CompTIA Network+ N10-009

Networking Ports and Protocols

Introduction

N10-009
Protocols
Ping
Port Scan
DNS Query

Welcome to the **Networking Ports and Protocols** Practice Lab. In this module, you will be provided with the instructions and devices needed to develop your hands-on skills.

Network ports and protocols are essential components of network communication, facilitating the exchange of data between devices. Ports are logical endpoints for communication, with each port corresponding to a specific service or application running on a device. Protocols, on the other hand, define the rules and formats for data exchange between devices.

In this module, you will explore and learn about network ports and protocols.

Learning Outcomes

In this module, you will complete the following exercises:

- Exercise 1 - Discover Protocols with Wireshark
- Exercise 2 - Network Port Scan

After completing this module, you should be able to:

- Conduct a Ping

- Conduct a DNS Query
- Make an SSH Connection
- Access an HTTP Web Server
- Conduct a Network Port Scan

Exam Objectives

The following exam objectives are covered in this module:

1.4 Explain common networking ports, protocols, services, and traffic types

- Protocols
- Ports
- Internet Protocol (IP) types

Lab Duration

It will take approximately **1 hour** to complete this lab.

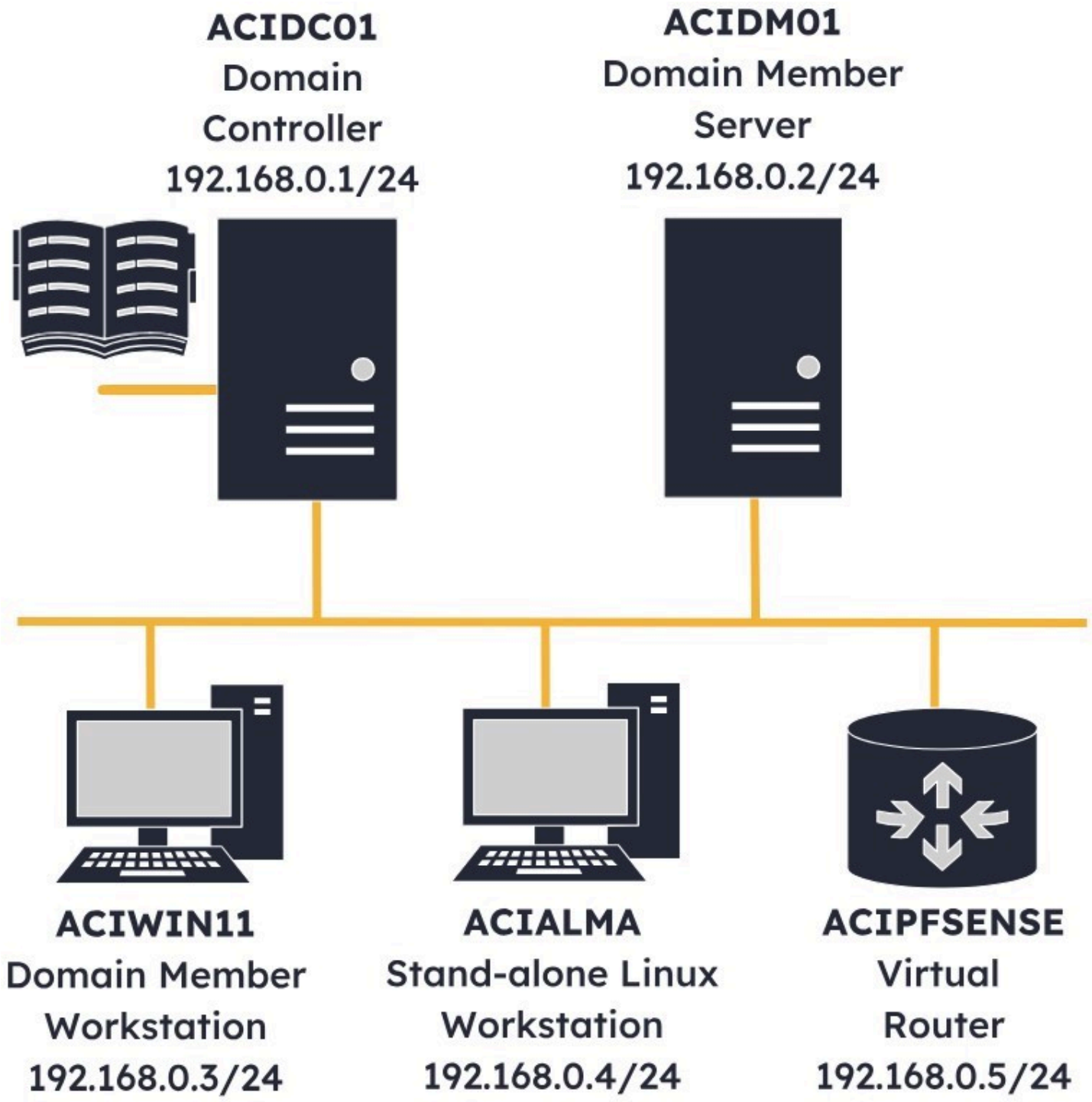
Help and Support

For more information on using Practice Labs, please see our **Help and Support** page. You can also raise a technical support ticket from this page.

Click **Next** to view the Lab topology used in this module.

Lab Topology

During your session, you will have access to the following lab configuration.



Depending on the exercises, you may or may not use all of the devices, but they are shown here in the layout to get an overall understanding of the topology of the lab.

- **ACIDC01 -** Windows Server 2022 - Domain Controller
- **ACIDM01 -** Windows Server 2022 - Domain Member Server
- **ACIWIN11 -** Windows 11 PRO - Domain Member Workstation
- **ACIALMA -** Alma Linux 9.3 - Stand-alone Linux Workstation
- **ACIPFSENSE -** PFSense v2.7.2 - Virtual Router

Click **Next** to proceed to the first exercise.

Exercise 1 - Discover Protocols with Wireshark

Wireshark is a network protocol analyzer. It allows users to capture and review network traffic on a specific network interface. It provides detailed information about network packets, including their source, destination, protocols used, and even the contents of the data payload.

In this exercise, you will utilize Wireshark to analyze network communication.

Learning Outcomes

After completing this exercise, you should be able to:

- Conduct a Ping
- Conduct a DNS Query
- Make an SSH Connection
- Access an HTTP Web Server

Your Devices

You will be using the following devices in this lab. Please power these on now.

- **ACIDC01 -** Windows Server 2022 - Domain Controller
- **ACIWIN11 -** Windows 11 PRO - Domain Member Workstation
- **ACIALMA -** Alma Linux 9.3 - Stand-alone Linux Workstation
- **ACIPFSENSE -** PFSense v2.7.2 - Virtual Router

Task 1 - Conduct a Ping

Ping is a tool that tests connectivity between hosts on an Internet Protocol (IP) network. It works by sending ICMP (Internet Control Message Protocol) echo request packets to the destination host and listening for an echo reply response. In addition to validating a communication path, ping measures the round-trip time for packets to travel from the source to the destination and back, providing network latency and packet loss information.

In this task, you will monitor and analyze a ping with Wireshark.

Step 1

Connect to **ACIWIN11.**

In the **Search** field, type the following:

```
wireshark
```

Select **Wireshark** from the **Best match** pop-up menu.

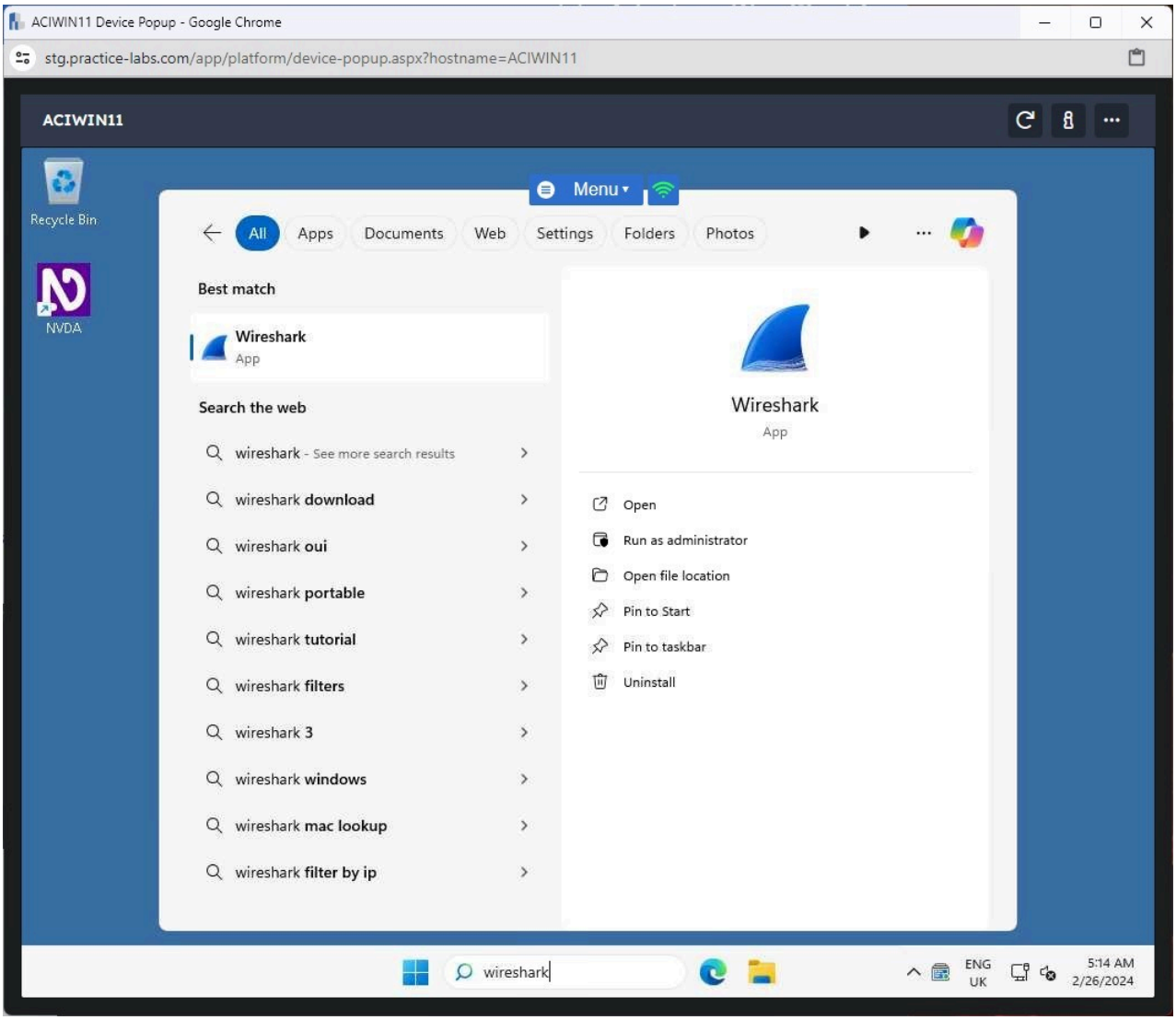


Figure 1.1 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Best match pop-up menu.

Step 2

In **Wireshark,** under the **Capture** menu, double-click on **Ethernet.**

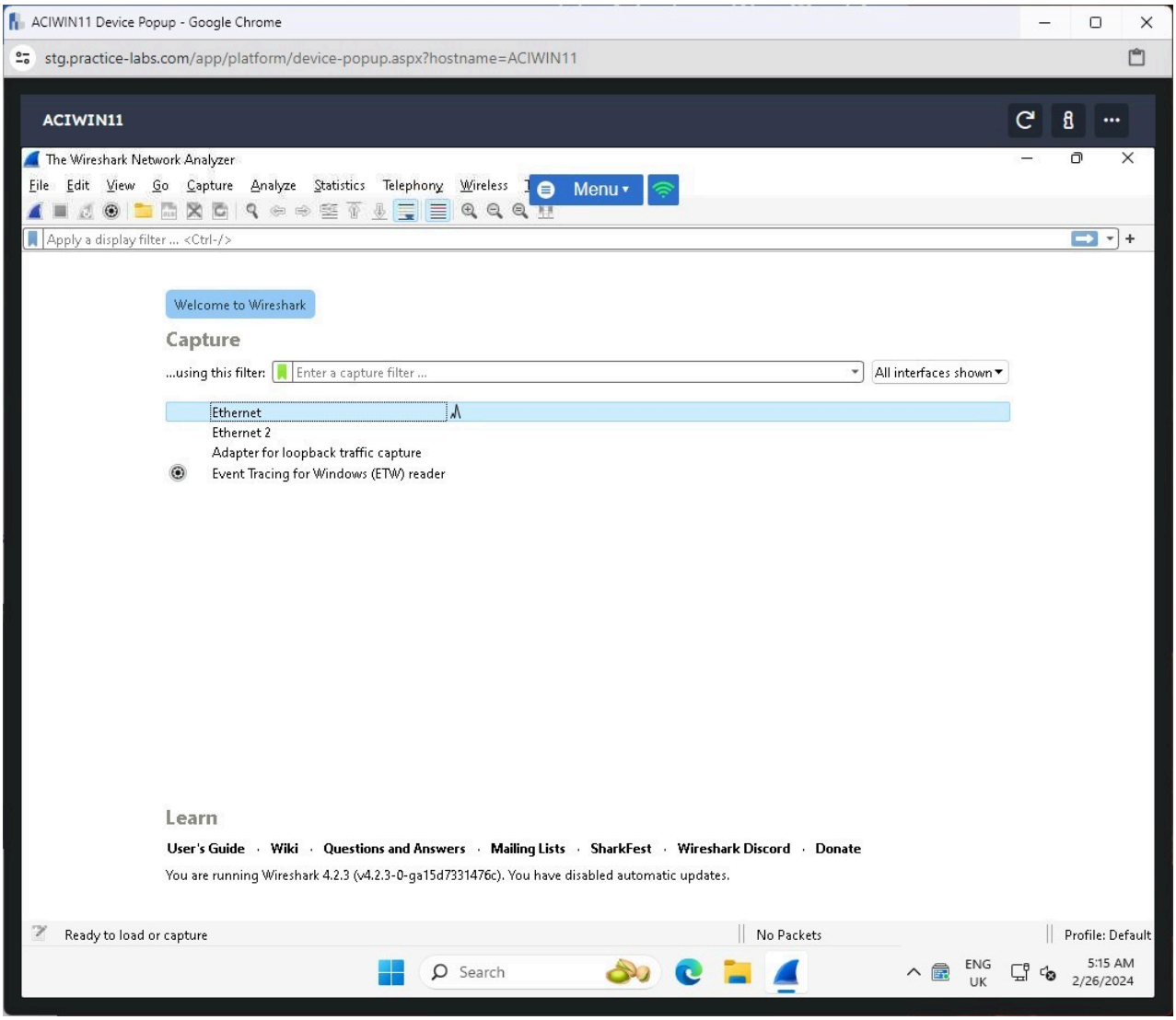


Figure 1.2 Screenshot of ACIWIN11: Displaying the Wireshark Capture menu and starting a capture of the Ethernet interface.

Note: *Wireshark allows users to select a specific network interface for packet capture, such as Ethernet, Ethernet 2, Wi-Fi, or virtual interfaces. This feature enables users to monitor and analyze, in real time, traffic on only the chosen network interface. As a result, Wireshark provides detailed insights into network activity, enabling users to diagnose issues, troubleshoot problems, and analyze network performance.*
In this instance, the Ethernet adapter is in use and should be used for the capturing of packets by Wireshark.

Step 3

In the **Taskbar - Search** field, type the following:

```
cmd
```

Select **Command Prompt** from the **Best match** pop-up menu.

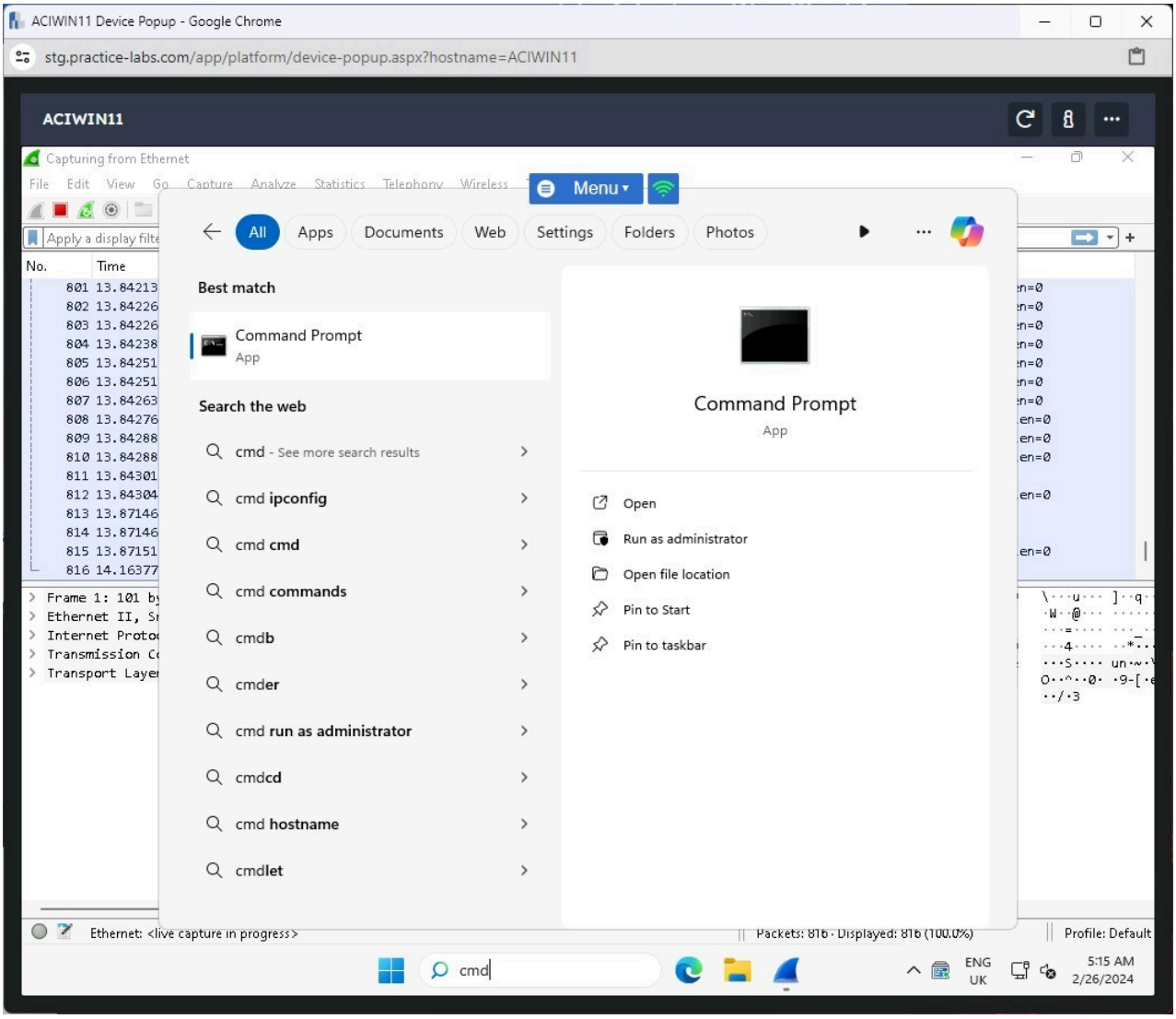


Figure 1.3 Screenshot of ACIWIN11: Displaying selecting Command Prompt from the Best match pop-up menu.

Step 4

In the **Command Prompt** window, type the following:

```
ping 192.168.0.1
```

Press **Enter.**

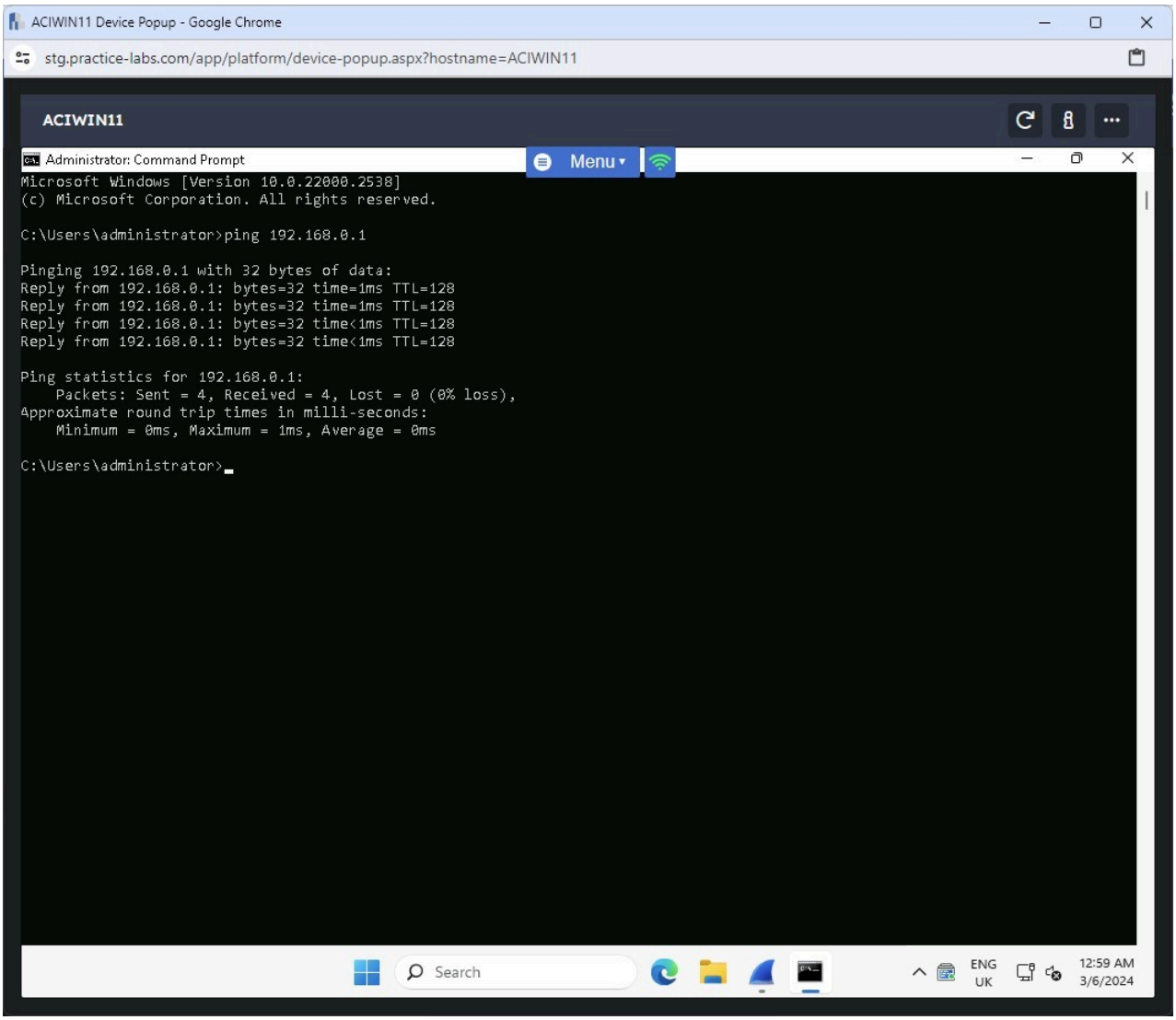


Figure 1.4 Screenshot of ACIWIN11: Displaying the Command Prompt and pinging ACIDC01.

Note: *The ping command sends ICMP echo request packets from the source (ACIWIN11) to the destination (ACIDC01). Upon the receipt of these packets, the destination device responds with ICMP echo reply packets, allowing the initiating source to measure the round-trip time and determine the status of the connection.*
By default, four pings are conducted to the destination.

Step 5

On the **Taskbar**, select **Wireshark.**

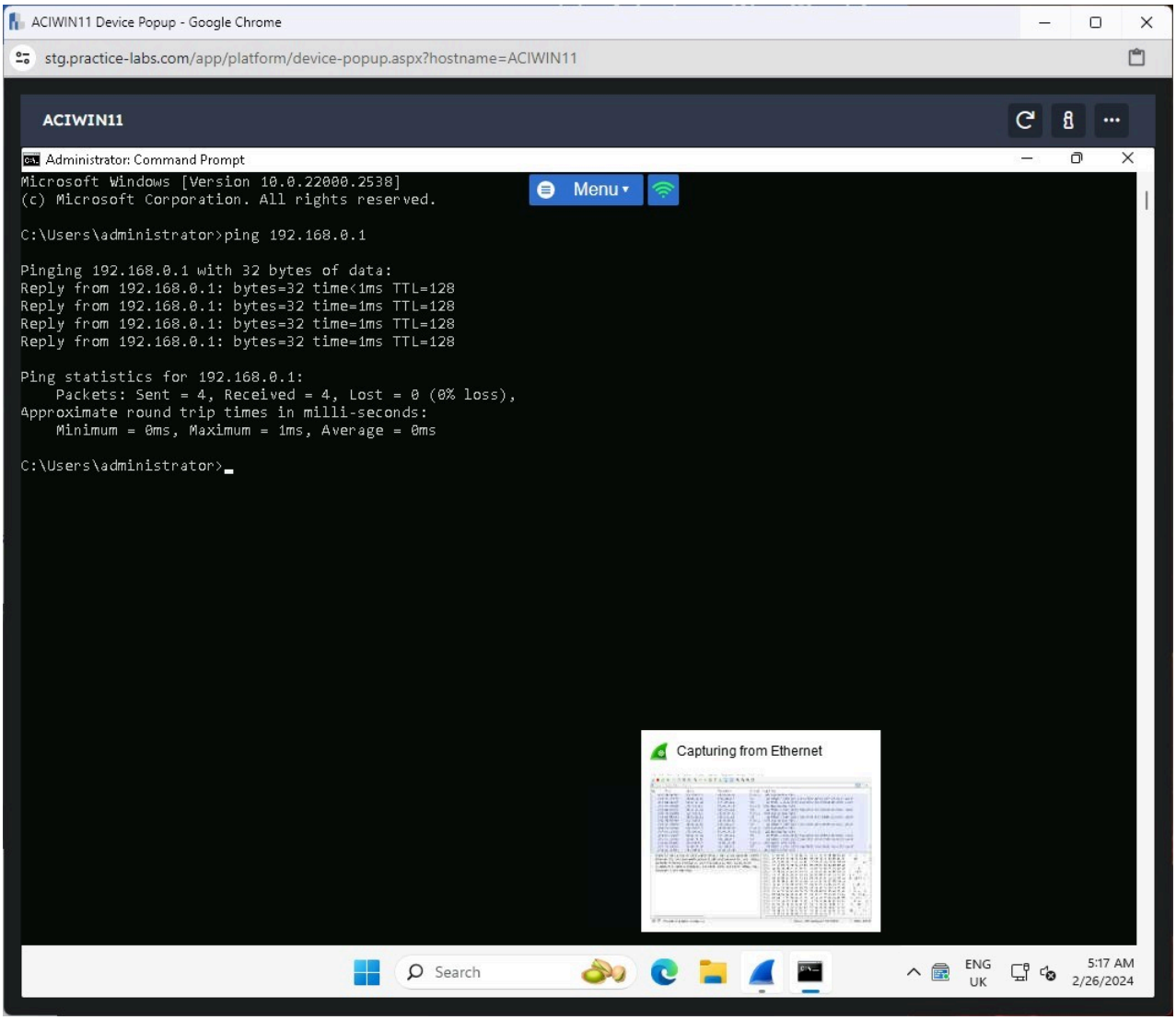


Figure 1.5 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Taskbar.

Step 6

In **Wireshark,** click on the red square to **Stop capturing packets.**

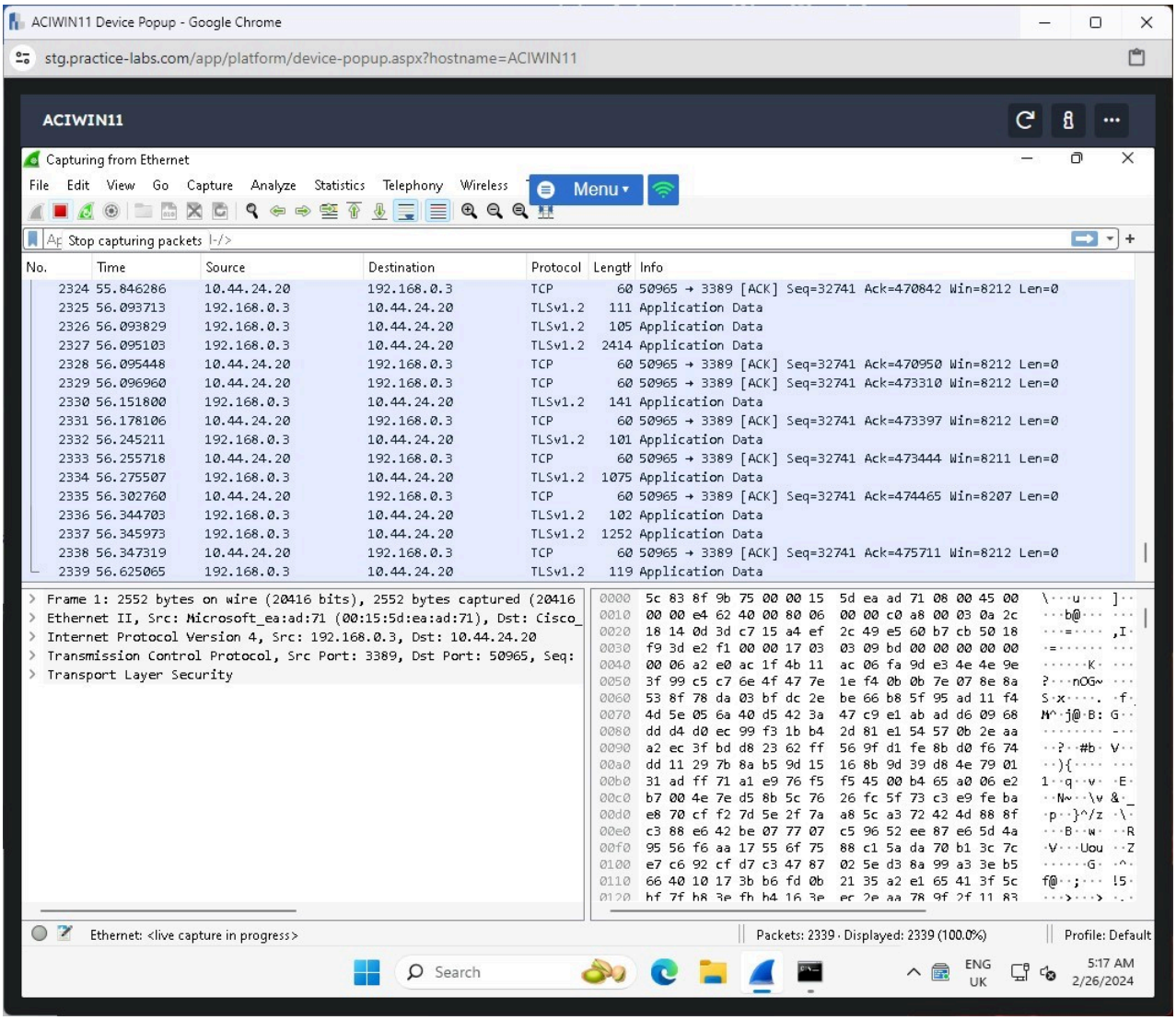


Figure 1.6 Screenshot of ACIWIN11: Displaying Wireshark and selecting the red square to Stop capturing packets.

Note: *Wireshark's GUI (Graphical User Interface) is comprised of three main sections:*
1. The Packet List pane (top), which displays captured packets in a tabular format and provides details such as packet number, time, source, destination, protocol, and length.
2. The Packet Details pane (bottom left), which presents a more in-depth analysis of the selected packet, offering a hierarchical view of its protocol layers and their respective field
3. The Packet Bytes pane (bottom right), which allows users to view the raw hexadecimal and ASCII representation of the packet's payload.

Step 7

In **Wireshark,** in the **Apply a display filter** field, type the following:

```
icmp
```

Press **Enter.**

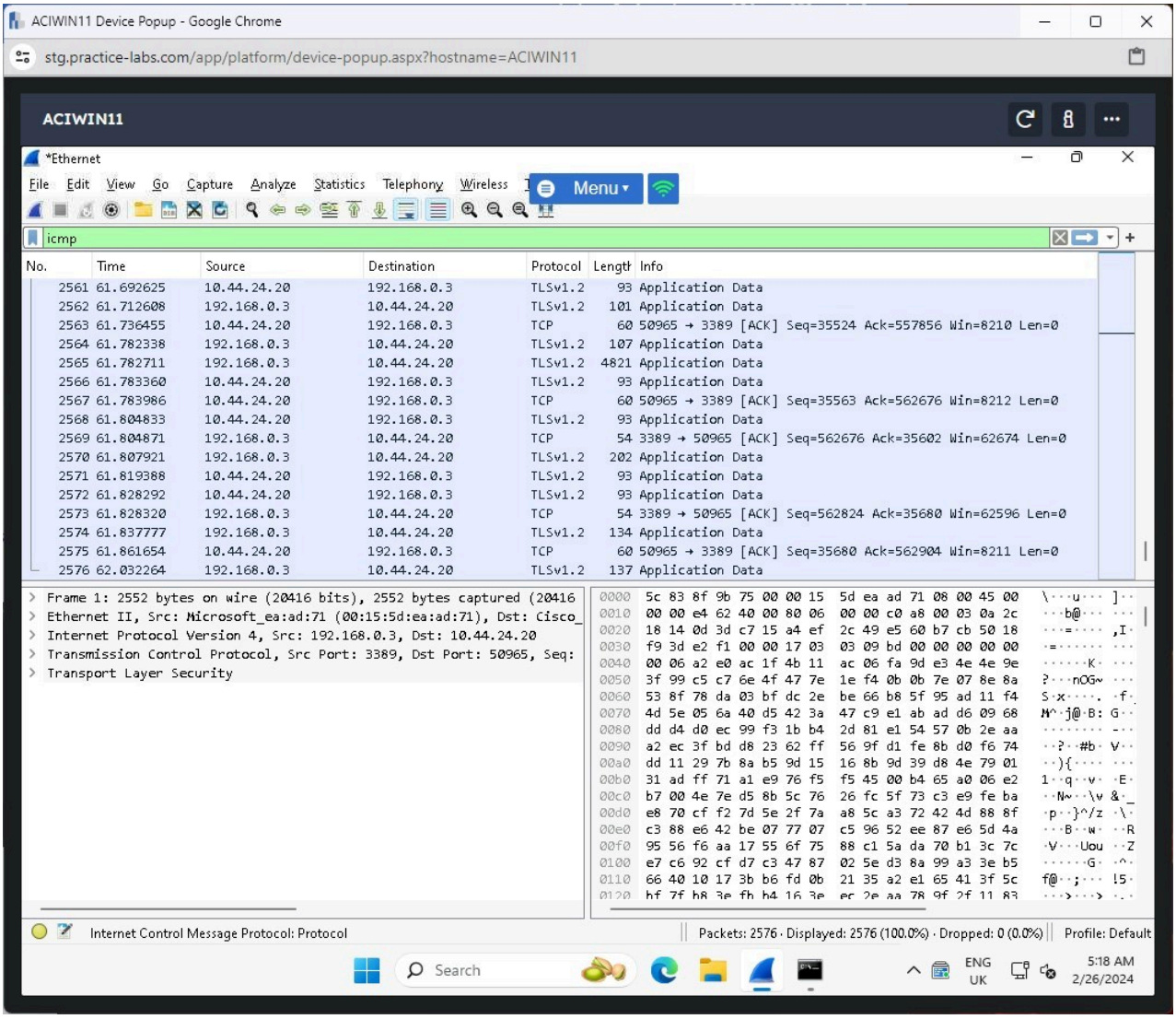


Figure 1.7 Screenshot of ACIWIN11: Displaying Wireshark and typing icmp in the Apply a display filter field.

Step 8

In **Wireshark,** select the first packet with a **Source** of **192.168.0.3** (the ACIWIN11 machine).

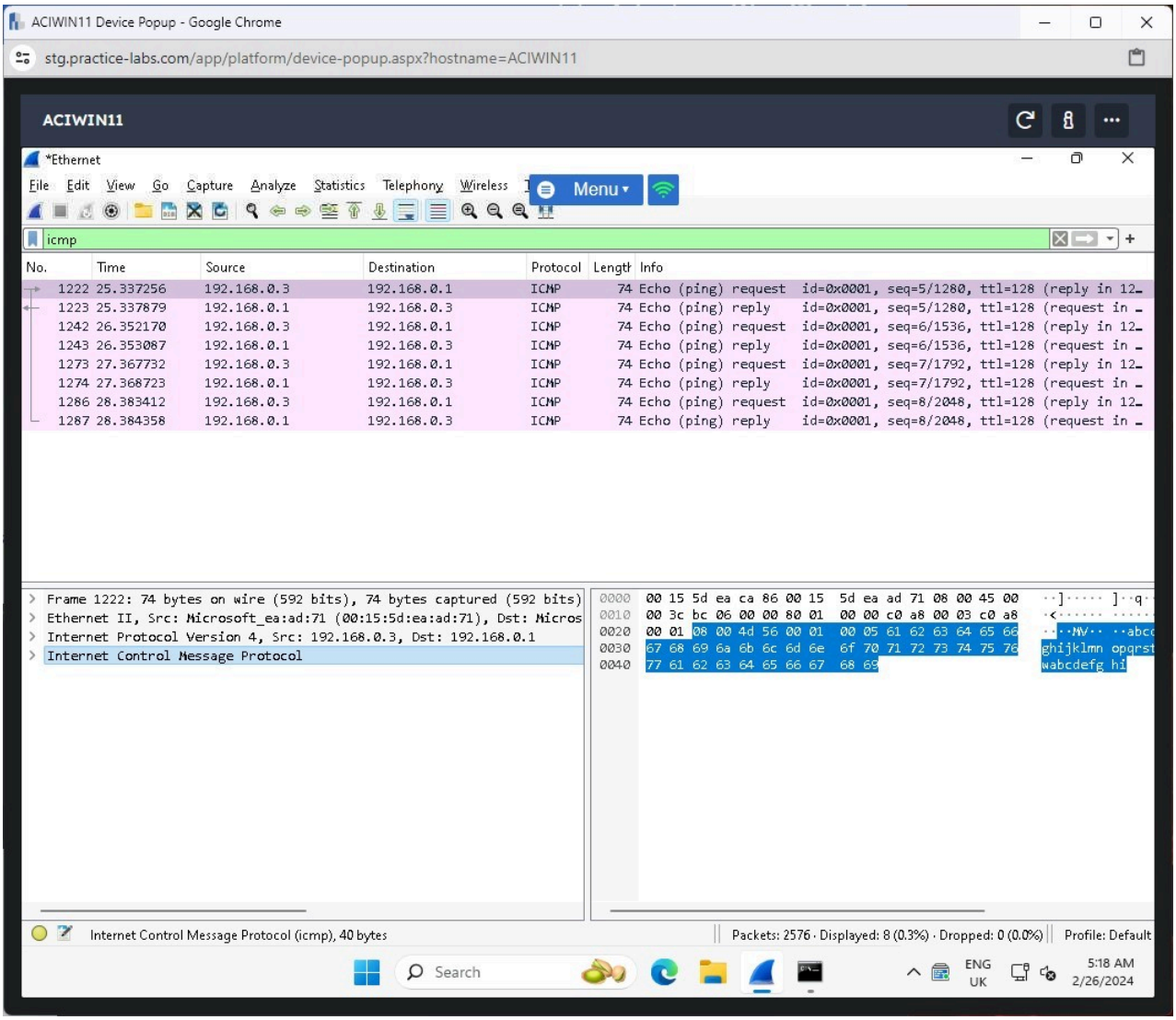


Figure 1.8 Screenshot of ACIWIN11: Displaying Wireshark and observing the first packet from 192.168.0.3.

> ***Note:*** *In Wireshark, selecting a specific frame in the Packets list pane allows users to analyze the details and contents of that packet in the Packet Details and Packets Bytes panes.*

Step 9

In **Wireshark**, expand the **Internet Control Message Protocol** field.

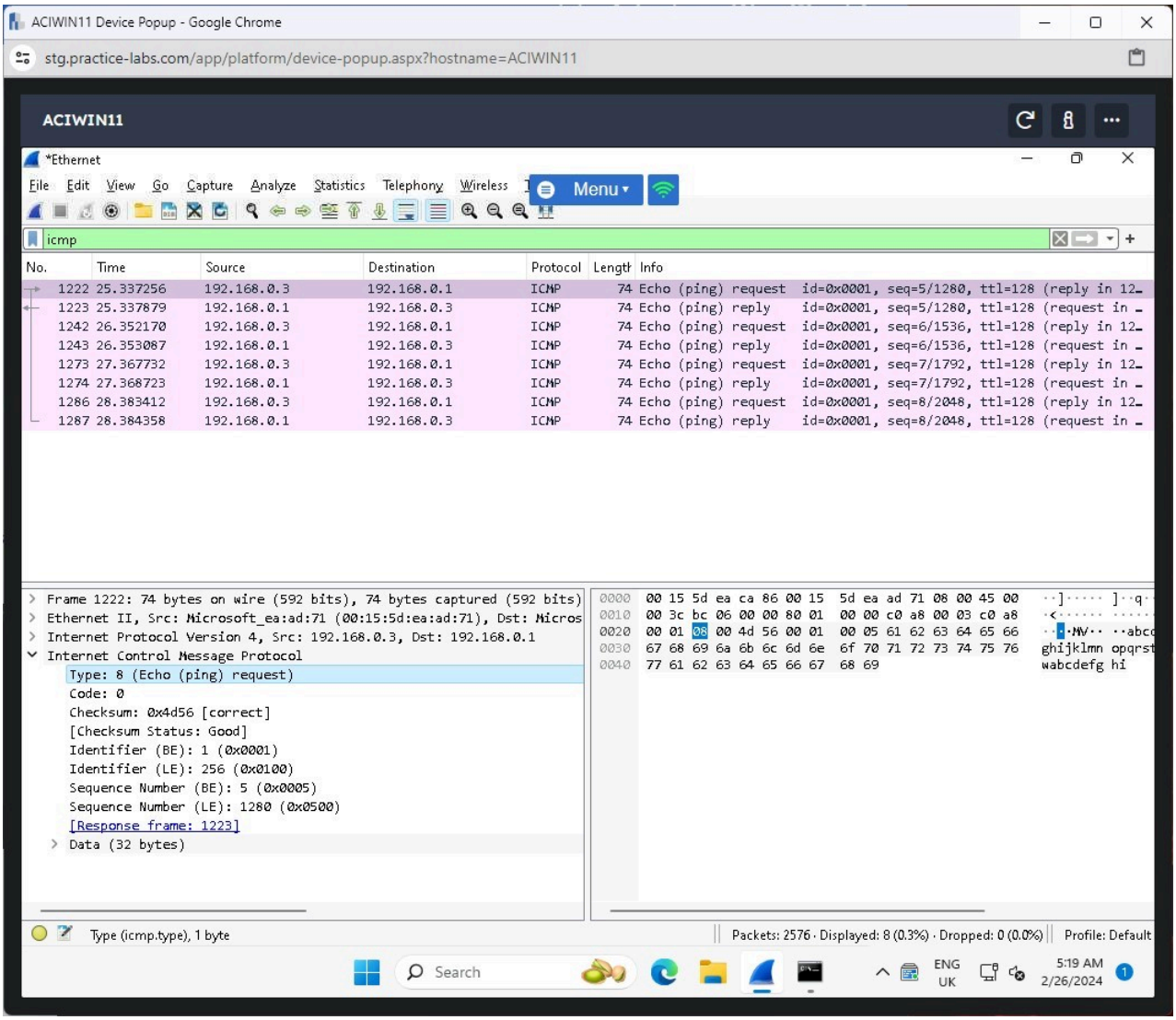


Figure 1.9 Screenshot of ACIWIN11: Displaying Wireshark and expanding the Internet Control Message Protocol field.

Note: *Ping utilizes the ICMP rather than a specific port for communication. As shown, when a ping command is initiated, the sender (Src: 192.168.0.3) generates an ICMP echo request packet and sends it to the target device's IP address (Dst: 192.168.0.1). The target device then responds with an ICMP echo reply packet, allowing the sender to measure the round-trip time and assess the connectivity between the two devices.*

Step 10

Close the **Wireshark** window.

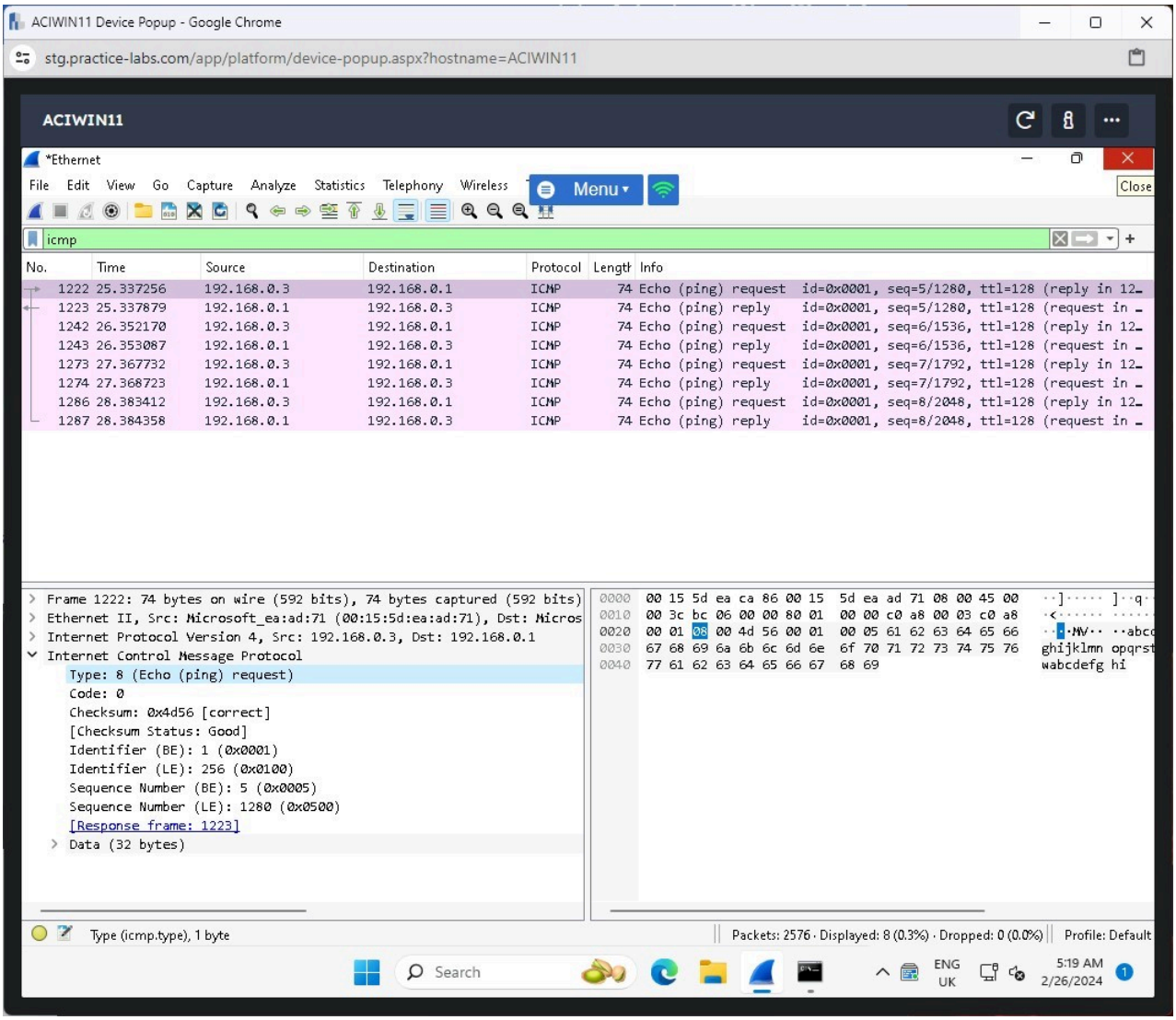


Figure 1.10 Screenshot of ACIWIN11: Displaying closing Wireshark.

Step 11

In the **Wireshark - Unsaved packets** pop-up window, click **Quit without Saving.**

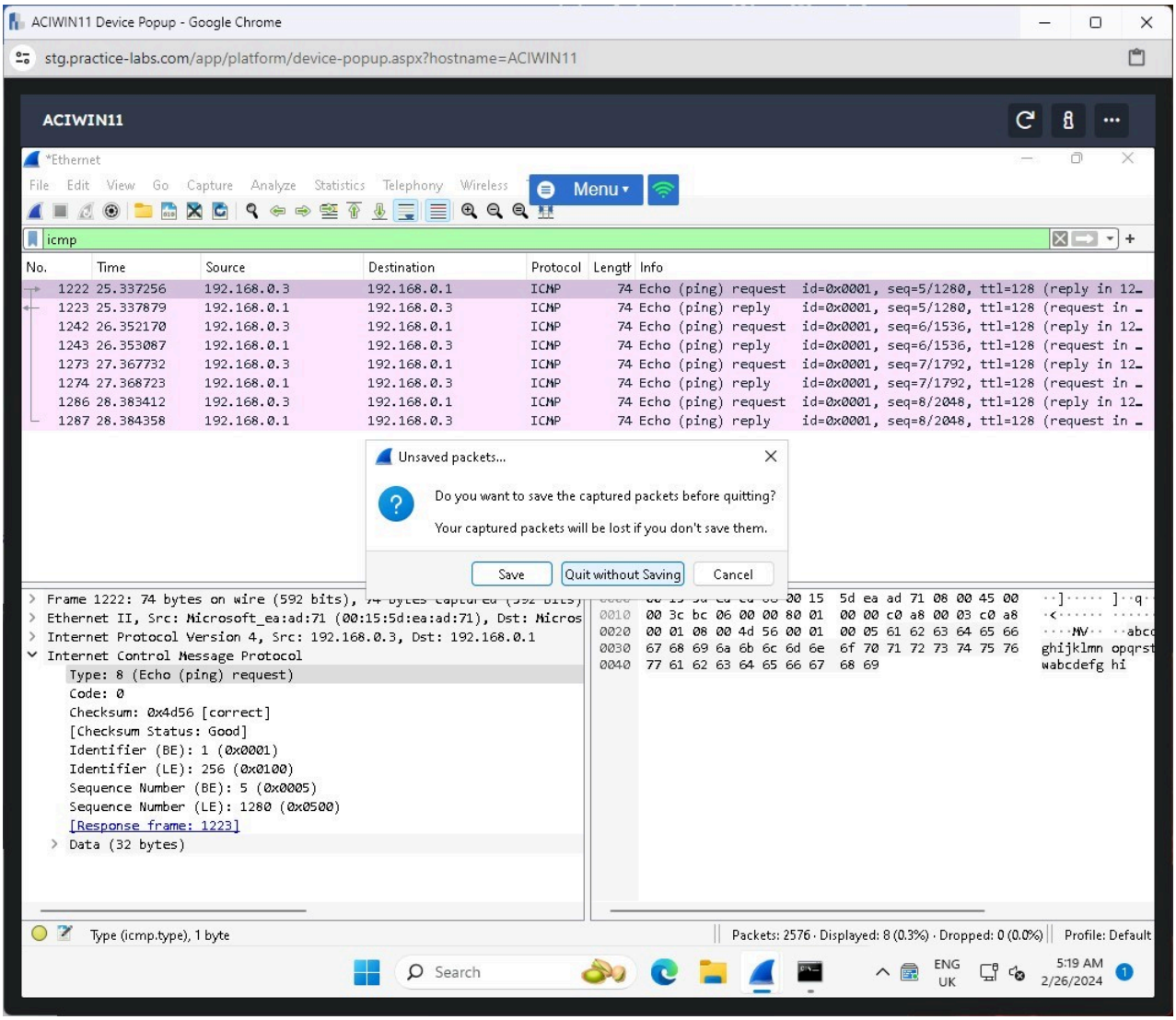


Figure 1.11 Screenshot of ACIWIN11: Displaying the Wireshark Unsaved packets pop-up and selecting Quit without Saving.

Step 12

Close the **Command Prompt** window.

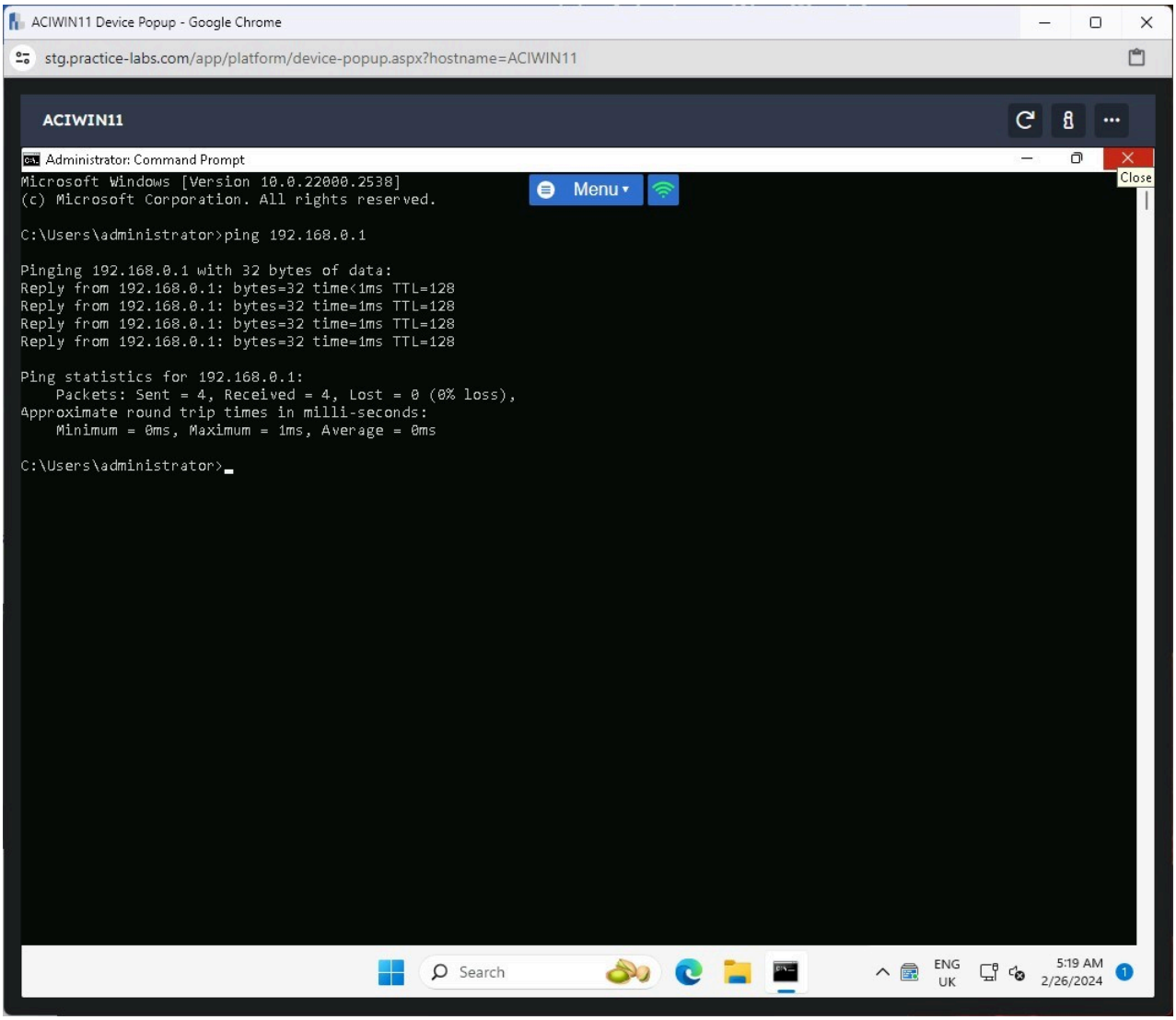


Figure 1.12 Screenshot of ACIWIN11: Displaying closing the Command Prompt window.

Task 2 - Conduct a DNS Query

DNS (Domain Name System) translates human-readable domain names, like www.acilearning.com, into IP addresses used to communicate over a network. It operates through a hierarchical structure of DNS servers and plays a critical role in enabling users to access websites, send emails, and perform online activities.

In this task, you will monitor and analyze a DNS query with Wireshark.

Step 1

Connect to **ACIWIN11.**

In the **Search** field, type the following:

```
wireshark
```

Select **Wireshark** from the **Best match** pop-up menu.

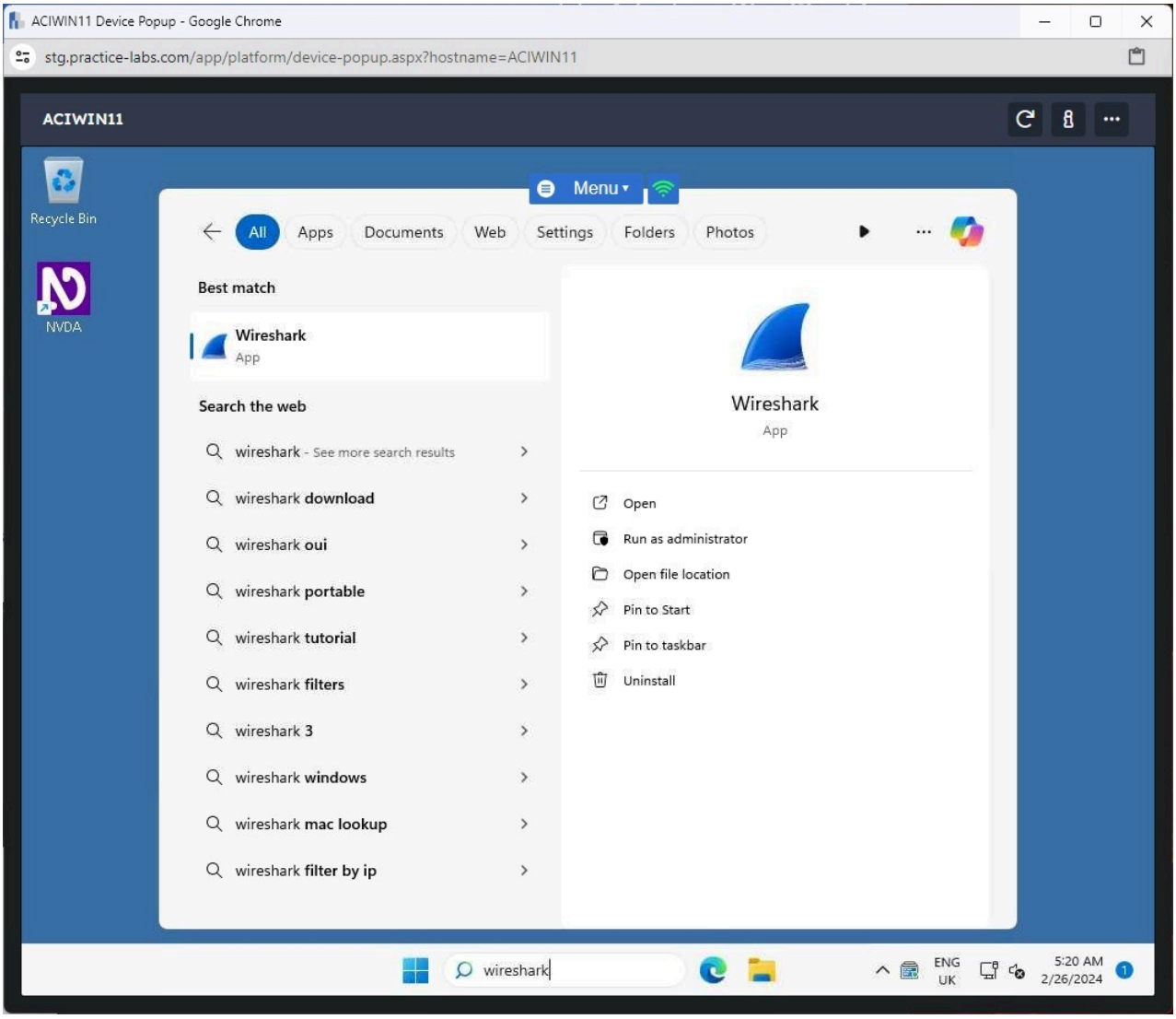


Figure 1.13 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Best match pop-up menu.

Step 2

In **Wireshark,** under the **Capture** menu, double-click on **Ethernet.**

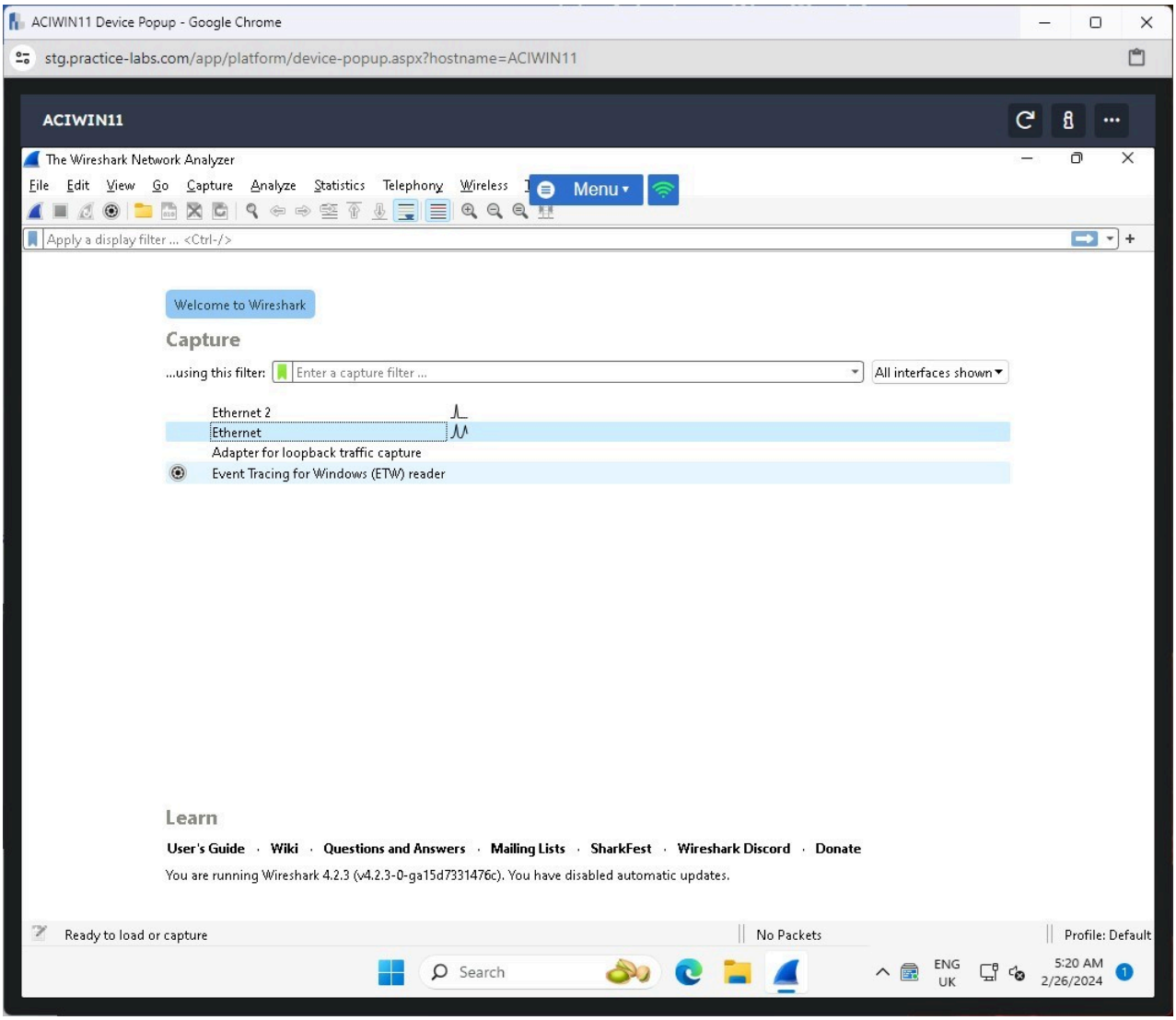


Figure 1.14 Screenshot of ACIWIN11: Displaying the Wireshark Capture menu and starting a capture of the Ethernet interface.

Step 3

In the **Taskbar - Search** field, type the following:

```
cmd
```

Select **Command Prompt** from the **Best match** pop-up menu.

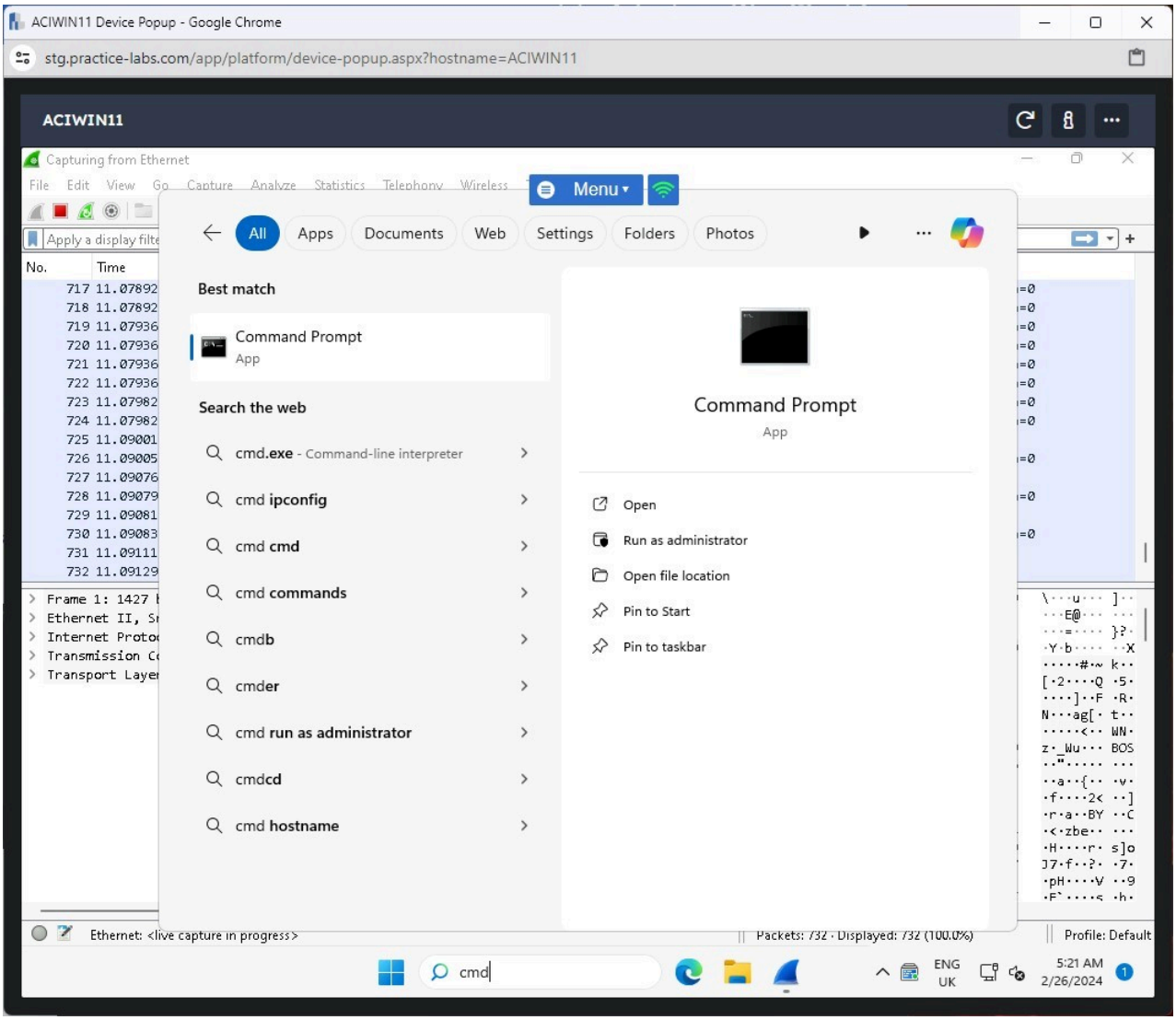


Figure 1.15 Screenshot of ACIWIN11: Displaying selecting Command Prompt from the Best match pop-up menu.

Step 4

In the **Command Prompt** window, type the following:

```
nslookup acidc01
```

Press **Enter.**

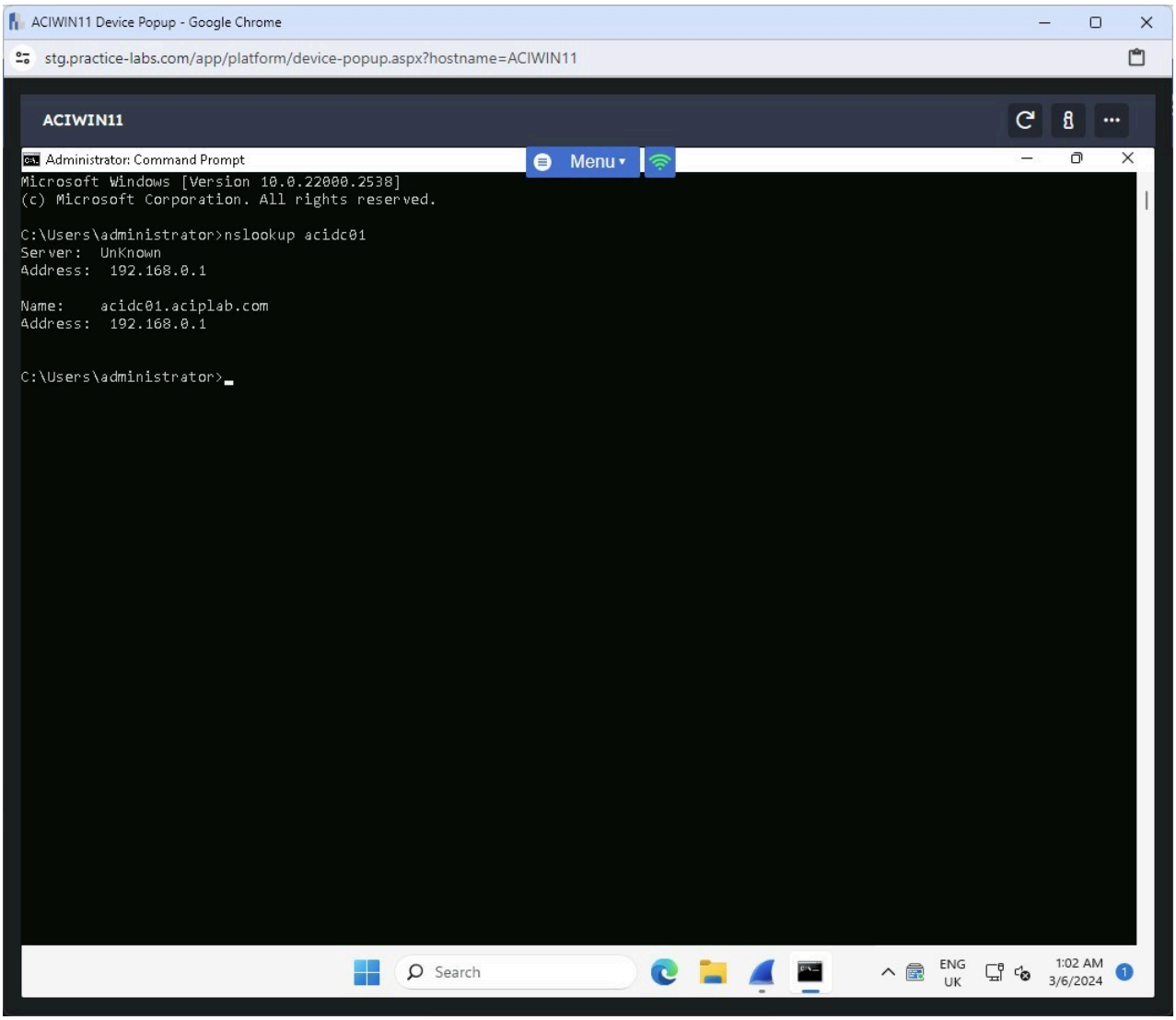


Figure 1.16 Screenshot of ACIWIN11: Displaying the Command Prompt window and conducting a dns query.

Note: nslookup *is a command-line tool that queries DNS servers. When used with a specific hostname, such as "acidc01", nslookup sends a DNS query to the DNS server to retrieve information about that hostname, including its IPv4 and IPv6 address.*
Notice the results indicate that ACIDC01 has an IP address of ***192.168.0.1****.*

Step 5

On the **Taskbar**, select **Wireshark.**

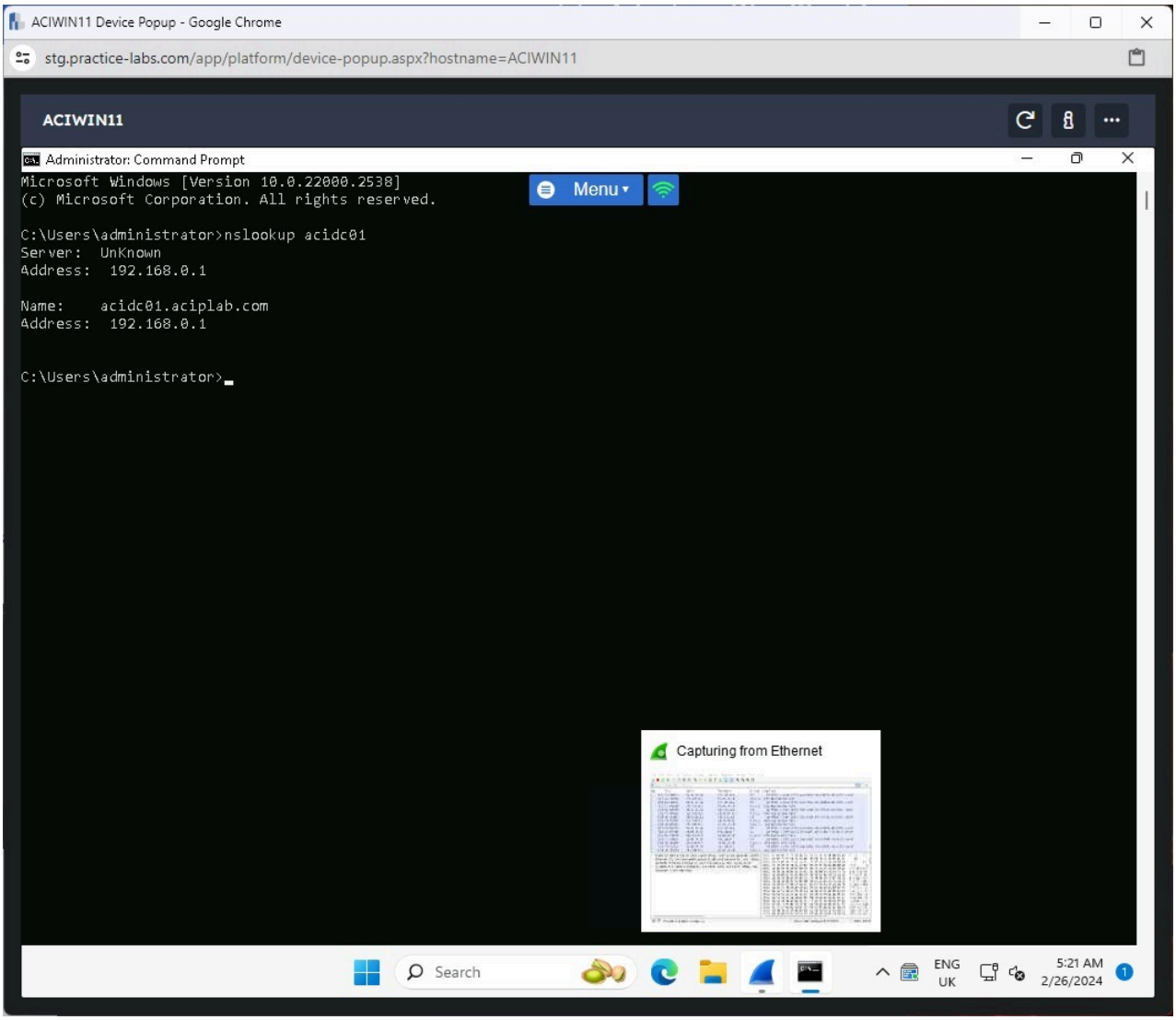


Figure 1.17 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Taskbar.

Step 6

In **Wireshark,** select the red square to **Stop capturing packets.**

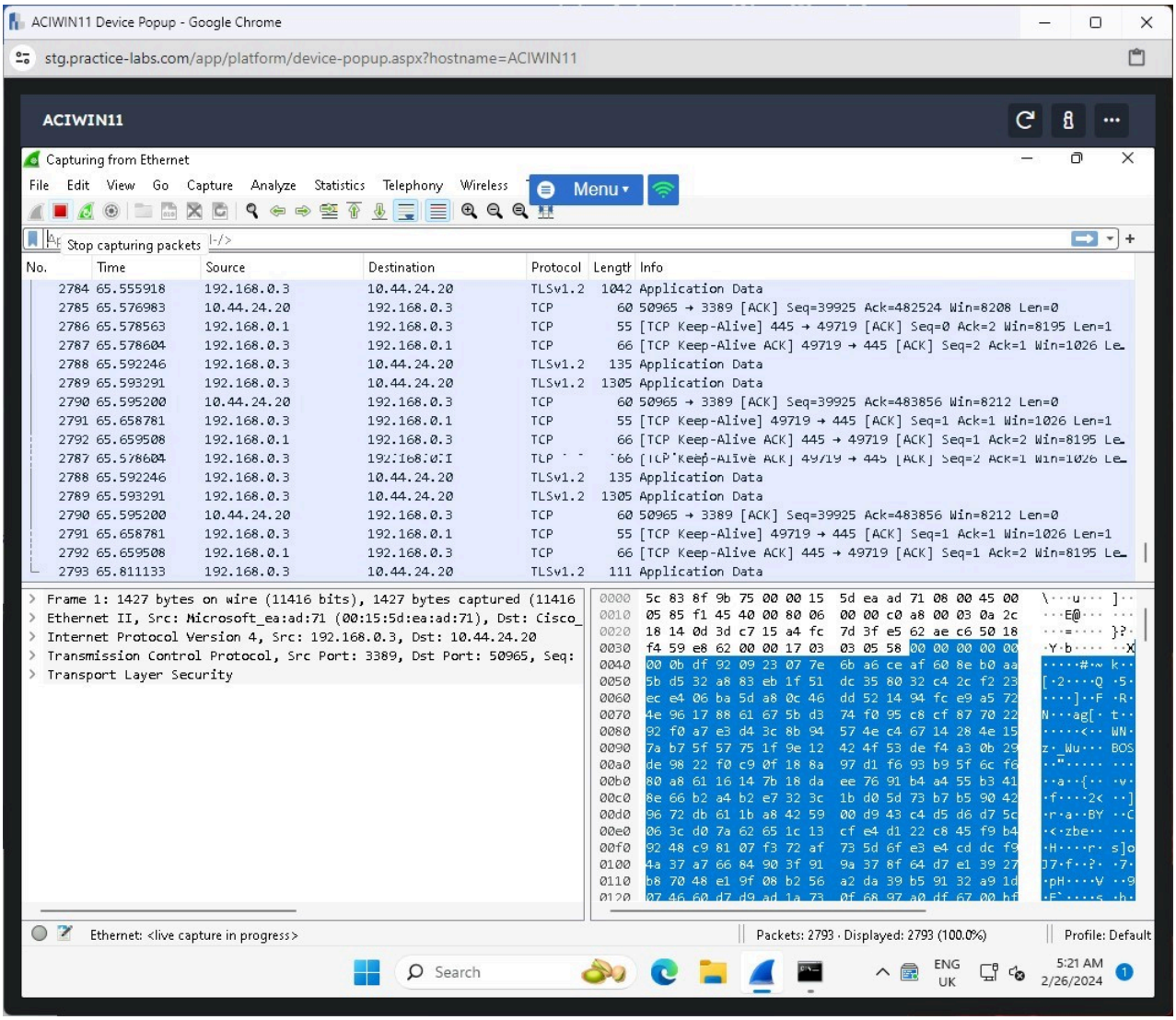


Figure 1.18 Screenshot of ACIWIN11: Displaying Wireshark and selecting the red square to Stop capturing packets.

Step 7

In **Wireshark,** in the **Apply a display filter** field, type the following:

```
dns
```

Press **Enter.**

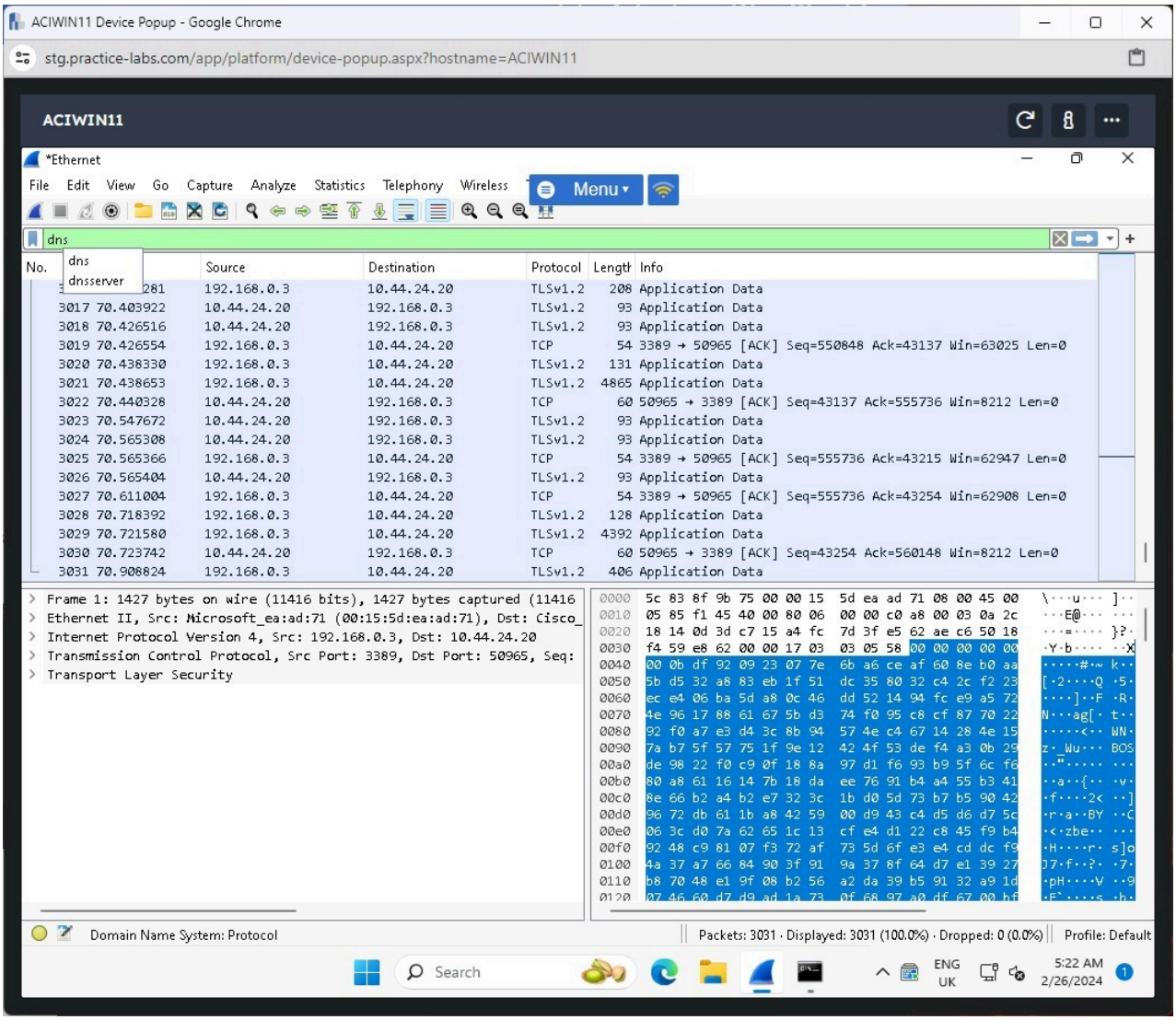


Figure 1.19 Screenshot of ACIWIN11: Displaying Wireshark and completing the Apply a display filter field.

Step 8

In **Wireshark**, select the first packet with a **Source** of **192.168.0.3** (the ACIWIN11 machine).

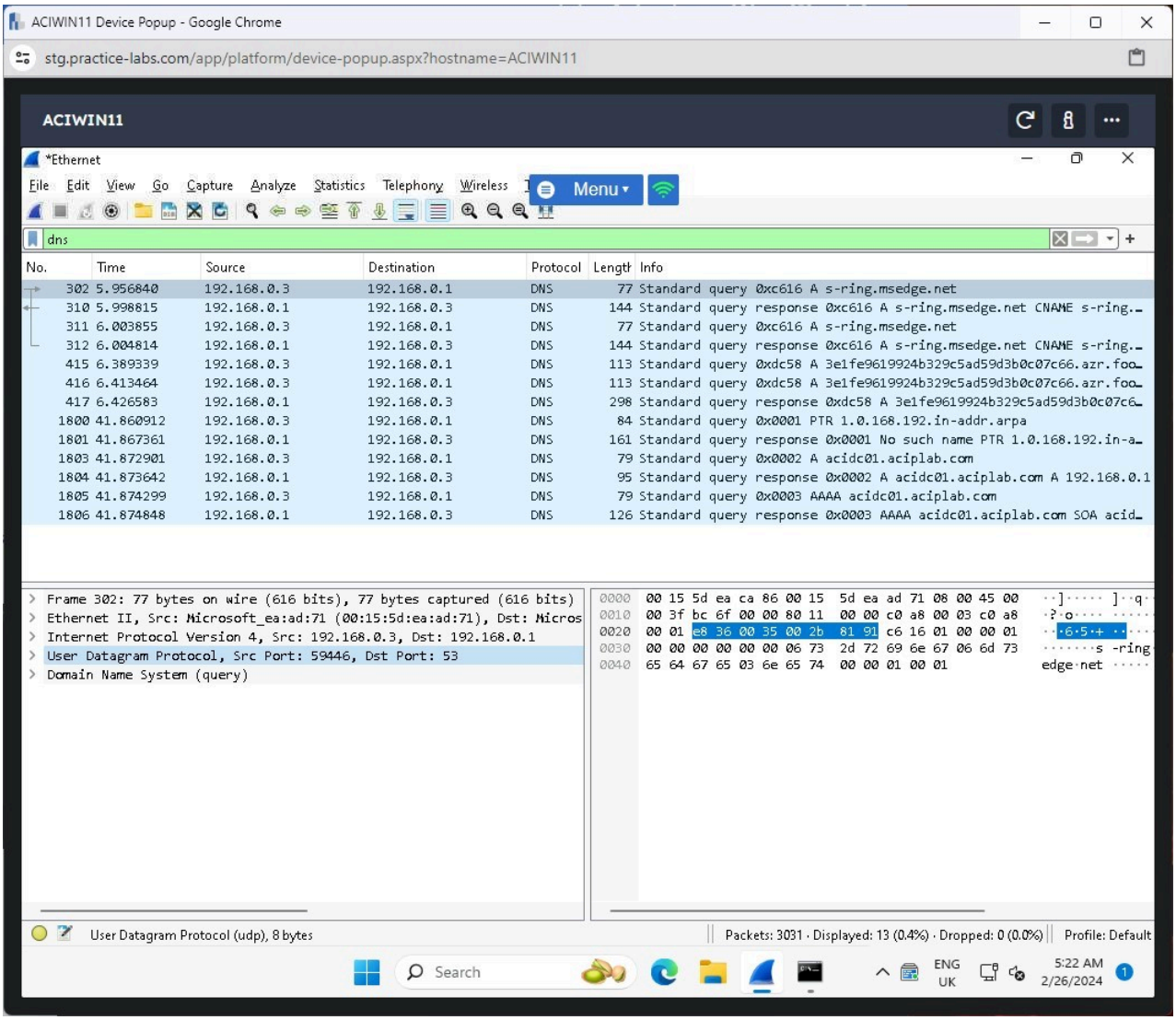


Figure 1.20 Screenshot of ACIWIN11: Displaying Wireshark and observing the first packet from 192.168.0.3.

Note: *In the Packet Details pane, the packet is defined as using the* ***User Datagram Protocol (UDP)*** *and is associated with the destination port* ***53****.*
UDP is a connectionless transport protocol used for sending datagrams over a network. UDP port 53 is used for DNS queries and responses between clients and servers, enabling the translation of domain names to IP addresses.

Step 9

In **Wireshark,** expand the **User Datagram Protocol** field.

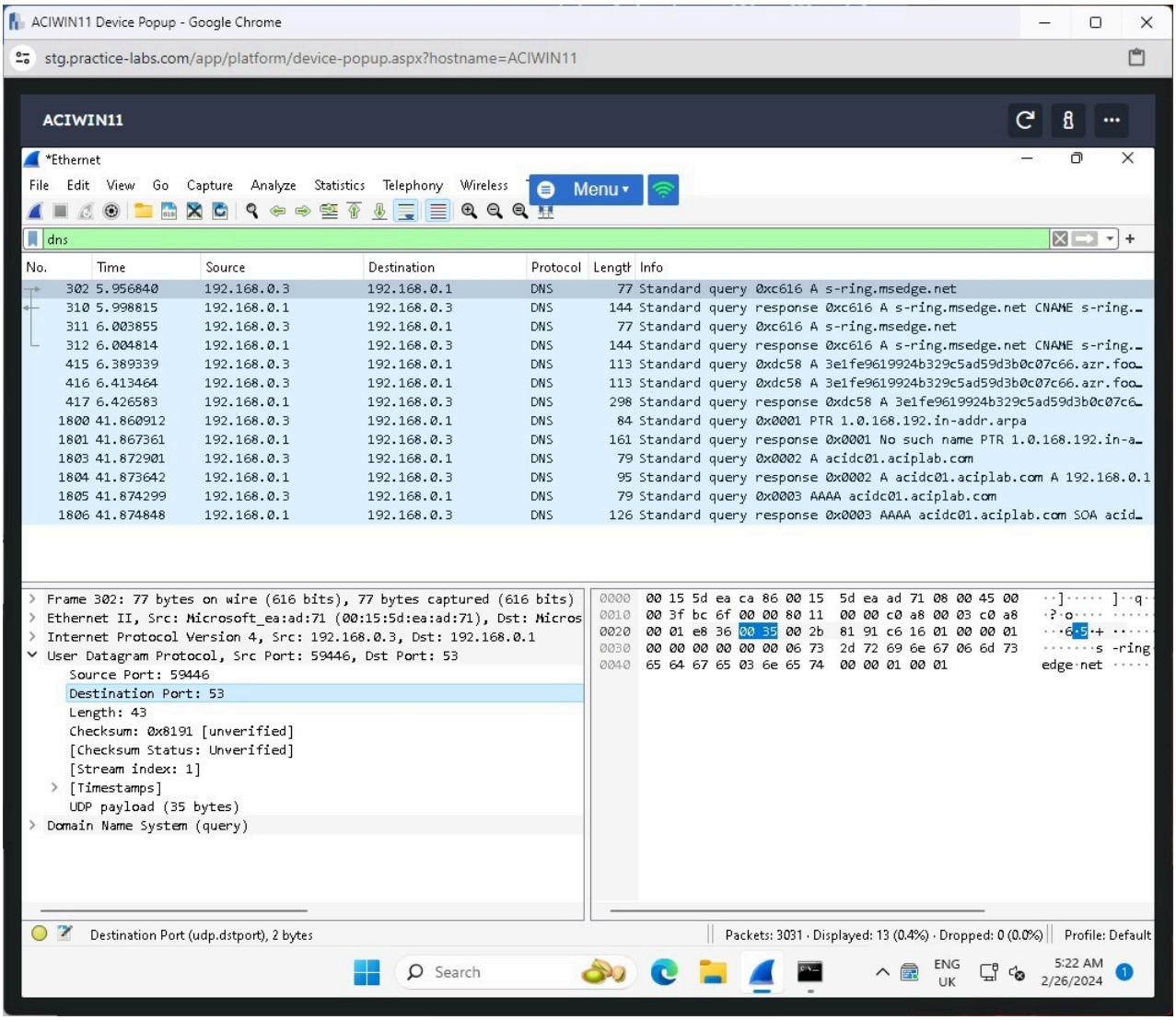


Figure 1.21 Screenshot of ACIWIN11: Displaying Wireshark and expanding the User Datagram Protocol field.

Step 10

Close **Wireshark.**

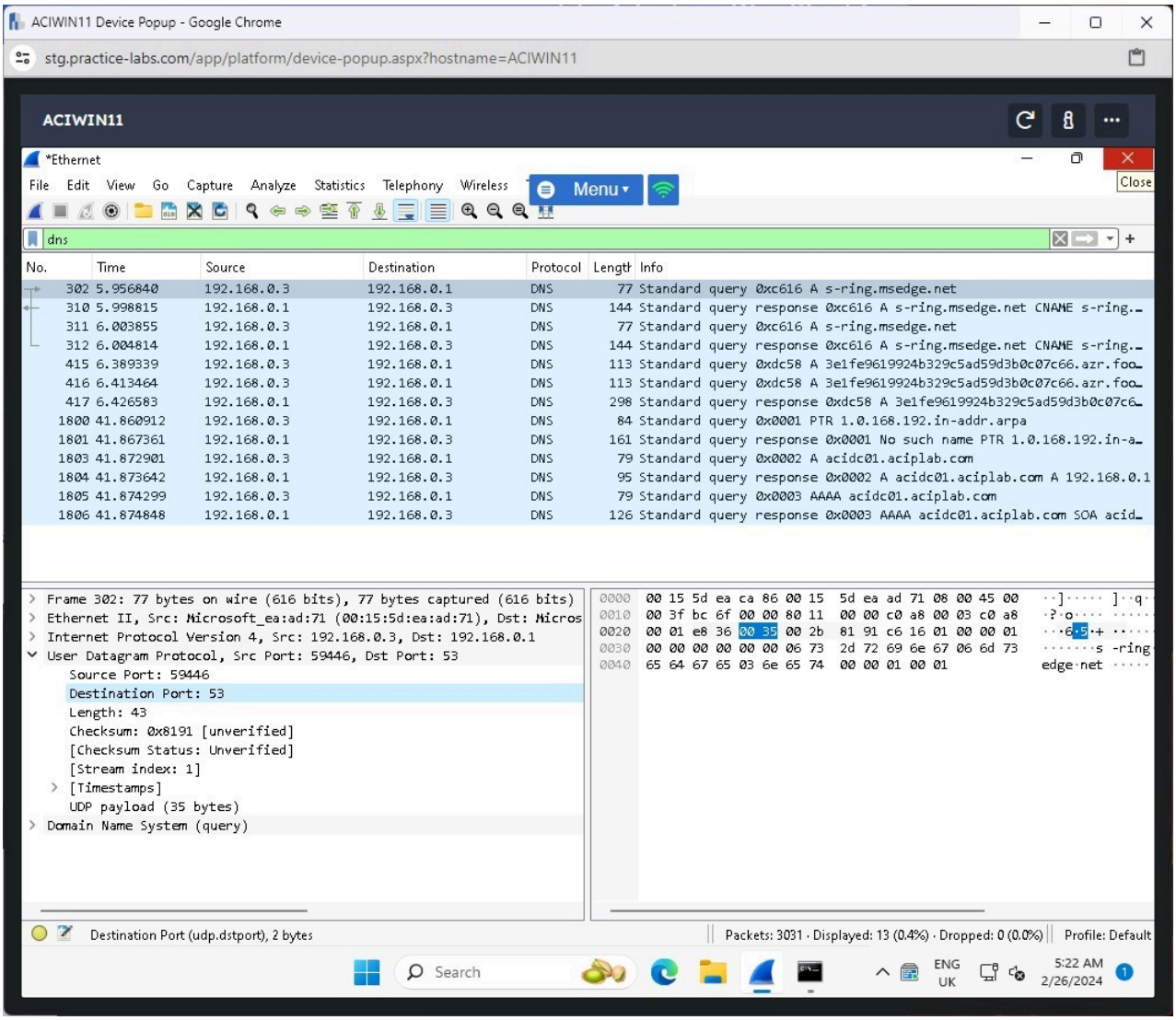


Figure 1.22 Screenshot of ACIWIN11: Displaying closing Wireshark.

Step 11

In the **Wireshark - Unsaved packets** pop-up window, click **Quit without Saving.**

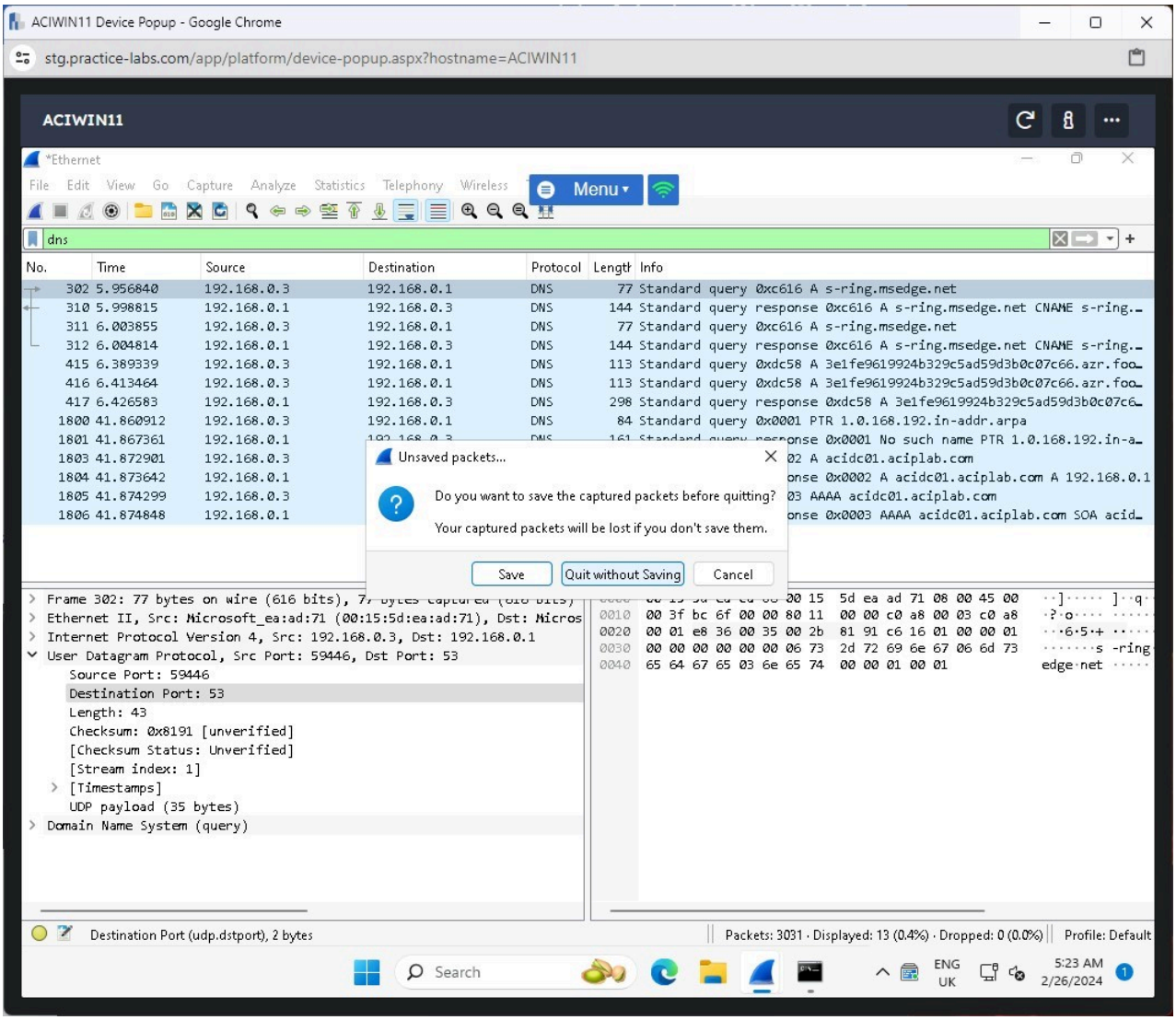


Figure 1.23 Screenshot of ACIWIN11: Displaying the Wireshark - Unsaved packets pop-up window and selecting Quit without Saving.

Step 12

Close the **Command Prompt** window.

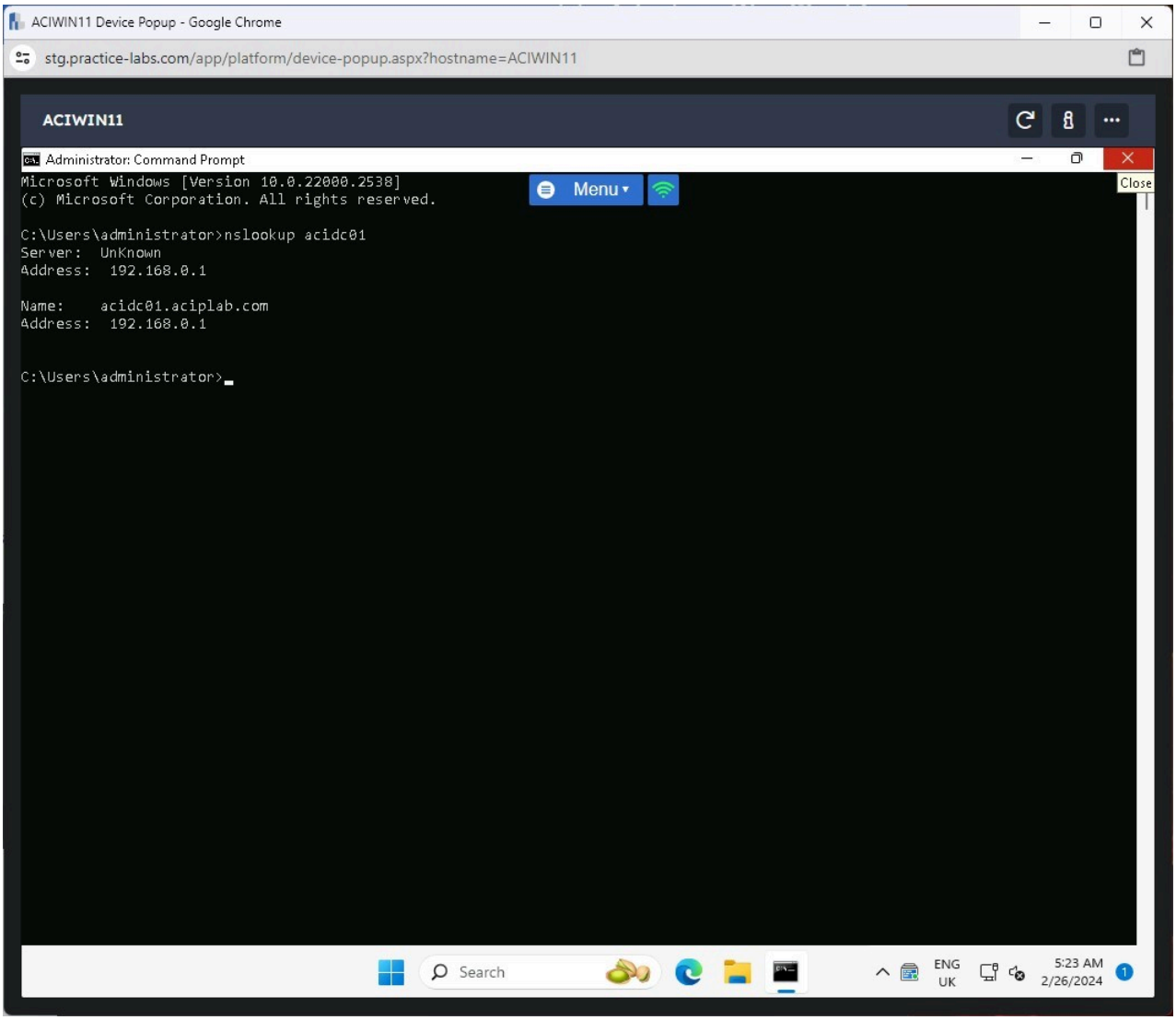


Figure 1.24 Screenshot of ACIWIN11: Displaying closing the Command Prompt window.

Task 3 - Make an SSH Connection

SSH (Secure Shell) is a network protocol for secure remote access to systems. It provides encrypted communication between the SSH client and SSH server, offering confidentiality, integrity, and authentication. Common uses for SSH include remote administration, file transfers, and tunneling network connections.

In this task, you will monitor and analyze an SSH connection with Wireshark.

Step 1

Connect to **ACIWIN11.**

In the **Search** field, type the following:

```
wireshark
```

Select **Wireshark** from the **Best match** pop-up menu.

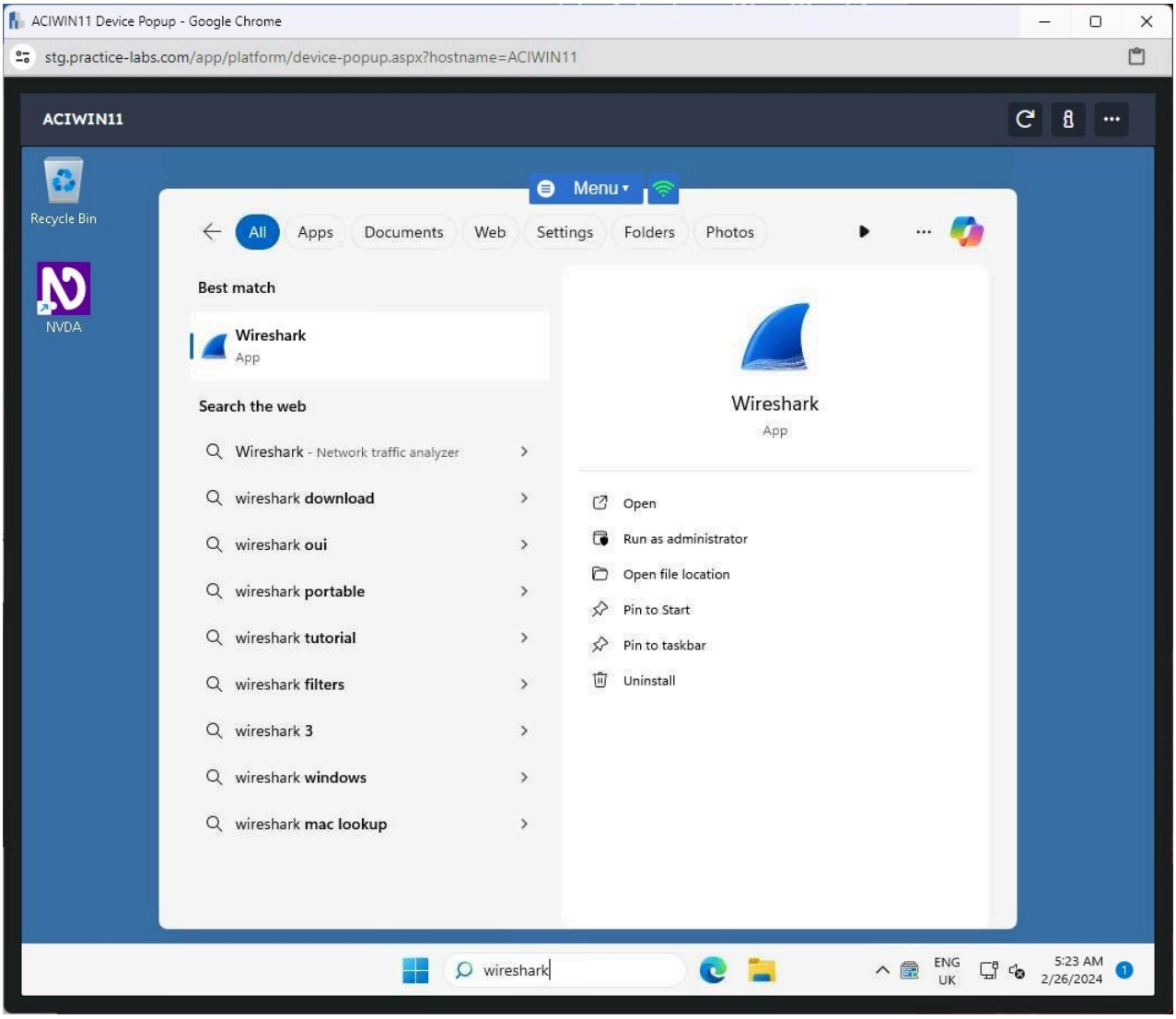


Figure 1.25 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Best match pop-up menu.

Step 2

In **Wireshark,** under the **Capture** menu, double-click on **Ethernet.**

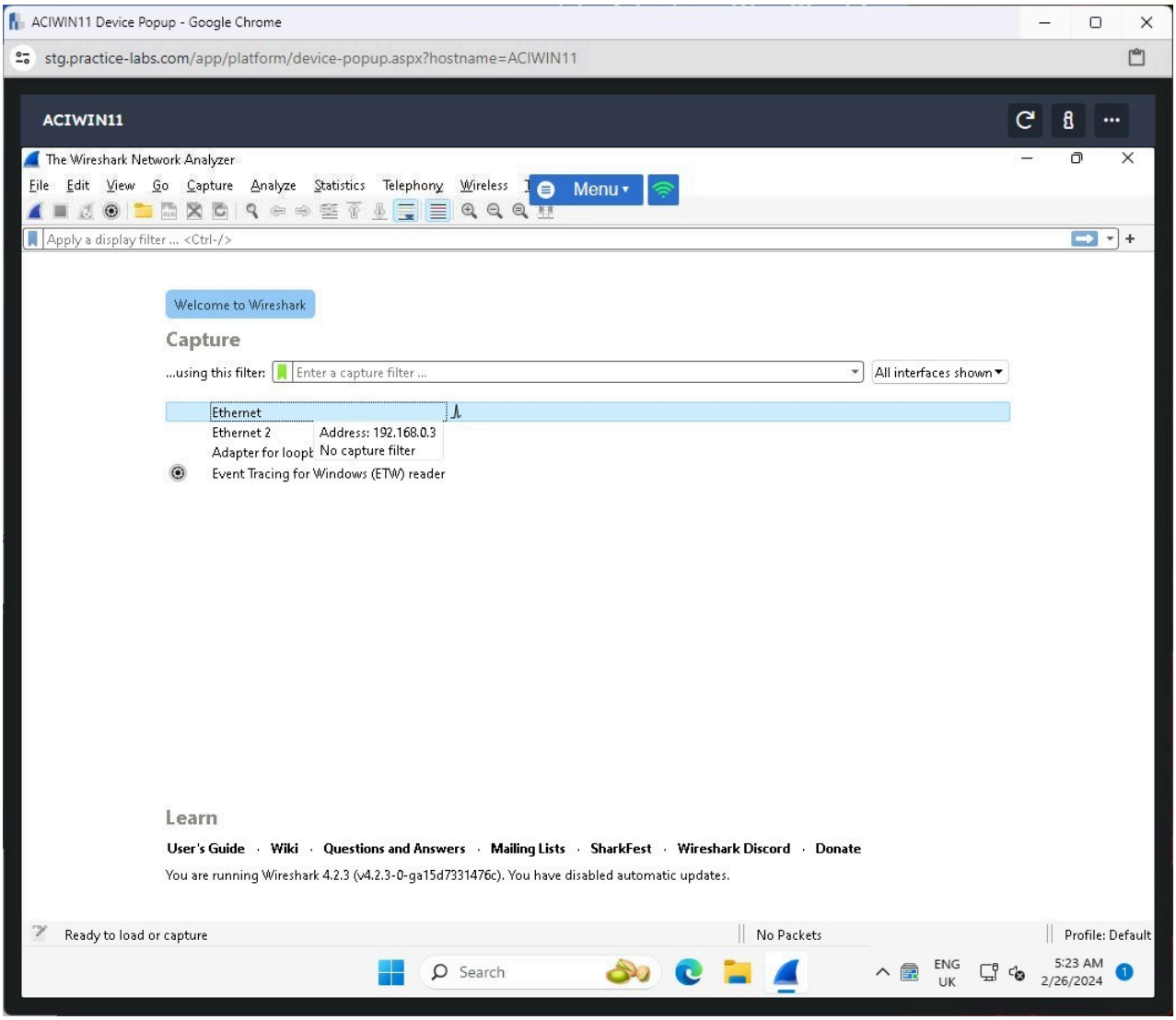


Figure 1.26 Screenshot of ACIWIN11: Displaying the Wireshark Capture menu and starting a capture of the Ethernet interface.

Step 3

In the **Taskbar - Search** field, type the following:

```
cmd
```

Select **Command Prompt** from the **Best match** pop-up menu.

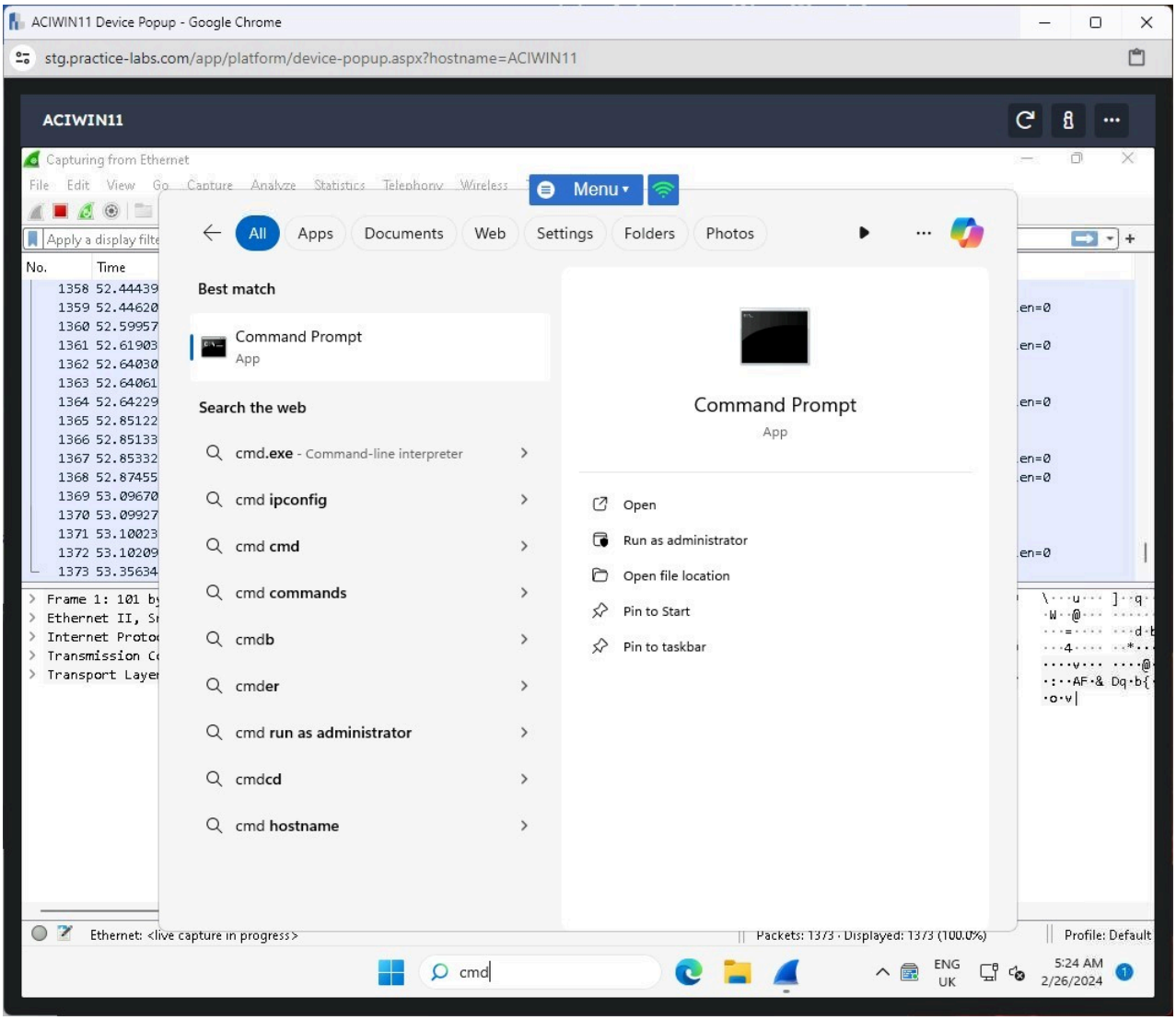


Figure 1.27 Screenshot of ACIWIN11: Displaying selecting Command Prompt from the Best match pop-up menu.

Step 4

In the **Command Prompt** window, type the following:

```
ssh admin@192.168.0.5
```

Press **Enter.**

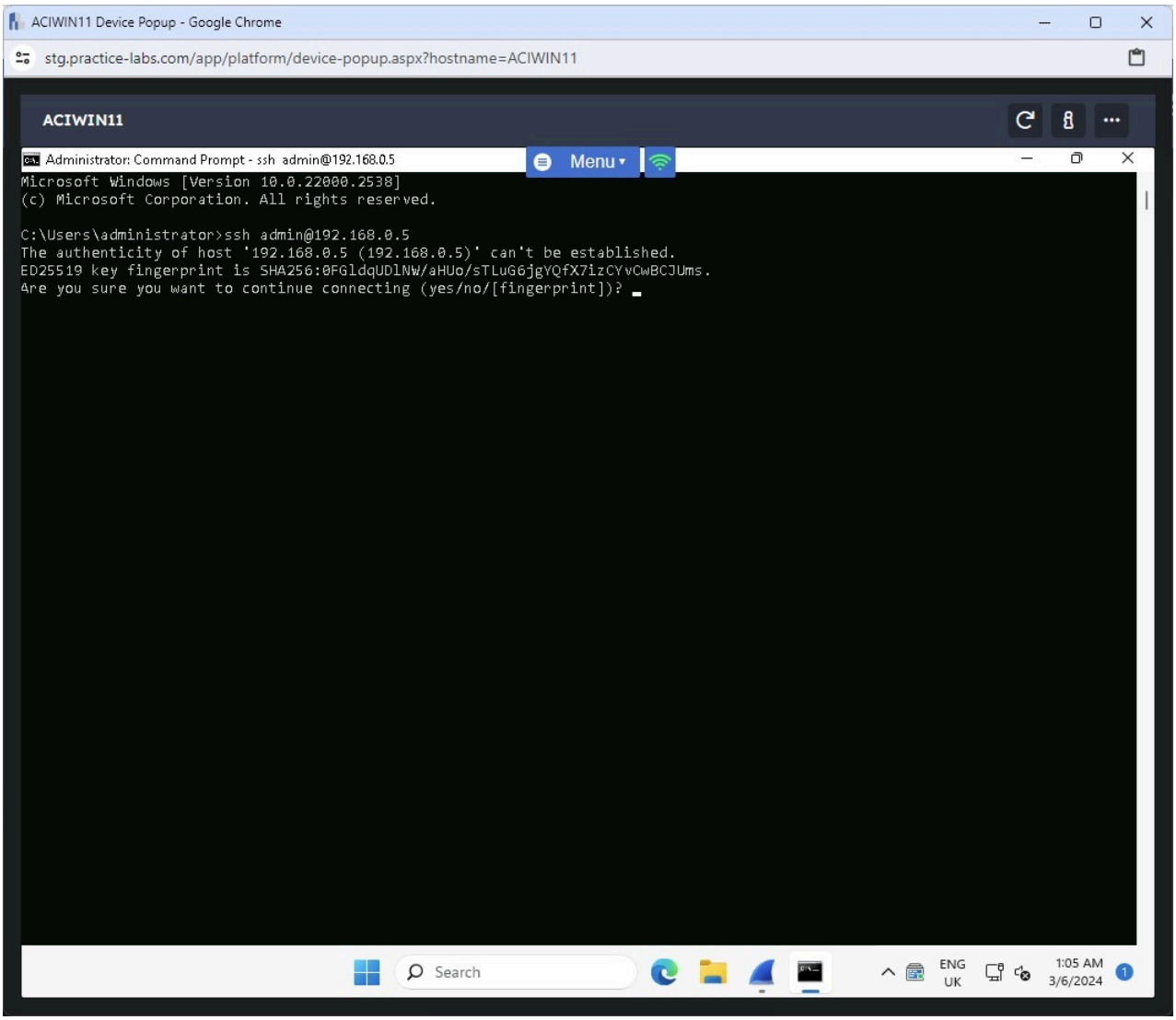


Figure 1.28 Screenshot of ACIWIN11: Displaying the Command Prompt window and initiating an SSH connection.

Note: *The command* ***ssh admin@192.168.0.5*** *initiates an SSH (Secure Shell) connection to the SSH server hosted on a device with the IP address 192.168.0.5. The command specifies the username "admin" to log in to the remote device. Once authenticated, the user, at the client machine, gains remote access to the device for administration and management tasks.*
UK and US keyboard layouts replace the @ sign with the “ sign.

Step 5

In the **Command Prompt** window, type the following:

```
yes
```

Press **Enter.**

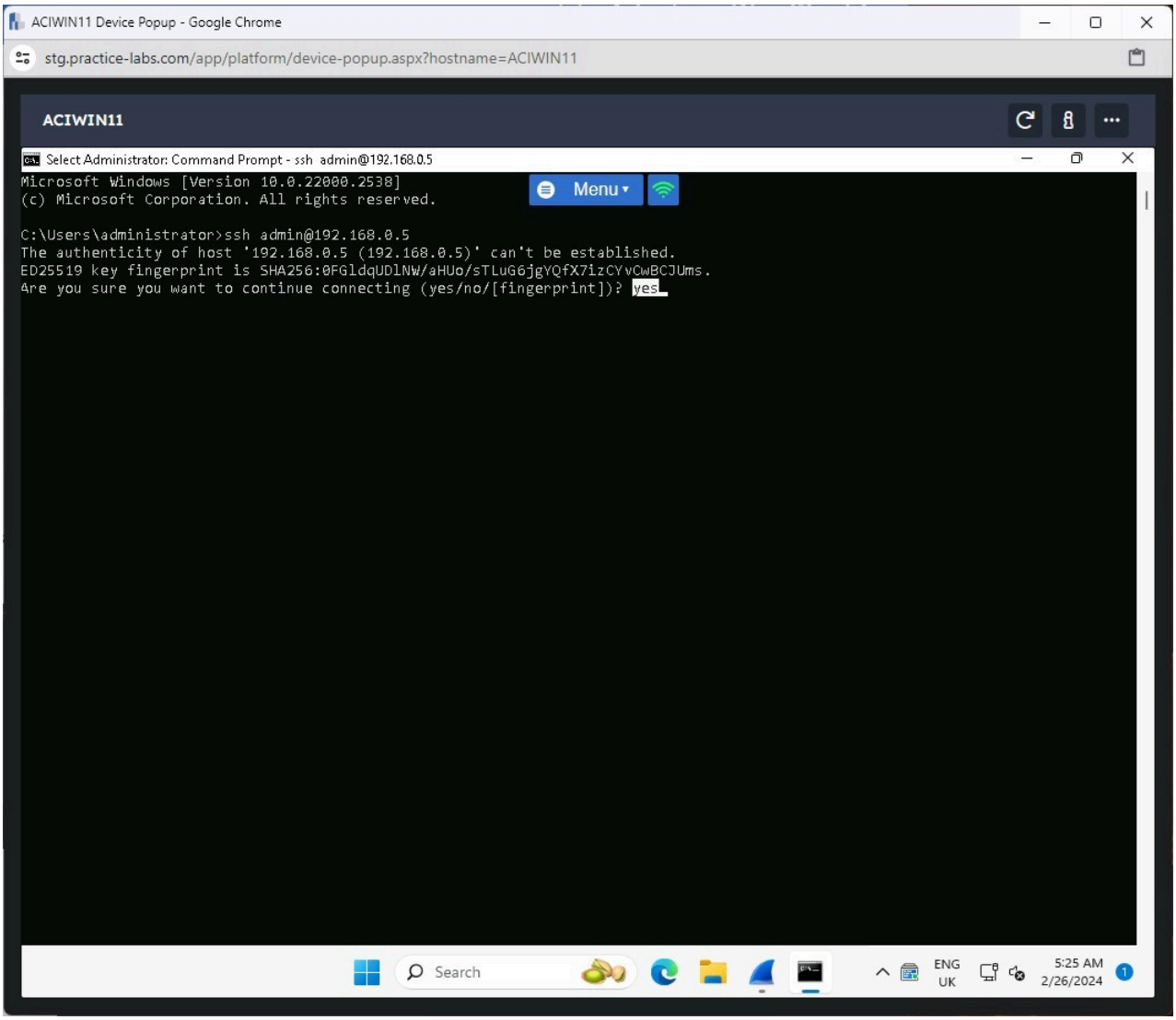


Figure 1.29 Screenshot of ACIWIN11: Displaying the Command Prompt window and accepting the SSH certificate.

Note: *When connecting to an SSH server for the first time, the client machine displays the server's key fingerprint as a security measure to confirm the server's identity. It then prompts the user with the message "Are you sure you want to continue?" to ensure they want to proceed with the connection. Verifying the fingerprint in this way helps prevent potential on-path attacks by letting the user confirm the server's identity before establishing the SSH connection.*

Step 6

In the **Command Prompt** window, type the following password:

```
Passw0rd
```

Press **Enter.**

> ***Note:*** *If the password prompt doesn't appear. Type the* ***ssh admin@192.168.0.5*** *command again and Press* ***Enter****.*

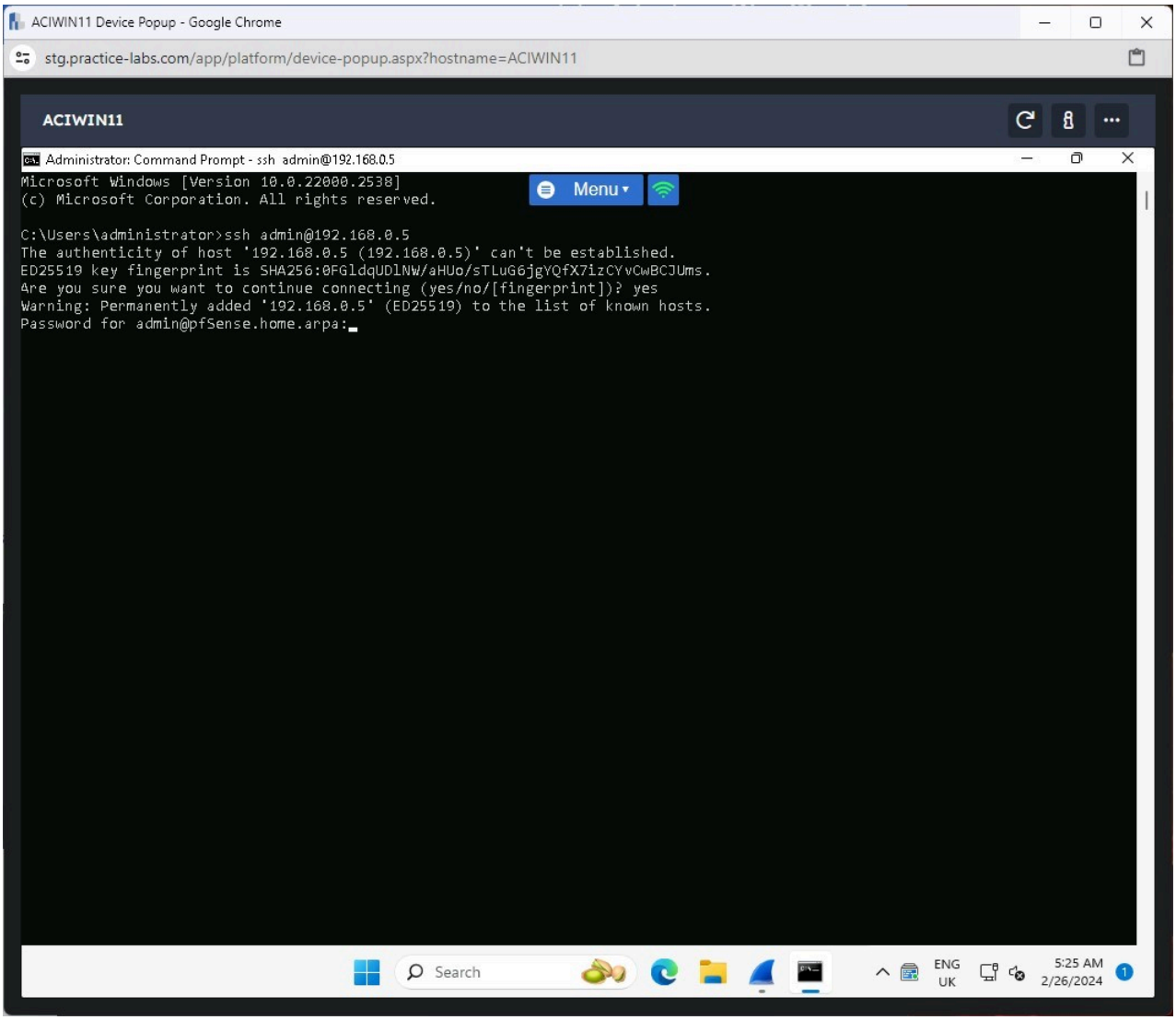


Figure 1.30 Screenshot of ACIWIN11: Displaying the Command Prompt window and providing the SSH login password for the admin account.

> ***Note:*** *The password will not be displayed when it is typed in. This is a security feature.*

Step 7

On the **Taskbar,** select **Wireshark.**

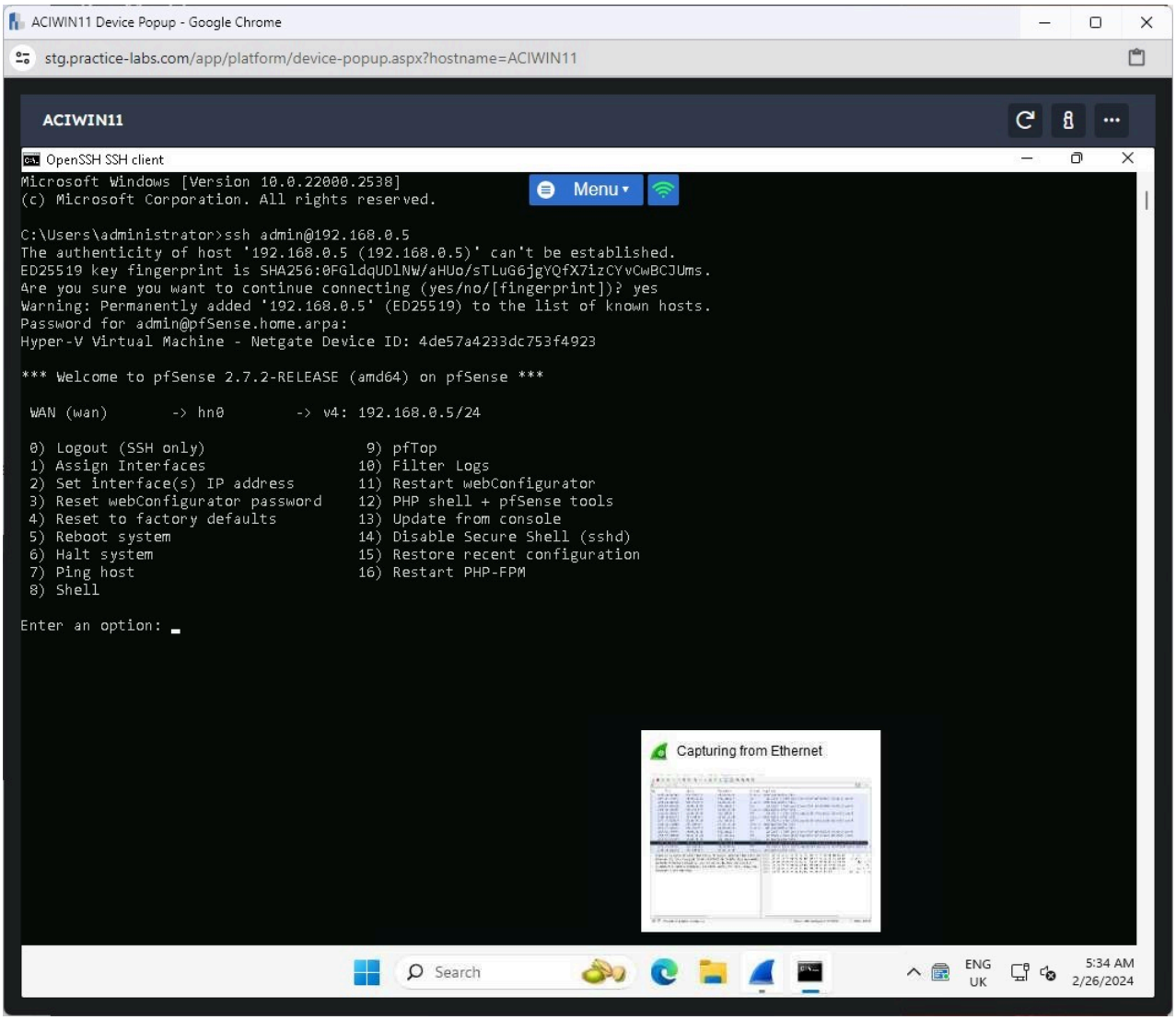


Figure 1.31 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Taskbar.

Step 8

In **Wireshark,** select the red square to **Stop capturing packets.**

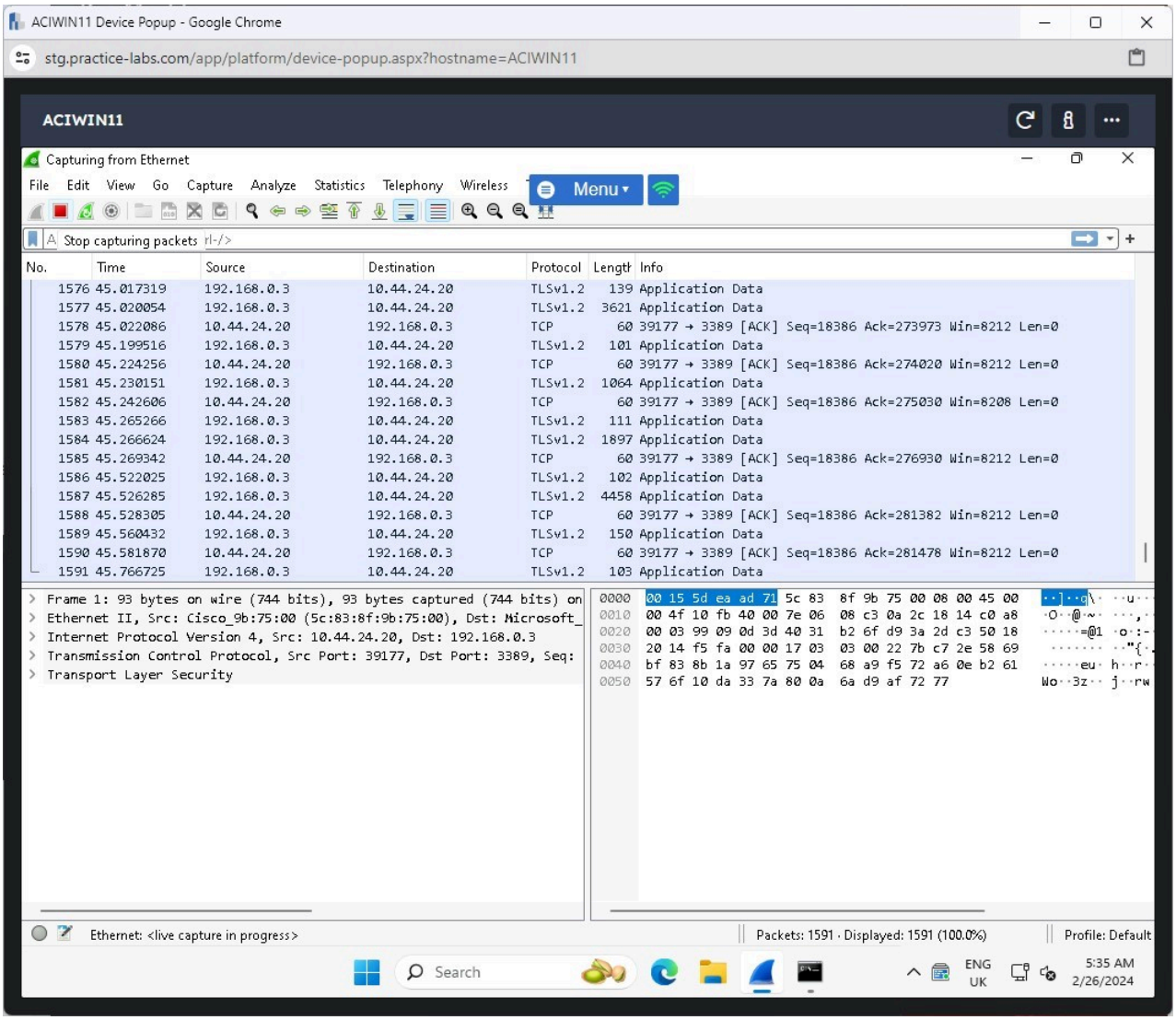


Figure 1.32 Screenshot of ACIWIN11: Displaying Wireshark and selecting the red square to Stop capturing packets.

Step 9

In **Wireshark,** in the **Apply a display filter** field, type the following:

```
ssh
```

Press **Enter.**

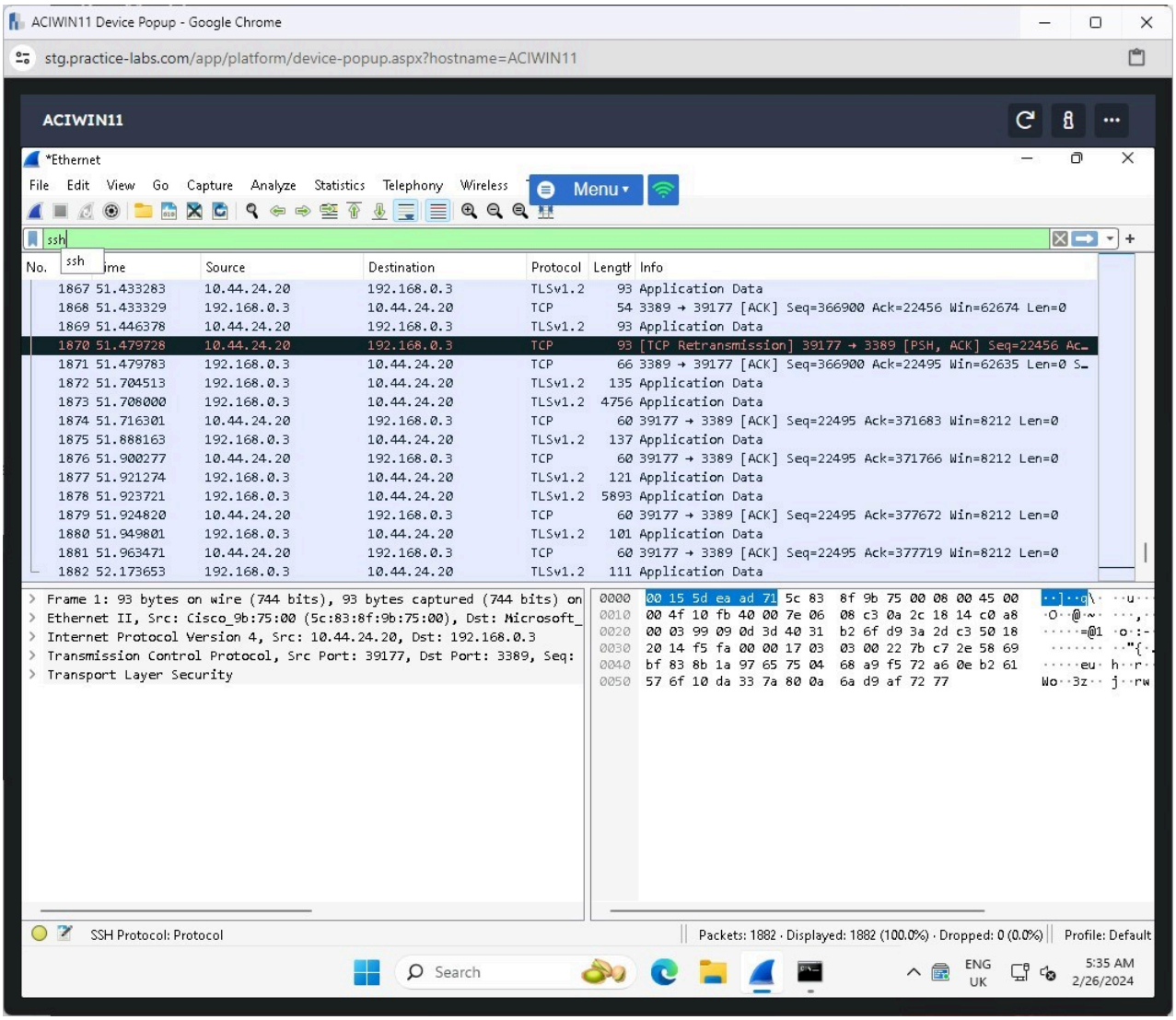


Figure 1.33 Screenshot of ACIWIN11: Displaying Wireshark and completing the Apply a display filter field.

Step 10

In **Wireshark,** select the first packet with a **Source** of **192.168.0.3** (the ACIWIN11 machine).

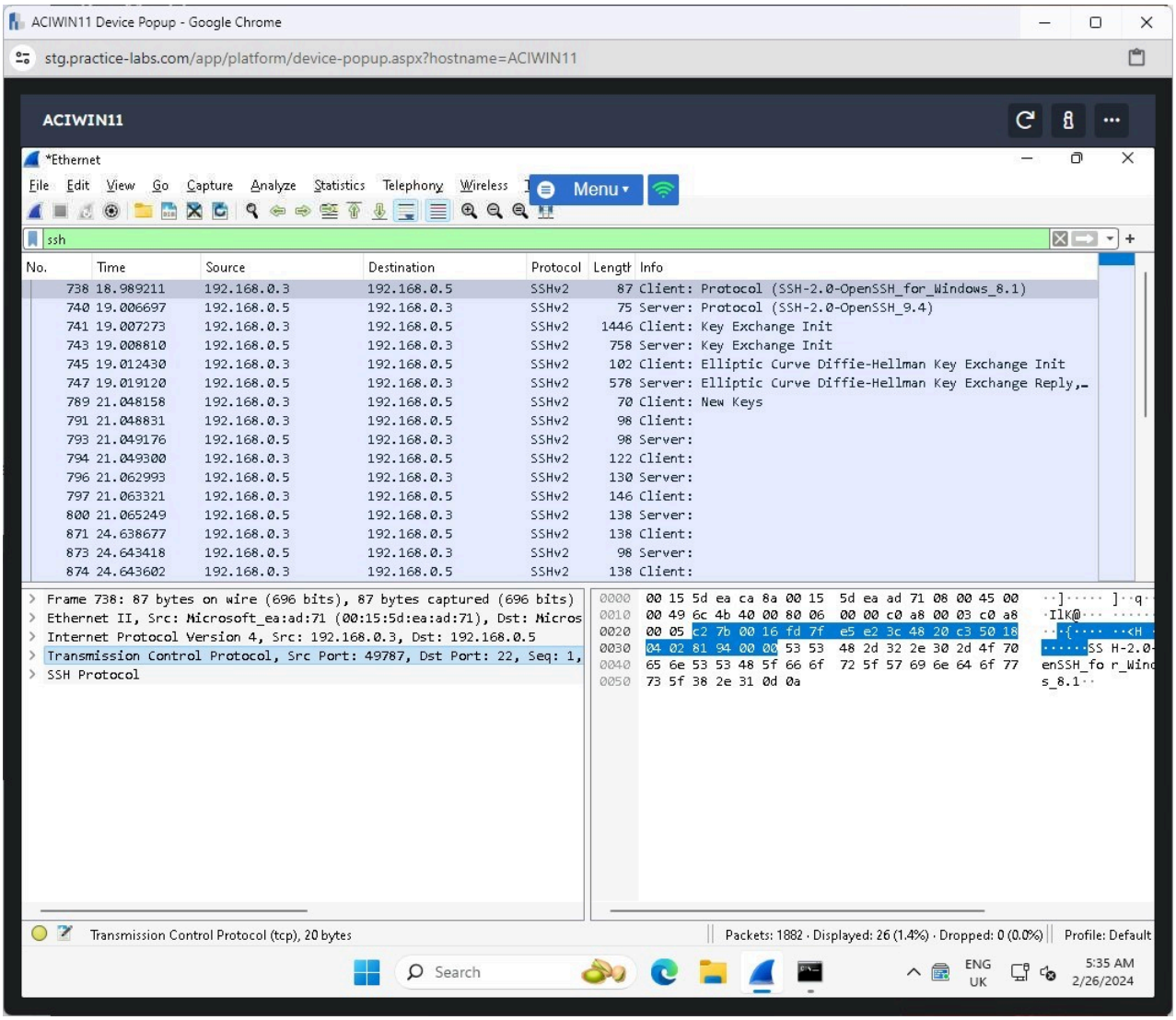


Figure 1.34 Screenshot of ACIWIN11: Displaying Wireshark and observing the first packet from 192.168.0.3.

Note: *In the Packet Details pane, the packet is defined as using the* ***Transmission Control Protocol (TCP)*** *and is associated with the destination port* ***22****.*
TCP is a connection-oriented protocol used for reliable data transmission. A packet with a destination port of 22 indicates SSH traffic, which is used for secure remote access and administration. SSH allows clients to establish encrypted connections for secure communication and remote shell access.

Step 11

In **Wireshark**, expand the **Transmission Control Protocol** field.

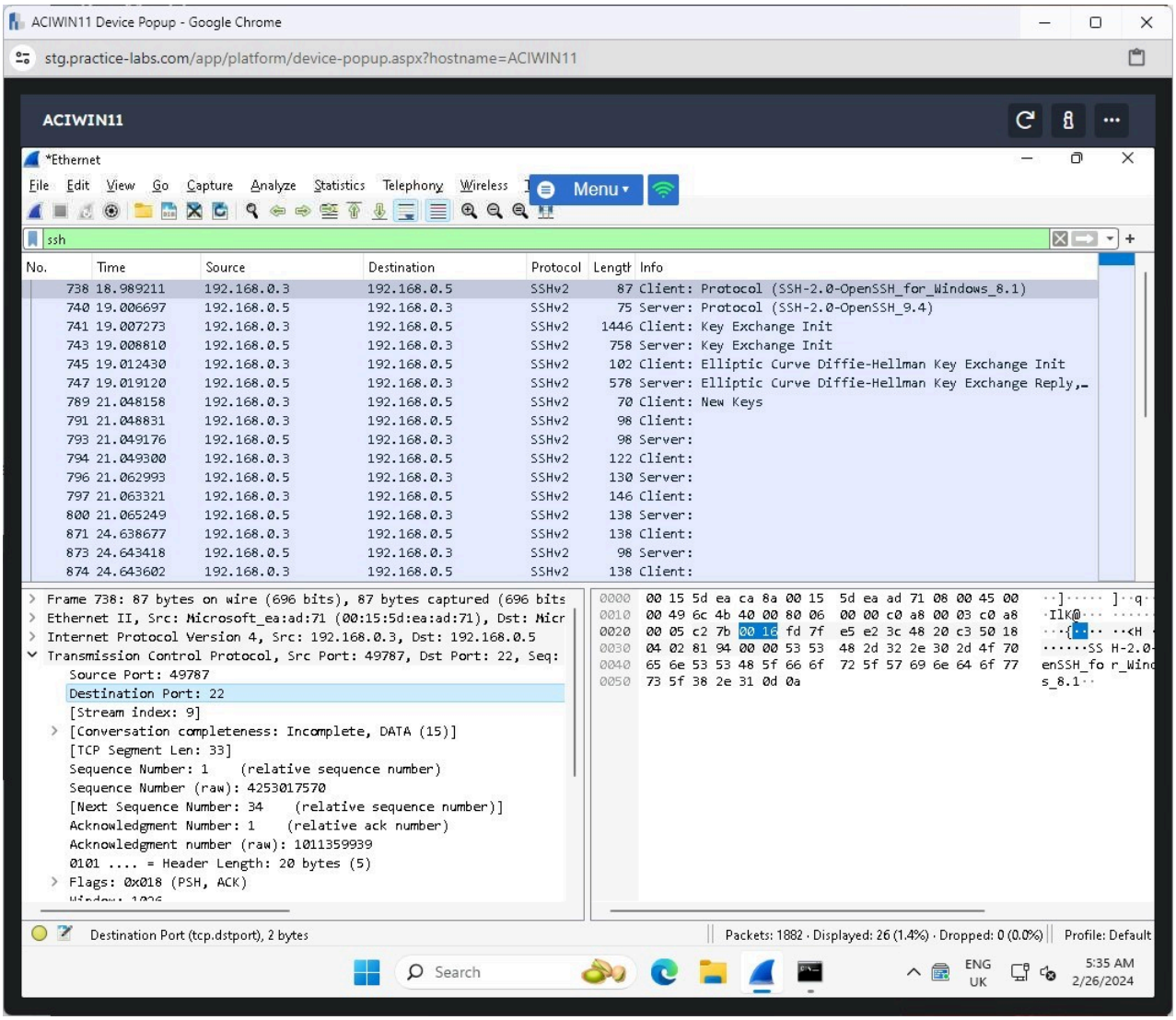


Figure 1.35 Screenshot of ACIWIN11: Displaying Wireshark and expanding the Transmission Control Protocol field.

Step 12

Close **Wireshark.**

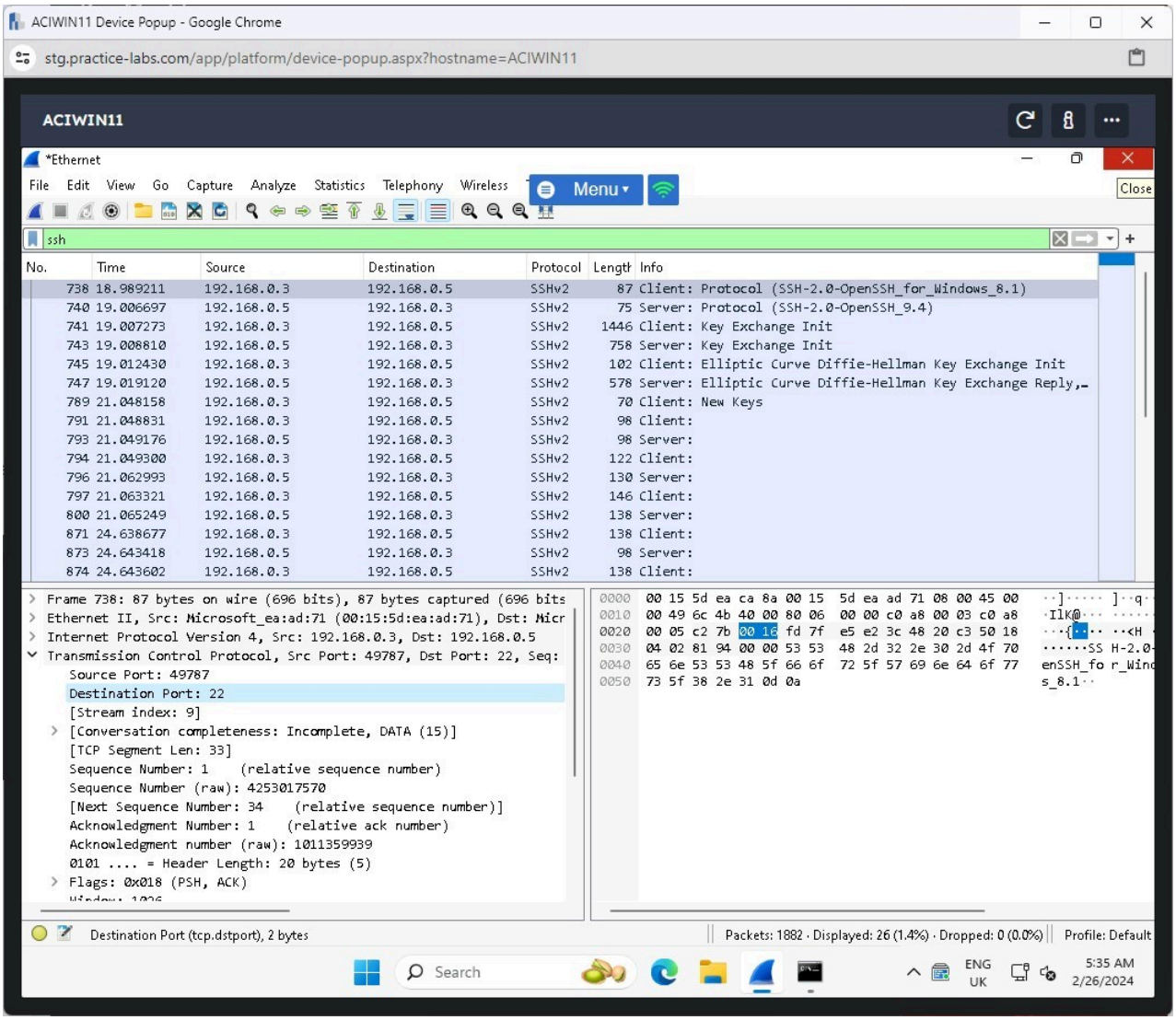


Figure 1.36 Screenshot of ACIWIN11: Displaying closing Wireshark.

Step 13

In the **Wireshark - Unsaved packets** pop-up window, click **Quit without Saving.**

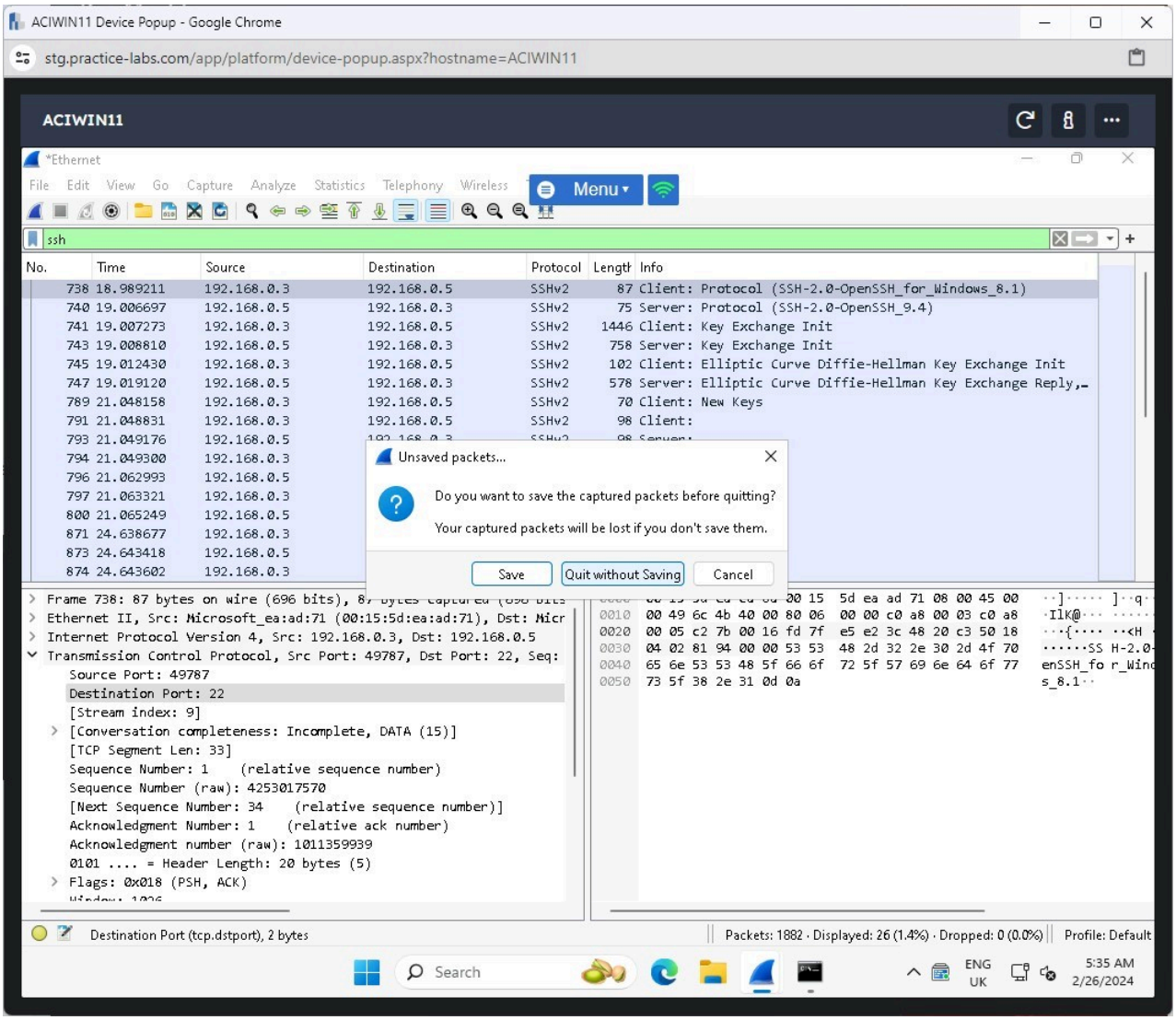


Figure 1.37 Screenshot of ACIWIN11: Displaying the Wireshark Unsaved packets pop-up and selecting Quit without Saving.

Step 14

Close the **Command Prompt** window.

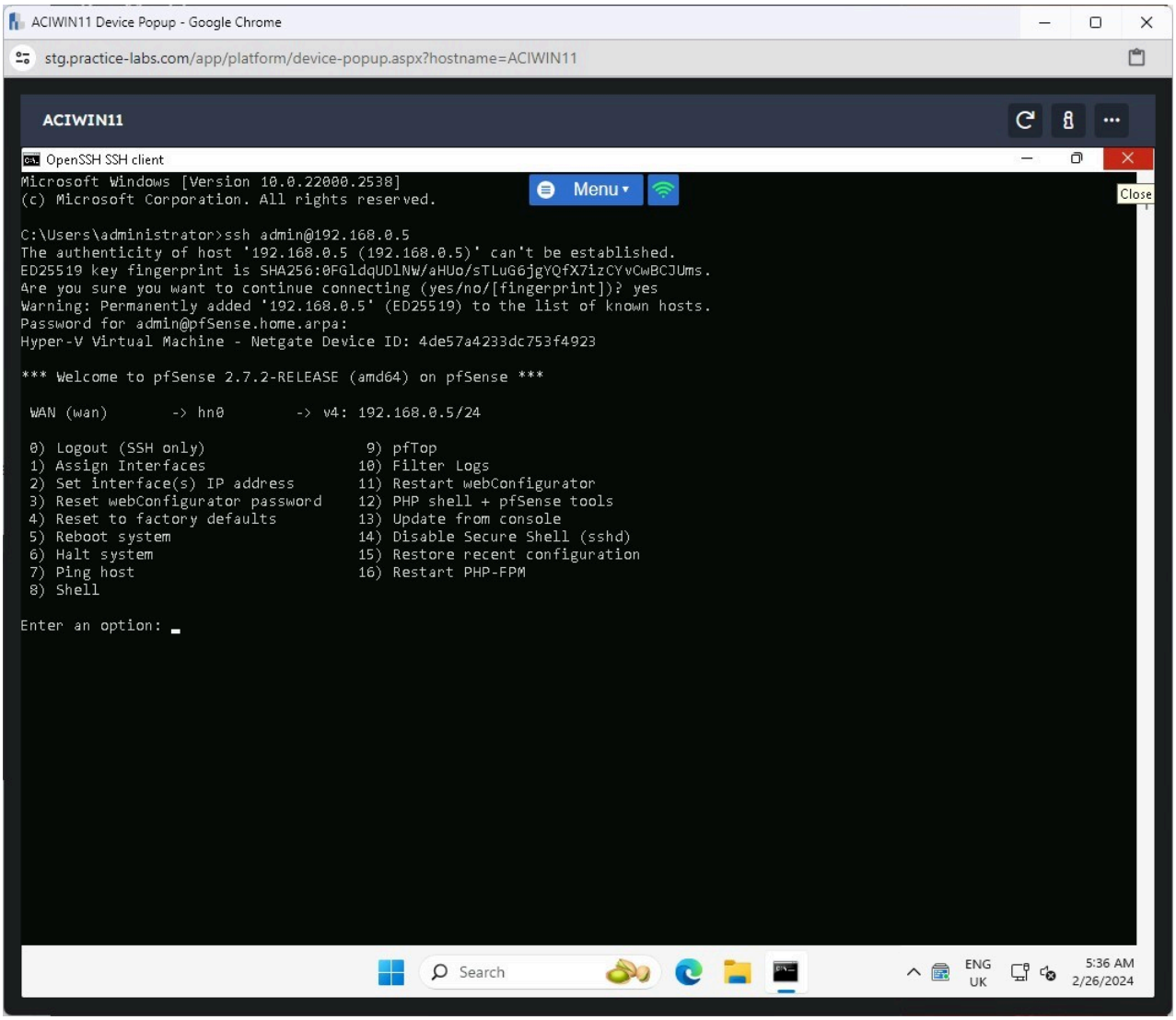


Figure 1.38 Screenshot of ACIWIN11: Displaying closing the Command Prompt window.

Task 4 - Access an HTTP Web Server

An HTTP web server is an application that serves web content, such as HTML files, to clients upon request. It listens for incoming HTTP requests on port 80 and responds with the requested web pages or resources.

In this task, you will monitor and analyze an HTTP connection with Wireshark.

Step 1

Connect to **ACIWIN11.**

In the **Search** field, type the following:

```
wireshark
```

Select **Wireshark** from the **Best match** pop-up menu.

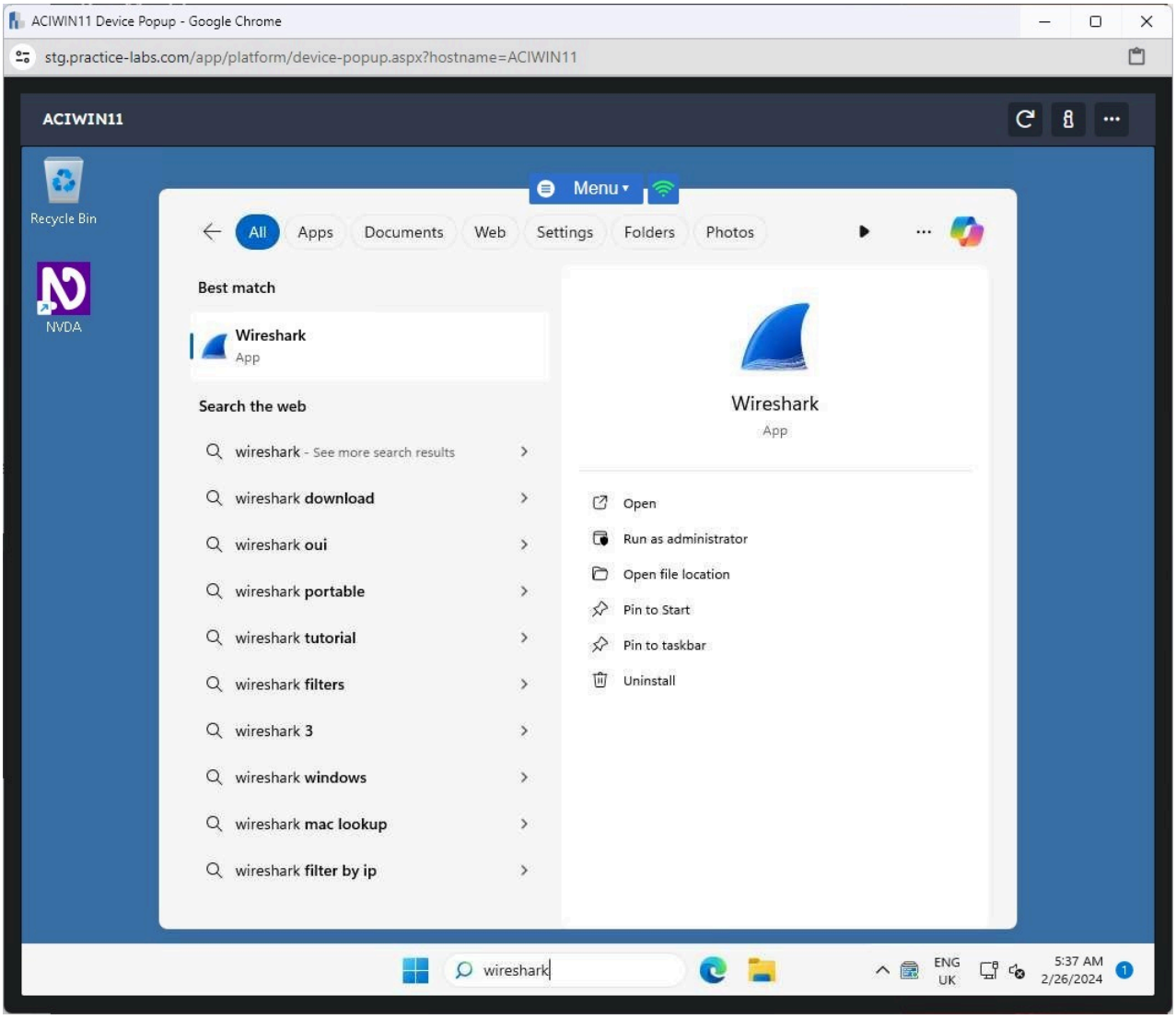


Figure 1.39 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Best match pop-up menu.

Step 2

In **Wireshark,** under the **Capture** menu, double-click on **Ethernet.**

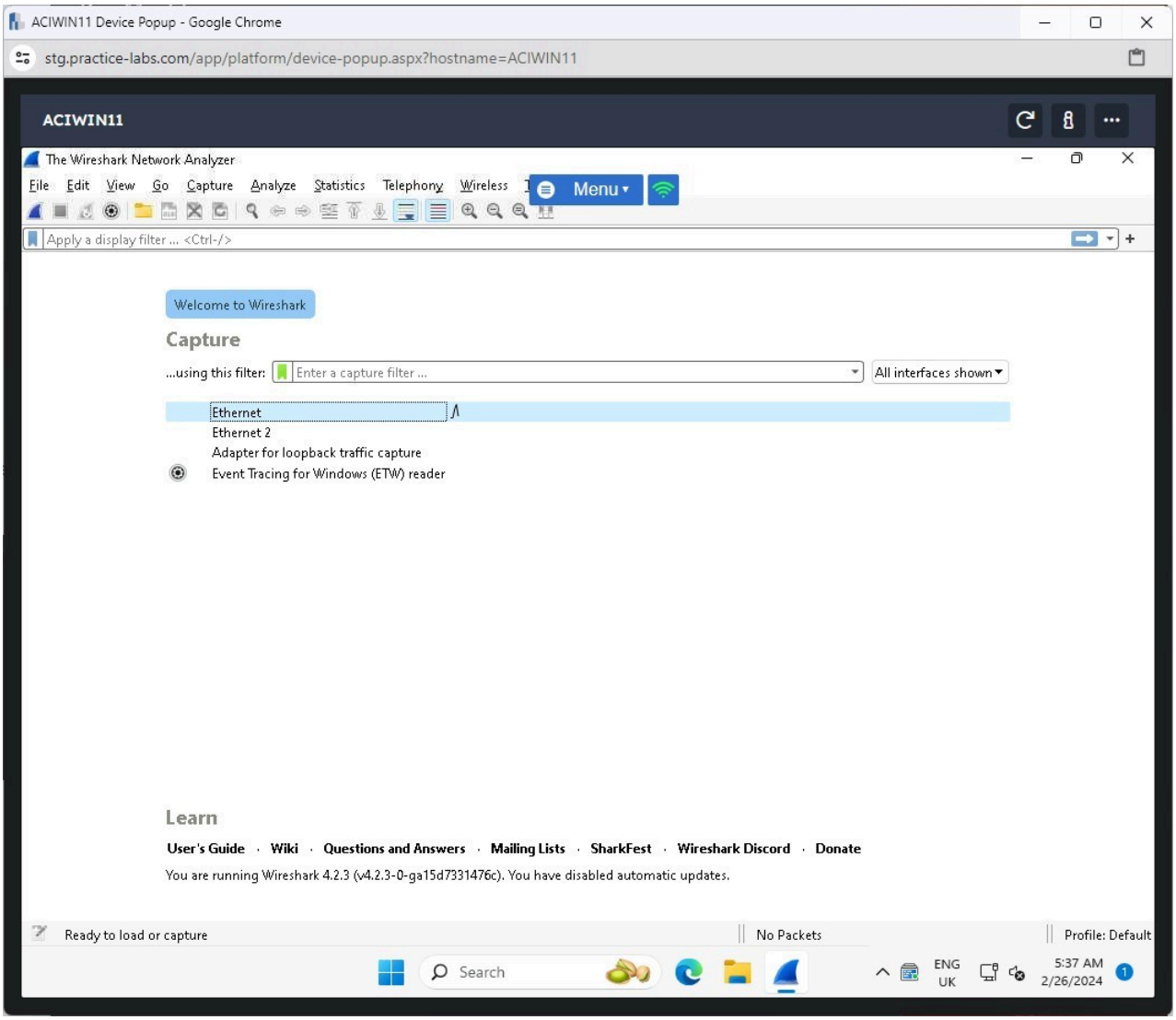


Figure 1.40 Screenshot of ACIWIN11: Displaying the Wireshark Capture menu and starting a capture of the Ethernet interface.

Step 3

In the **Taskbar,** select **Microsoft Edge.**

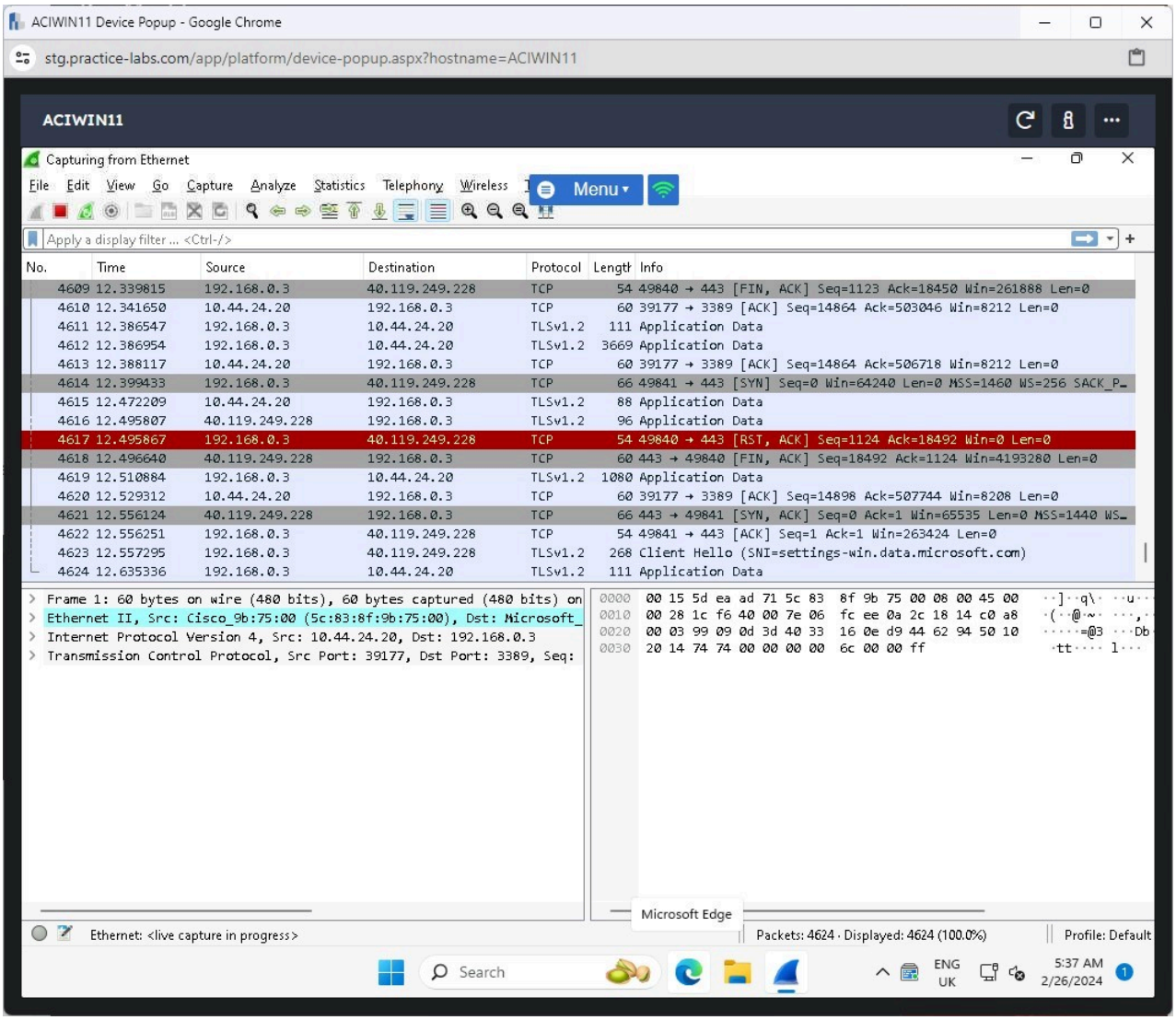


Figure 1.41 Screenshot of ACIWIN11: Displaying selecting Microsoft Edge from the Taskbar.

Step 4

In **Microsoft Edge**, type the following in the **URL** field:

```
192.168.0.4
```

Press **Enter.**

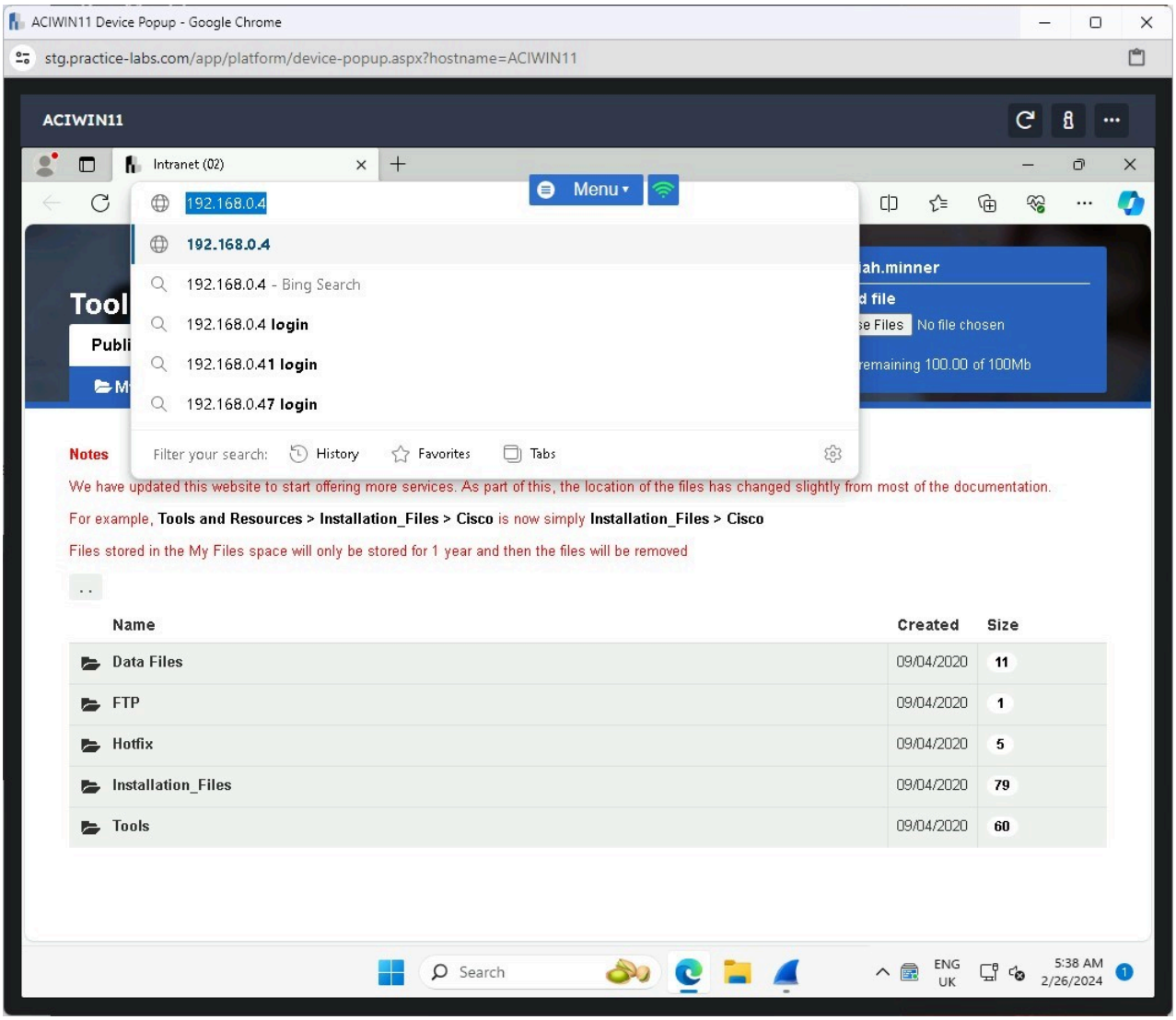


Figure 1.42 Screenshot of ACIWIN11: Displaying Microsoft Edge and navigating to 192.168.0.4.

> ***Note:*** *ACIALMA (192.168.0.4) hosts a web server on port 80. This can be accessed from ACIWIN11 through Microsoft Edge.*

Step 5

On the **Taskbar**, select **Wireshark.**

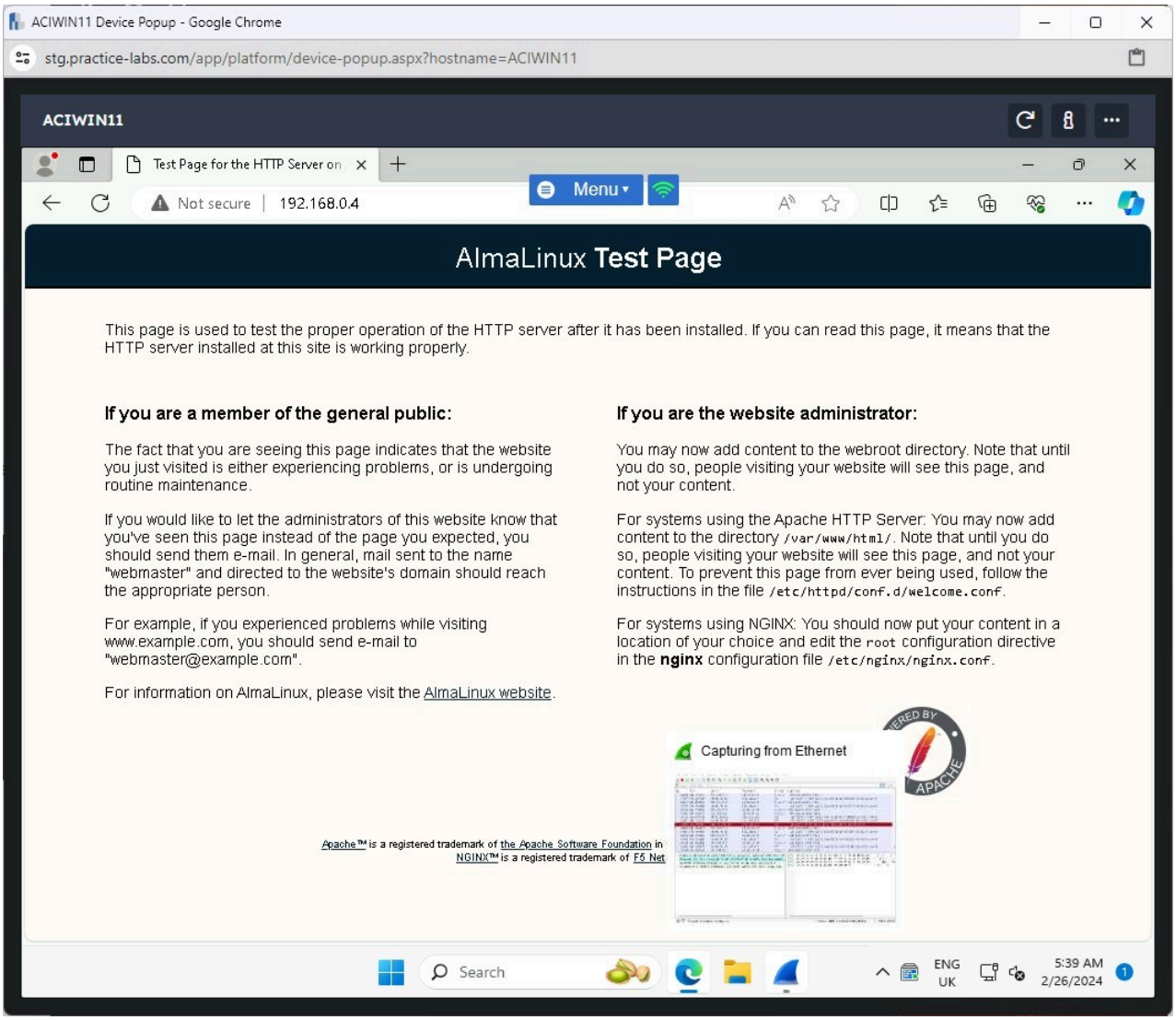


Figure 1.43 Screenshot of ACIWIN11: Displaying selecting Wireshark from the Taskbar.

Step 6

In **Wireshark,** select the red square to **Stop capturing packets.**

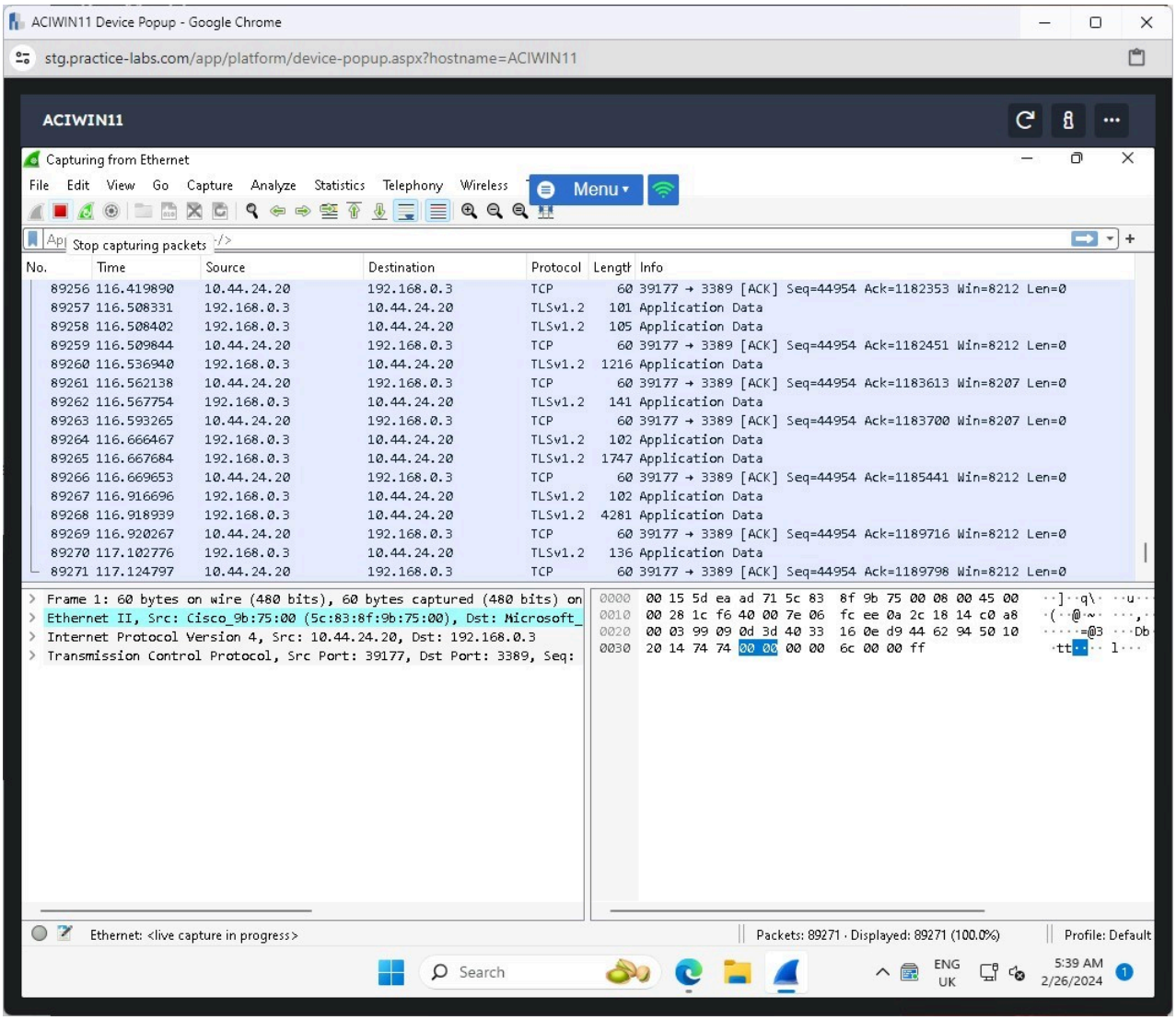


Figure 1.44 Screenshot of ACIWIN11: Displaying Wireshark and selecting the red square to Stop capturing packets.

Step 7

In **Wireshark,** in the **Apply a display filter** field, type the following:

```
http
```

Press **Enter.**

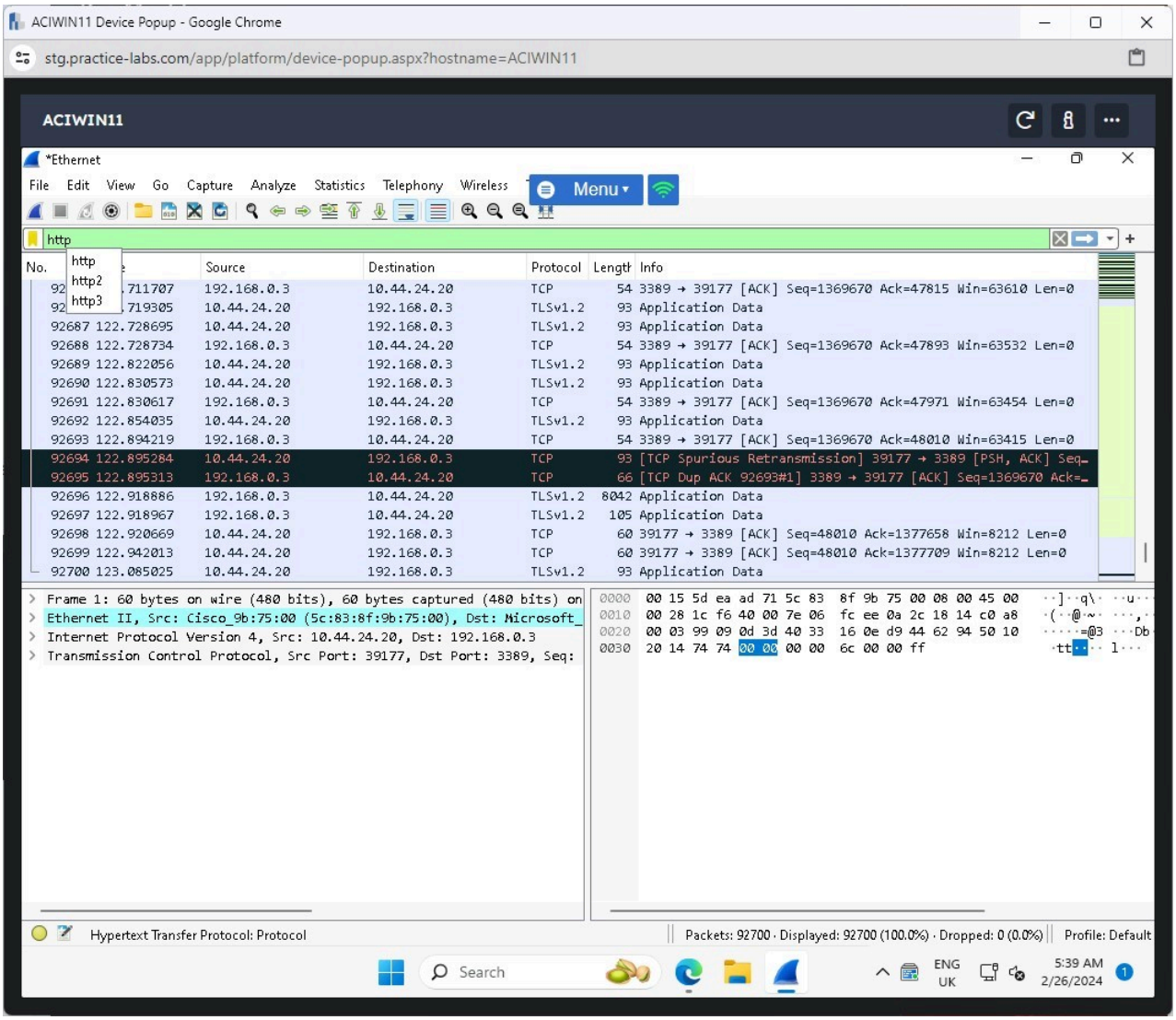


Figure 1.45 Screenshot of ACIWIN11: Displaying Wireshark and completing the Apply a display filter field.

Step 8

In **Wireshark**, select the first packet with a **Source** of **192.168.0.3** (the ACIWIN11 machine).

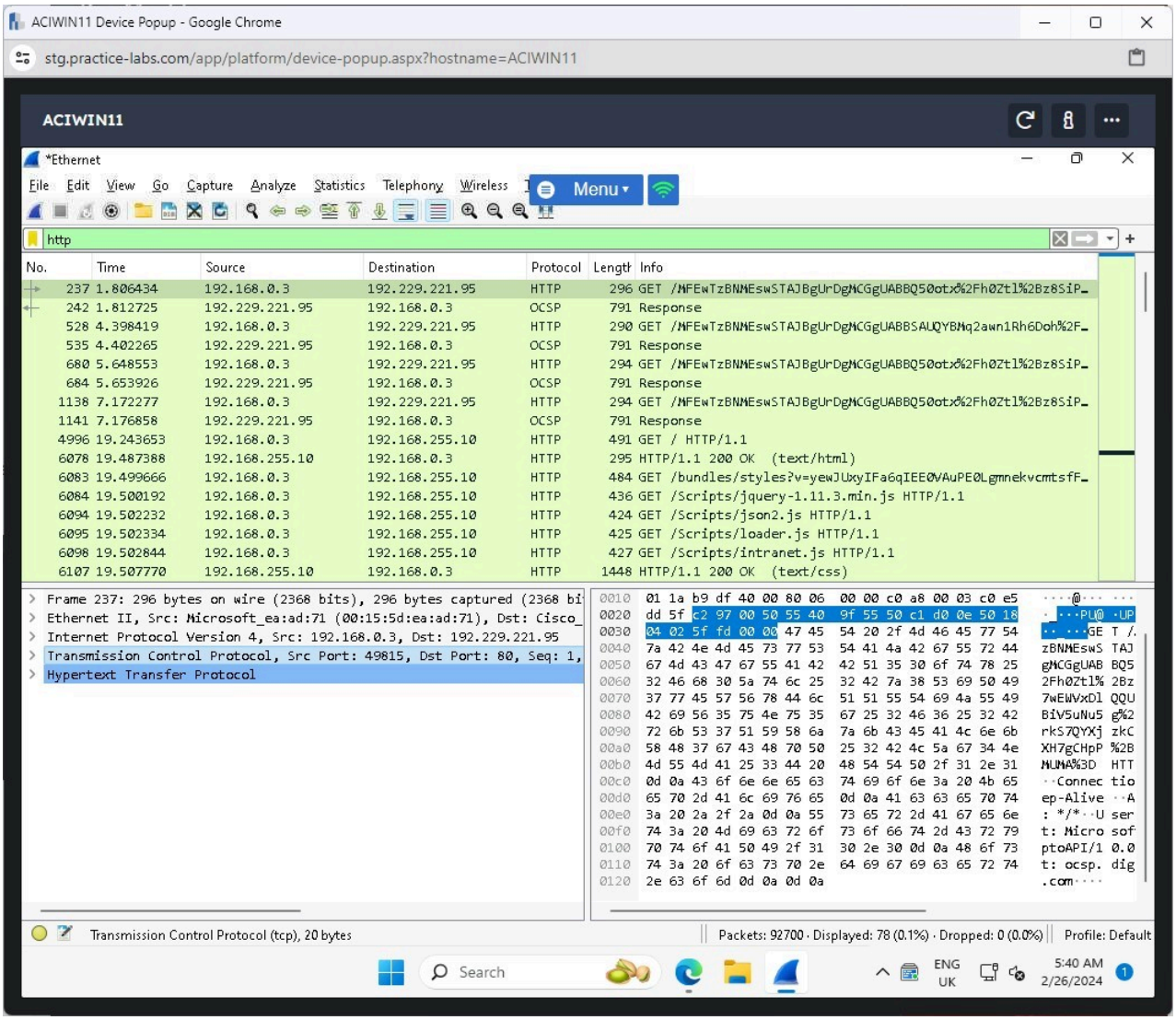


Figure 1.46 Screenshot of ACIWIN11: Displaying Wireshark and observing the first packet from 192.168.0.3.

> ***Note:*** *In the Packet Details pane, the packet is defined as using the* ***Transmission Control Protocol (TCP)*** *and is associated with the destination port* ***80****.*
> *HTTP (Hypertext Transfer Protocol) uses port 80 as its default port for communication between web servers and clients.*

Step 9

In **Wireshark,** expand the **Transmission Control Protocol** field.

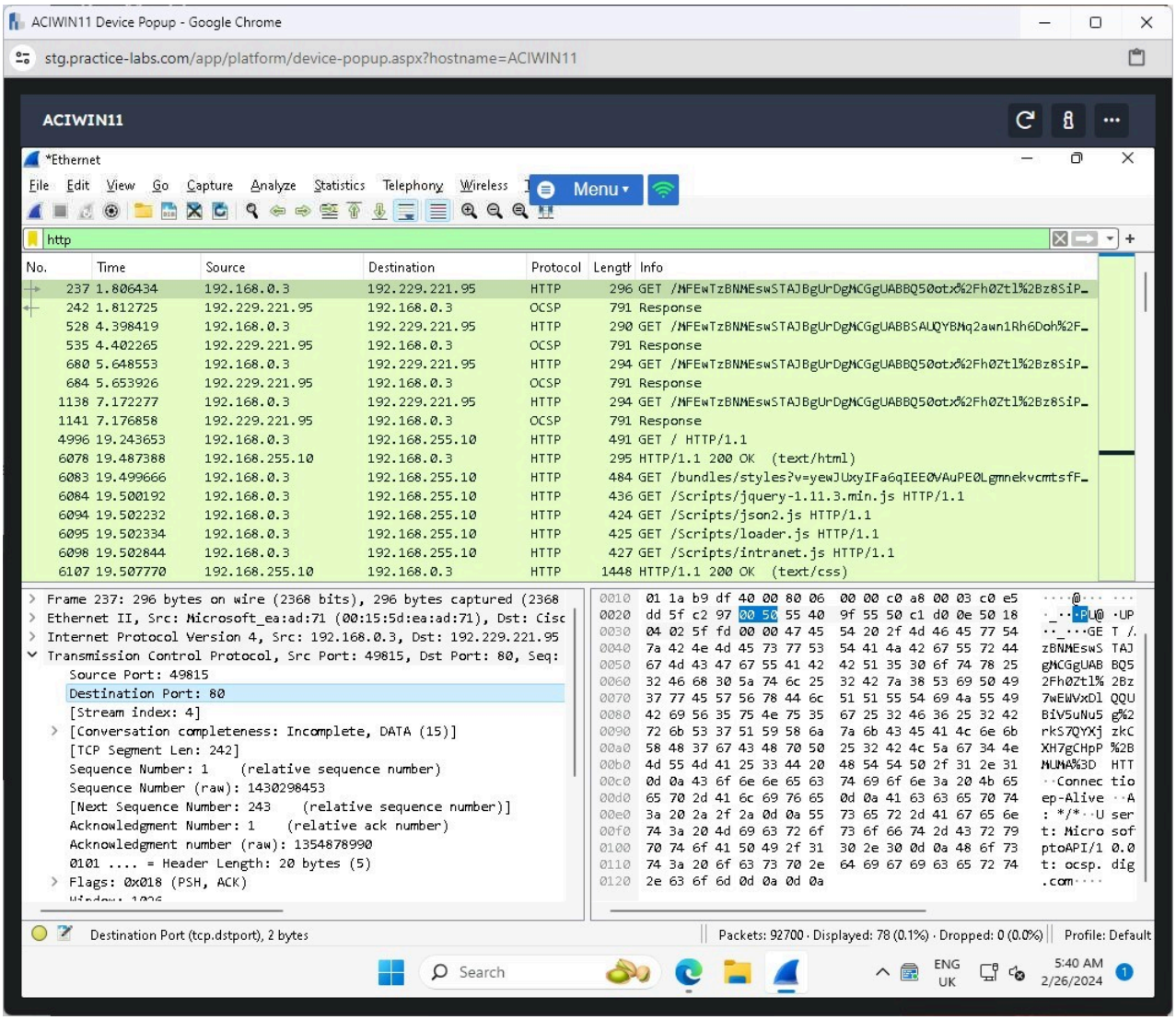


Figure 1.47 Screenshot of ACIWIN11: Displaying Wireshark and expanding the Transmission Control Protocol field.

Step 10

Close **Wireshark.**

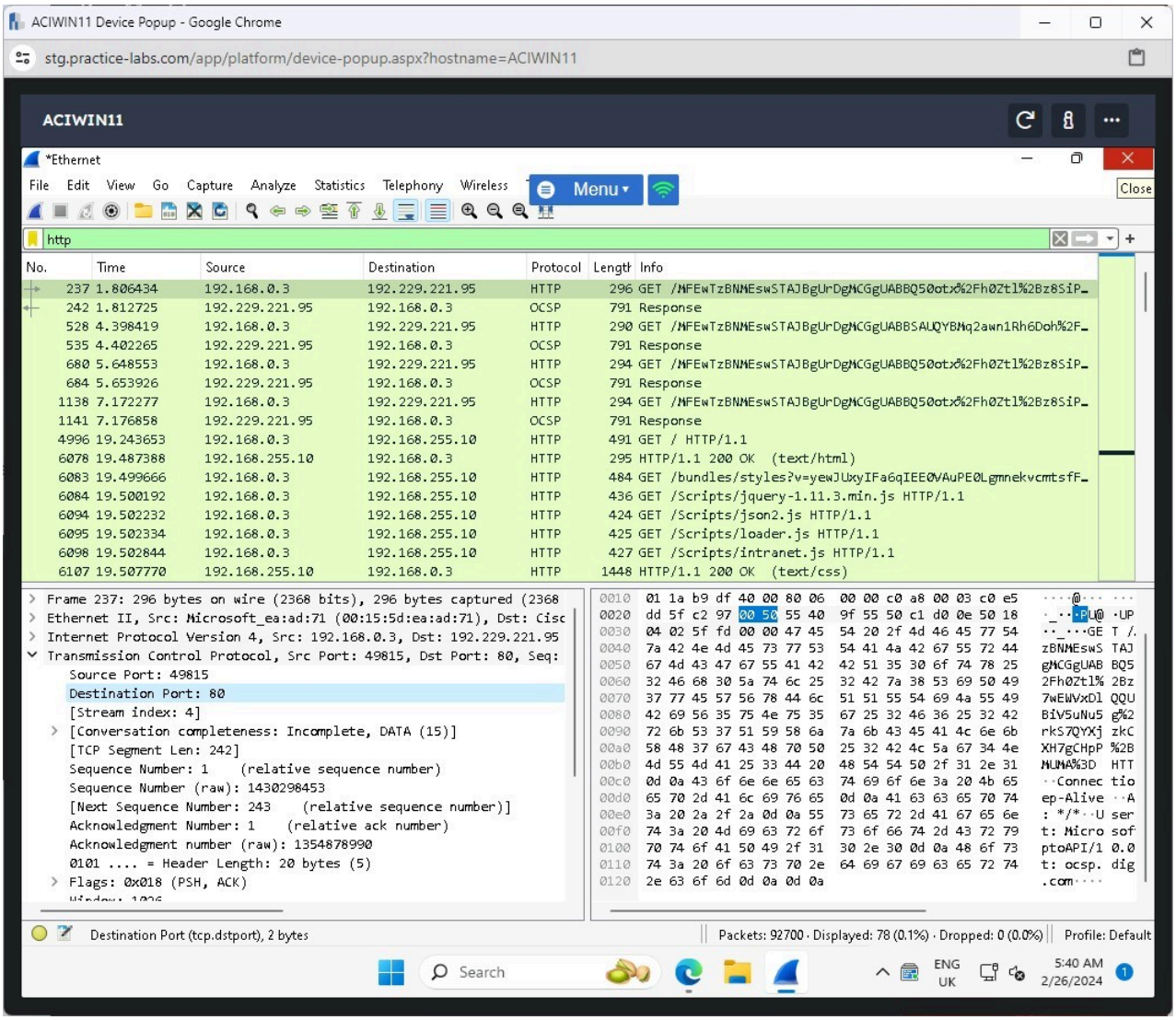


Figure 1.48 Screenshot of ACIWIN11: Displaying closing Wireshark.

Step 11

In the **Wireshark - Unsaved packets** pop-up window, click **Quit without Saving.**

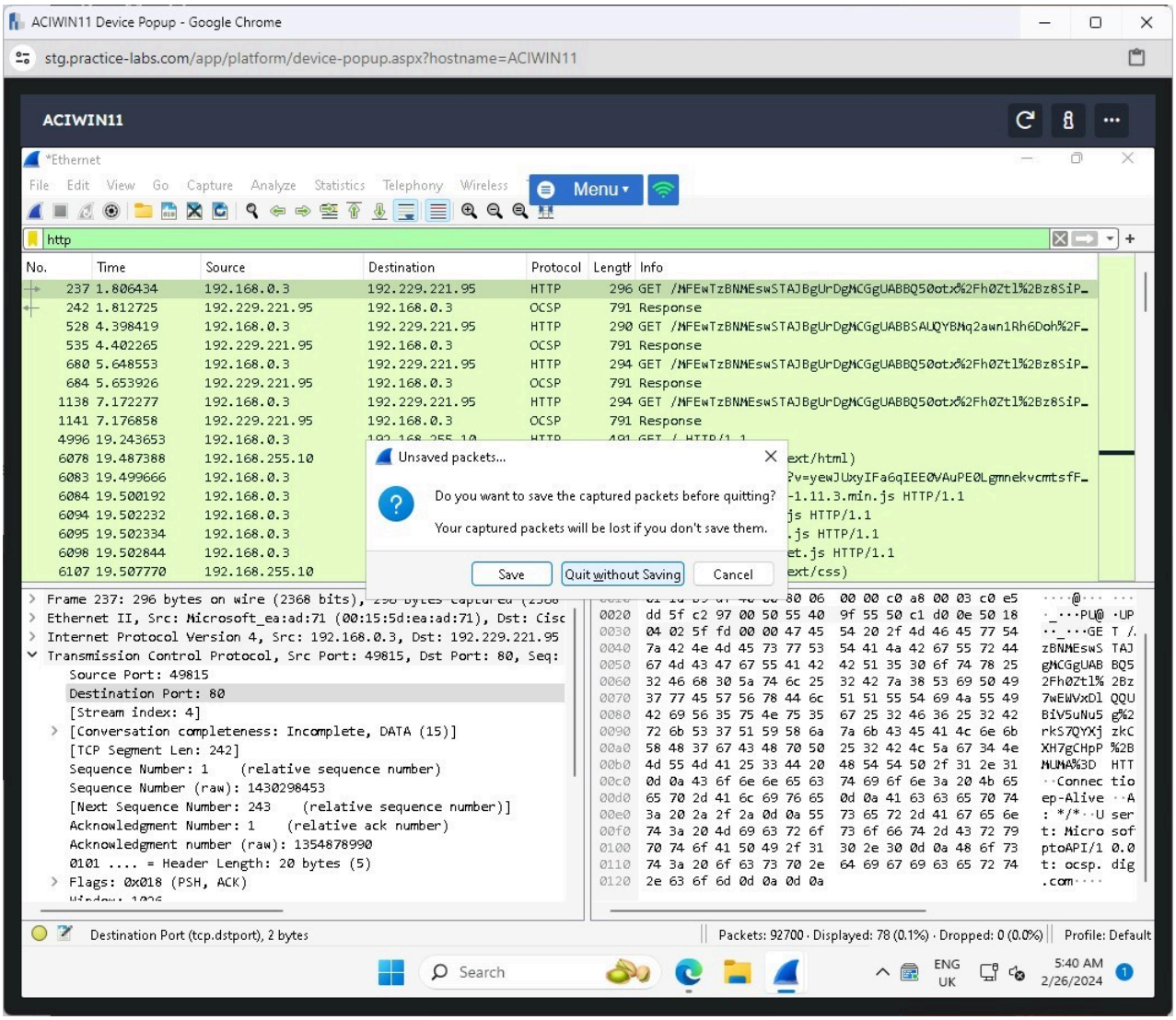


Figure 1.49 Screenshot of ACIWIN11: Displaying the Wireshark - Unsaved packets pop-up window and selecting Quit without Saving.

Step 12

Close **Microsoft Edge.**

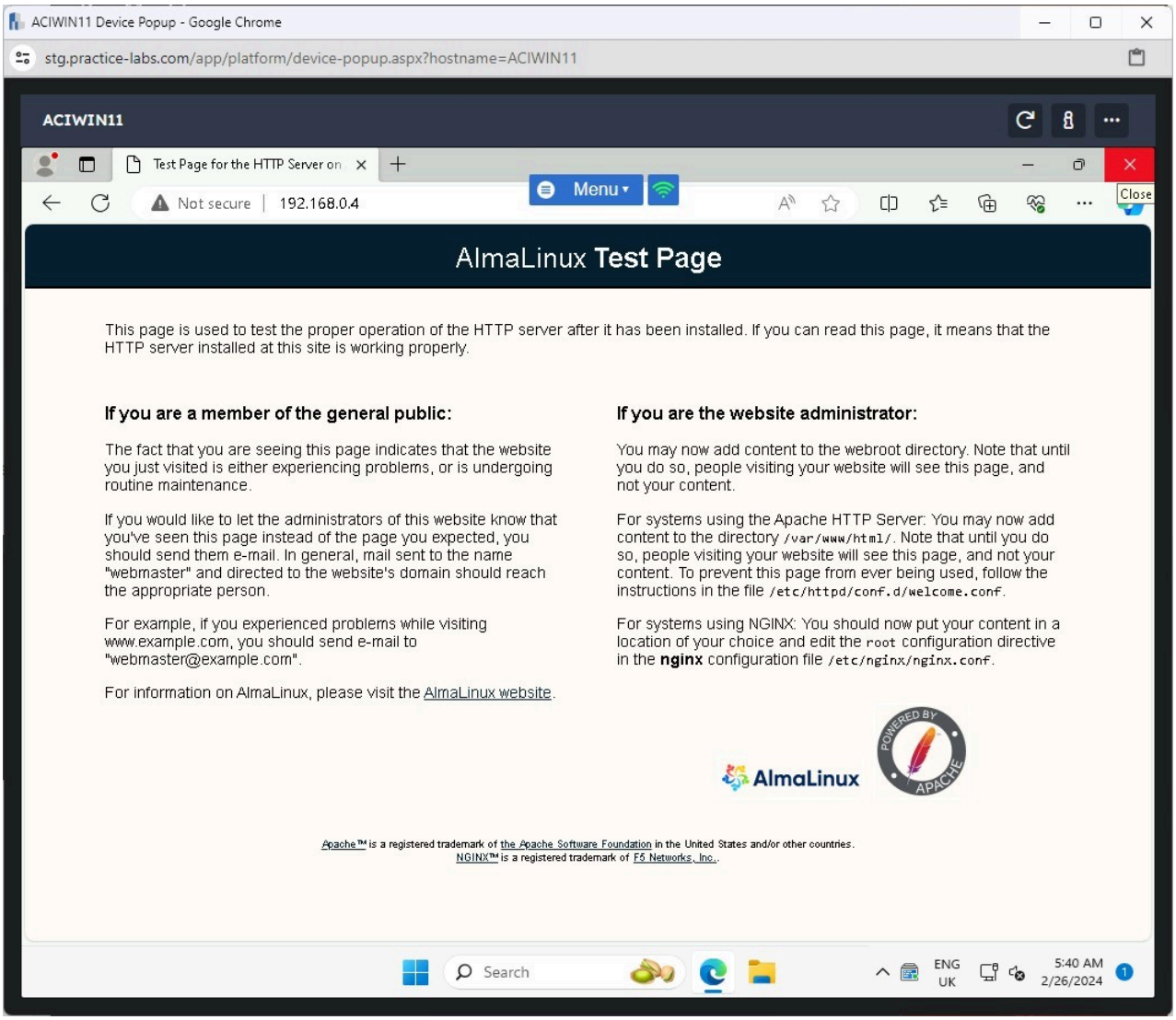


Figure 1.50 Screenshot of ACIWIN11: Displaying closing Microsoft Edge.

Exercise 2 - Network Port Scan

A network port scan is used to discover open ports on network machines. It involves sending packets to a range of port numbers on a target IP address and analyzing the responses to determine which ports are open, closed, or filtered. Port scanning is commonly used for security assessments, network troubleshooting, and reconnaissance purposes to identify potential vulnerabilities or misconfigurations.

In this exercise, you will conduct a network port scan using the Network Mapper (nmap).

Learning Outcomes

After completing this exercise, you should be able to:

- Conduct a Network Port Scan

Your Devices

You will be using the following devices in this lab. Please power these on now.

- **ACIDC01 -** Windows Server 2022 - Domain Controller
- **ACIDM01 -** Windows Server 2022 - Domain Member Server
- **ACIWIN11 -** Windows 11 PRO - Domain Member Workstation
- **ACIALMA -** Alma Linux 9.3 - Stand-alone Linux Workstation
- **ACIPFSENSE -** PFSense v2.7.2 - Virtual Router

Task 1 - Conduct a Network Port Scan

Nmap is an open-source network scanning tool used for locating hosts and services on a network. It employs various scanning techniques to identify open ports, services, and potential vulnerabilities in network systems.

When conducting a default scan with nmap, a TCP SYN scan on the 1,000 most common TCP ports is performed. To do this, nmap sends TCP SYN packets to the target ports and analyzes the responses to determine their status, whether open, closed, or filtered. This default scan provides an overview of the target's open ports and is commonly used for initial reconnaissance in network security assessments and configuration checks by network administrators.

In this task, you will conduct a default network scan of the 192.168.0.0/24 network.

Step 1

Connect to **ACIWIN11.**

In the **Taskbar - Search** field, type the following:

```
cmd
```

Select **Command Prompt** from the **Best match** pop-up menu.

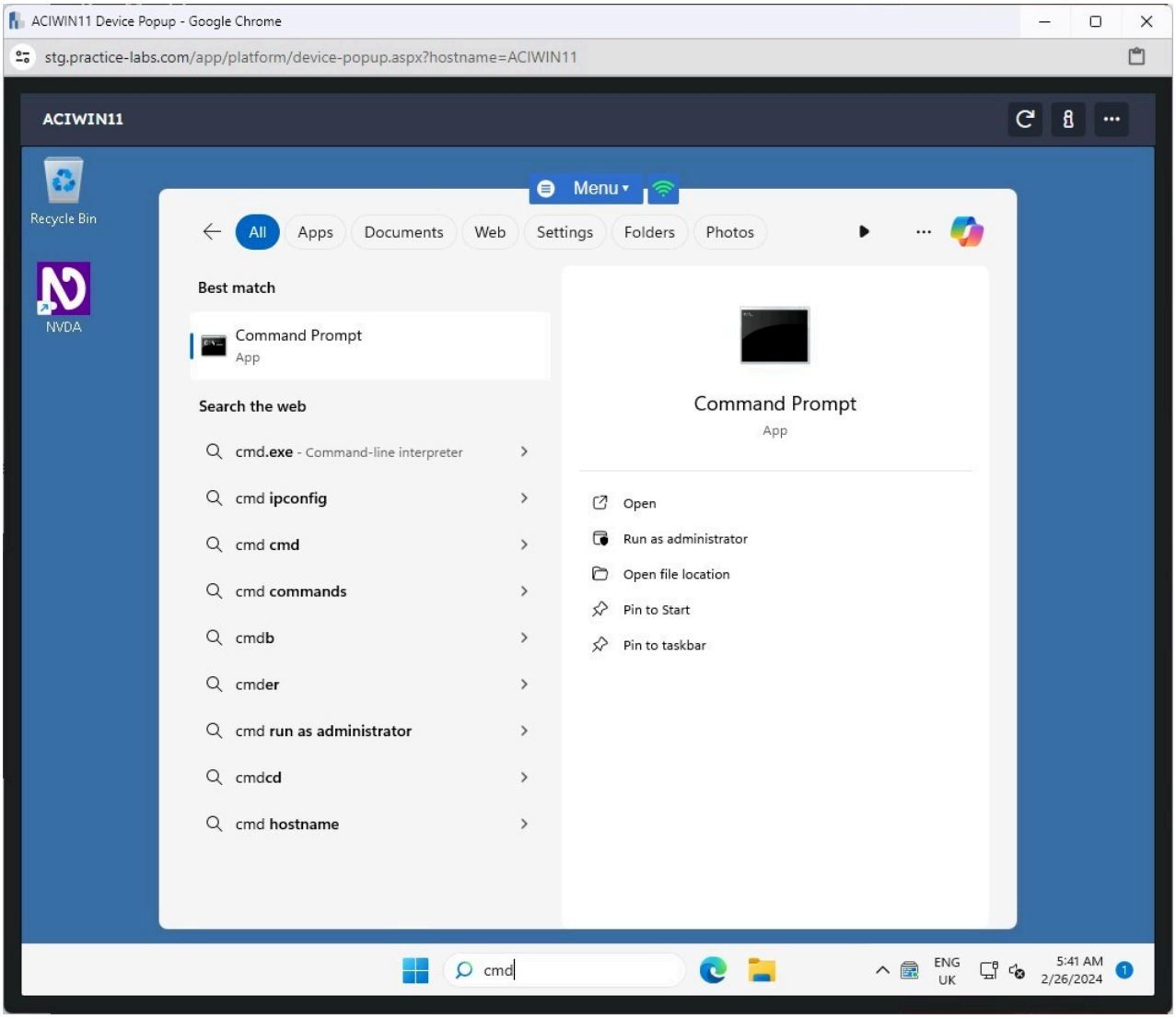


Figure 2.1 Screenshot of ACIWIN11: Displaying selecting Command Prompt from the Best match pop-up menu.

Step 2

In the **Command Prompt,** type the following:

```
nmap 192.168.0.0/24
```

Press **Enter.**

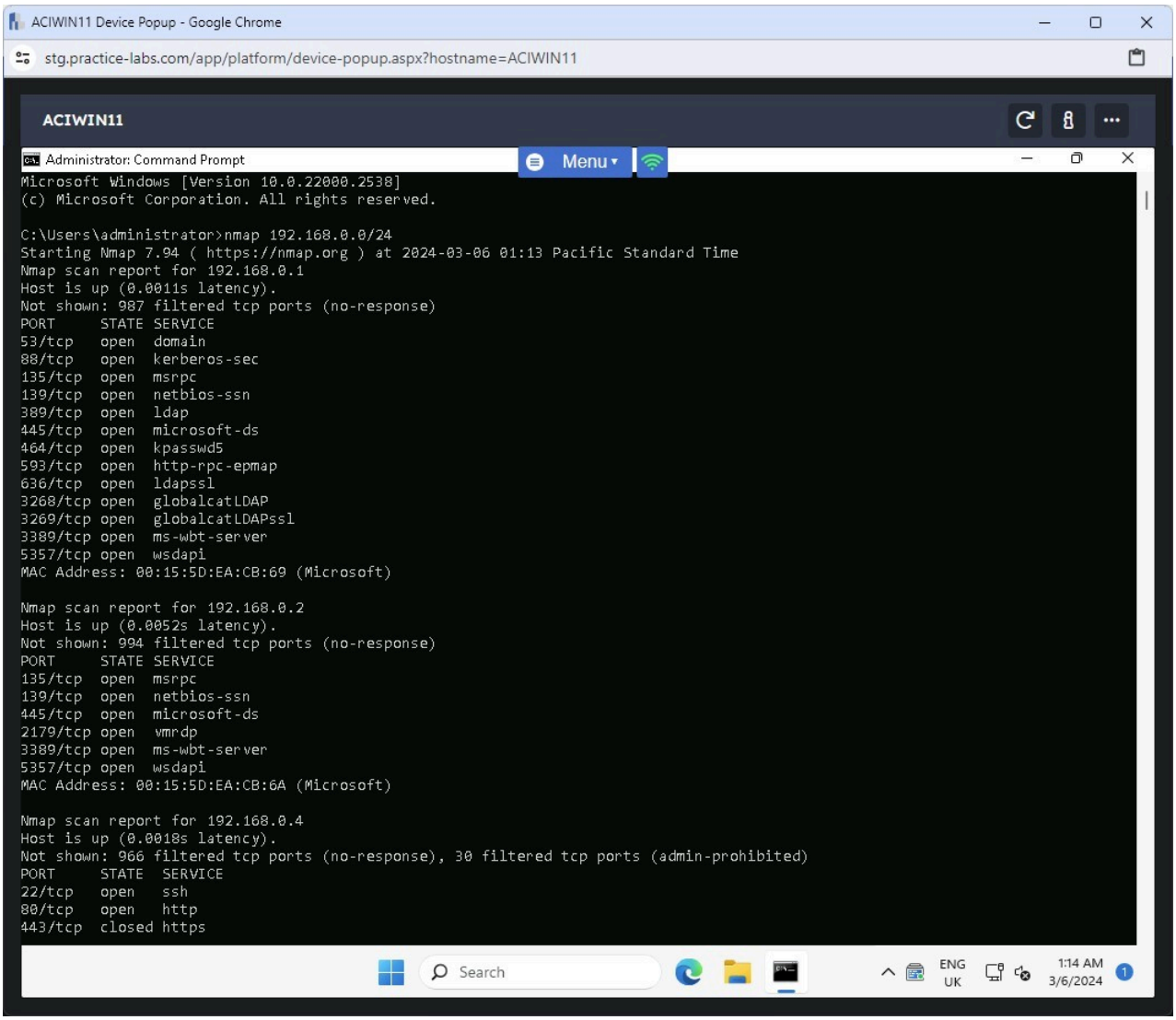


Figure 2.2 Screenshot of ACIWIN11: Displaying the Command Prompt window and conducting an nmap port scan.

Note: *The command* ***nmap 192.168.0.0/24*** *initiates a network scan using Nmap on the range of IP addresses from 192.168.0.1 to 192.168.0.254. This scan aims to discover active hosts and open ports within the specified subnet.*
The results are displayed per machine and identify the IP address and MAC address of the machine along with the Port, State of the Port (open/closed/filtered), and the Service being hosted on that Port.
Scan through the results of the nmap scan with a focus on identifying ports and associated services.

Step 3

Close the **Command Prompt** window.

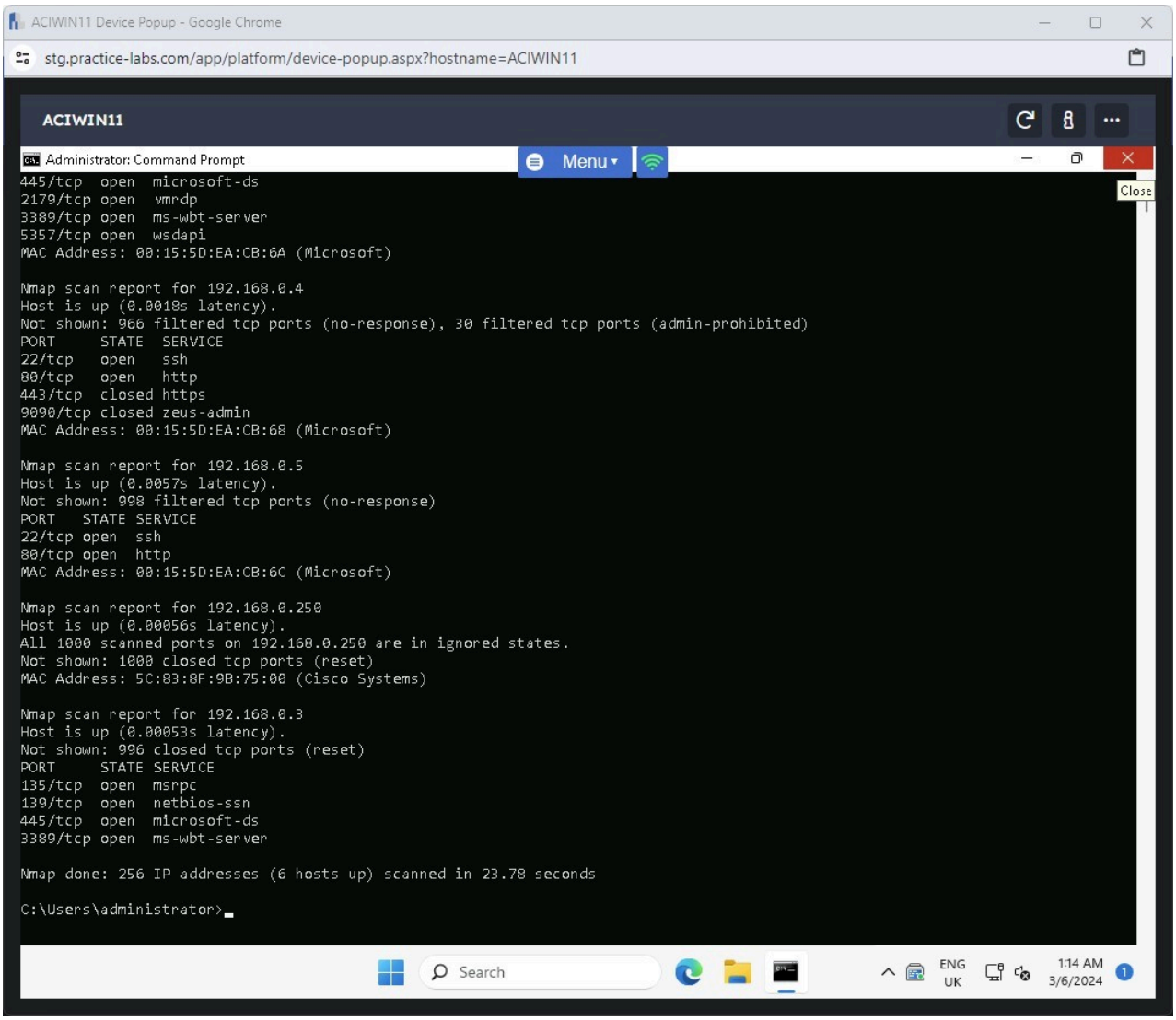


Figure 2.3 Screenshot of ACIWIN11: Displaying closing the Command Prompt window.

Keep all devices that you have powered on in their current state and proceed to the **Review** section.

Review

Well done, you have completed the **Networking Ports and Protocols** Practice Lab.

Throughout this module, you analyzed packets with Wireshark and conducted a network port scan, enabling an understanding of some key protocols and services, including DNS, SSH, HTTP, ICMP, TCP, and UDP.

Summary

You completed the following exercises:

- Exercise 1 - Discover Protocols with Wireshark
- Exercise 2 - Network Port Scan

You should now be able to:

- Conduct a Ping
- Conduct a DNS Query
- Make an SSH Connection
- Access an HTTP Web Server
- Conduct a Network Port Scan

Feedback

Shutdown all virtual machines used in this lab. Alternatively, you can log out of the lab platform.