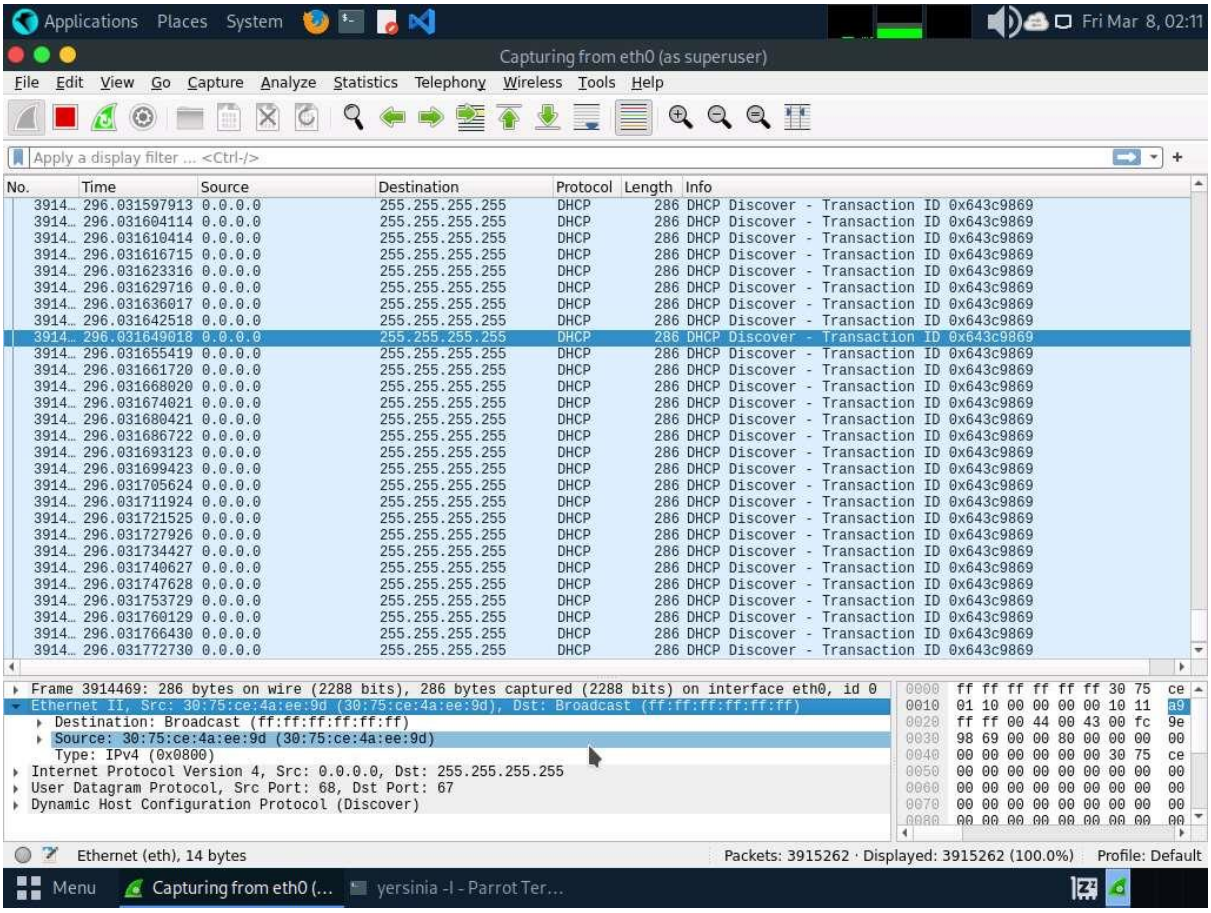

13. After a few seconds, press **q** to stop the attack and terminate Yersinia, as shown in the screenshot.




14. Now, switch to the **Wireshark** window and observe the huge number of captured **DHCP** packets, as shown in the screenshot.




15. Click on any DHCP packet and expand the **Ethernet II** node in the packet details section. Information regarding the source and destination MAC addresses is displayed, as shown in the screenshot.




16. Close the **Wireshark** window. If an **Unsaved packets…** pop-up appears, click **Stop and Quit without Saving**.

17. This concludes the demonstration of how to perform a DHCP starvation attack using Yersinia.

18. Close all open windows and document all the acquired information.

**Question 8.1.2.1**

Use Yersinia on the Parrot Security machine to perform a DHCP starvation attack. What is the default source port used by Yersinia in the DHCP mode?

# Lab 2: Perform Network Sniffing using Various Sniffing Tools

**Lab Scenario**

Data traversing an HTTP channel flows in plain-text format and is therefore prone to MITM attacks. Network administrators can use sniffers for helpful purposes such as to troubleshoot network problems, examine security problems, and debug protocol implementations. However, an attacker can use sniffing tools such as Wireshark to sniff the traffic flowing between the client and the server. The traffic obtained by the attacker might contain sensitive information such as login credentials, which can then be used to perform malicious activities such as user-session impersonation.

An attacker needs to manipulate the functionality of the switch to see all traffic passing through it. A packet sniffing program (also known as a sniffer) can only capture data packets from within a given subnet, which means that it cannot sniff packets from another network. Often, any laptop can plug into a network and gain access to it. Many enterprises leave their switch ports open. A packet sniffer placed on a network in promiscuous mode can capture and analyze all network traffic. Sniffing programs turn off the filter employed by Ethernet network interface cards (NICs) to prevent the host machine from seeing other stations' traffic. Thus, sniffing programs can see everyone's traffic.

The information gathered in the previous step may be insufficient to reveal the potential vulnerabilities of the target. There may be more information to help find loopholes in the target. An ethical hacker needs to perform network security assessments and suggest proper troubleshooting techniques to mitigate attacks. This lab provides hands-on experience of how to use sniffing tools to sniff network traffic and capture it on a remote interface.

**Lab Objectives**

- Perform password sniffing using Wireshark

**Overview of Network Sniffing Tools**

System administrators use automated tools to monitor their networks, but attackers misuse these tools to sniff network data. Network sniffing tools can be used to perform a detailed network analysis. When protecting a network, it is important to have as many details about the packet traffic as possible. By actively scanning the network, a threat hunter can stay vigilant and respond quickly to attacks.

## Task 1: Perform Password Sniffing using Wireshark

Wireshark is a network packet analyzer used to capture network packets and display packet data in detail. The tool uses Winpcap to capture packets on its own supported networks. It captures live network traffic from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, and FDDI networks. The captured files can be

programmatically edited via the command-line. A set of filters for customized data displays can be refined using a display filter.

Here, we will use the Wireshark tool to perform password sniffing.

In this task, we will use the **Windows Server 2019** (**10.10.1.19**) machine as the host machine and the **Windows 11** (**10.10.1.11**) machine as the target machine.

1. Click Windows Server 2019 to switch to the **Windows Server 2019** machine and login with **Administrator**/**Pa$$w0rd**.

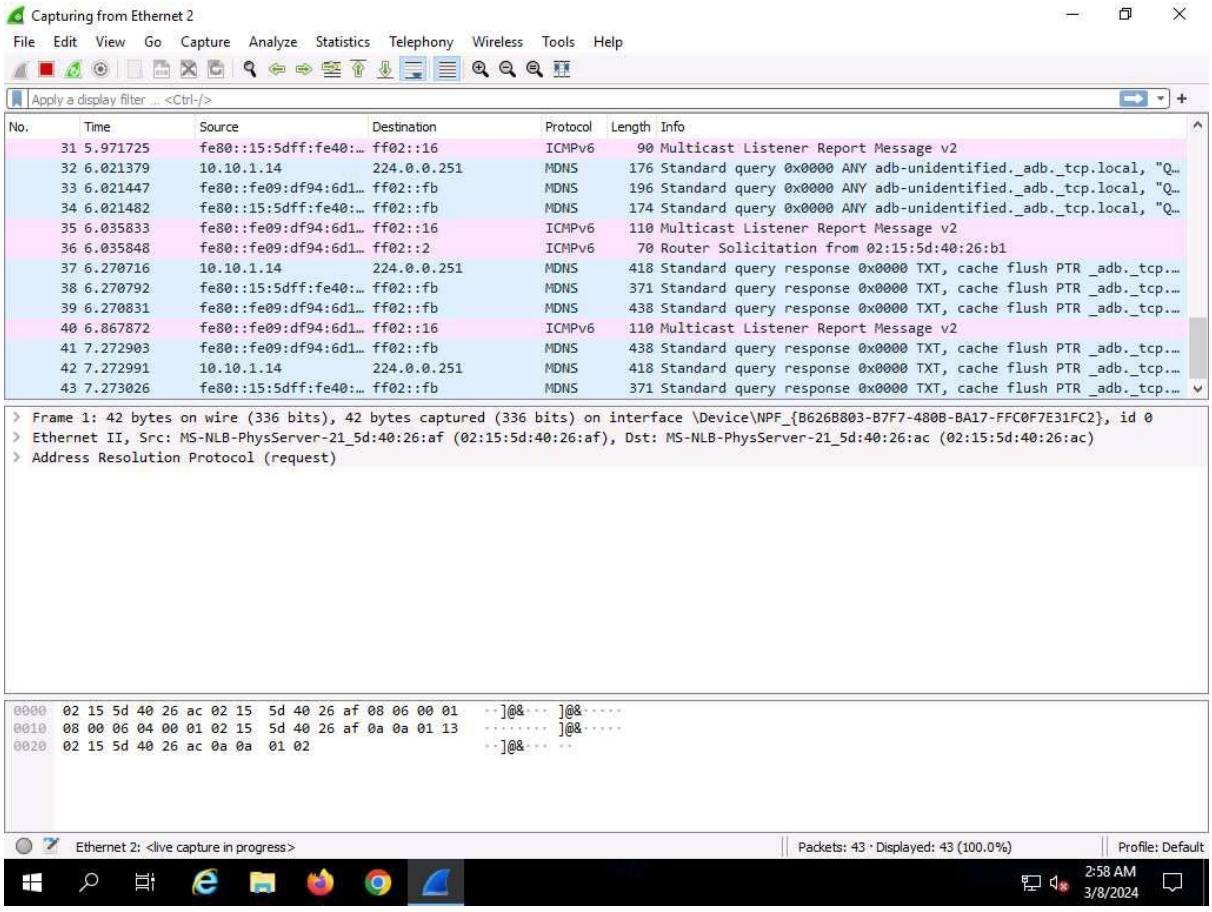
   Networks screen appears, click **Yes** to allow your PC to be discoverable by other PCs and devices on the network.

2. Search **Wireshark** from search bar and launch it.

   If the **Software update** window appears, click **Remind me later**.

3. **The Wireshark Network Analyzer** window appears, start capturing the network traffic on the primary network interface (here, **Ethernet 2**).

4. **Wireshark** starts capturing all packets generated while traffic is received by or sent from your machine.



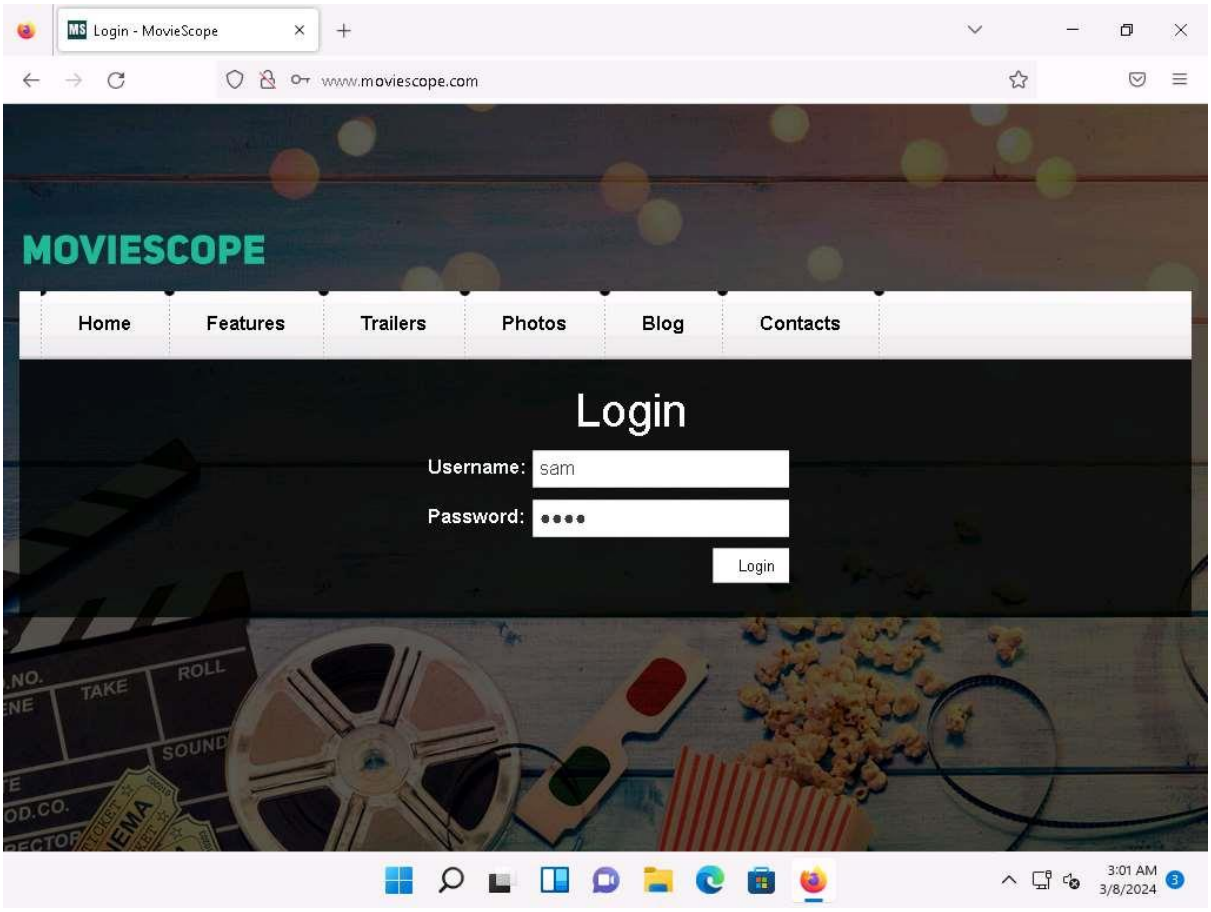
5. Now, click Windows 11 to switch to the **Windows 11** machine, login using **Admin/Pa$$w0rd**.

   Alternatively, you can also click **Pa$$w0rd** under **Windows 11** machine thumbnail in the **Resources** pane.

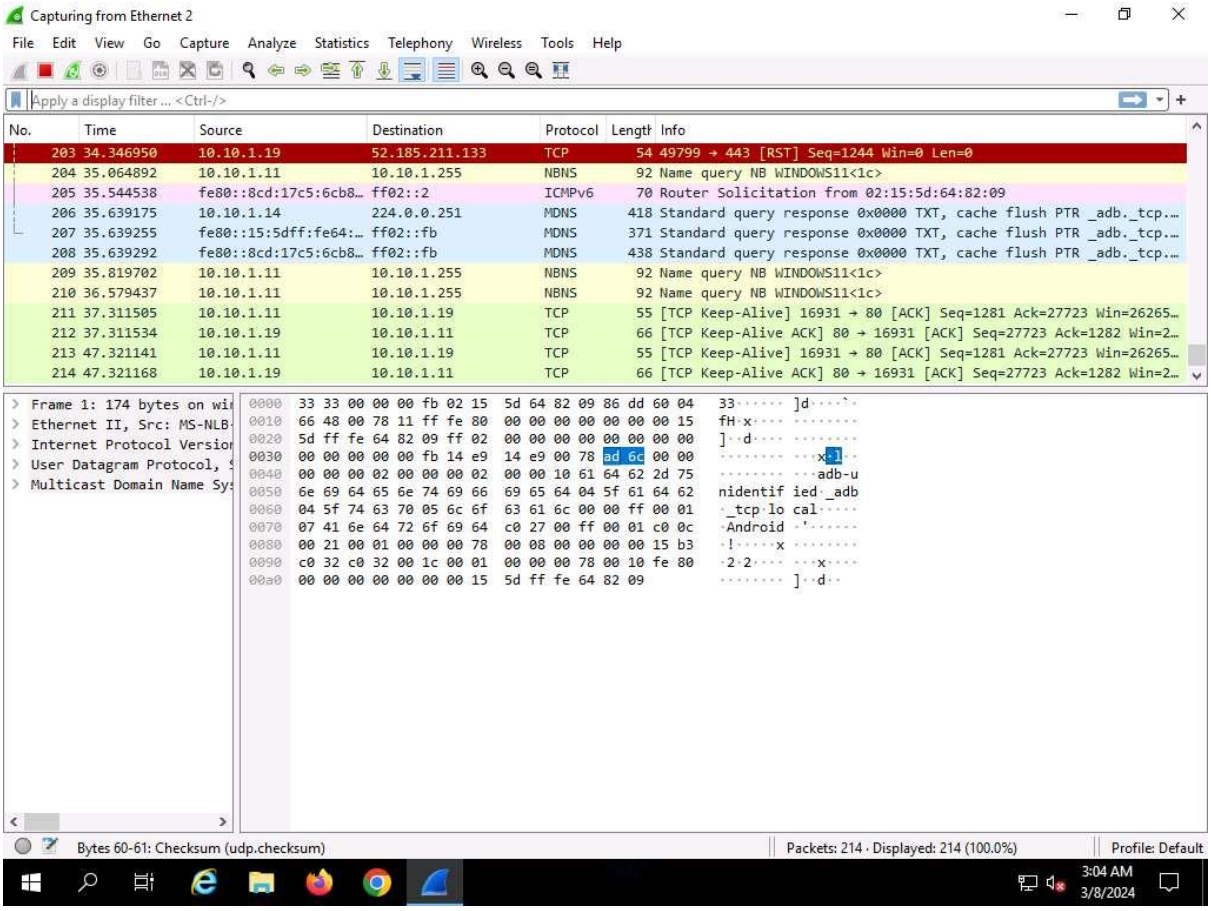
   If **Welcome to Windows** wizard appears, click **Continue** and in **Sign in with Microsoft** wizard, click **Cancel**.
   Networks screen appears, click **Yes** to allow your PC to be discoverable by other PCs and devices on the network.

6. Open any web browser, and go to **http://www.moviescope.com/** (here, we are using **Mozilla Firefox**).

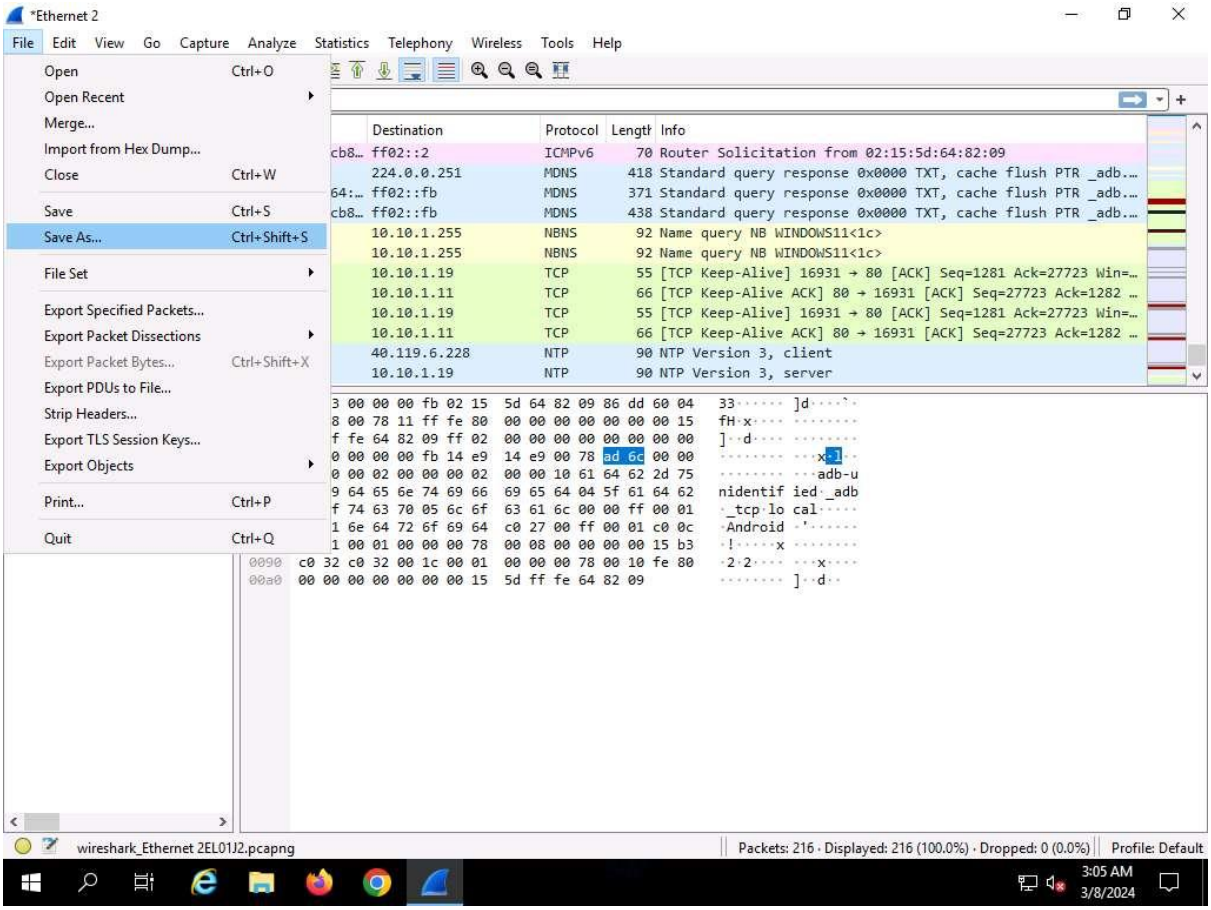
7. The **MOVIESCOPE** home page appears; login using **sam**/**test**.



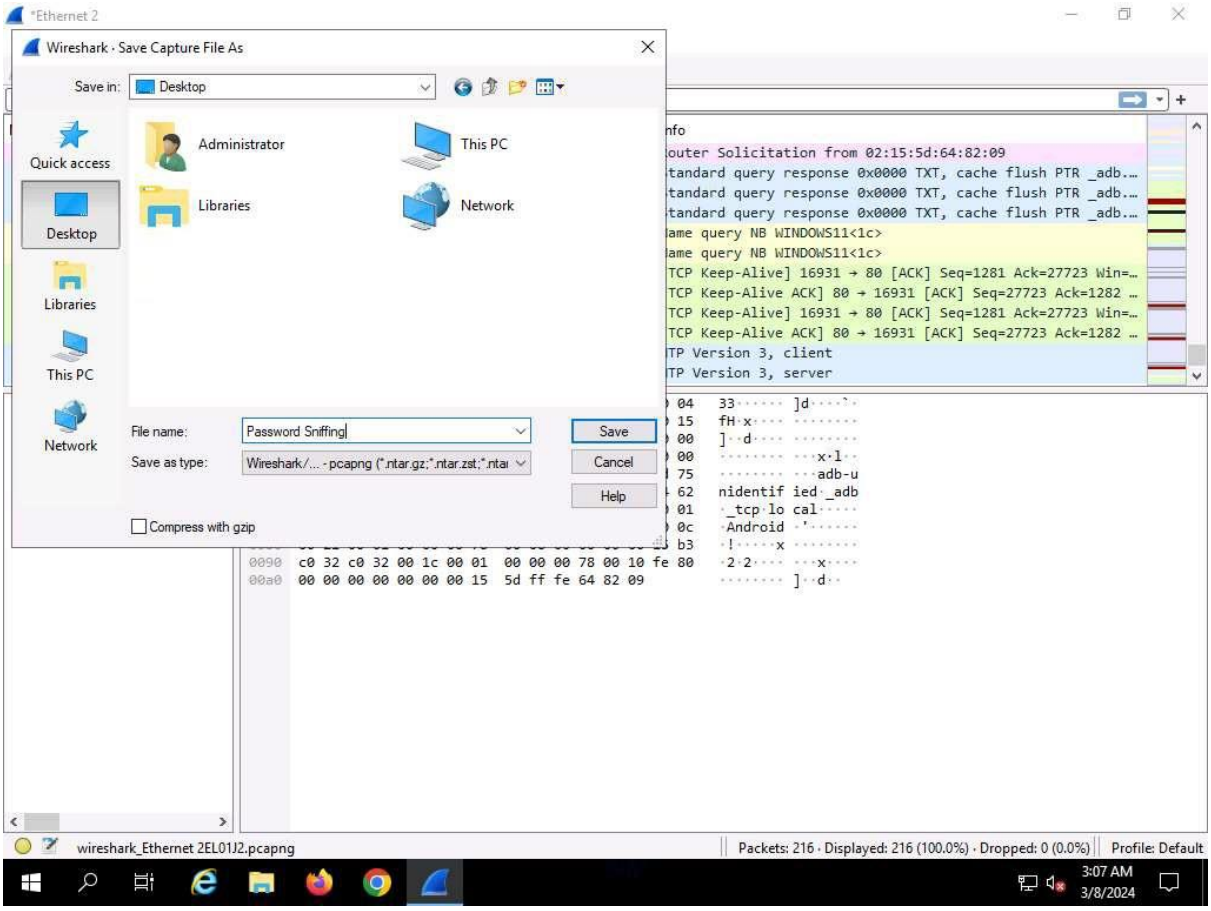

8. Click Windows Server 2019 to switch back to **Windows Server 2019** machine, and in the **Wireshark** window, click the **Stop capturing packets** icon on the toolbar.




9. Click **File --> Save As…** from the top-left corner of the window to save the captured packets.



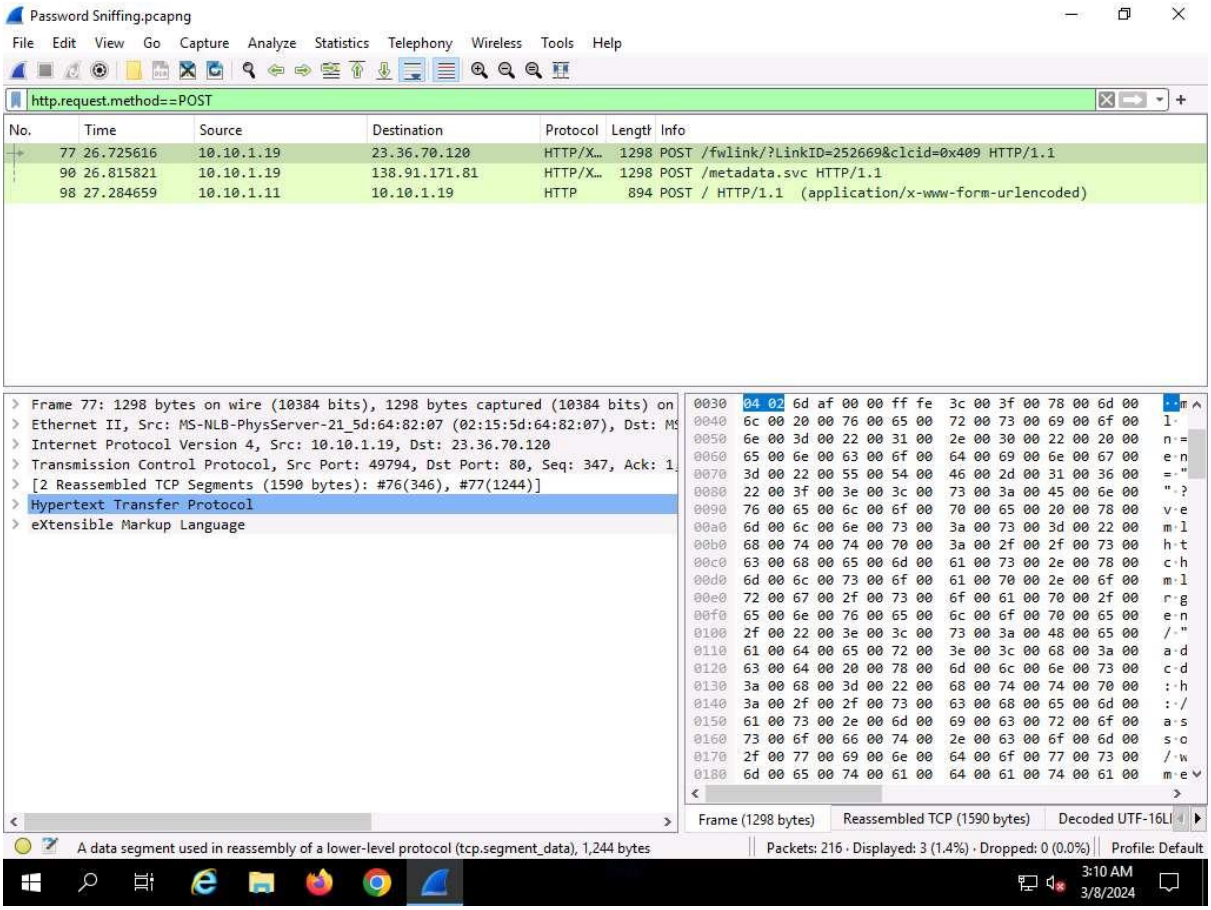

10. The **Wireshark: Save Capture File As** window appears. Select any location to save the file, specify **File name** as **Password Sniffing**, and click **Save**.



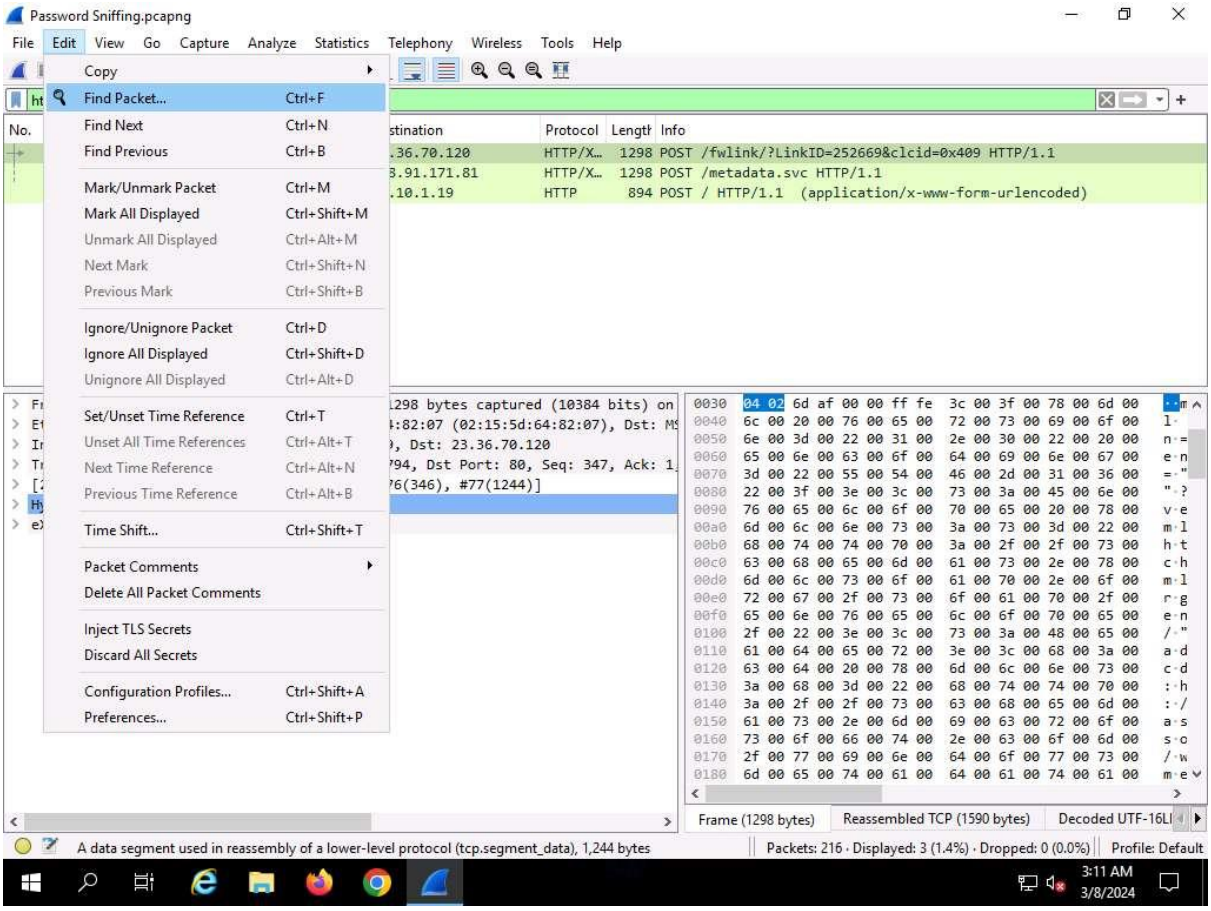

11. In the **Apply a display filter field**, type **http.request.method == POST** and click the arrow icon (**-->**) to apply the filter.

    Applying this syntax helps you narrow down the search for http POST traffic.

12. Wireshark only filters **http POST** traffic packets, as shown in the screenshot.



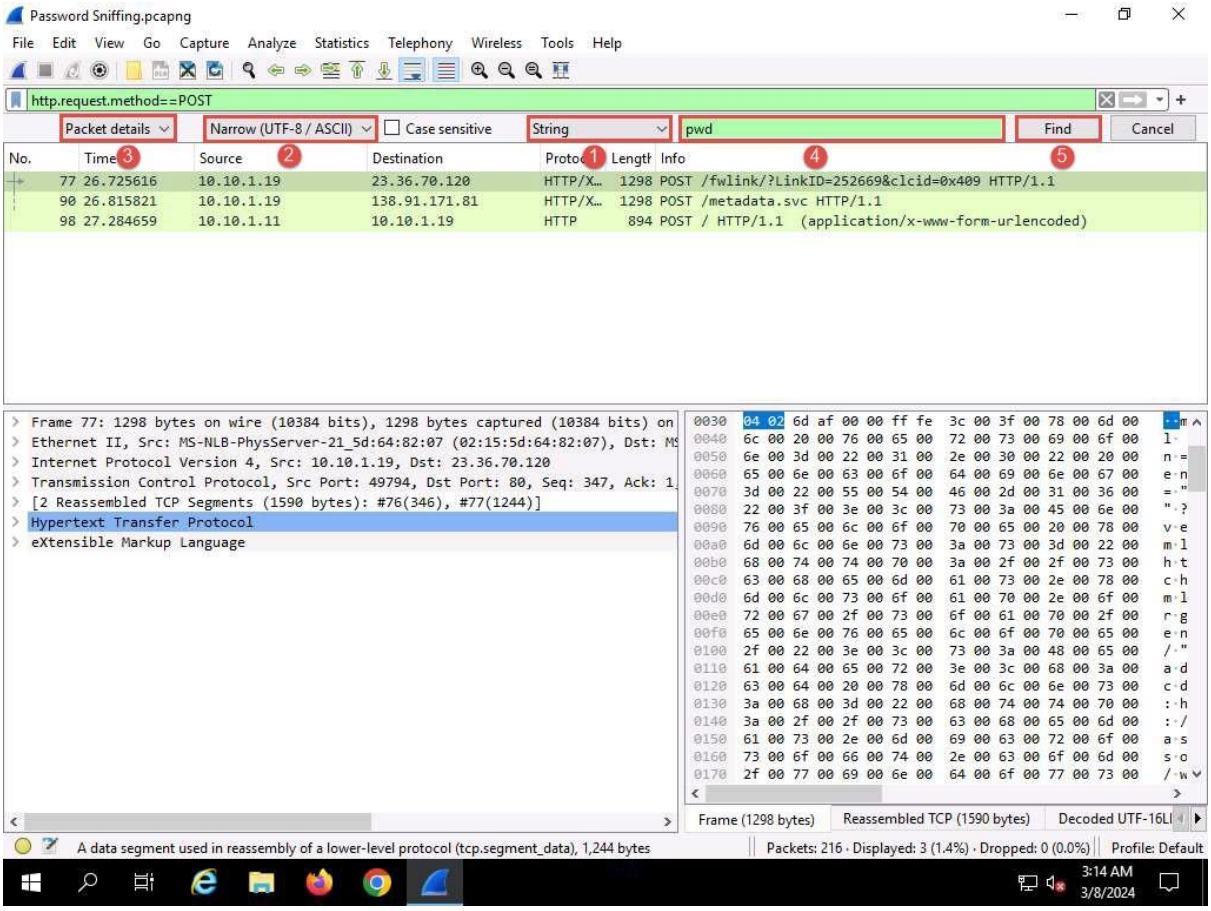

13. Now, navigate to **Edit** --> **Find Packet** from menu bar.




14. The **Find Packet** section appears below the display filter field.

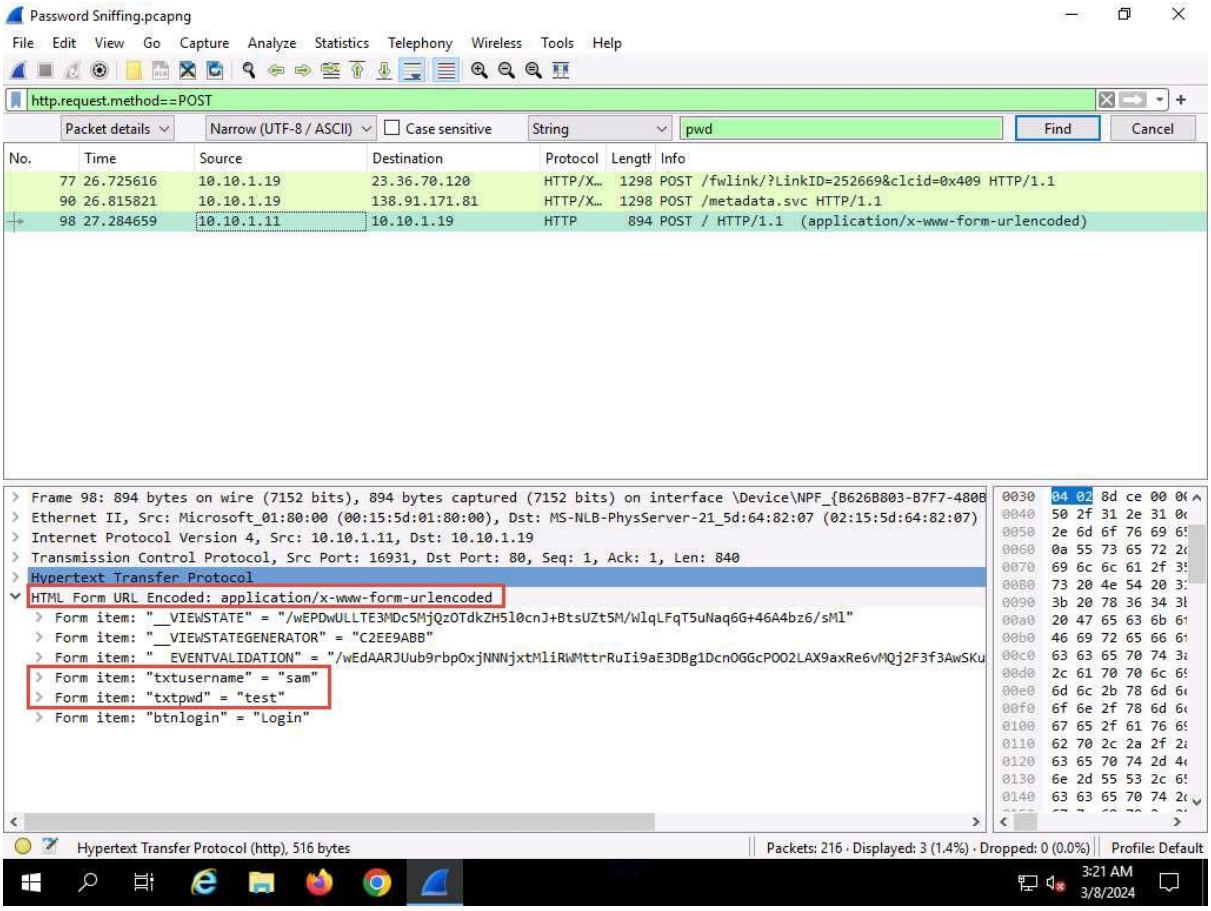
15. Click **Display filter**, select **String** from the drop-down options, click **Narrow & Wide** and select **Narrow (UTF-8 / ASCII)** from the drop-down options and click **Packet list**, select **Packet details** from the drop-down options.

16. In the field next to **String**, type **pwd** and click the **Find** button.



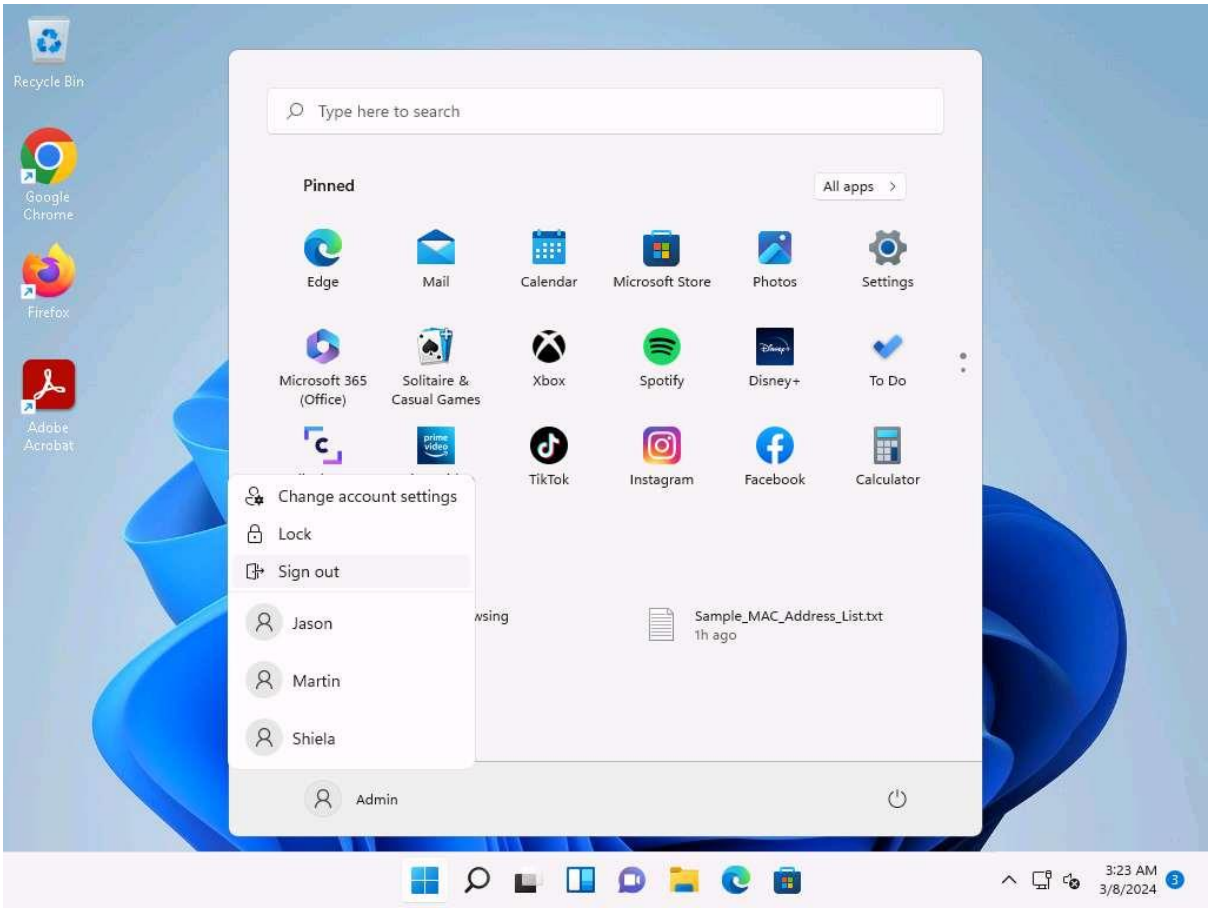

17. **Wireshark** will now display the sniffed password from the captured packets.

18. Expand the **HTML Form URL Encoded: application/x-www-form-urlencoded** node from the packet details section, and view the captured username and password, as shown in the screenshot.




19. Close the **Wireshark** window.

20. Click Windows 11 to switch to the **Windows 11** machine, close the web browser, and sign out from the **Admin** account.

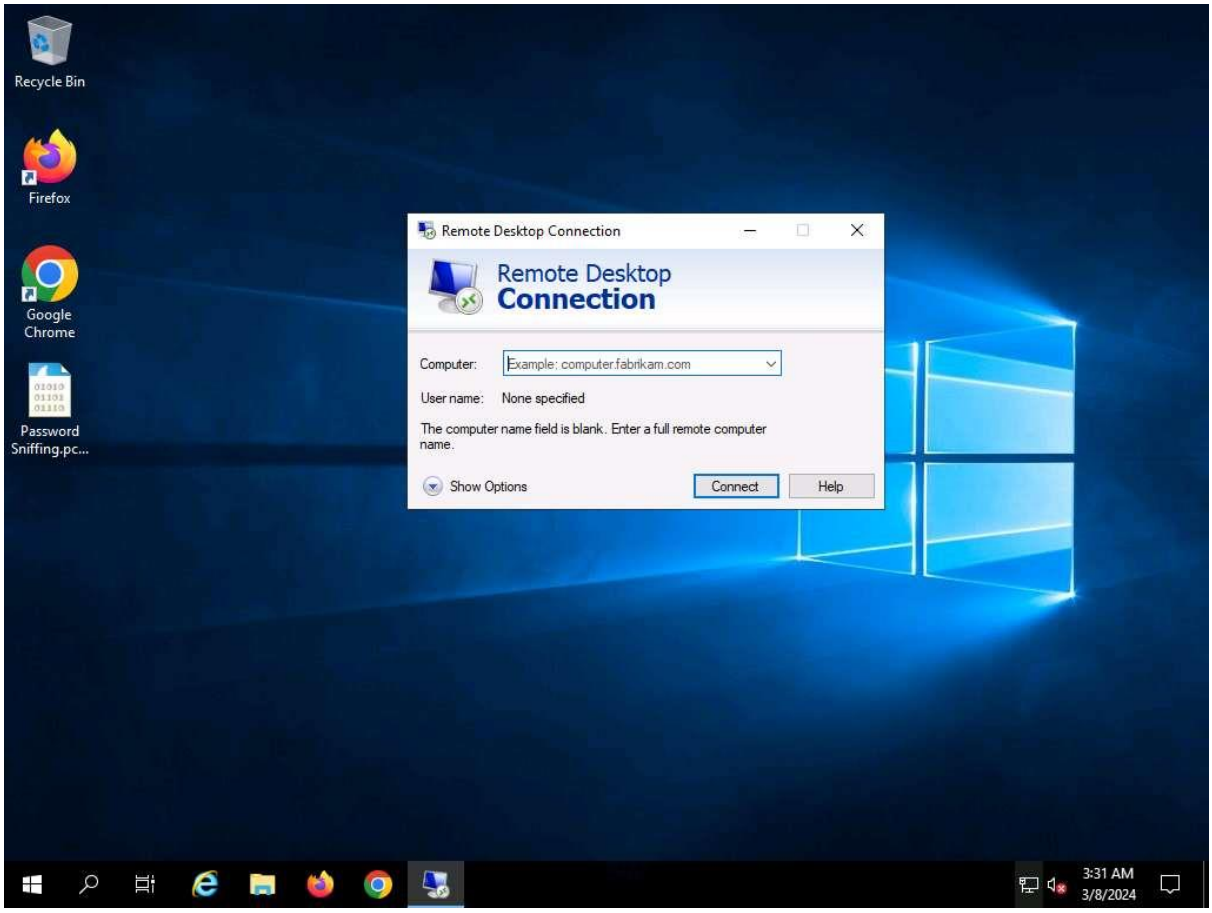



21. Click Windows Server 2019 to switch back to the **Windows Server 2019** machine.

22. Search **Remote Desktop Connection** from search bar and launch it.

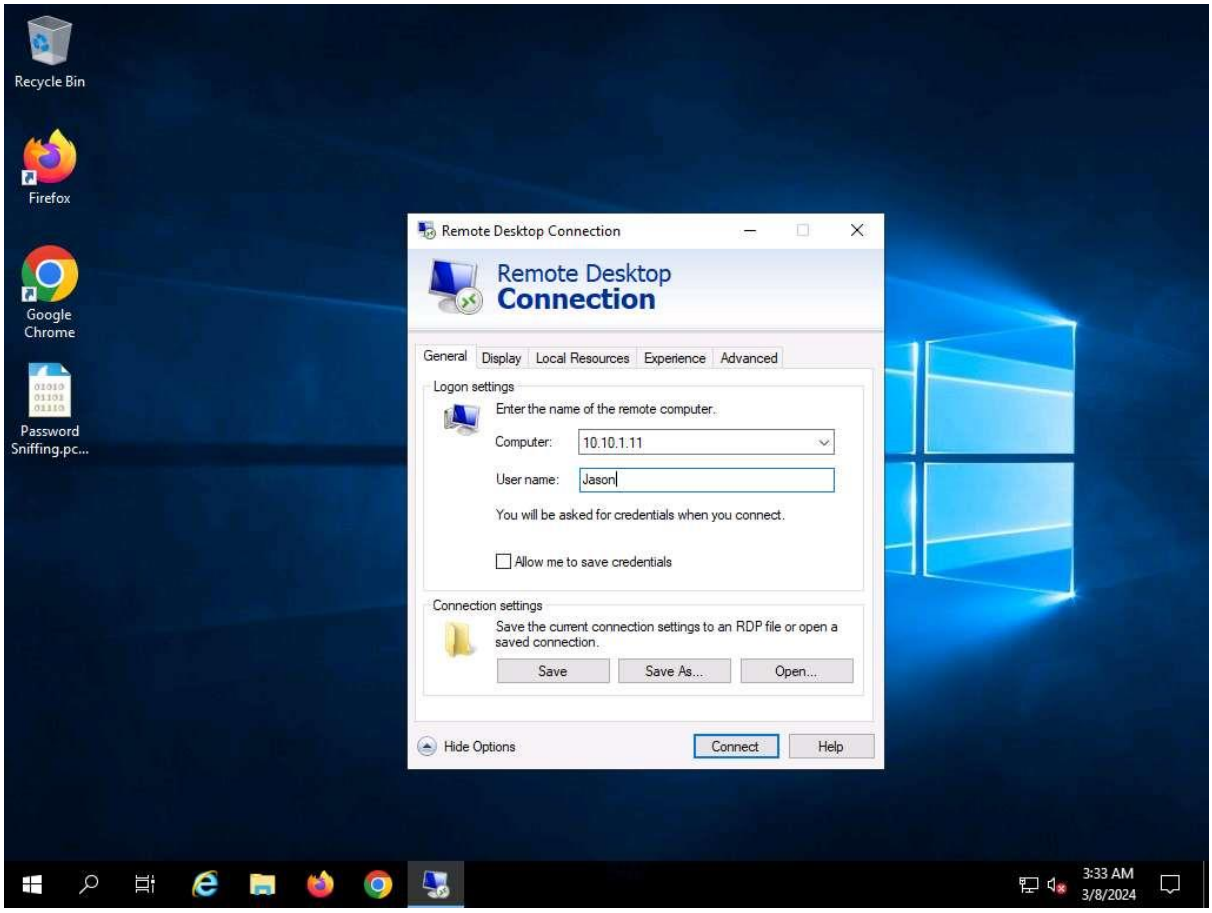
23. The **Remote Desktop Connection** dialog-box appears; click **Show Options**.

    If some previously accessed IP address appears in the **Computer** field, delete it.



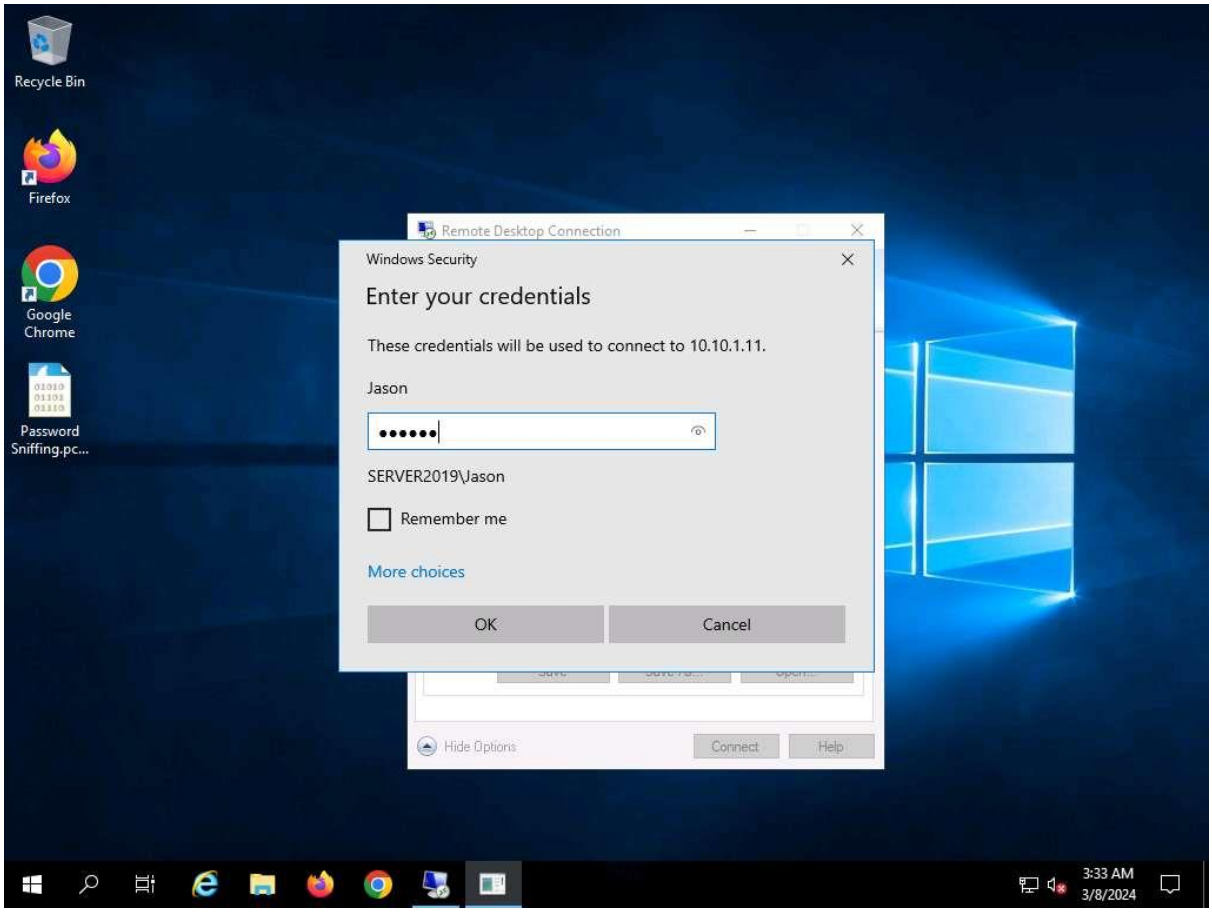

24. The dialog-box expands; under the **General** tab, type **10.10.1.11** in the **Computer** field and **Jason** in the **User name** field; click **Connect**.

    The IP address and username might differ in your lab environment. The target system credentials (**Jason** and **qwerty**) we are using here are obtained in the previous labs.

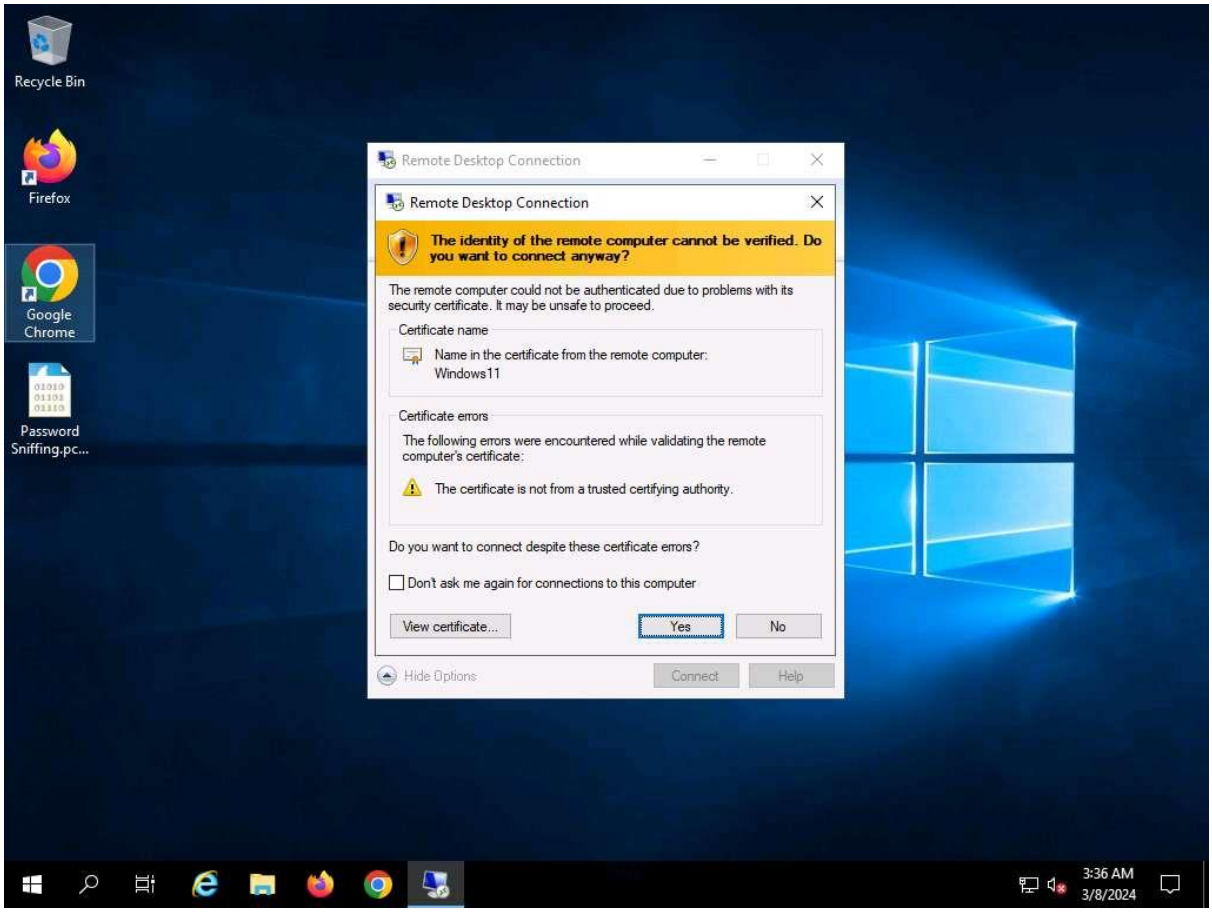



25. The **Windows Security** pop-up appears. Enter **Password** (**qwerty**) and click **OK**.

    If **Remember me** option is checked uncheck it.




26. The **Remote Desktop Connection** pop-up appears; click **Yes**.

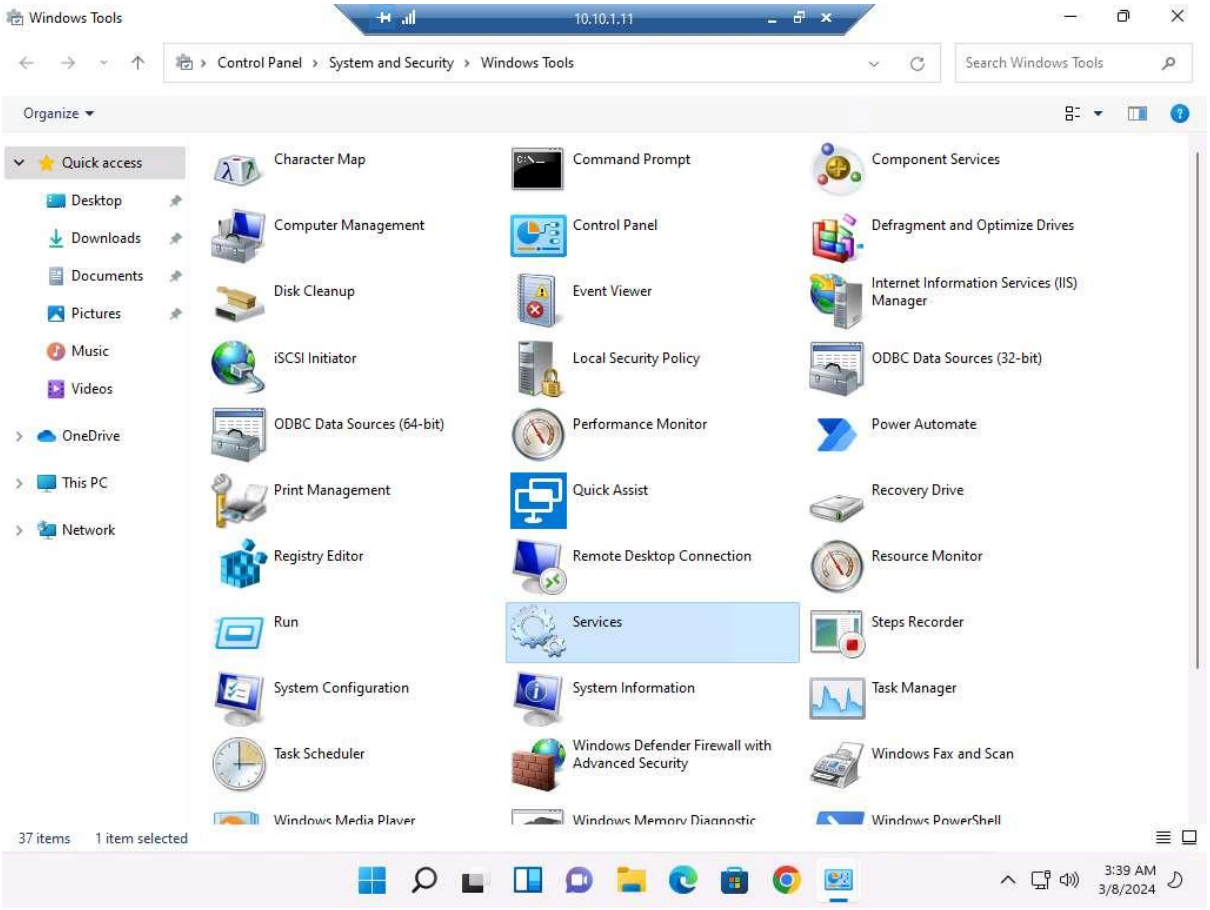



27. A remote connection to the target system (**Windows 11**) appears.

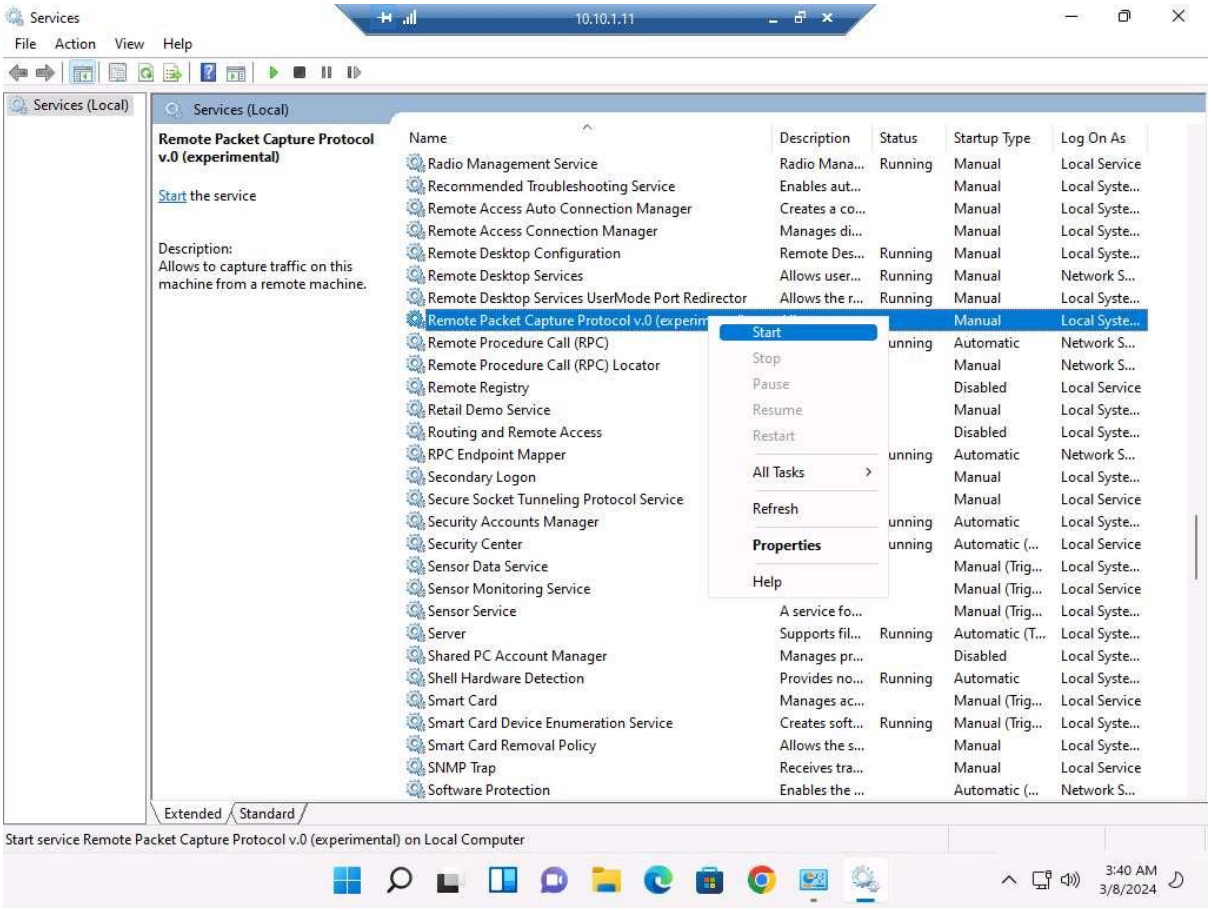
    If a **Choose privacy settings for your device** window appears, click on **Next** in the next window click on **Next** and in the next window click on **Accept**.

28. In the **Desktop** window, click windows **Search** icon and search for **Control Panel** in the search bar and launch it.

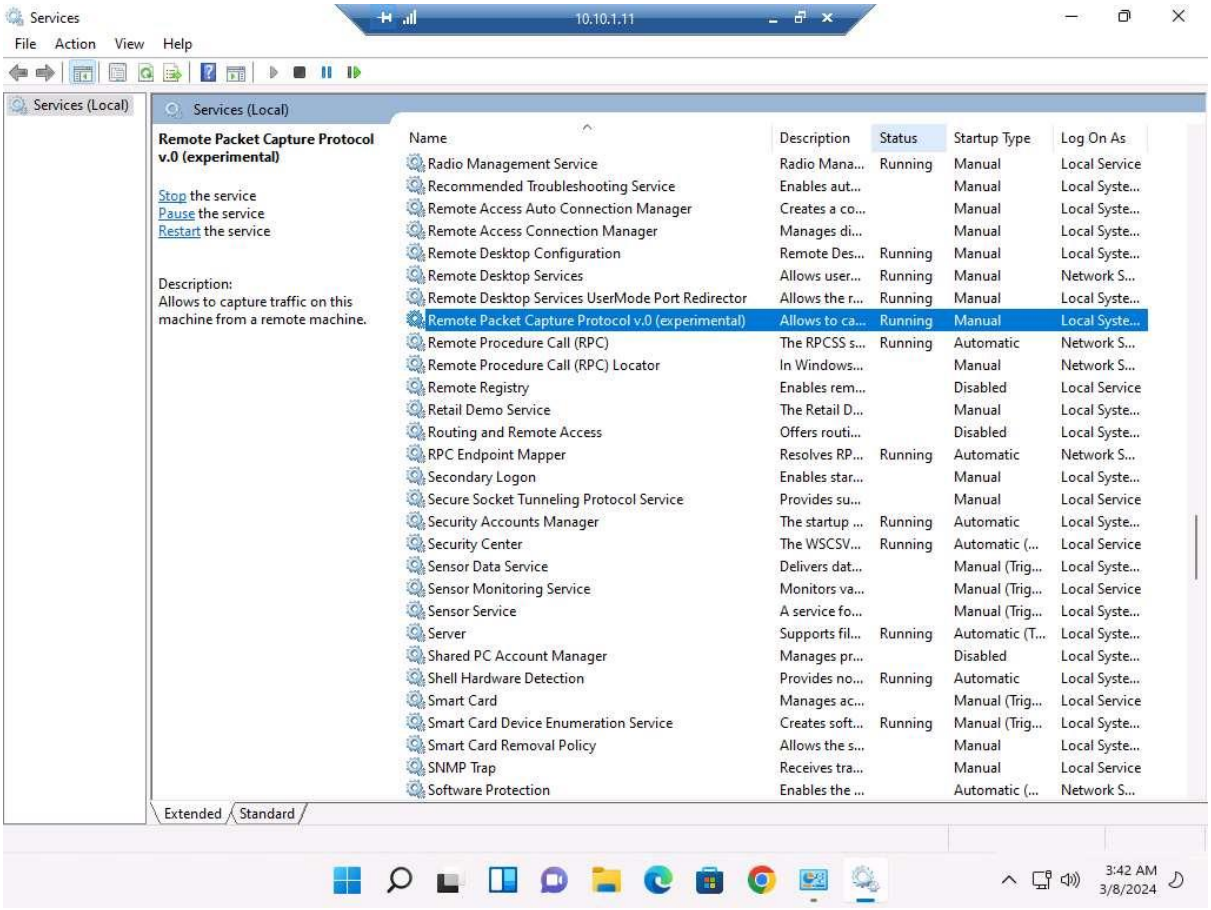
29. The **Control Panel** window appears; navigate to **System and Security --> Windows Tools**. In the **Windows Tools** control panel, double-click **Services**.




30. The **Services** window appears. Choose **Remote Packet Capture Protocol v.0 (experimental)**, right-click the service, and click **Start**.



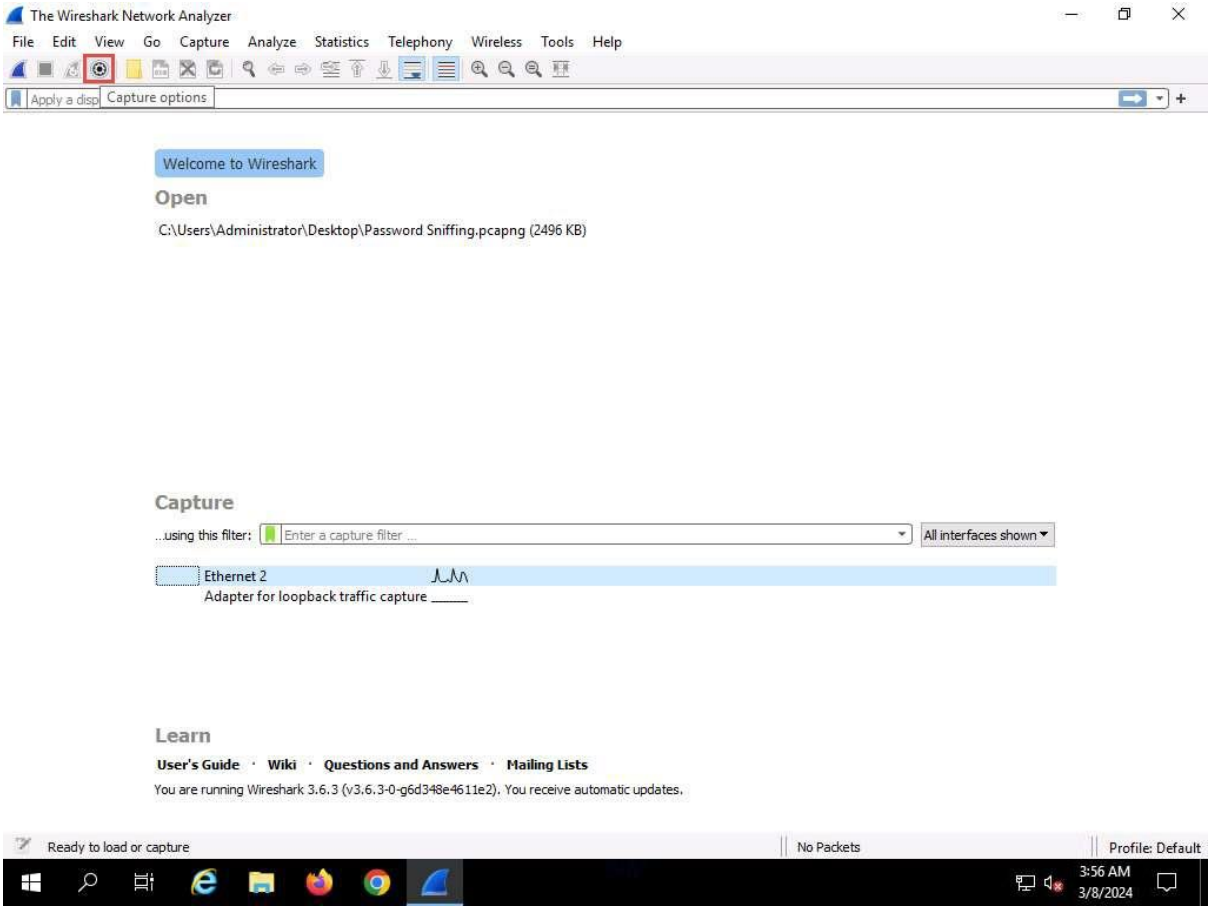

31. The **Status** of the **Remote Packet Capture Protocol v.0 (experimental)** service will change to **Running**, as shown in the screenshot.



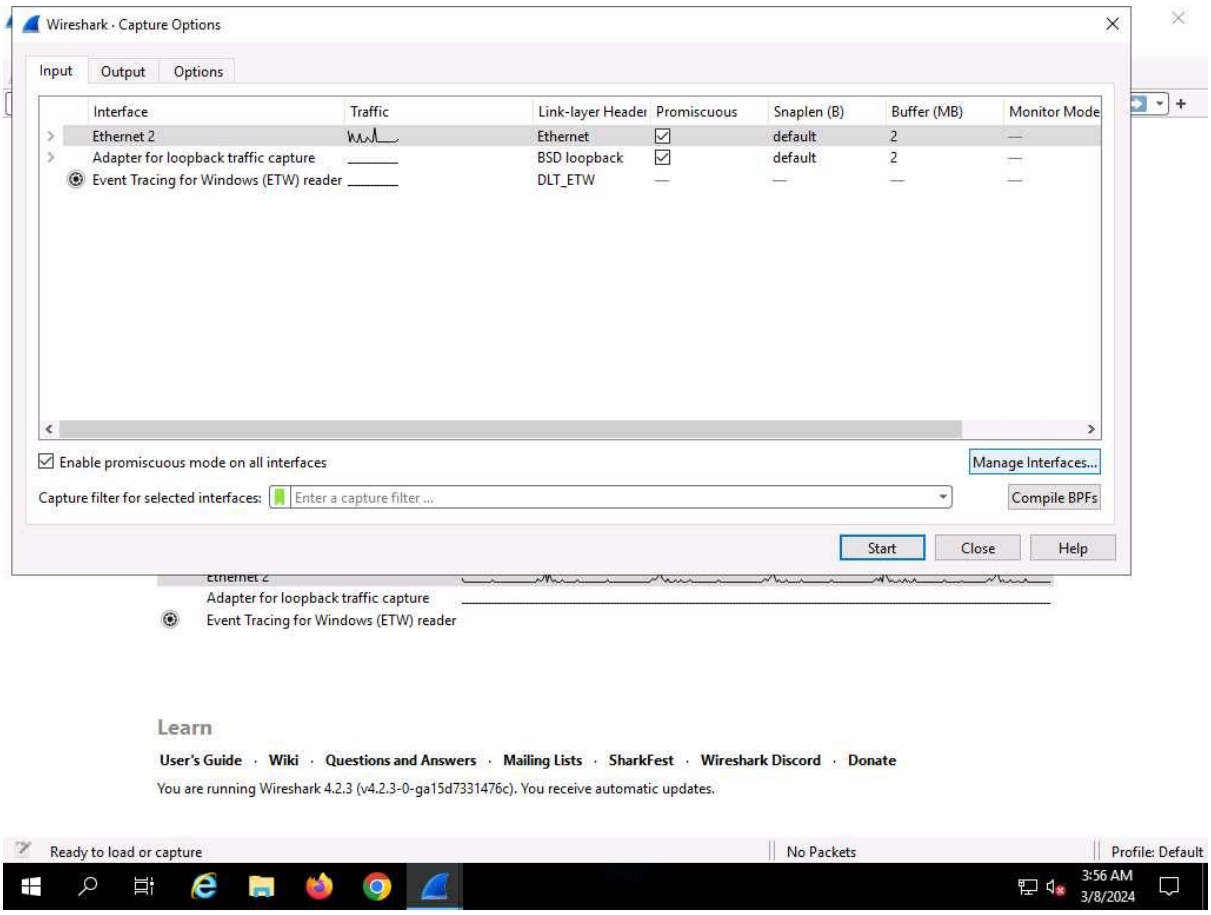

32. Close all open windows on the **Windows 11** machine and close **Remote Desktop Connection**.

    If a **Remote Desktop Connection** pop-up appears, click **OK**.

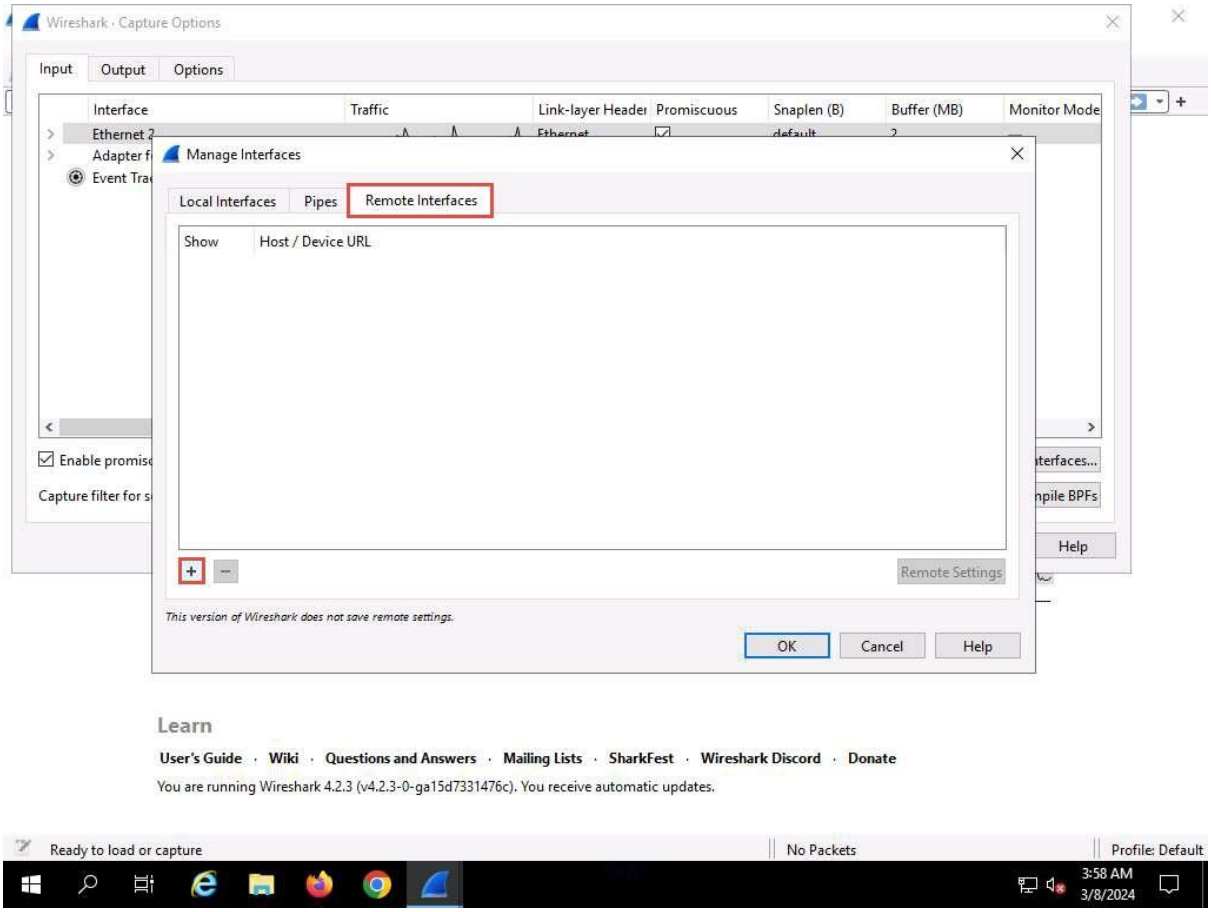
33. Now, in **Windows Server 2019**, launch **Wireshark** and click on **Capture options** icon from the toolbar.




34. The **Wireshark**. **Capture Options** window appears; click the **Manage Interfaces…** button.



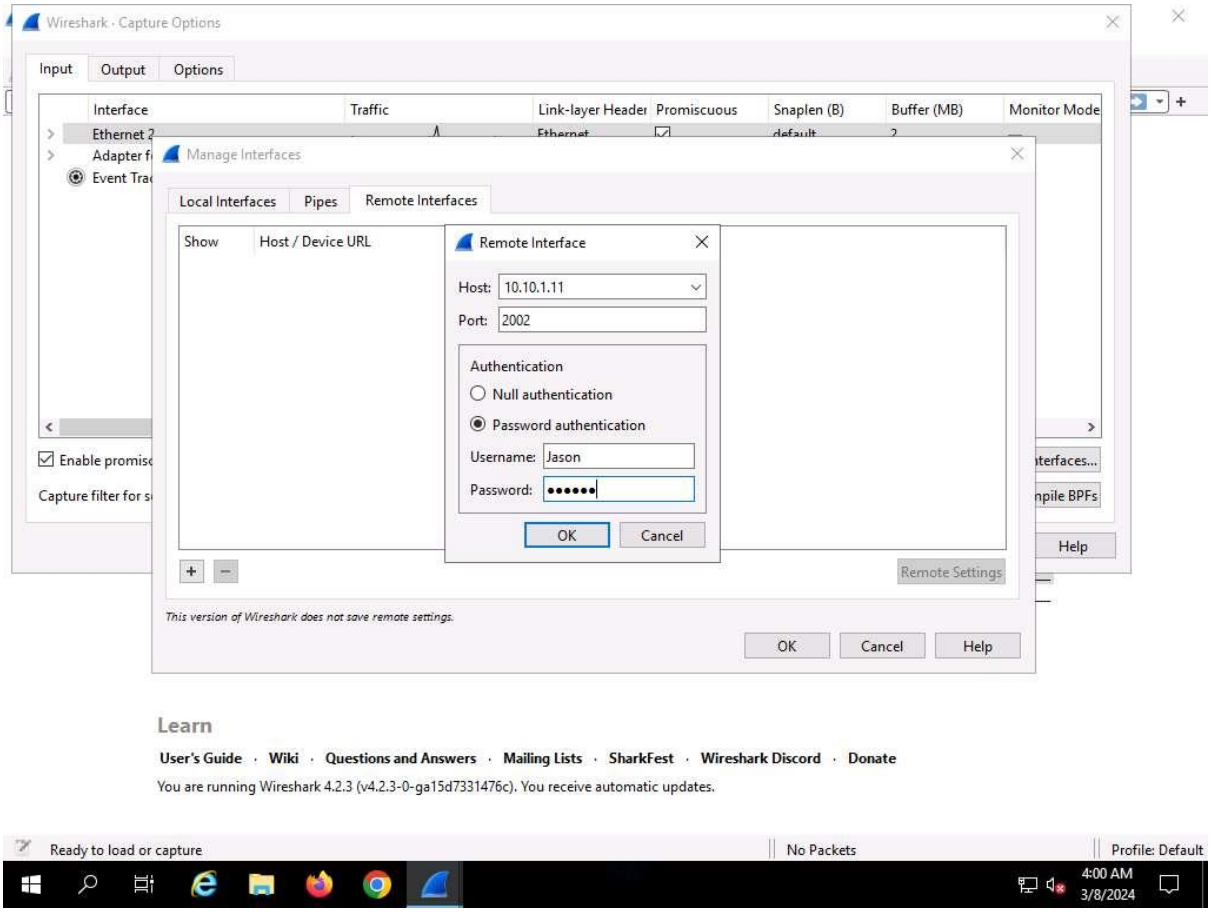

35. The **Manage Interfaces** window appears; click the **Remote Interfaces** tab, and then the **Add a remote host and its interface** icon (+).



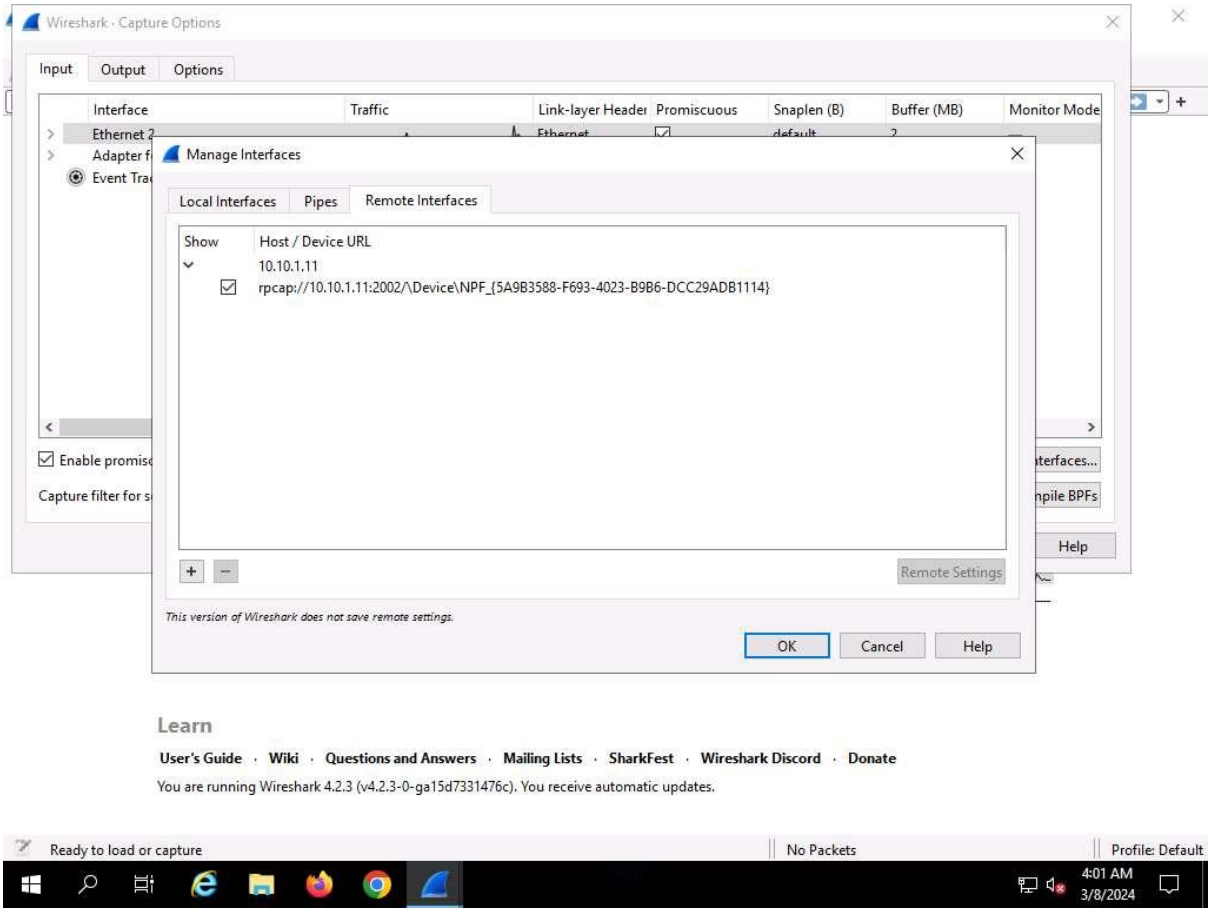

36. The **Remote Interface** window appears. In the **Host** text field, enter the IP address of the target machine (here, **10.10.1.11**); and in the **Port** field, enter the port number as **2002**.

37. Under the **Authentication** section, select the **Password authentication** radio button and enter the target machine's user credentials (here, **Jason** and **qwerty**); click **OK**.

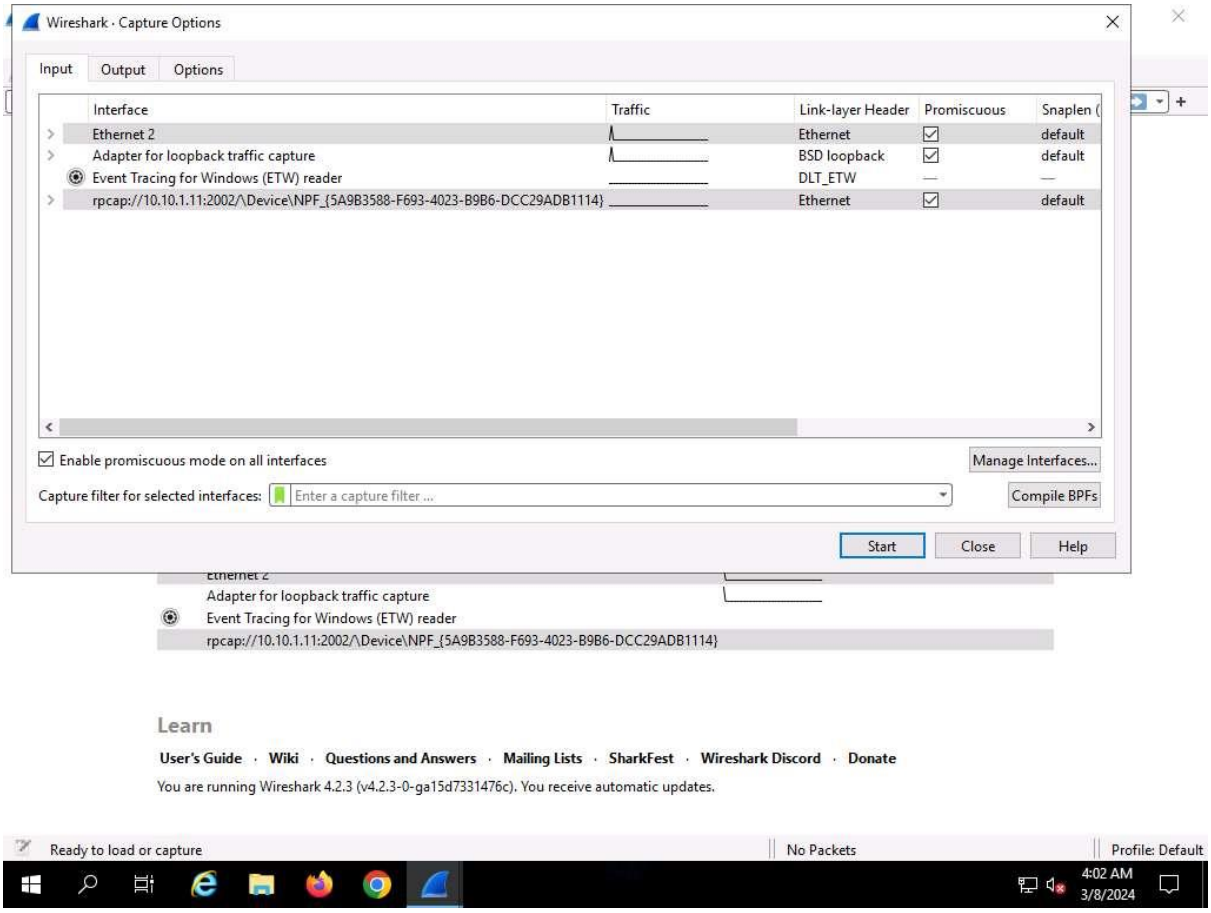
    The IP address and user credentials may differ when you perform this task.




38. A new remote interface is added to the **Manage Interfaces** window; click **OK**.



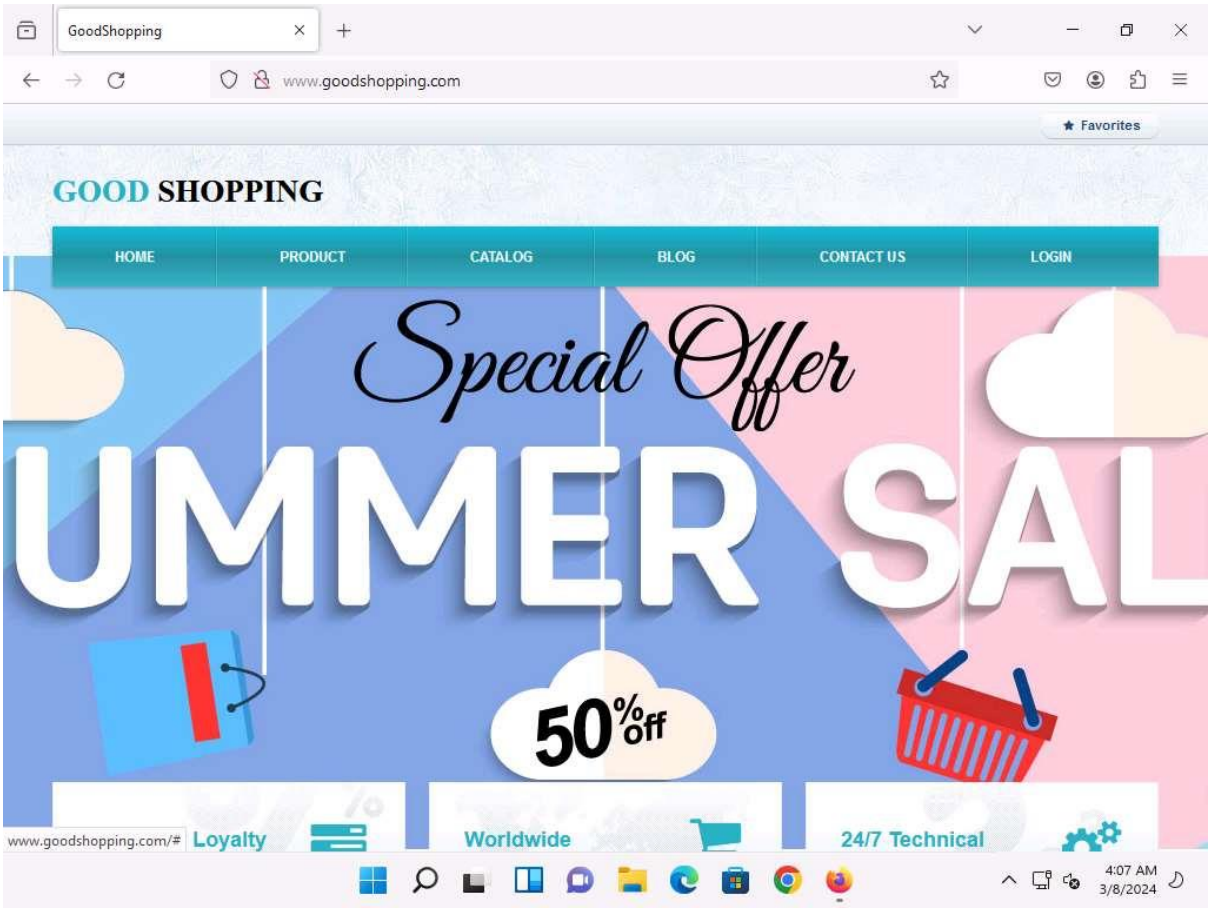

39. The newly added remote interface appears in the **Wireshark**. **Capture Options** window; click **Start**.



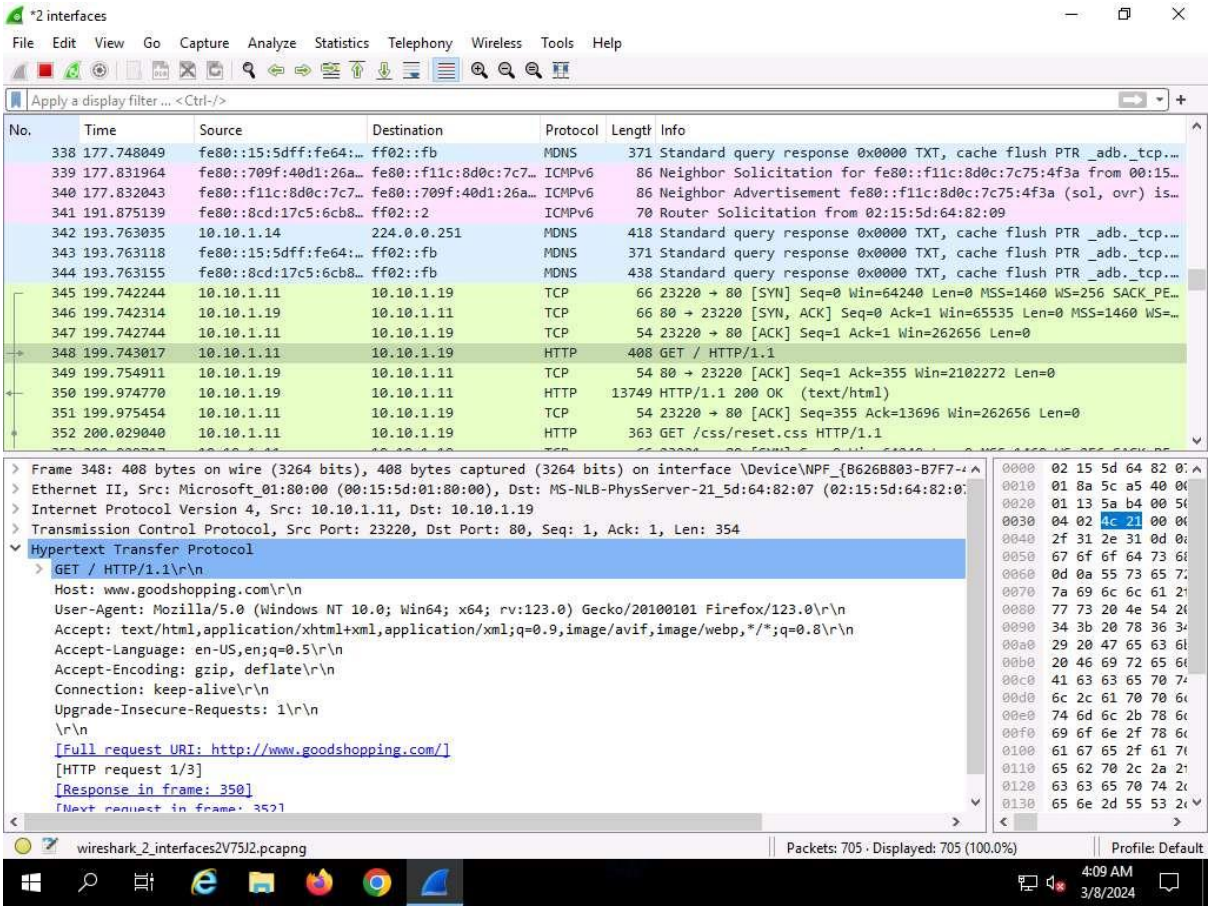

40. Click Windows 11 to switch to the **Windows 11** machine, and login using **Jason/qwerty**. Here, you are signing in as the victim.

41. Acting as the target, open any web browser go to **http://www.goodshopping.com** (here, we are using **Mozilla Firefox**).

    Although we are only browsing the Internet here, you could also log in to your account and sniff the credentials.




42. Click Windows Server 2019 to switch back to the **Windows Server 2019** machine. **Wireshark** starts capturing packets as soon as the user (here, you) begins browsing the Internet, the shown in the screenshot.




43. After a while, click the **Stop capturing packet** icon on the toolbar to stop live packet capture.

44. This way, you can use Wireshark to capture traffic on a remote interface.

    In real-time, when attackers gain the credentials of a victim's machine, they attempt to capture its remote interface and monitor the traffic its user browses to reveal confidential user information.

45. This concludes the demonstration of how to perform password sniffing using Wireshark.

46. Close all open windows and document all the acquired information.

**Question 8.2.1.1**

Use the Wireshark tool to perform password sniffing. Which Wireshark display filter shows HTTP POST traffic?

# Lab 3: Detect Network Sniffing

**Lab Scenario**

The previous labs demonstrated how an attacker carries out sniffing with different techniques and tools. This lab helps you understand possible defensive techniques used to defend a target network against sniffing attacks.

A professional ethical hacker or pen tester should be able to detect network sniffing in the network. A sniffer on a network only captures data and runs in promiscuous mode, so it is not easy to detect. Promiscuous mode allows a network device to intercept and read each network packet that arrives in its entirety. The sniffer leaves no trace, since it does not transmit data. Therefore, to detect sniffing attempts, you must use the various network sniffing detection techniques and tools discussed in this lab.

**Lab Objectives**

- Detect ARP poisoning and promiscuous mode in a switch-based network

**Overview of Detecting Network Sniffing**

Network sniffing involves using sniffer tools that enable the real-time monitoring and analysis of data packets flowing over computer networks. These network sniffers can be detected by using various techniques such as:

- **Ping Method**: Identifies if a system on the network is running in promiscuous mode
- **DNS Method**: Identifies sniffers in the network by analyzing the increase in network traffic
- **ARP Method**: Sends a non-broadcast ARP to all nodes in the network; a node on the network running in promiscuous mode will cache the local ARP address

# Task 1: Detect ARP Poisoning and Promiscuous Mode in a Switch-Based Network

ARP poisoning involves forging many ARP request and reply packets to overload a switch. ARP cache poisoning is the method of attacking a LAN network by updating the target computer's ARP cache with both forged ARP request and reply packets designed to change the Layer 2 Ethernet MAC address (that of the network card) to one that the attacker can monitor. Attackers use ARP poisoning to sniff on the target network. Attackers can thus steal sensitive information, prevent network and web access, and perform DoS and MITM attacks.

Promiscuous mode allows a network device to intercept and read each network packet that arrives in its entirety. The sniffer toggles the NIC of a system to promiscuous mode, so that it listens to all data transmitted on its segment. A sniffer can constantly monitor all network traffic to

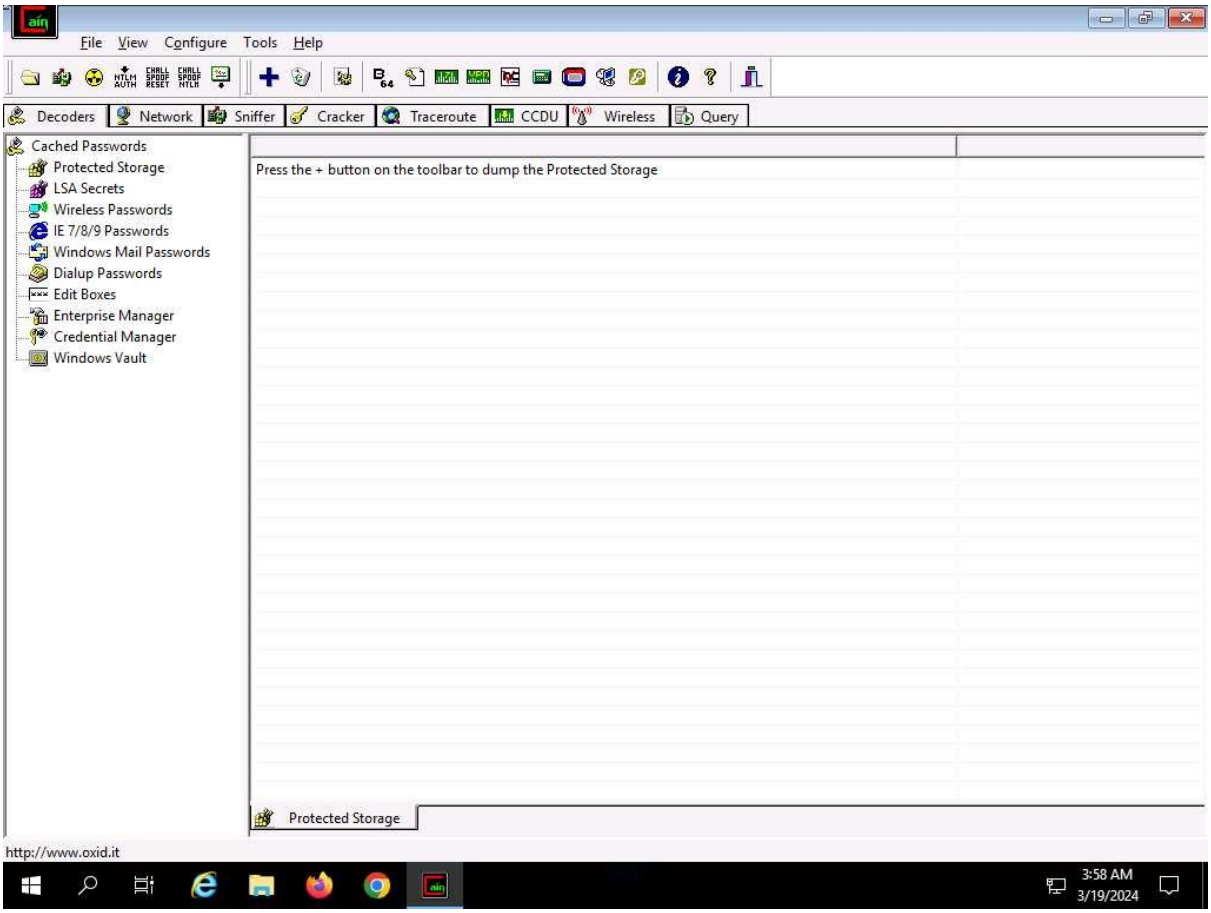
a computer through the NIC by decoding the information encapsulated in the data packet. Promiscuous mode in the network can be detected using various tools.

The ethical hacker and pen tester must assess the organization or target of evaluation for ARP poisoning vulnerabilities.

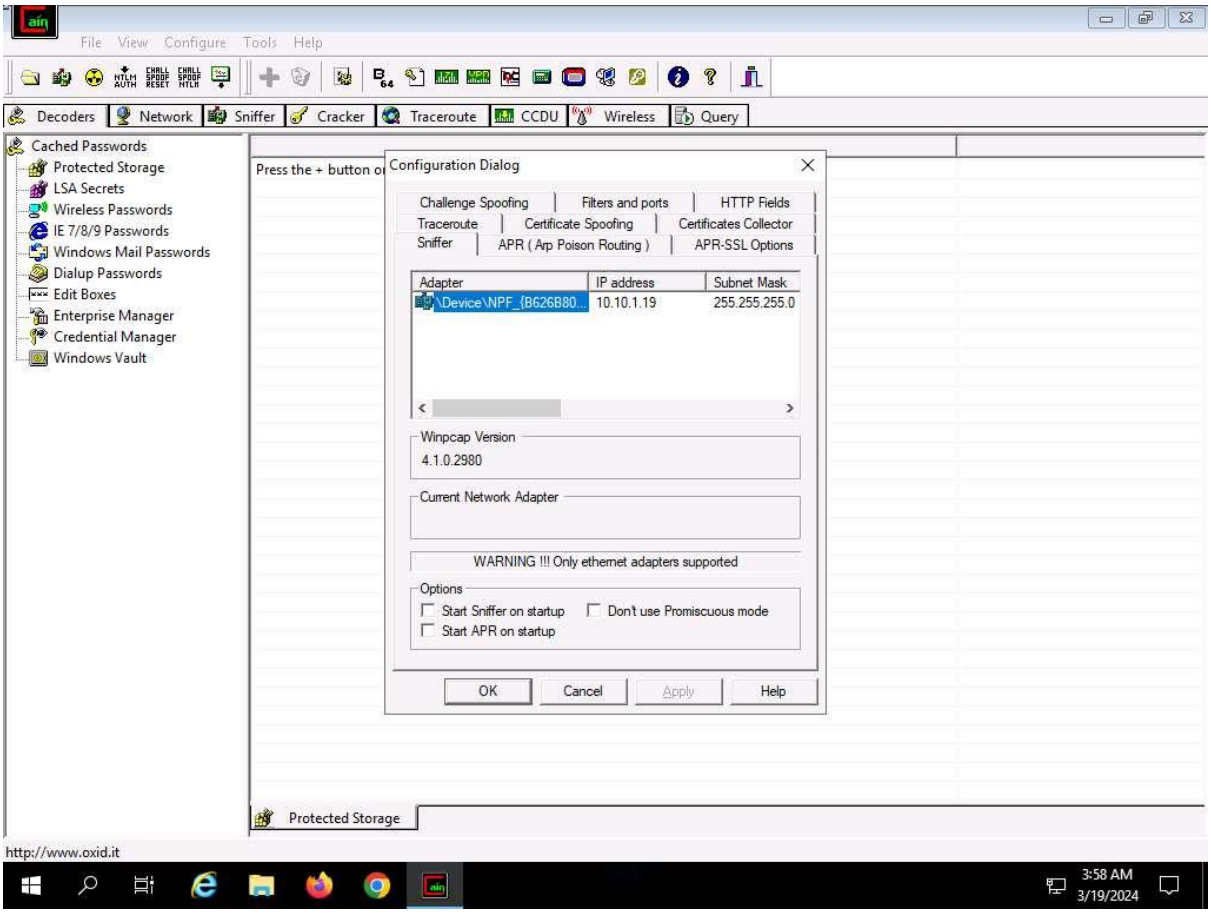
Here, we will detect ARP poisoning in a switch-based network using Wireshark and we will use the Nmap Scripting Engine (NSE) to check if a system on a local Ethernet has its network card in promiscuous mode.

In this task, we will use the **Windows Server 2019** machine as the host machine to perform ARP poisoning, and will sniff traffic flowing between the **Windows 11** and **Parrot Security** machines. We will use the same machine (**Windows Server 2019**) to detect ARP poisoning and use the **Windows 11** machine to detect promiscuous mode in the network.

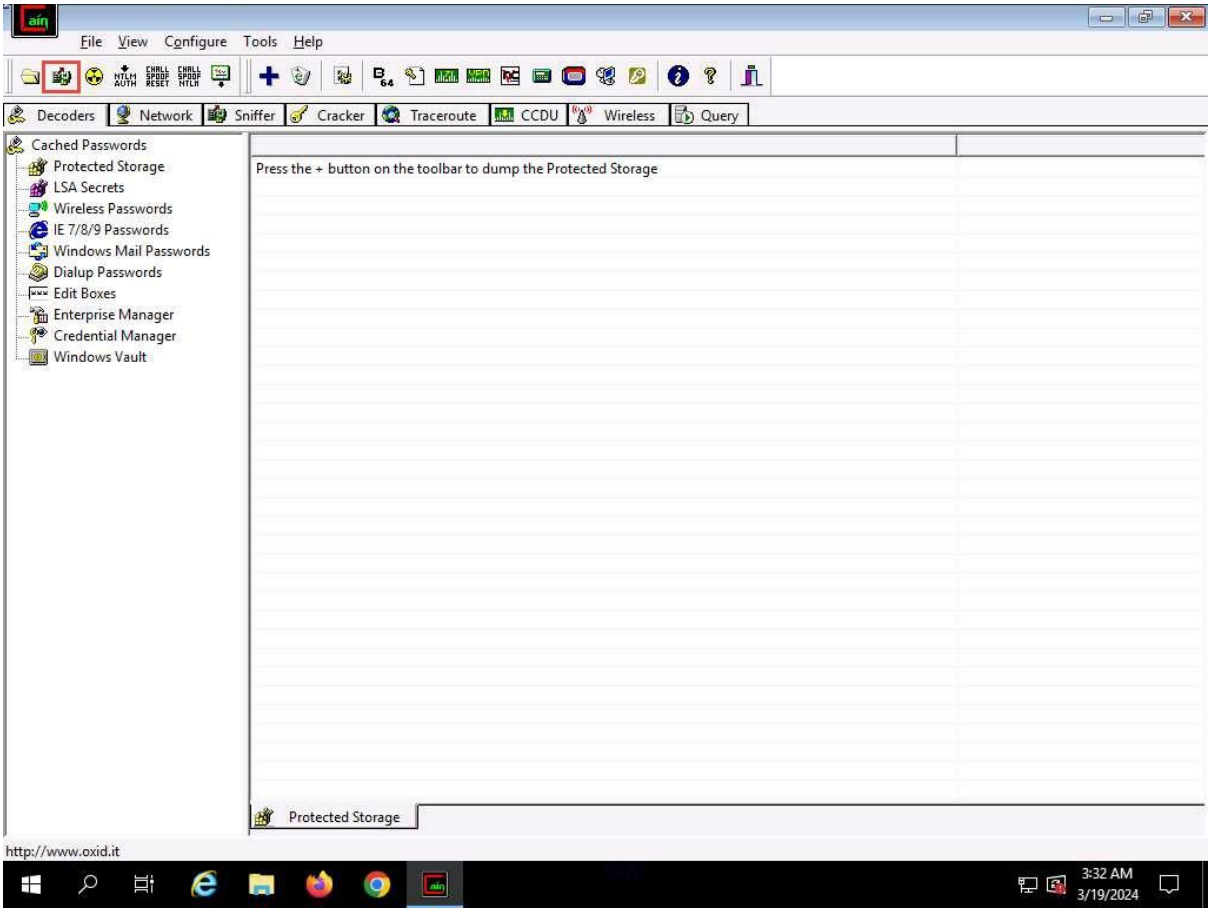
1. Click Windows Server 2019 to switch to the **Windows Server 2019** machine.
2. In the **Desktop** window, click windows **Search** icon and search for **cain** in the search bar and launch it.
3. The **Cain & Abel** main window appears, click **Configure** from the menu bar to configure an ethernet card.



4. The **Configuration Dialog** window appears. The **Sniffer** tab is selected by default. Ensure that the **Adapter** associated with the **IP address** of the machine is selected and click **OK**.




5. Click the **Start/Stop Sniffer** icon on the toolbar to begin sniffing.

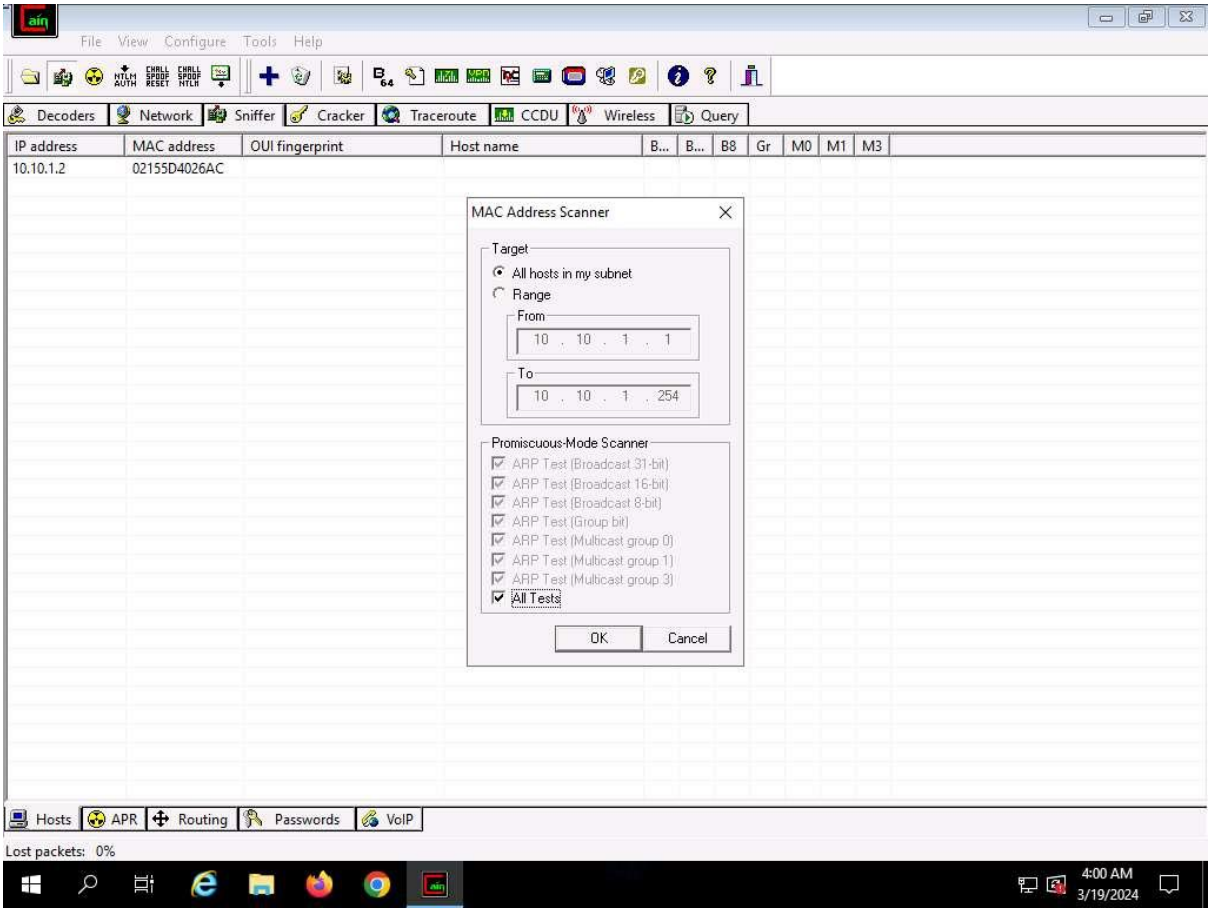



6. The **Cain** pop-up appears with a **Warning** message, click **OK**.

7. Now, click the **Sniffer** tab.

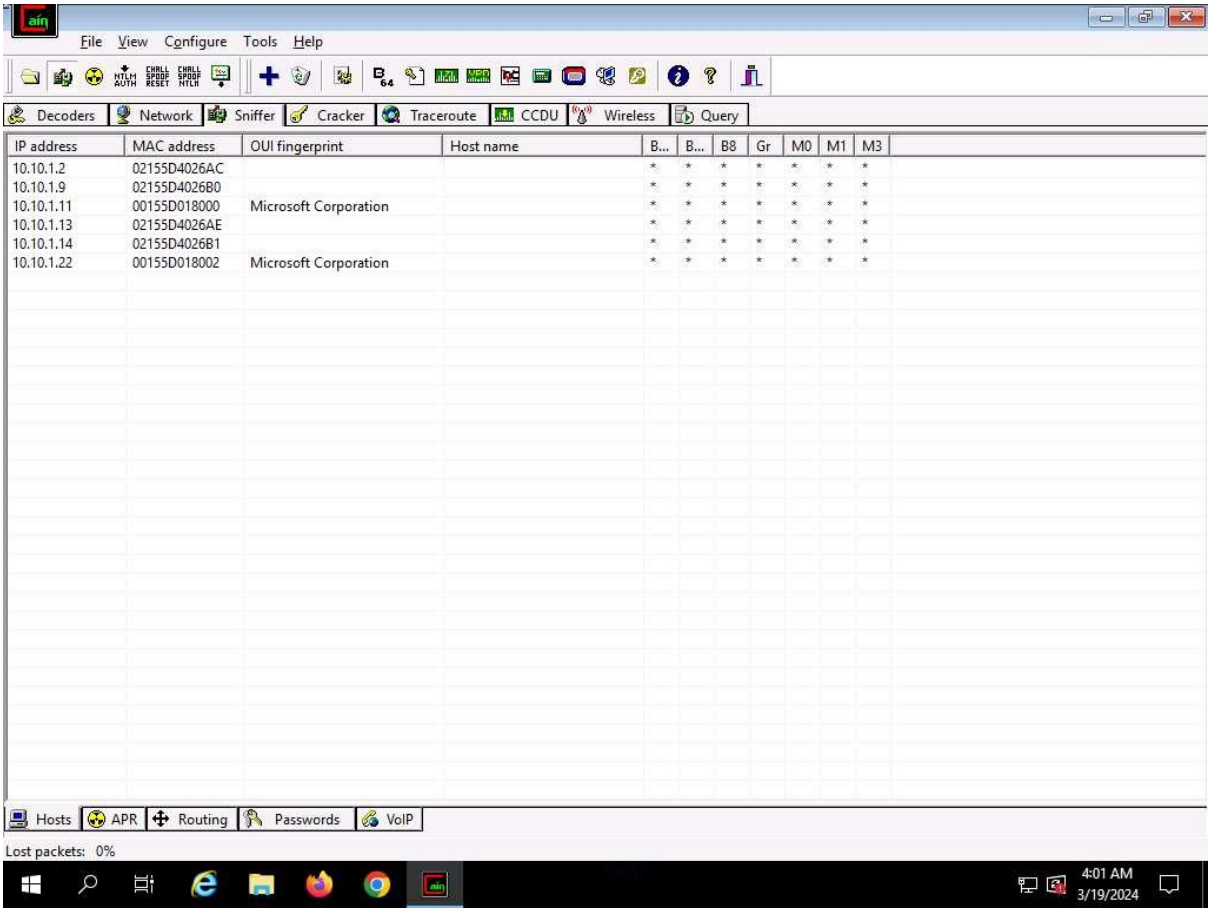
8. Click the plus (**+**) icon or right-click in the window and select **Scan MAC Addresses** to scan the network for hosts.

9. The **MAC Address Scanner** window appears. Check the **All hosts in my subnet** radio button. Select the **All Tests** checkbox; then, click **OK**.



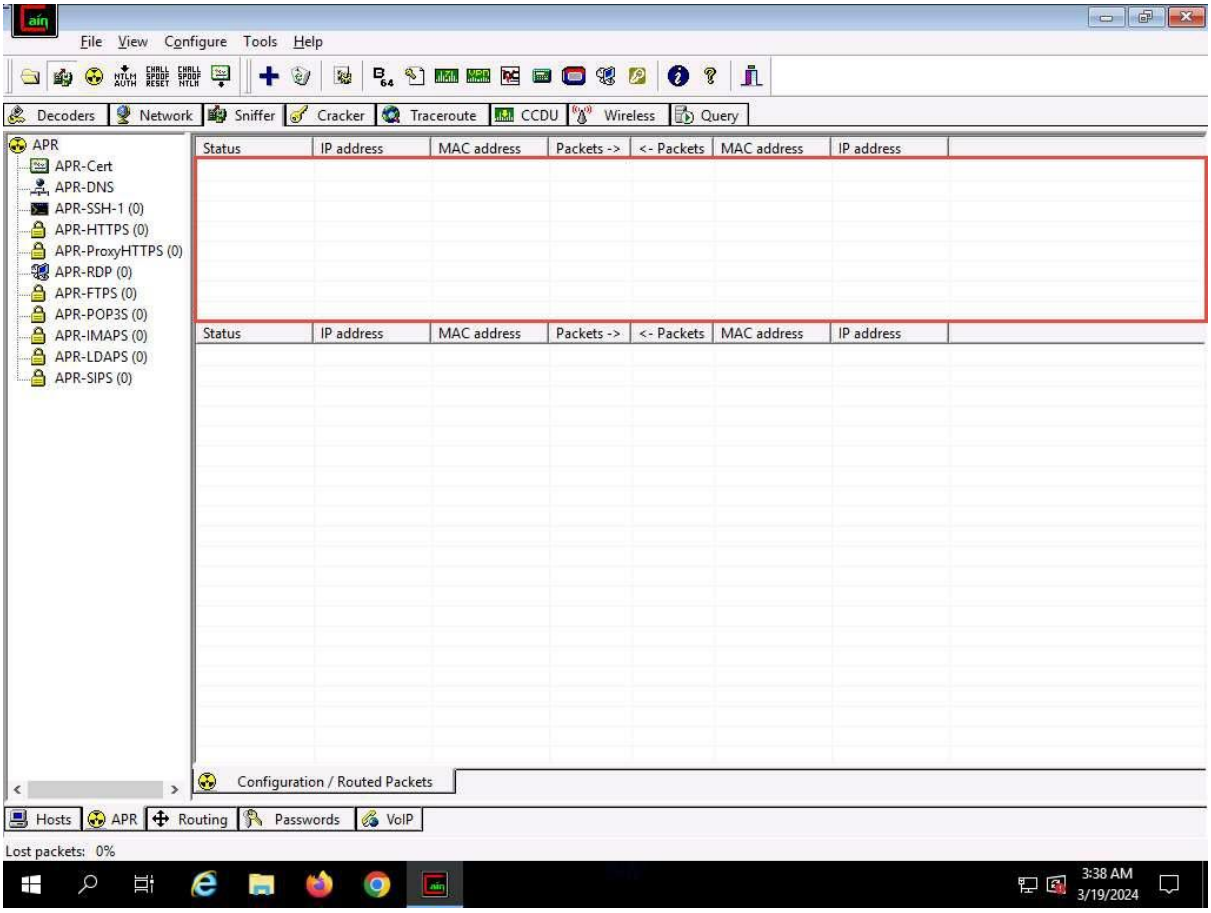

10. Cain & Abel starts scanning for MAC addresses and lists all those found.

11. After the completion of the scan, a list of all active IP addresses along with their corresponding MAC addresses is displayed, as shown in the screenshot.



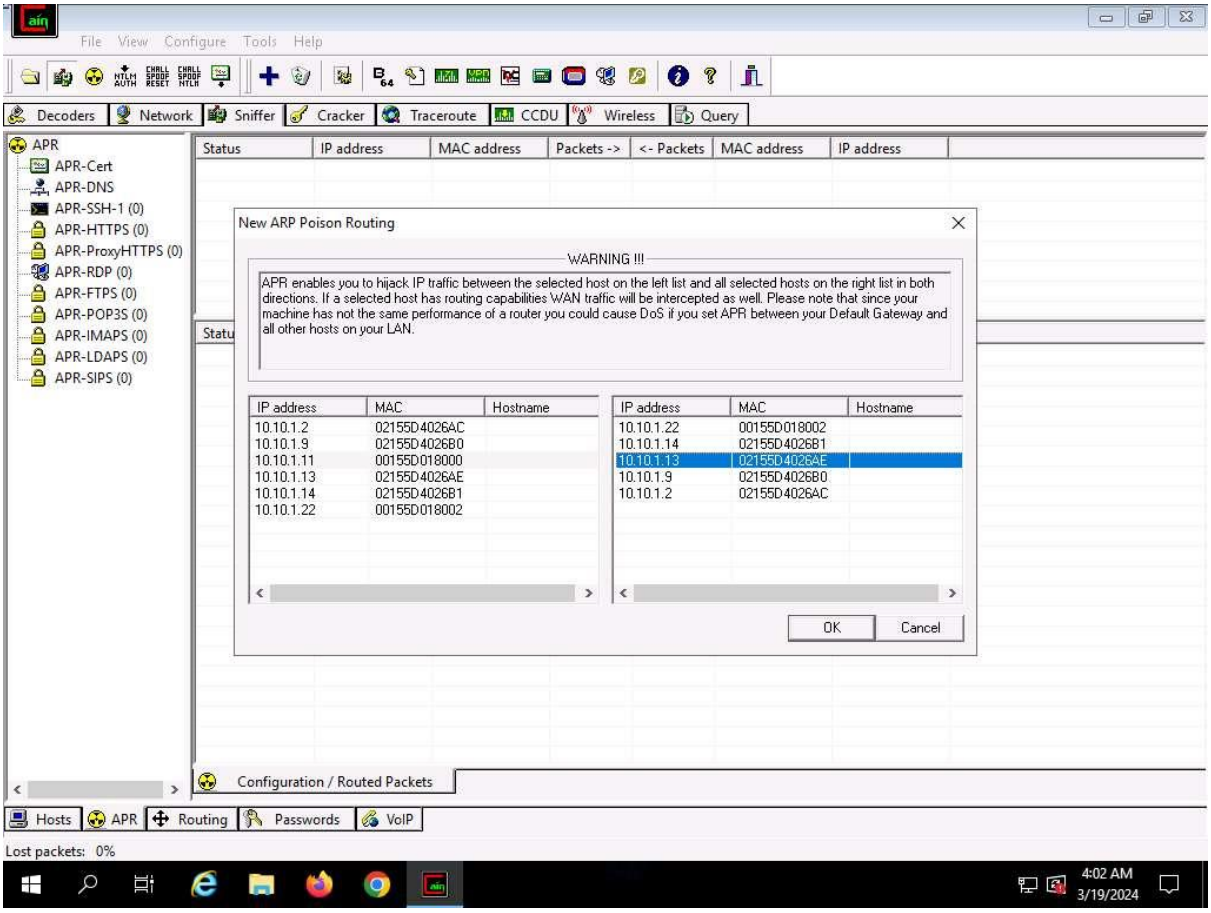

12. Now, click the **APR** tab at the bottom of the window.

13. APR options appear in the left-hand pane. Click anywhere on the topmost section in the right-hand pane to activate the plus (**+**) icon.




14. Click the plus (**+**) icon; a **New ARP Poison Routing** window appears; from which we can add IPs to listen to traffic.

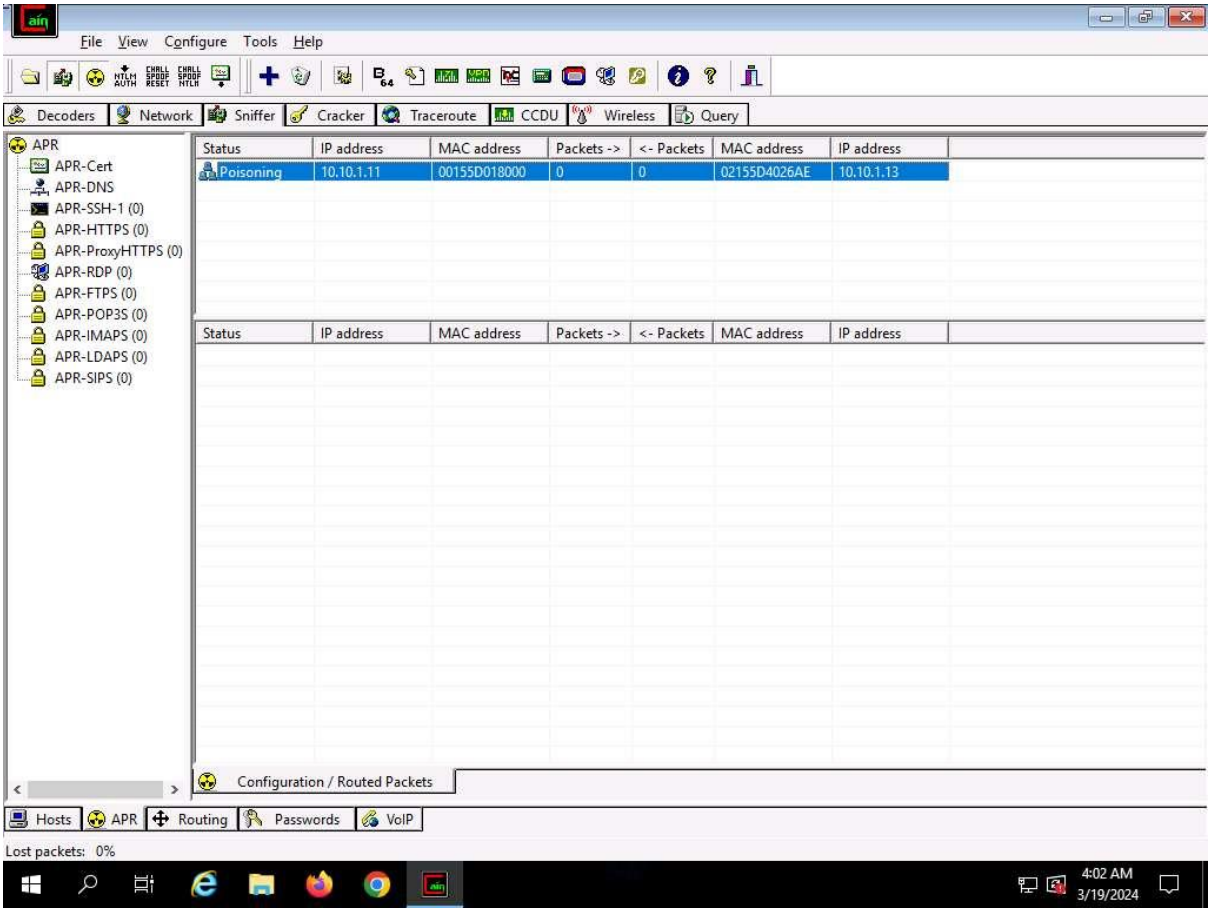
15. To monitor the traffic between two systems (here, **Windows 11** and **Parrot Security**), from the left-hand pane, click to select **10.10.1.11** (**Windows 11**) and from the right-hand pane, click **10.10.1.13** (**Parrot Security**); click **OK**. By doing so, you are setting Cain to perform ARP poisoning between the first and second targets.




16. Click to select the created target IP address scan that is displayed in the **Configuration / Routed Packets** tab.

17. Click on the **Start/Stop APR** icon to start capturing ARP packets.

18. After clicking on the **Start/Stop APR** icon, Cain & Abel starts ARP **poisoning** and the status of the scan changes to Poisoning, as shown in the screenshot.




19. Cain & Abel intercepts the traffic traversing between these two machines.

20. To generate traffic between the machines, you need to ping one target machine using the other.

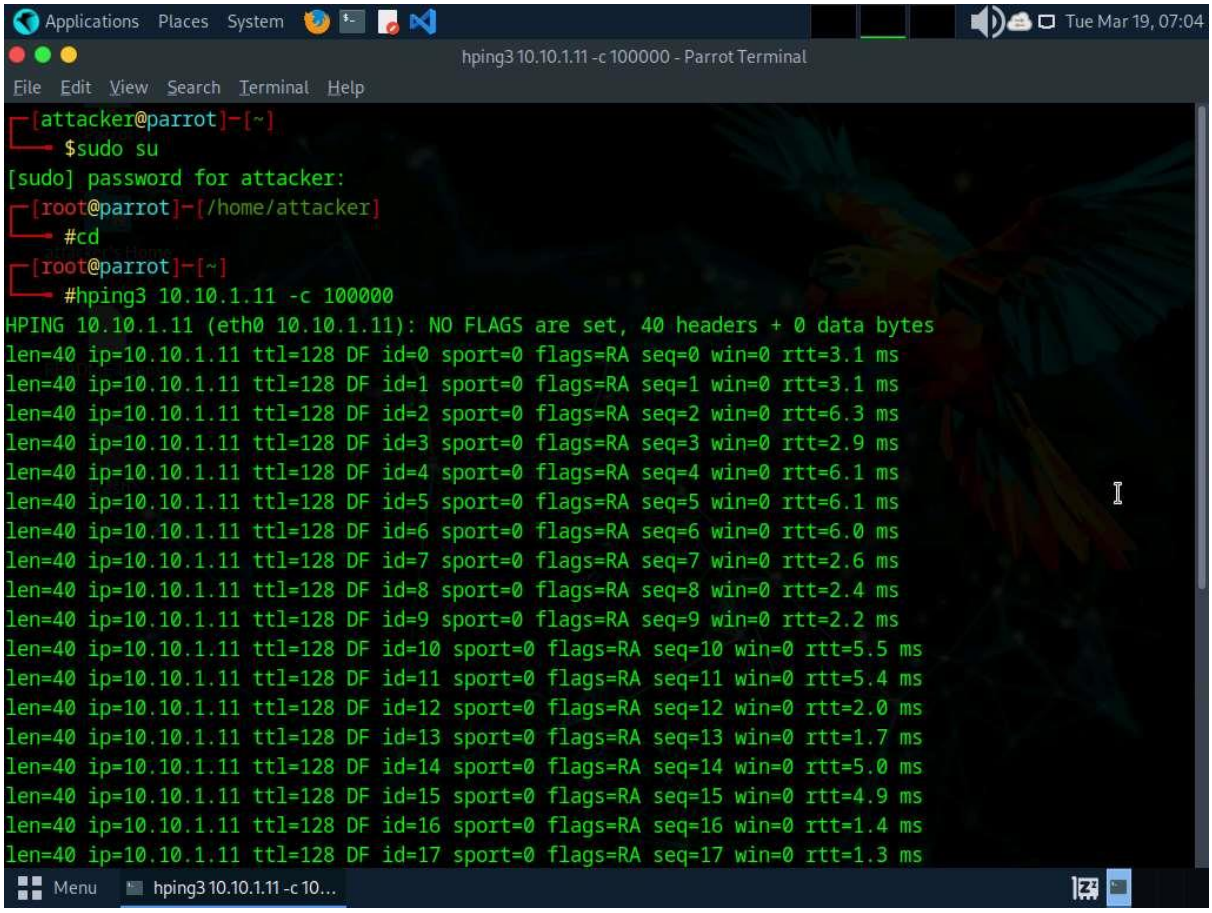
21. Click Parrot Security to switch to the **Parrot Security** machine.

22. Open a **Terminal** window and execute **sudo su** to run the programs as a root user (When prompted, enter the password **toor**). Run **cd** command to jump to root directory.

23. Run **hping3 [Target IP Address] -c 100000** command (here, target IP address is **10.10.1.11 [Windows 11]**).

    **-c**: specifies the packet count.

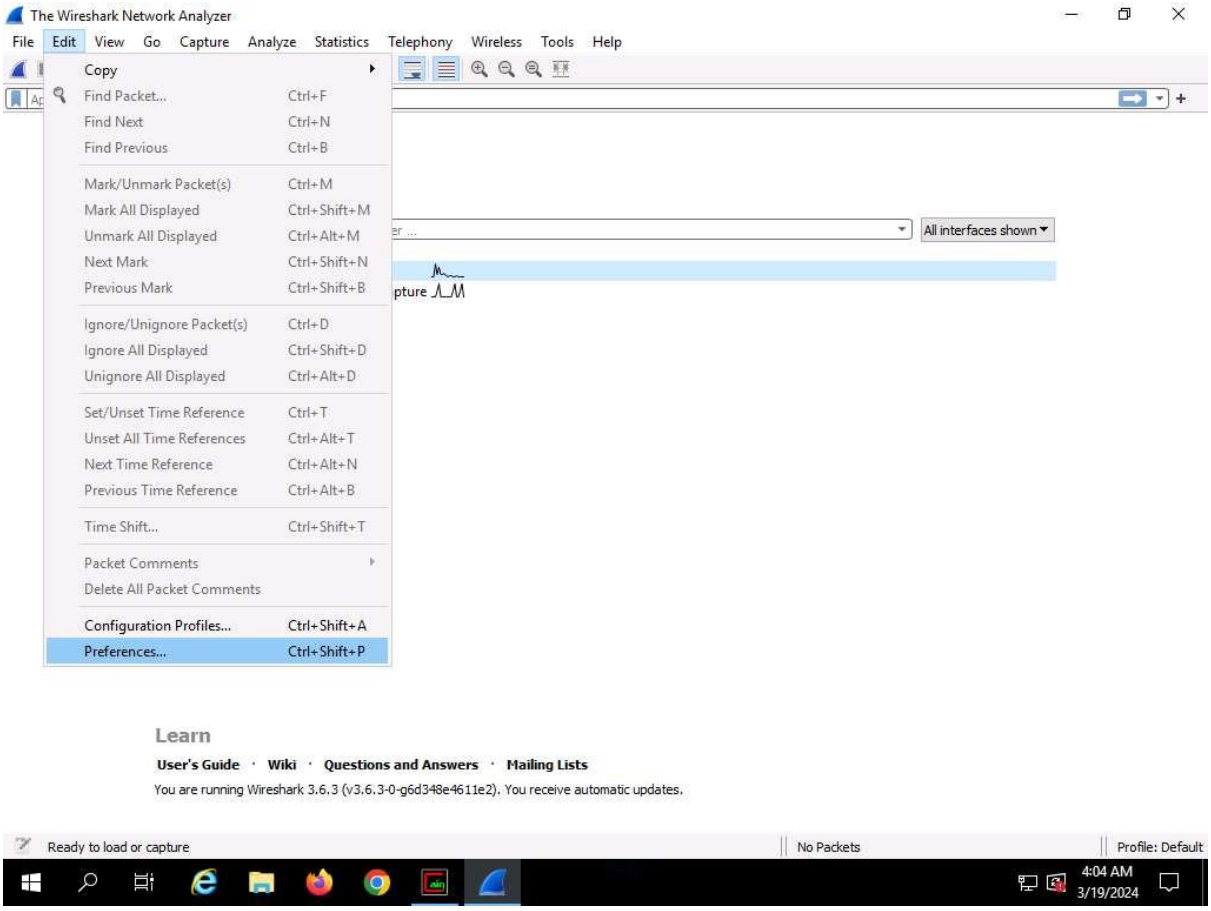
24. This command will start pinging the target machine (**Windows 11**) with 100,000 packets.




25. Leave the command running and immediately click Windows Server 2019 to switch to the **Windows Server 2019** machine.

26. In the **Desktop** window, click windows **Search** icon and search for **wireshark** in the search bar and launch it.

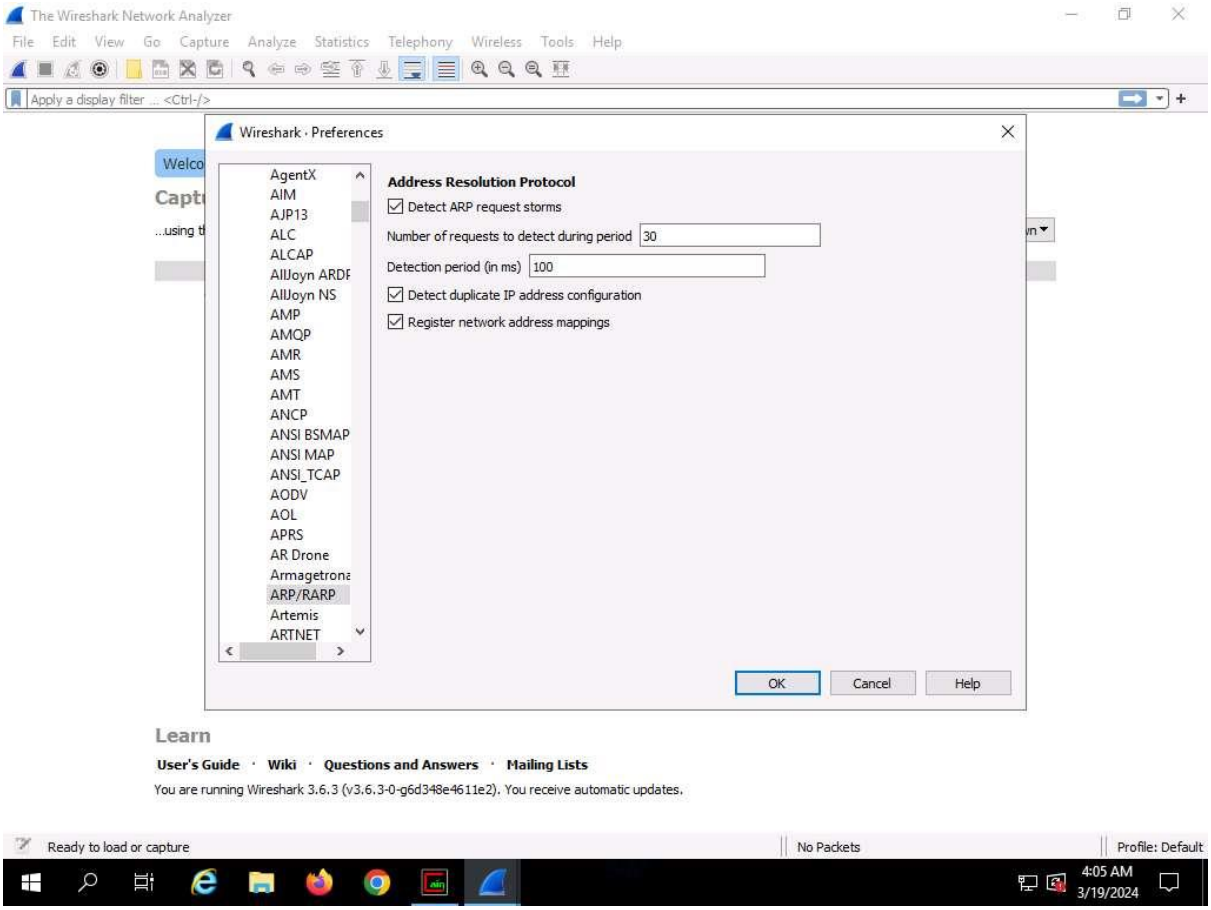
27. The **Wireshark Network Analyzer** window appears; click **Edit** in the menu bar and select **Preferences…**.



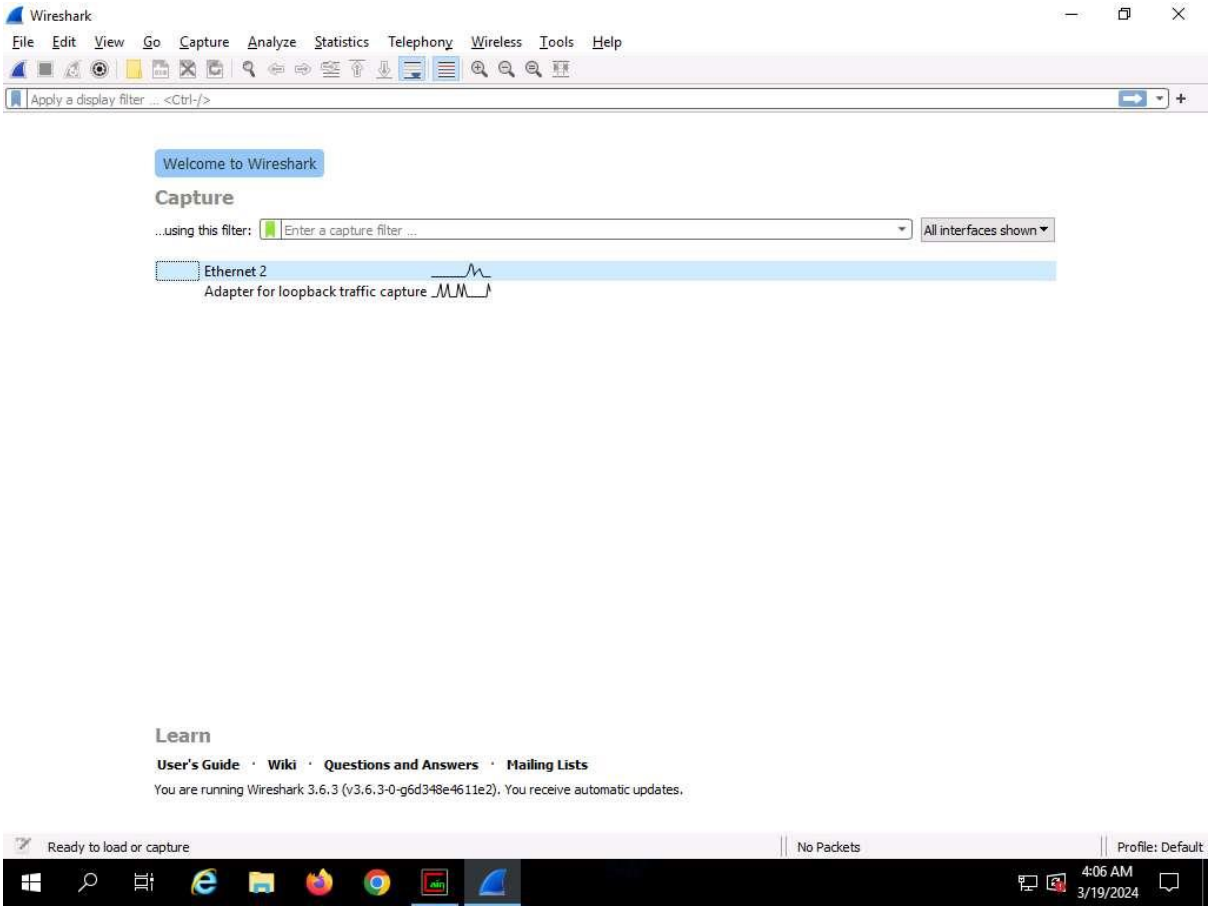

28. The **Wireshark . Preferences** window appears; expand the **Protocols** node.

29. Scroll-down in the **Protocols** node and select the **ARP/RARP** option.

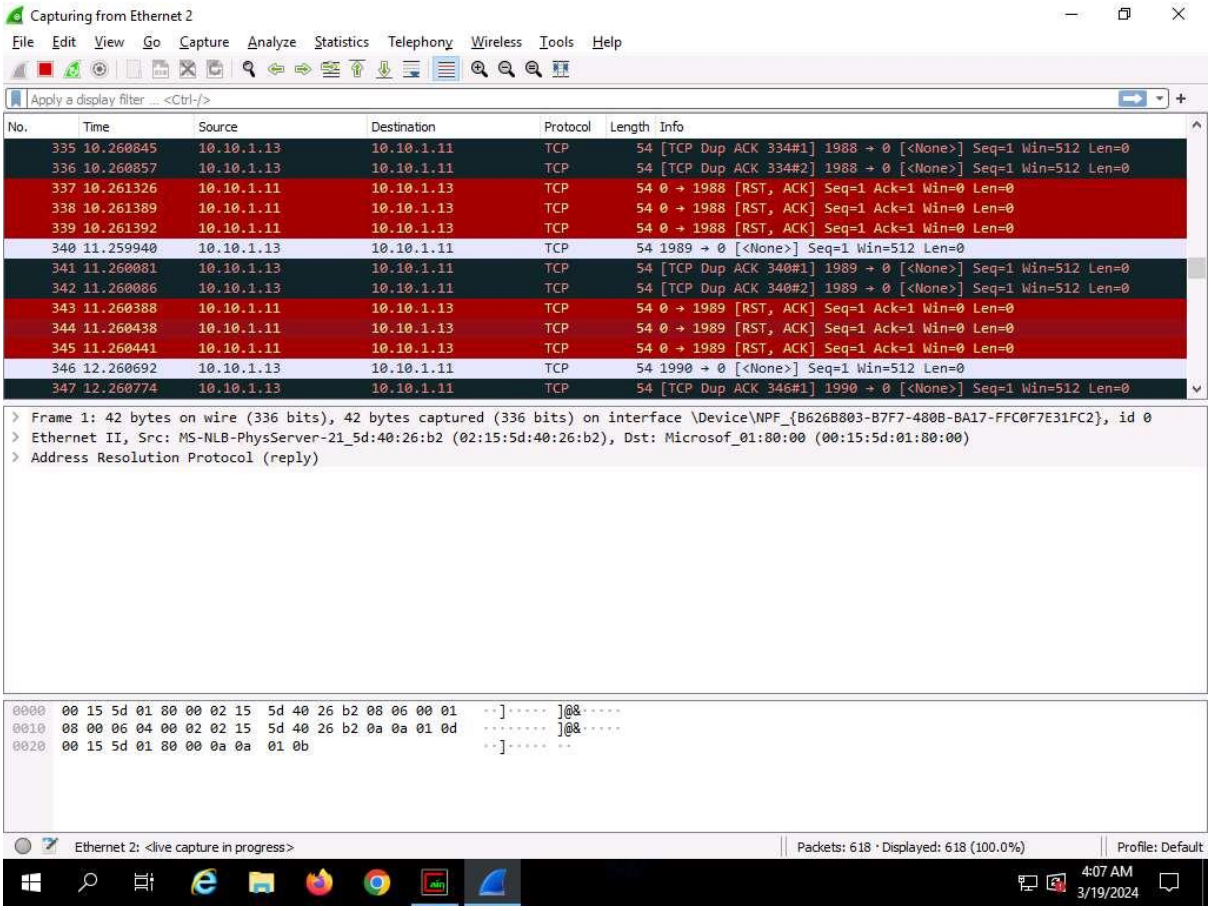
30. From the right-hand pane, click the **Detect ARP request storms** checkbox and ensure that the **Detect duplicate IP address configuration** checkbox is checked; click **OK**.



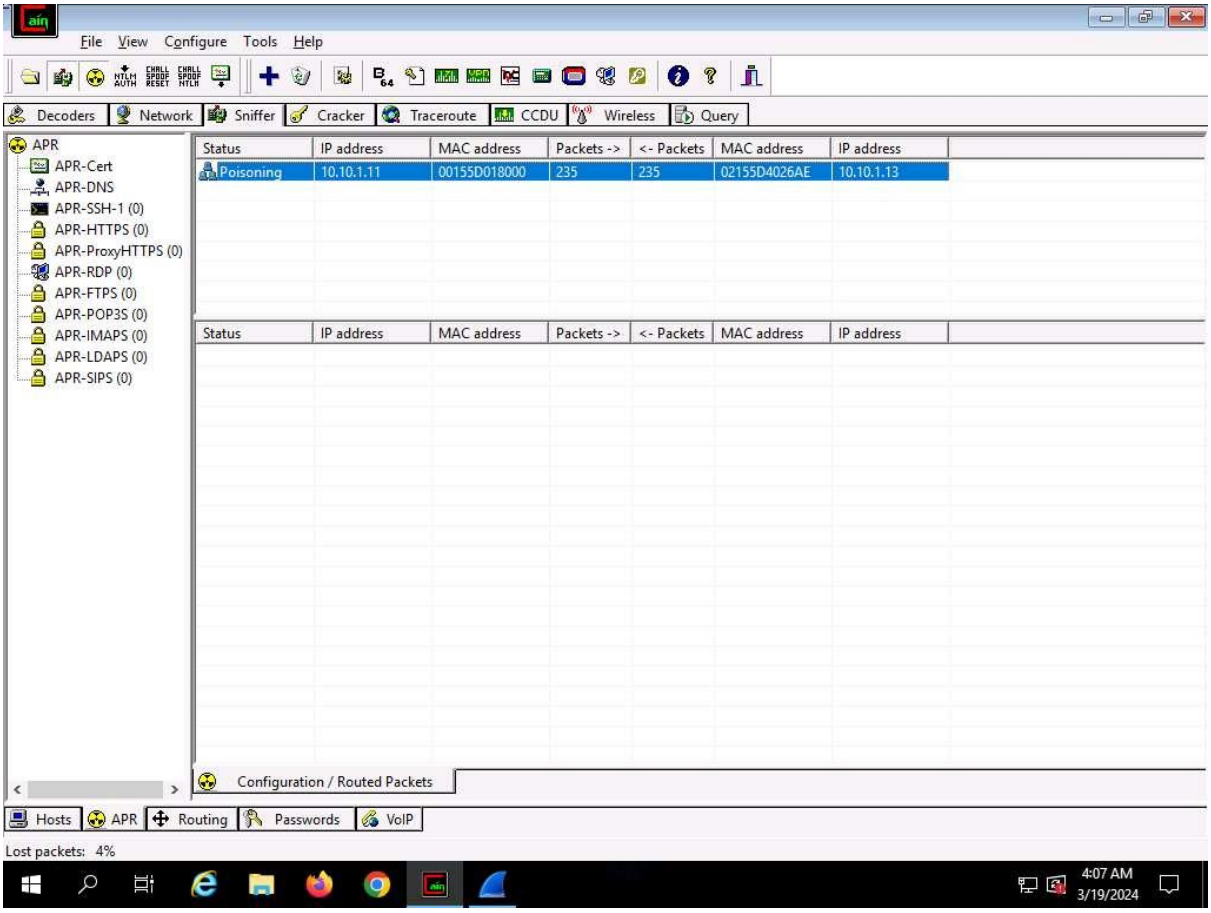

31. Now, double-click on the adapter associated with your network (here, **Ethernet2**) to start capturing the network packets.




32. **Wireshark** begins to capture the traffic between the two machines, as shown in the screenshot.

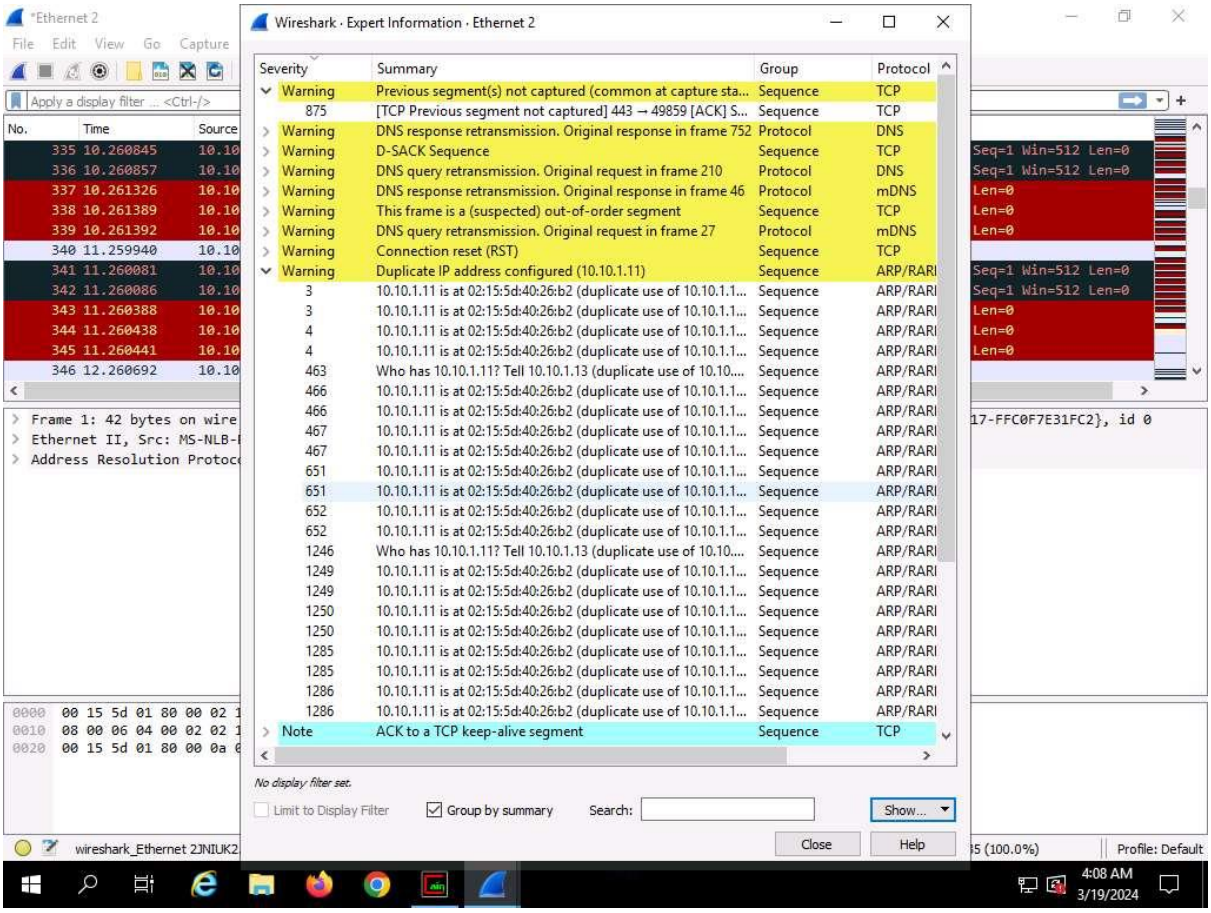



33. Switch to the **Cain & Abel** window to observe the packets flowing between the two machines.



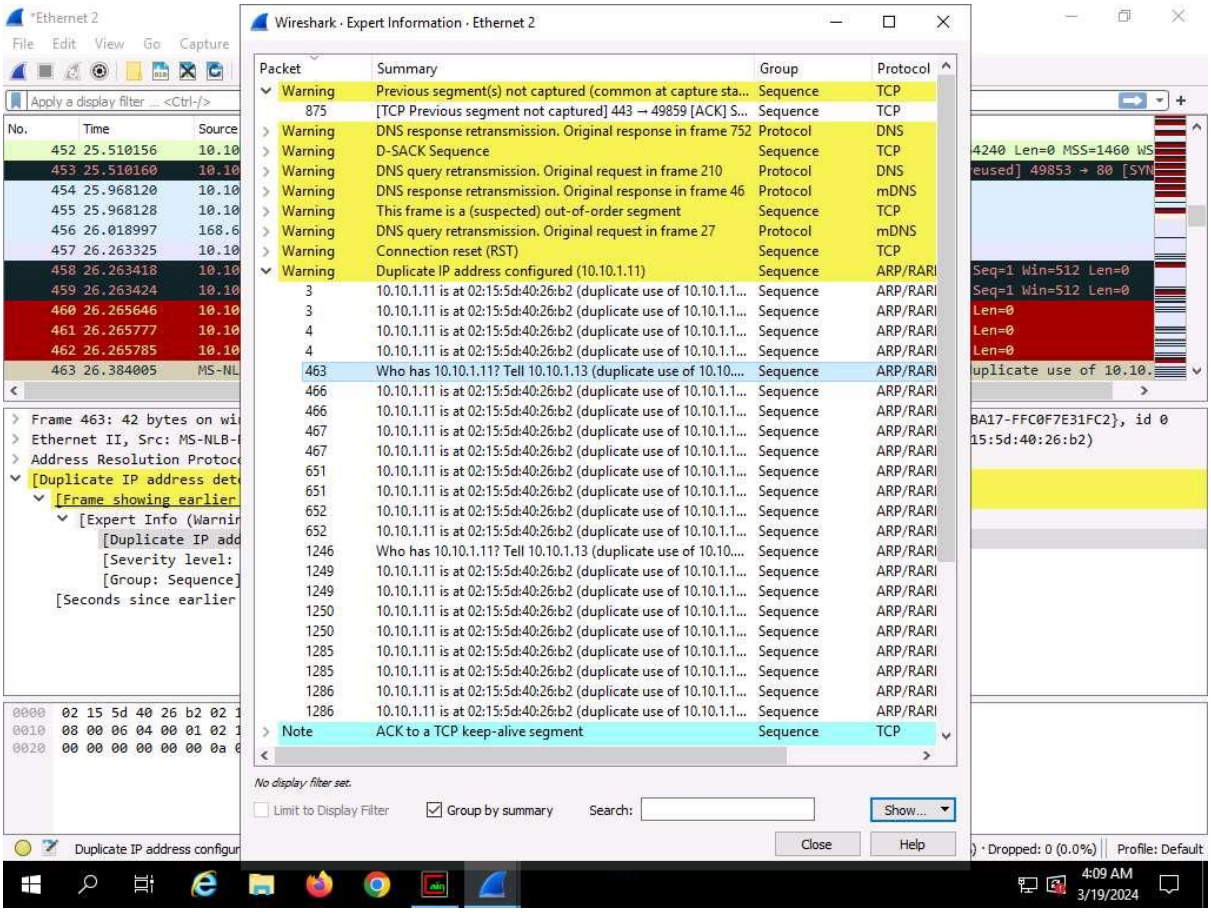

34. Now, switch to **Wireshark** and click the **Stop packet capturing** icon to stop the packet capturing.

35. Click **Analyze** from the menu bar and select **Expert Information** from the drop-down options. The **Wireshark . Expert Information** window appears; click to expand the **Warning** node labeled **Duplicate IP address configured (10.10.1.11)**, running on the **ARP/RARP** protocol.



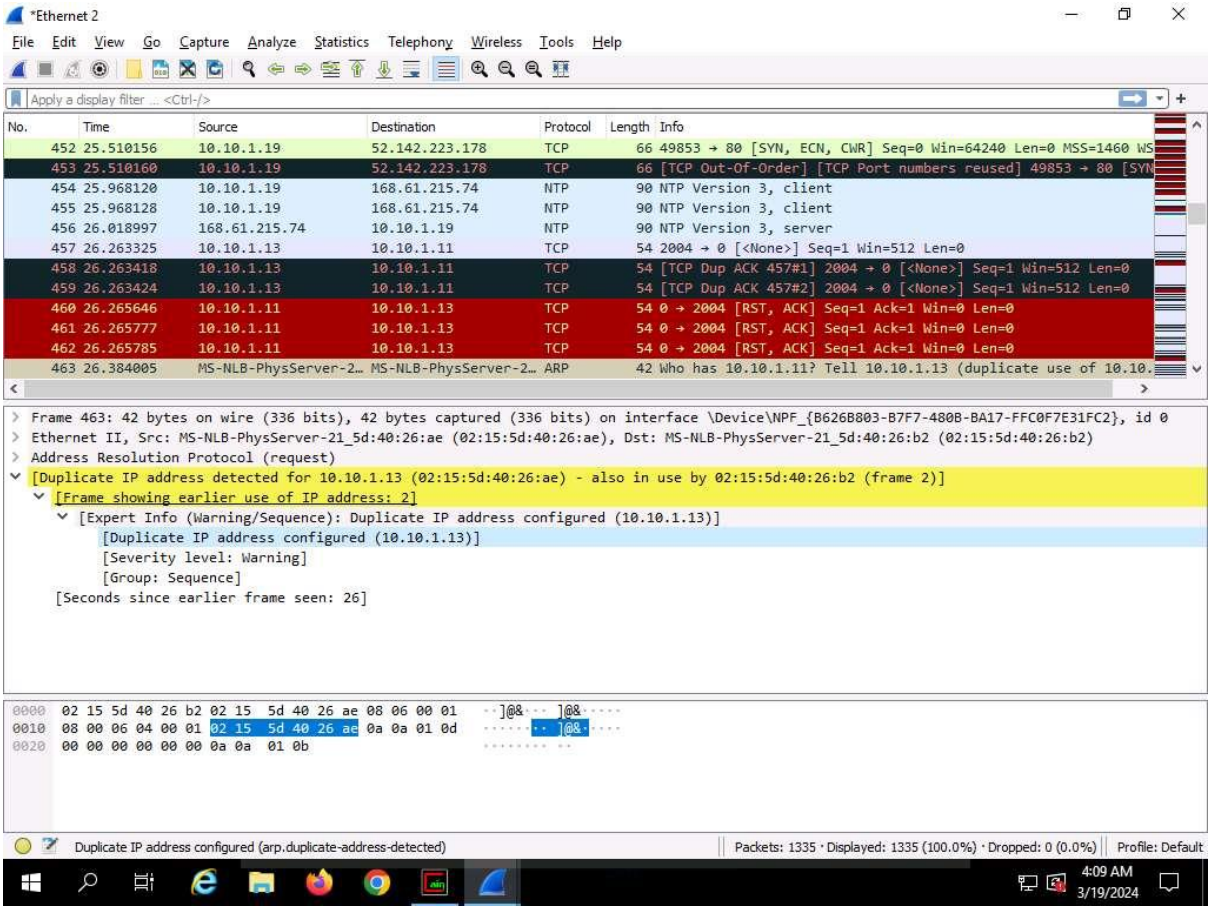

36. Arrange the **Wireshark . Expert Information** window above the **Wireshark** window so that you can view the packet number and the **Packet details** section.

37. In the **Wireshark . Expert Information** window, click any packet (here, **463**).




38. On selecting the packet number, **Wireshark** highlights the packet, and its associated information is displayed under the packet details section. Close the **Wireshark . Expert Information** window.

39. The warnings highlighted in yellow indicate that duplicate IP addresses have been detected at one MAC address, as shown in the screenshot.




ARP spoofing succeeds by changing the IP address of the attacker's computer to the IP address of the target computer. A forged ARP request and reply packet find a place in the target ARP cache in this process. As the ARP reply has been forged, the destination computer (target) sends frames to the attacker's computer, where the attacker can modify the frames before sending them to the source machine (User A) in an MITM attack. At this point, the attacker can launch a DoS attack by associating a non-existent MAC address with the IP address of the gateway or may passively sniff the traffic, and then forward it to the target destination.

40. This concludes the demonstration of detecting ARP poisoning in a switch-based network.

41. Close the **Wireshark** window and leave all other windows running.

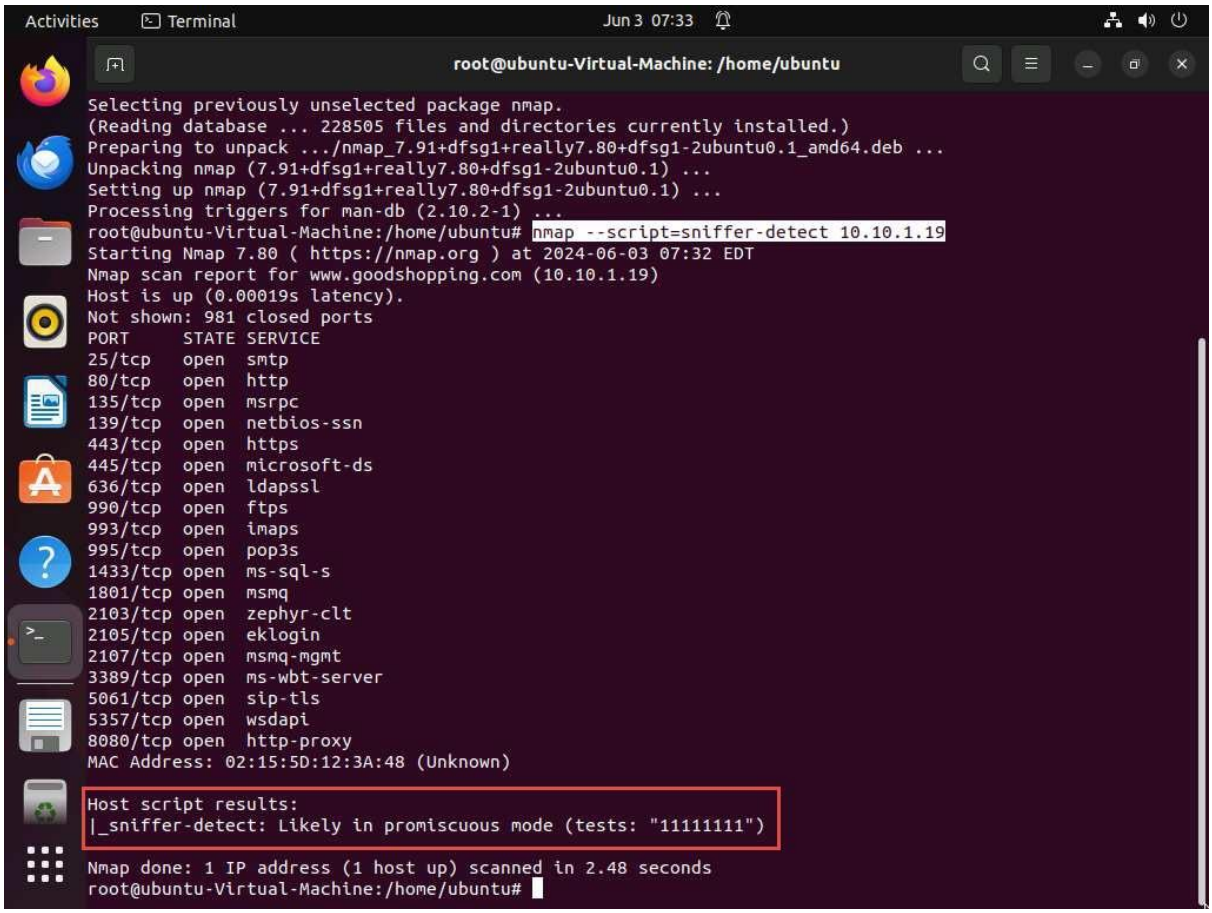
42. Now, we shall perform promiscuous mode detection using Nmap.

43. Now, Click Ubuntu to switch to the **Ubuntu** machine and login with **Ubuntu**/**toor**.

44. In the **Ubuntu** machine, open a **Terminal** window and execute **sudo su** to run the programs as a root user (When prompted, enter the password **toor**)

45. Run **nmap --script=sniffer-detect [Target IP Address/ IP Address Range]** (here, target IP address is **10.10.1.19 [Windows Server 2019]**) to start scanning.

46. The scan results appear, displaying **Likely in promiscuous mode** under the **Host script results** section. This indicates that the target system is in promiscuous mode.




47. Close the terminal window and document all the acquired information.

48. Close all open windows in all machines (ensure that ARP poisoning is not running in **Windows Server 2019**), and document all the acquired information.

**Question 8.3.1.1**

Use Cain and Abel on the Windows Server 2019 machine to perform ARP poisoning, and sniff traffic between the Windows 11 and Parrot Security machines. Further, use Wireshark on the same Windows Server 2019 machine to detect ARP poisoning. What is the severity level of ARP/RARP packets as shown in the expert information window of Wireshark?

**Question 8.3.1.2**

Use the Nmap Scripting Engine (NSE) to check if a system on the local Ethernet has its network card in the promiscuous mode. Which Nmap NSE script detects if a network interface is in the promiscuous mode?