Module 03: Scanning Networks

Scenario

Earlier, you gathered all possible information about the target such as organization information (employee details, partner details, web links, etc.), network information (domains, sub-domains, sub sub-domains, IP addresses, network topology, etc.), and system information (OS details, user accounts, passwords, etc.).

Now, as an ethical hacker, or as a penetration tester (hereafter, pen tester), your next step will be to perform port scanning and network scanning on the IP addresses that you obtained in the information-gathering phase. This will help you to identify an entry point into the target network.

Scanning itself is not the actual intrusion, but an extended form of reconnaissance in which the ethical hacker and pen tester learns more about the target, including information about open ports and services, OSes, and any configuration lapses. The information gleaned from this reconnaissance helps you to select strategies for the attack on the target system or network.

This is one of the most important phases of intelligence gathering, which enables you to create a profile of the target organization. In the process of scanning, you attempt to gather information, including the specific IP addresses of the target system that can be accessed over the network (live hosts), open ports, and respective services running on the open ports and vulnerabilities in the live hosts.

Port scanning will help you identify open ports and services running on specific ports, which involves connecting to Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) system ports. Port scanning is also used to discover the vulnerabilities in the services running on a port.

The labs in this module will give you real-time experience in gathering information about the target organization using various network scanning and port scanning techniques.

Objective

The objective of this lab is to conduct network scanning, port scanning, analyzing the network vulnerabilities, etc.

Network scans are needed to:

- Check live systems and open ports
- Identify services running in live systems
- Perform banner grabbing/OS fingerprinting
- Identify network vulnerabilities

Overview of Scanning Networks

Network scanning is the process of gathering additional detailed information about the target by using highly complex and aggressive reconnaissance techniques. The purpose of scanning is to discover exploitable communication channels, probe as many listeners as possible, and keep track of the responsive ones.

Types of scanning:

- **Port Scanning**: Lists open ports and services
- **Network Scanning**: Lists the active hosts and IP addresses
- **Vulnerability Scanning**: Shows the presence of known weaknesses

Lab Tasks

Ethical hackers and pen testers use numerous tools and techniques to scan the target network. Recommended labs that will assist you in learning various network scanning techniques include:

1. Perform host discovery
    - Perform host discovery using Nmap
    - Perform host discovery using Angry IP Scanner
2. Perform port and service discovery
    - Perform port and service discovery using MegaPing
    - Perform port and service discovery using NetScanTools Pro
    - Perform port scanning using sx tool

- Explore various network scanning techniques using Nmap
- Explore various network scanning techniques using Hping3

3. Perform OS discovery

- Identify the target system's OS with Time-to-Live (TTL) and TCP window sizes using Wireshark
- Perform OS discovery using Nmap Script Engine (NSE)
- Perform OS discovery using Unicornscan

4. Scan beyond IDS and Firewall

- Scan beyond IDS/firewall using various evasion techniques
- Create custom packets using Colasoft Packet Builder to scan beyond the IDS/firewall
- Create custom UDP and TCP packets using Hping3 to scan beyond the IDS/firewall

5. Perform network scanning using various scanning tools

- Scan a target network using Metasploit

Lab 1: Perform Host Discovery

Lab Scenario

As a professional ethical hacker or pen tester, you should be able to scan and detect the active network systems/devices in the target network. During the network scanning phase of security assessment, your first task is to scan the network systems/devices connected to the target network within a specified IP range and check for live systems in the target network.

Lab Objectives

- Perform host discovery using Nmap
- Perform host discovery using Angry IP Scanner

Overview of Host Discovery

Host discovery is considered the primary task in the network scanning process. It is used to discover the active/live hosts in a network. It provides an accurate status of the systems in the network, which, in turn, reduces the time spent on scanning every port on every system in a sea of IP addresses in order to identify whether the target host is up.

The following are examples of host discovery techniques:

- ARP ping scan
- UDP ping scan
- ICMP ping scan (ICMP ECHO ping, ICMP timestamp, ping ICMP, and address mask ping)
- TCP ping scan (TCP SYN ping and TCP ACK ping)
- IP protocol ping scan

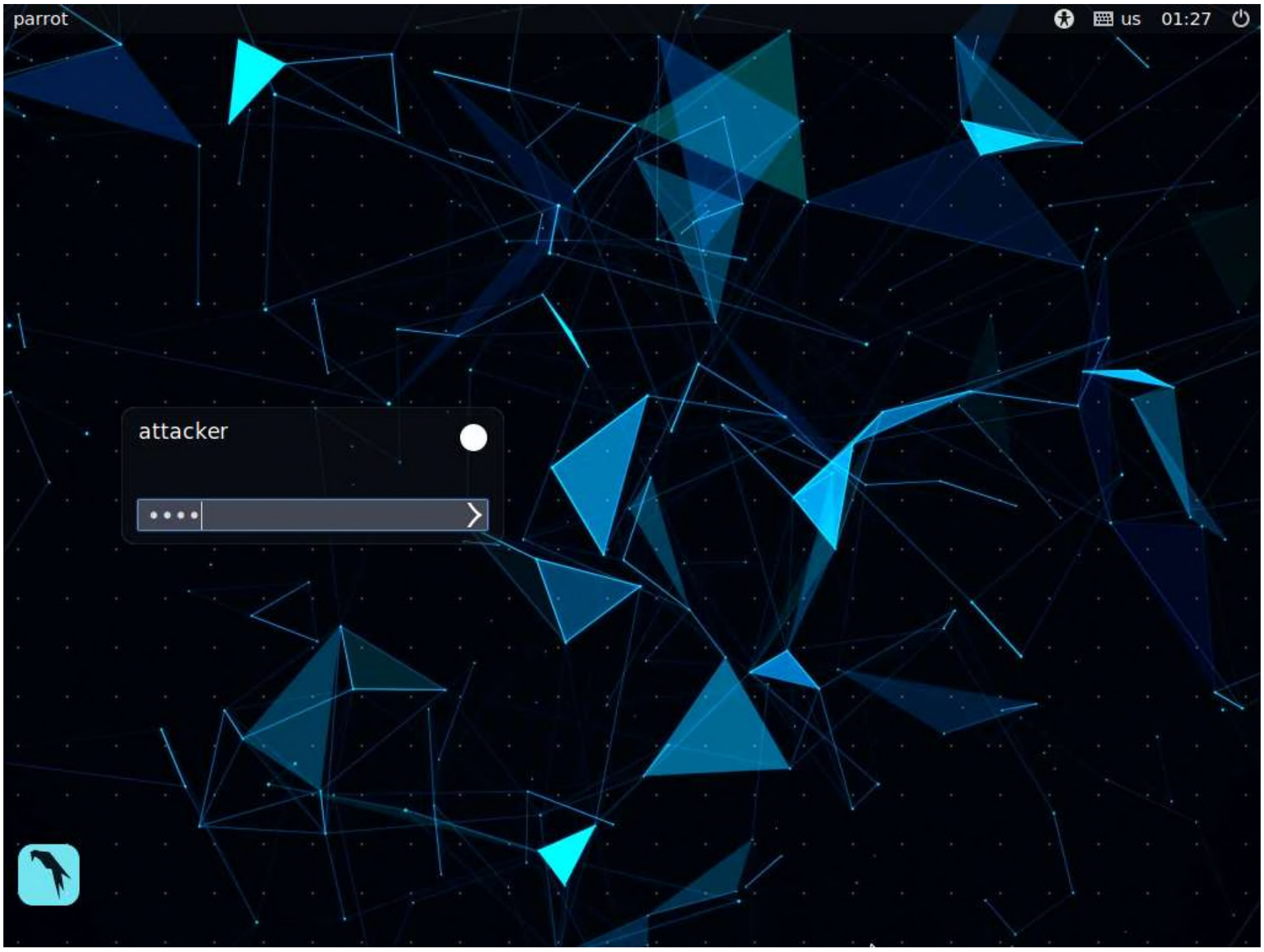
Task 1: Perform Host Discovery using Nmap

Nmap is a utility used for network discovery, network administration, and security auditing. It is also used to perform tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime.

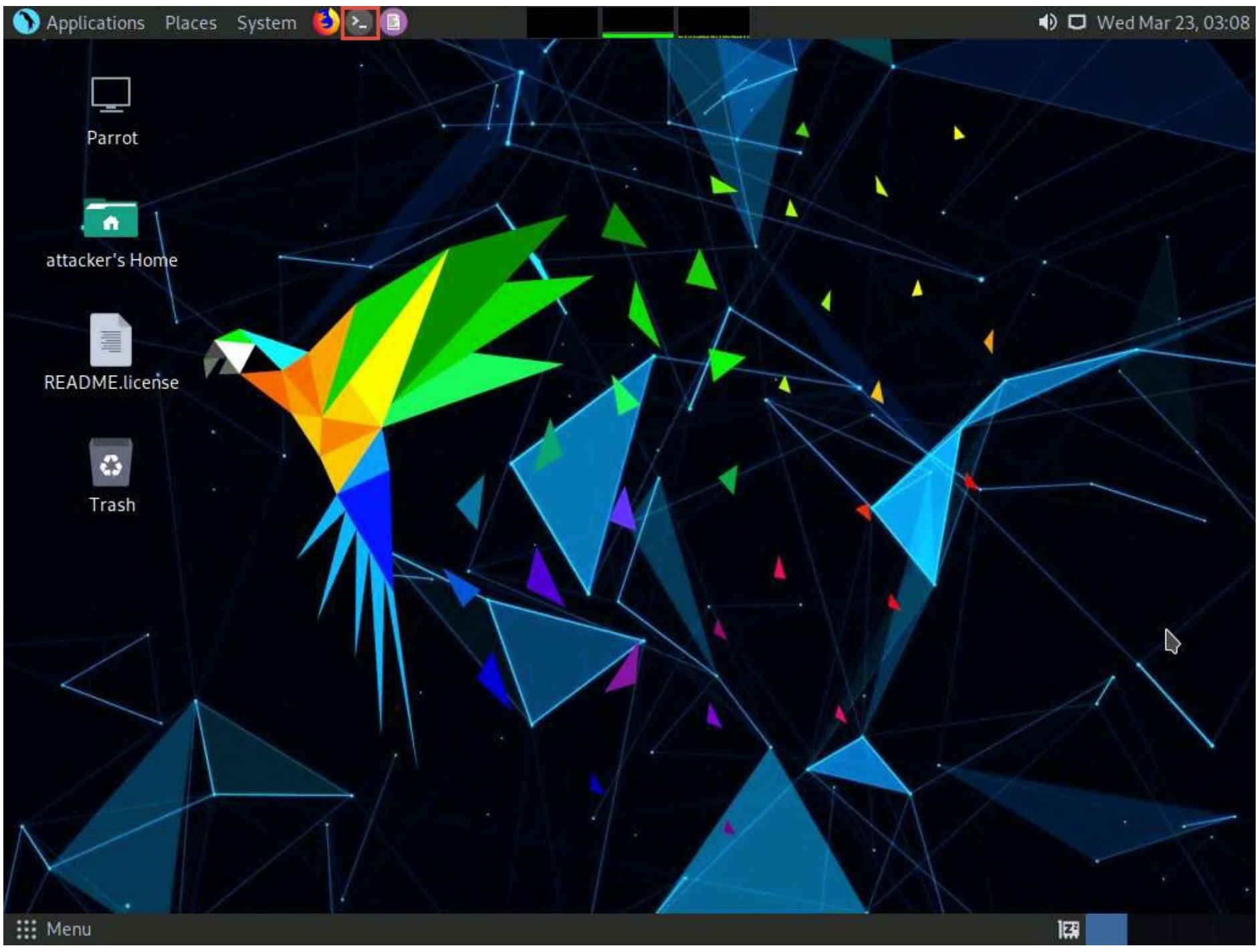
Here, we will use Nmap to discover a list of live hosts in the target network. We can use Nmap to scan the active hosts in the target network using various host discovery techniques such as ARP ping scan, UDP ping scan, ICMP ECHO ping scan, ICMP ECHO ping sweep, etc.

1. By default the **Parrot Security** machine is selected.
2. In the login page, the **attacker** username will be selected by default. Enter password as **toor** in the **Password** field and press **Enter** to log in to the machine.

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



3. Click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.



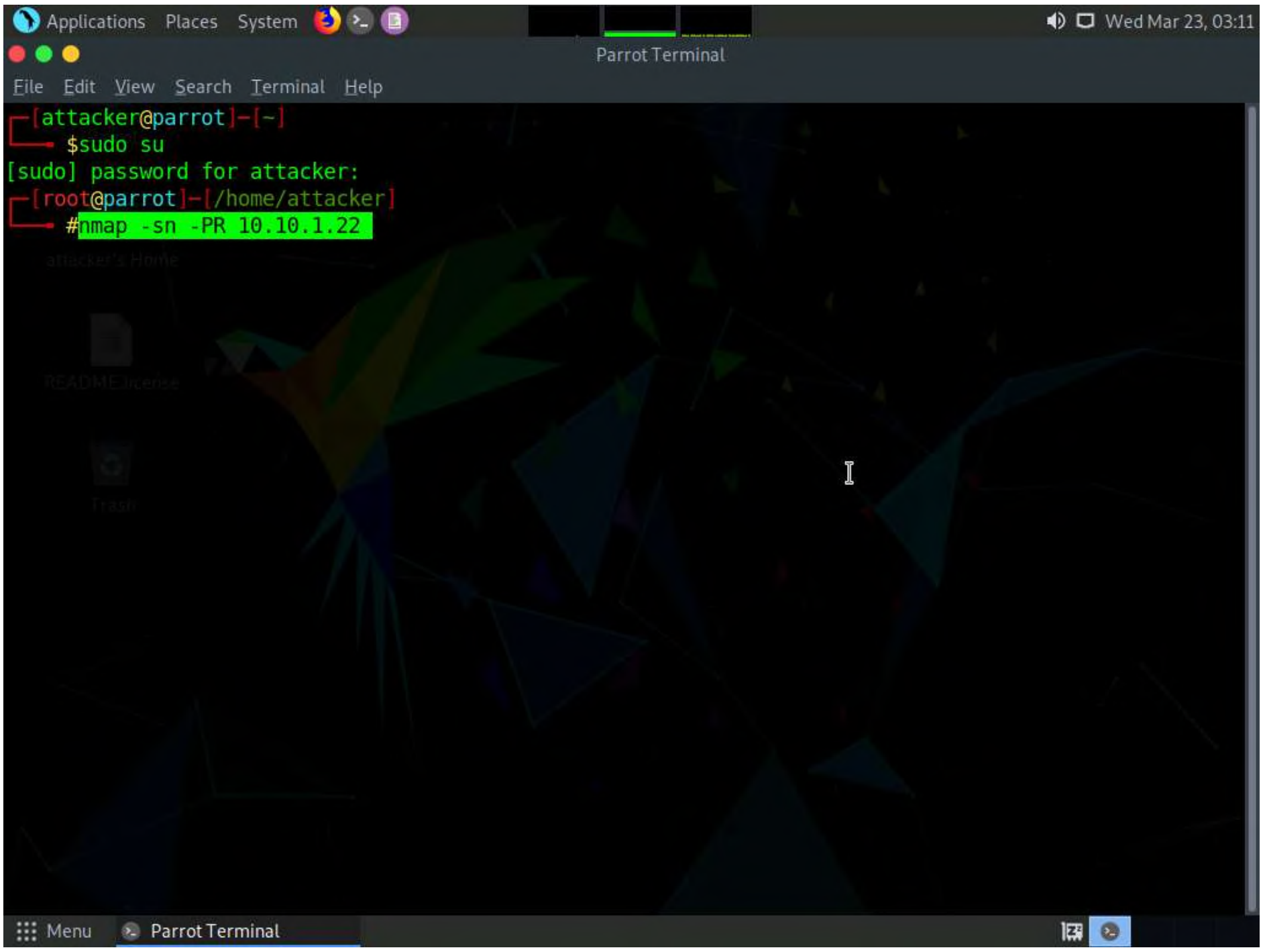
4. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

5. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

Note: The password that you type will not be visible.

6. In the terminal window, type the command **nmap -sn -PR [Target IP Address]** (here, the target IP address is **10.10.1.22**) and press **Enter**.

Note: **-sn**: disables port scan and **-PR**: performs ARP ping scan.

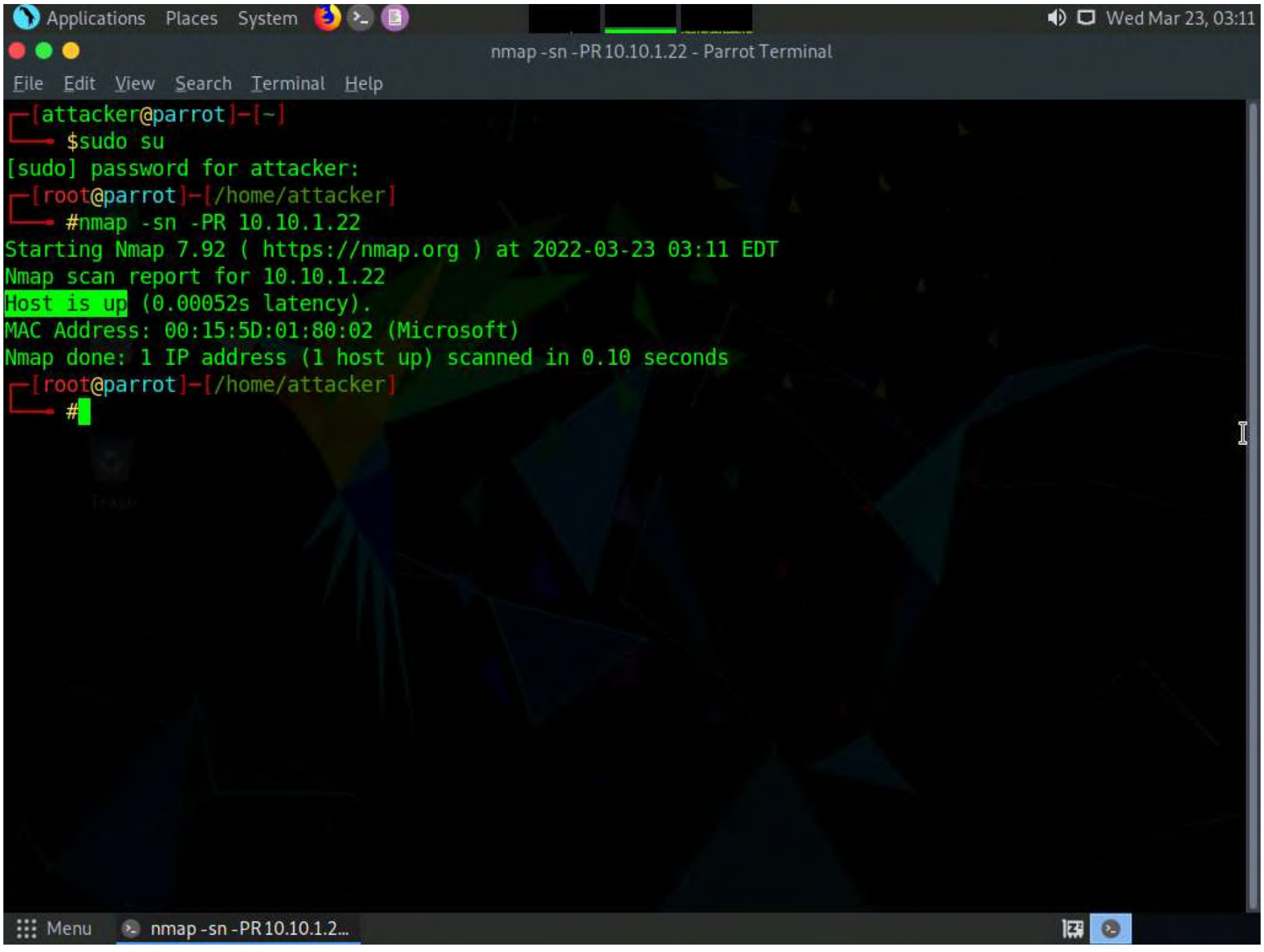


7. The scan results appear, indicating that the target **Host is up**, as shown in the screenshot.

Note: In this lab, we are targeting the **Windows Server 2022** (**10.10.1.22**) machine.

Note: The ARP ping scan probes ARP request to target host; an ARP response means that the host is active.

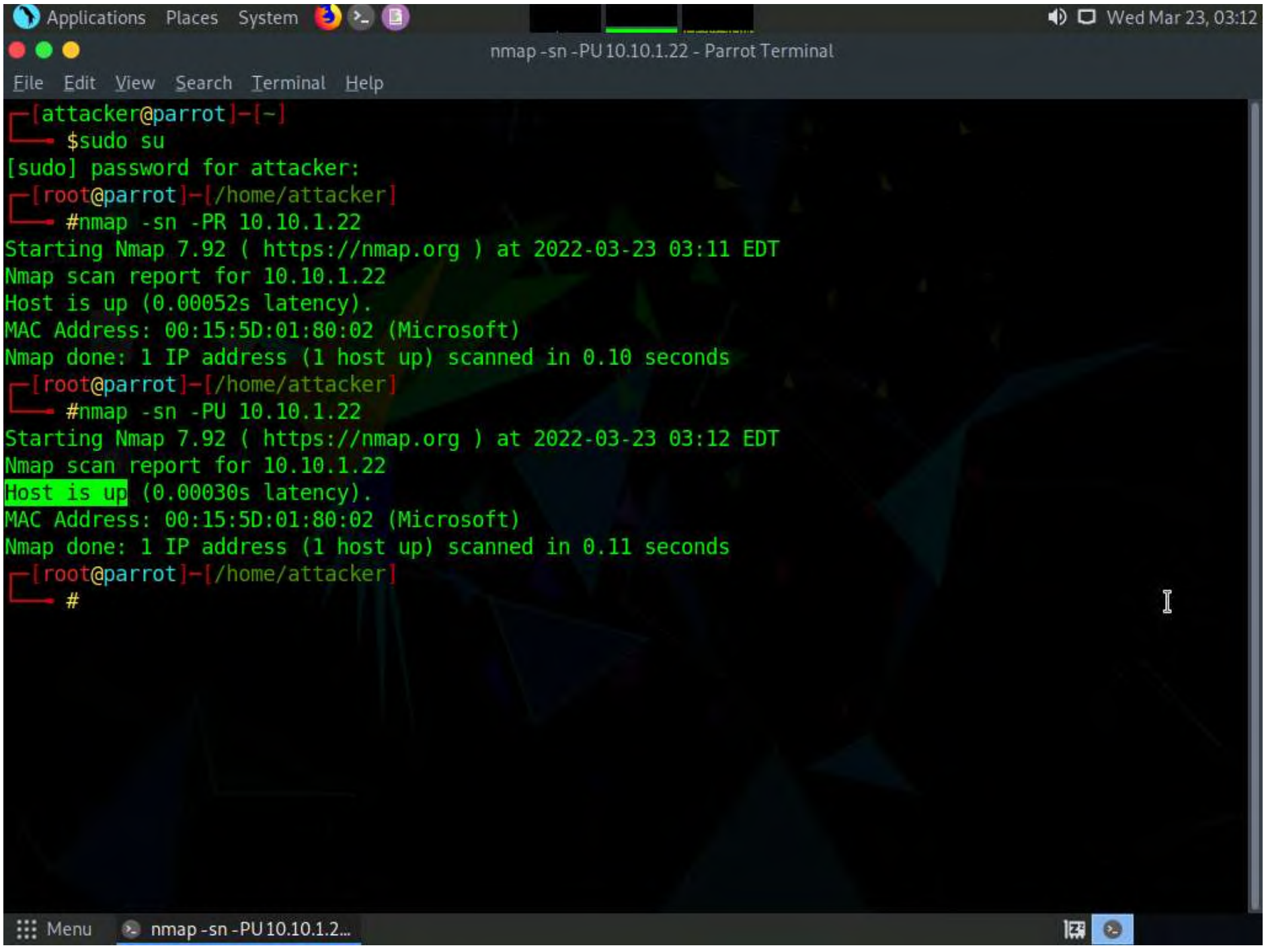
Note: The MAC address might differ when you perform this task.



8. In the terminal window, type **nmap -sn -PU [Target IP Address]**, (here, the target IP address is **10.10.1.22**) and press **Enter**. The scan results appear, indicating the target **Host is up**, as shown in the screenshot.

Note: **-PU**: performs the UDP ping scan.

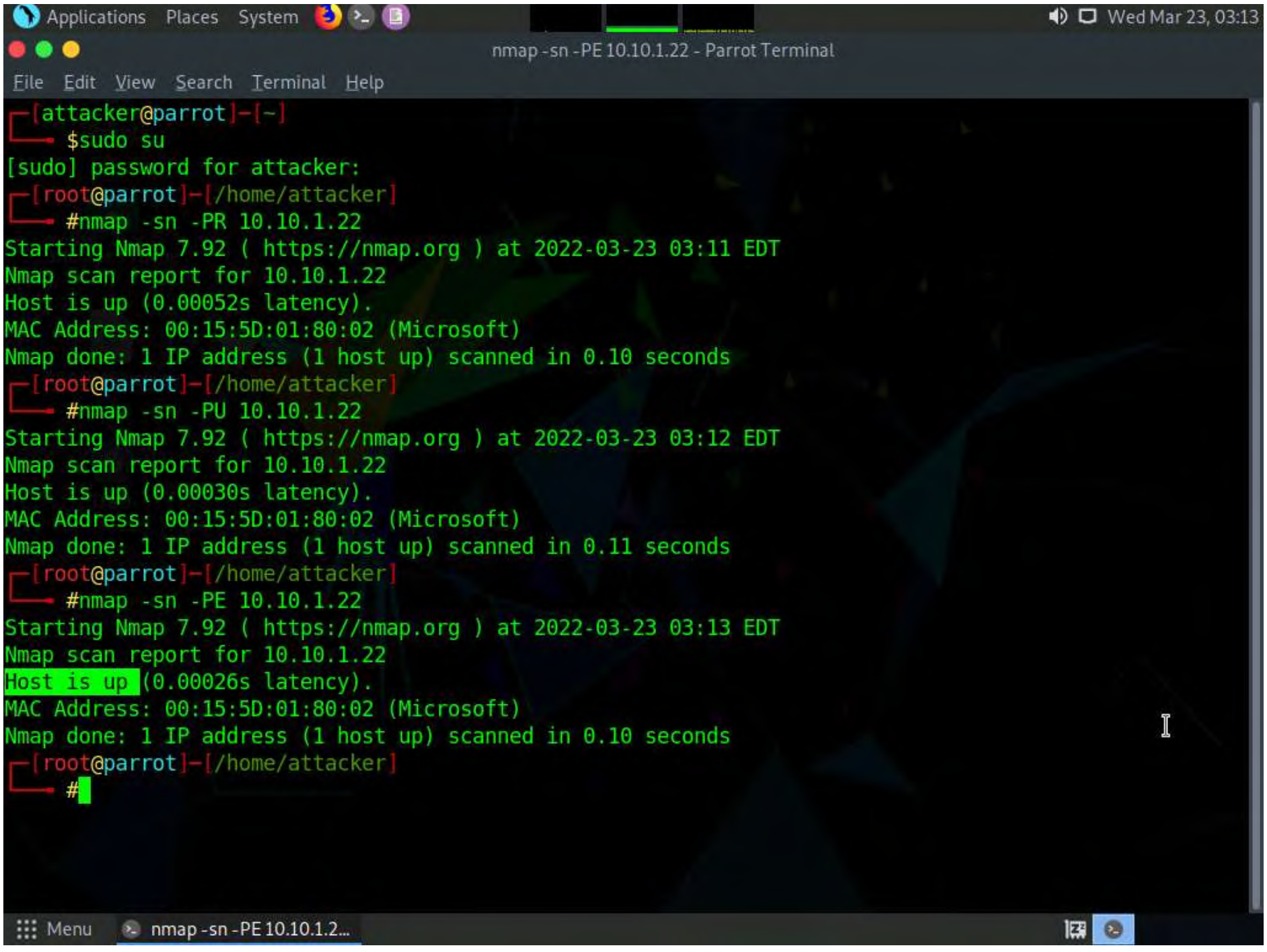
Note: The UDP ping scan sends UDP packets to the target host; a UDP response means that the host is active. If the target host is offline or unreachable, various error messages such as “host/network unreachable” or “TTL exceeded” could be returned.



9. Now, we will perform the ICMP ECHO ping scan. In the terminal window, type **nmap -sn -PE [Target IP Address]**, (here, the target IP address is **10.10.1.22**) and press **Enter**. The scan results appear, indicating that the target **Host is up**, as shown in the screenshot.

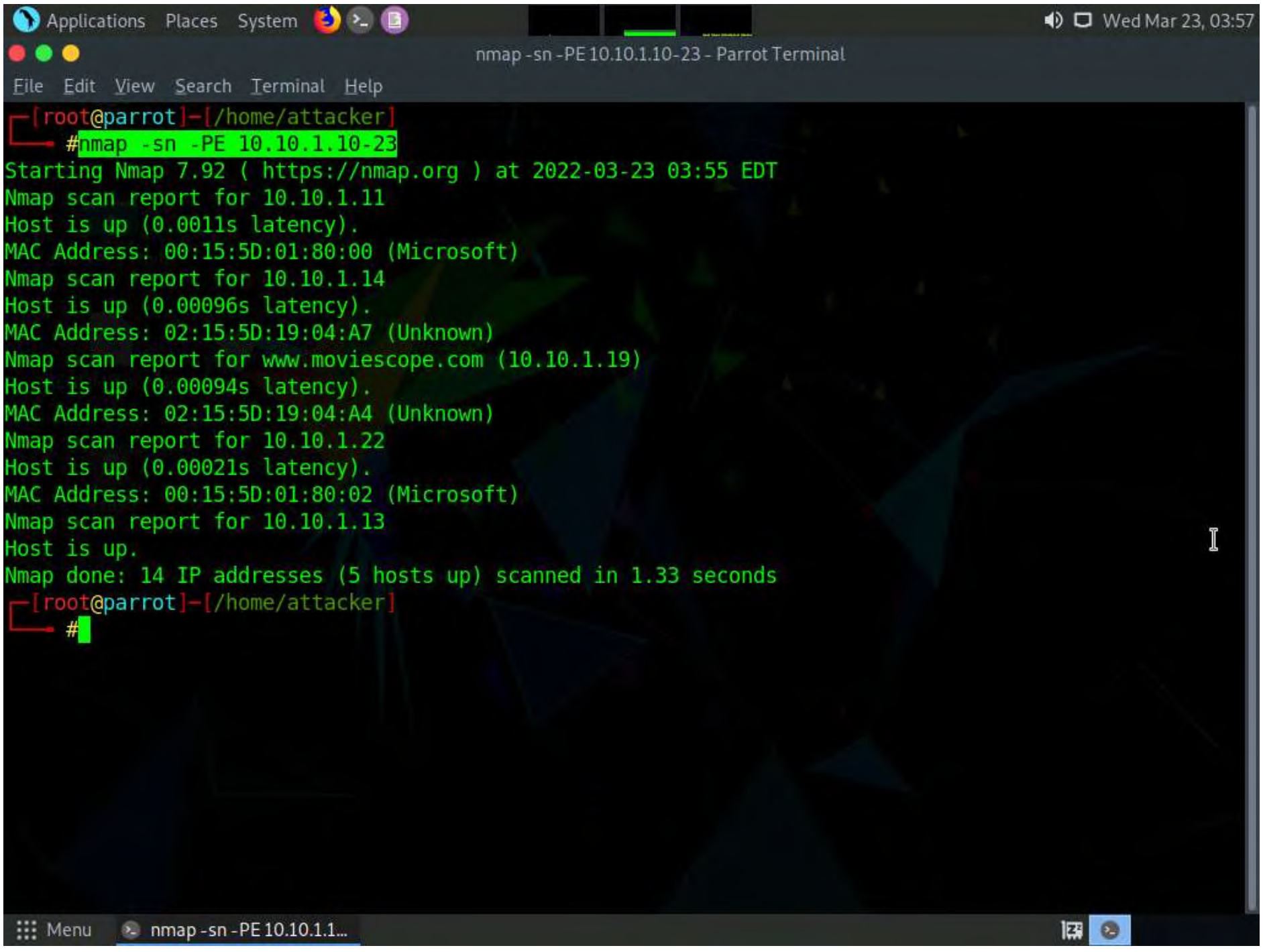
Note: **-PE**: performs the ICMP ECHO ping scan.

Note: The ICMP ECHO ping scan involves sending ICMP ECHO requests to a host. If the target host is alive, it will return an ICMP ECHO reply. This scan is useful for locating active devices or determining if the ICMP is passing through a firewall.



10. Now, we will perform an ICMP ECHO ping sweep to discover live hosts from a range of target IP addresses. In the terminal window, type **nmap -sn -PE [Target Range of IP Addresses]** (here, the target range of IP addresses is **10.10.1.10-23**) and press **Enter**. The scan results appear, indicating the target **Host is up**, as shown in the screenshot.

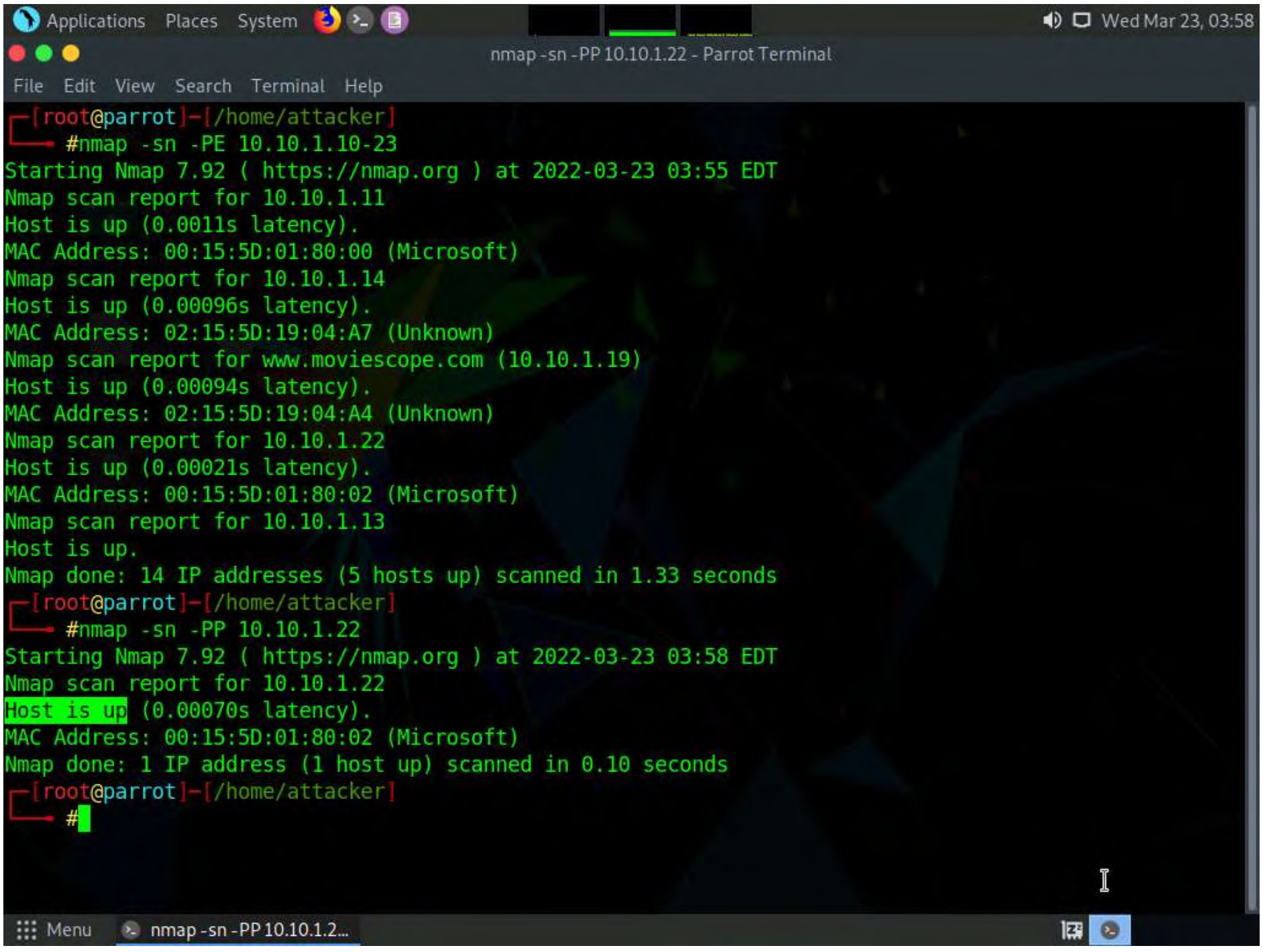
Note: In this lab task, we are scanning **Windows 11**, **Windows Server 2022**, **Windows Server 2019**, and **Android** machines. Note: The ICMP ECHO ping sweep is used to determine the live hosts from a range of IP addresses by sending ICMP ECHO requests to multiple hosts. If a host is alive, it will return an ICMP ECHO reply.



11. In the terminal window, type **nmap -sn -PP [Target IP Address]**, (here, the target IP address is **10.10.1.22**) and press **Enter**. The scan results appear, indicating the target **Host is up**, as shown in the screenshot.

Note: **-PP**: performs the ICMP timestamp ping scan.

Note: ICMP timestamp ping is an optional and additional type of ICMP ping whereby the attackers query a timestamp message to acquire the information related to the current time from the target host machine.



12. Apart from the aforementioned network scanning techniques, you can also use the following scanning techniques to perform a host discovery on a target network.

    - **ICMP Address Mask Ping Scan**: This technique is an alternative for the traditional ICMP ECHO ping scan, which are used to determine whether the target host is live specifically when administrators block the ICMP ECHO pings.

      # nmap -sn -PM [target IP address]

    - **TCP SYN Ping Scan**: This technique sends empty TCP SYN packets to the target host, ACK response means that the host is active.

      # nmap -sn -PS [target IP address]

    - **TCP ACK Ping Scan**: This technique sends empty TCP ACK packets to the target host; an RST response means that the host is active.

      # nmap -sn -PA [target IP address]

    - **IP Protocol Ping Scan**: This technique sends different probe packets of different IP protocols to the target host, any response from any probe indicates that a host is active.

      # nmap -sn -PO [target IP address]

13. This concludes the demonstration of discovering the target host(s) in the target network using various host discovery techniques.
14. Close all open windows and document all the acquired information.

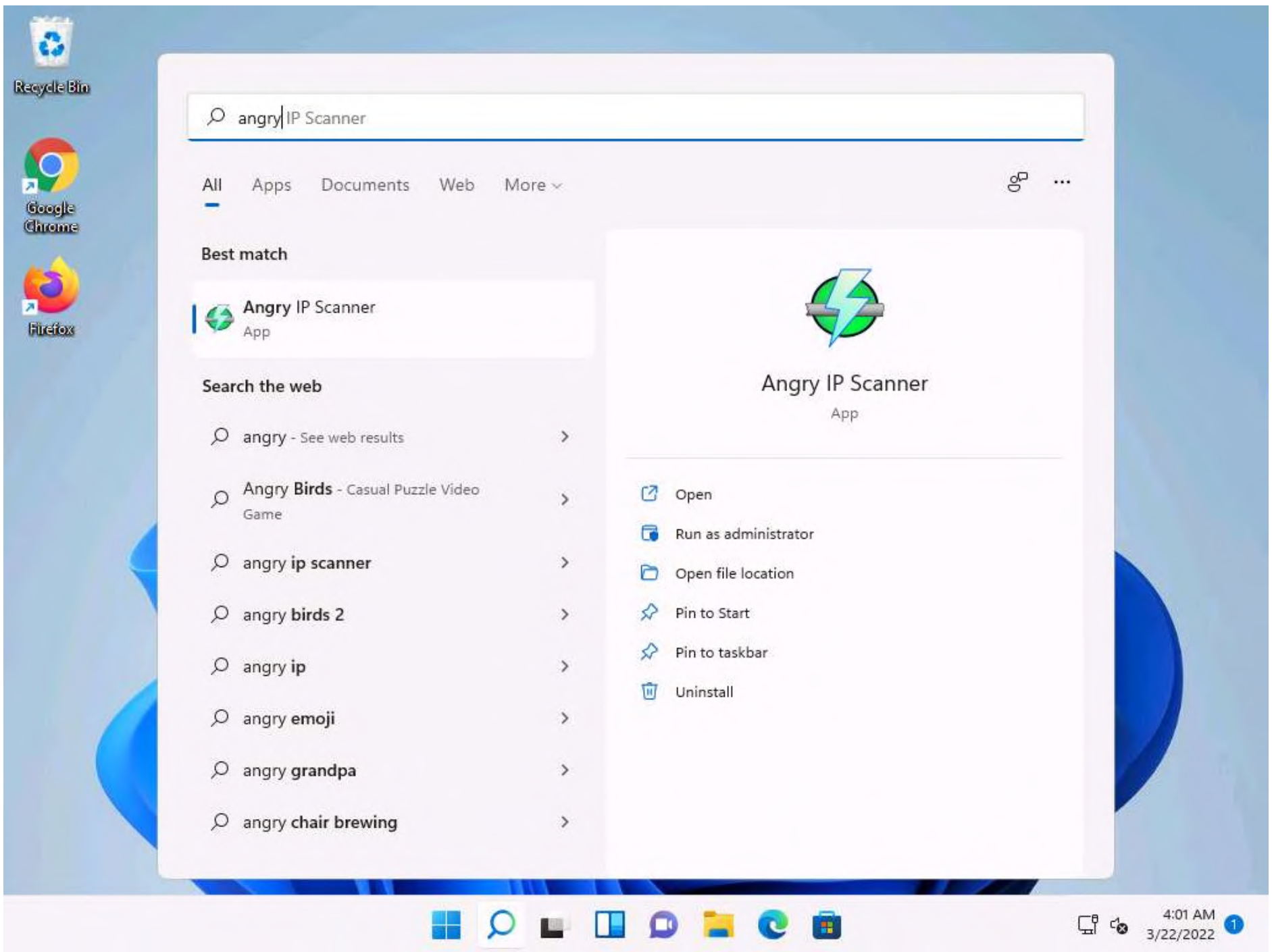
Task 2: Perform Host Discovery using Angry IP Scanner

Angry IP Scanner is an open-source and cross-platform network scanner designed to scan IP addresses as well as ports. It simply pings each IP address to check if it is alive; then, optionally by resolving its hostname, determines the MAC address, scans ports, etc. The amount of gathered data about each host can be extended with plugins.

Here, we will use the Angry IP Scanner tool to discover the active hosts in the target network.

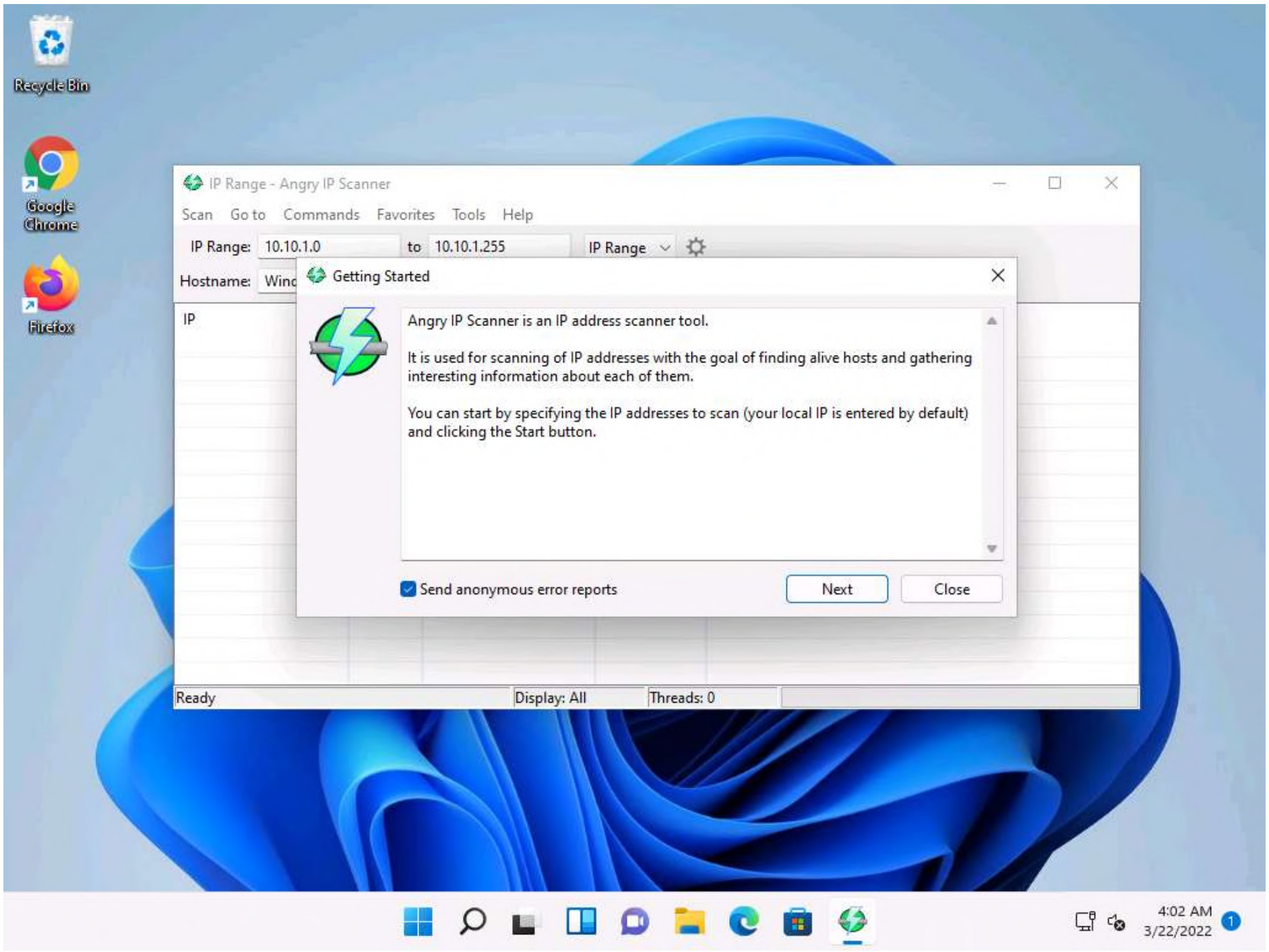
1. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine.

2. Click **Search** icon ( ) on the **Desktop**. Type **angry** in the search field, the **Angry IP Scanner** appears in the result, click **Open** to launch it.

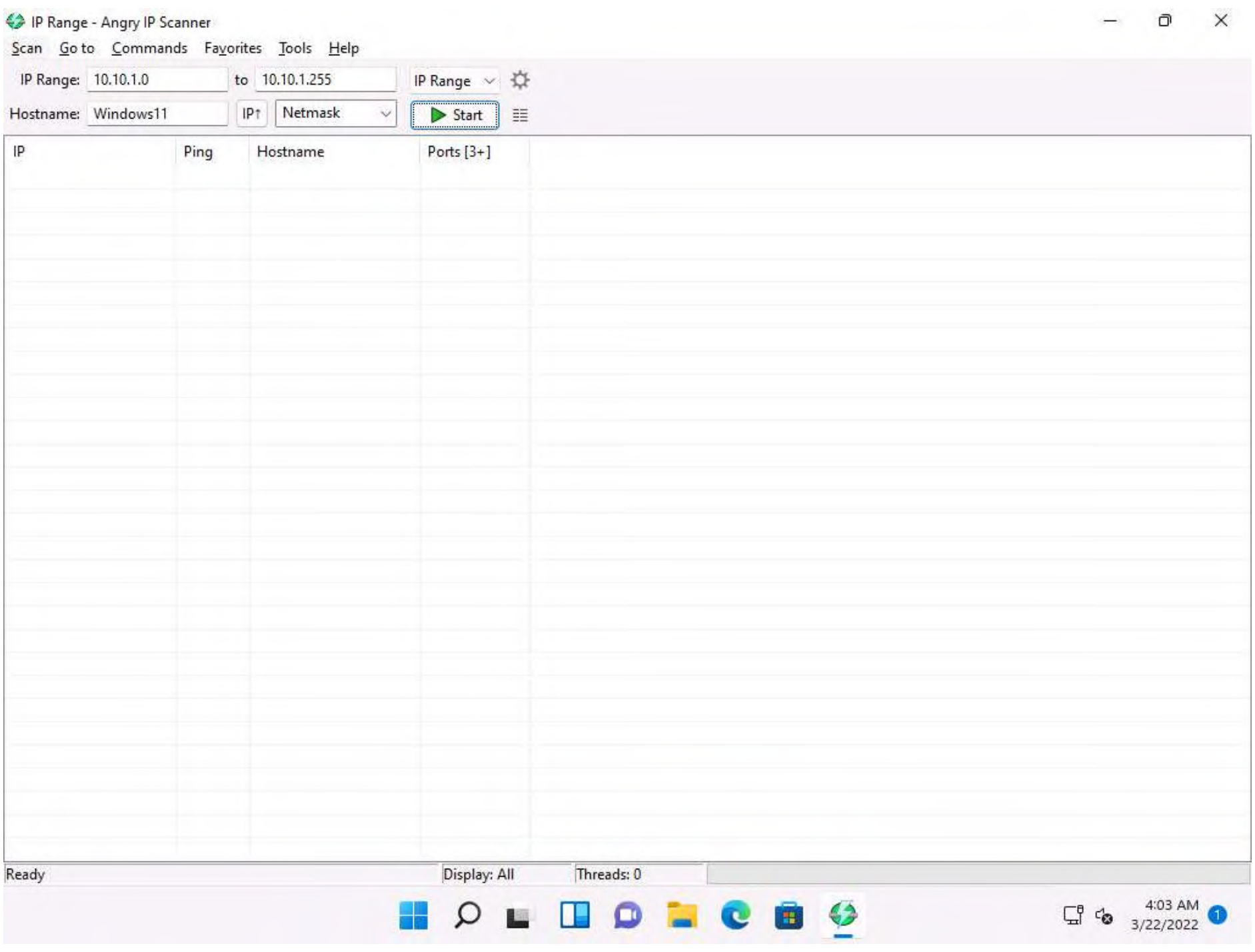


3. **Angry IP Scanner** starts, and a **Getting Started** window pops up. Click **Next**, follow the wizard, and click **Close**.

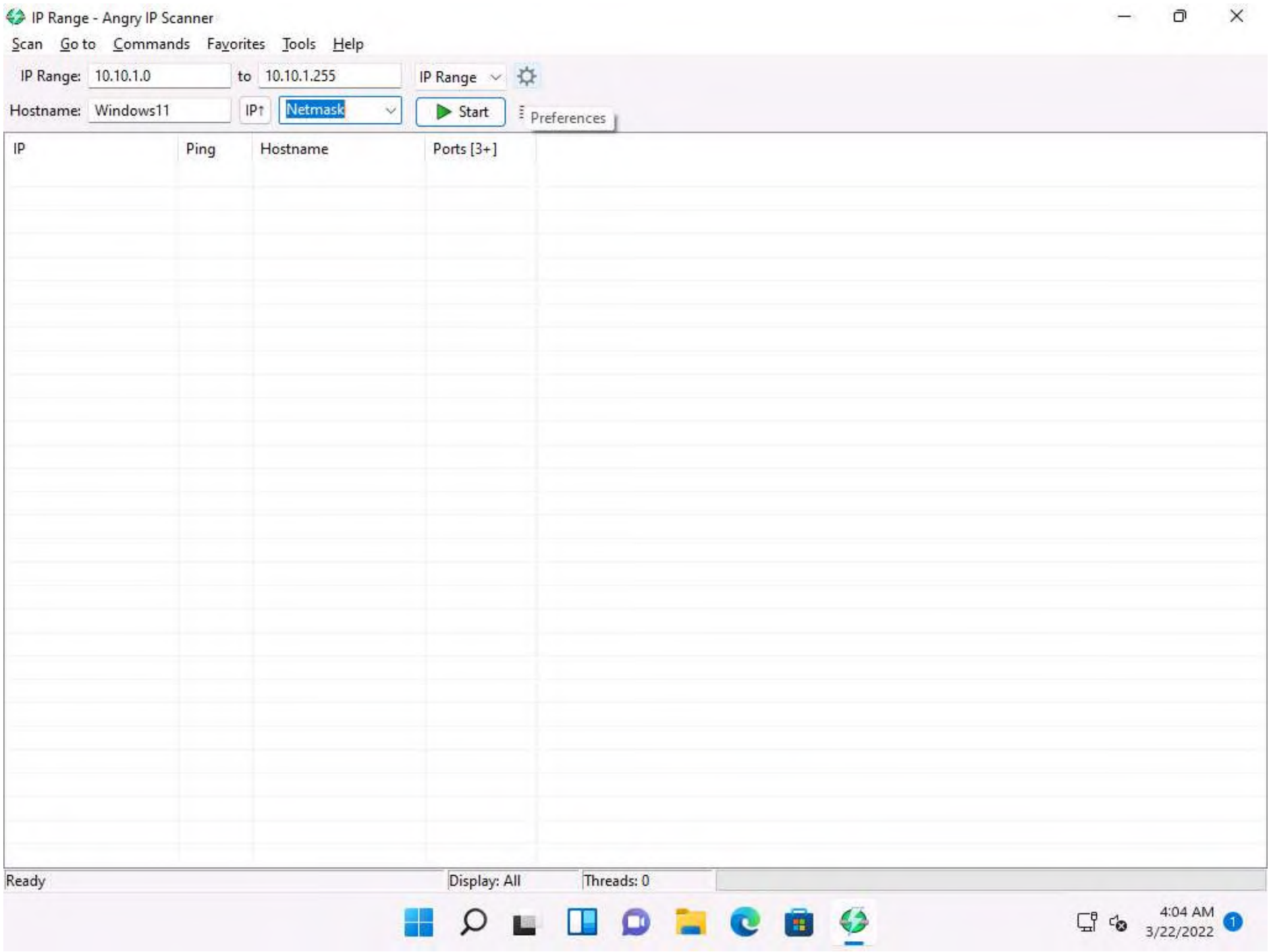
Note: If **Open File - Security Warning** window appears, click **Run**.



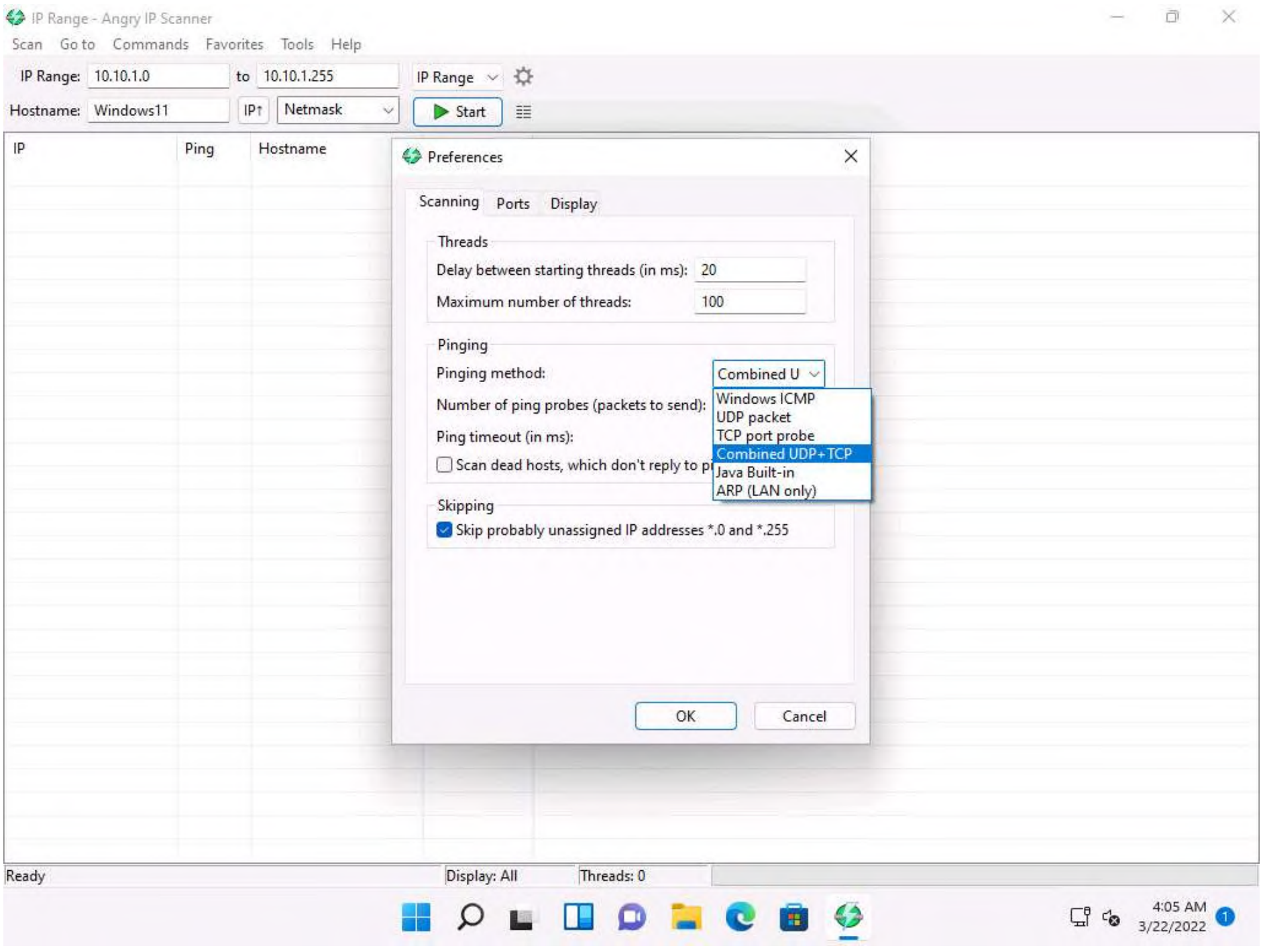
4. The **IP Range - Angry IP Scanner** window appears, as shown in the screenshot.



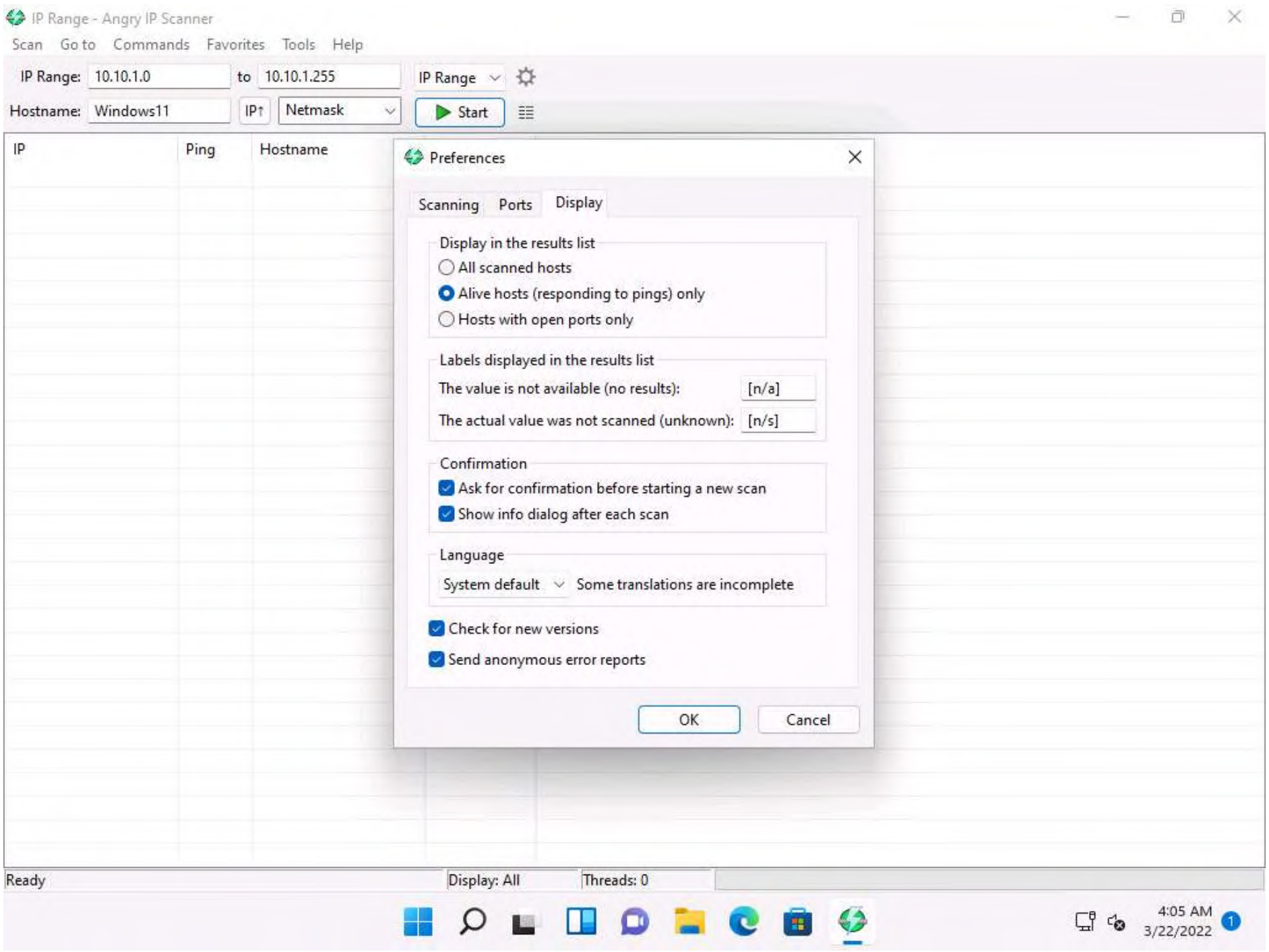
5. In the **IP Range** fields, type the IP range as **10.10.1.0** to **10.10.1.255** and click the **Preferences** icon beside the **IP Range** menu, as shown in the screenshot.



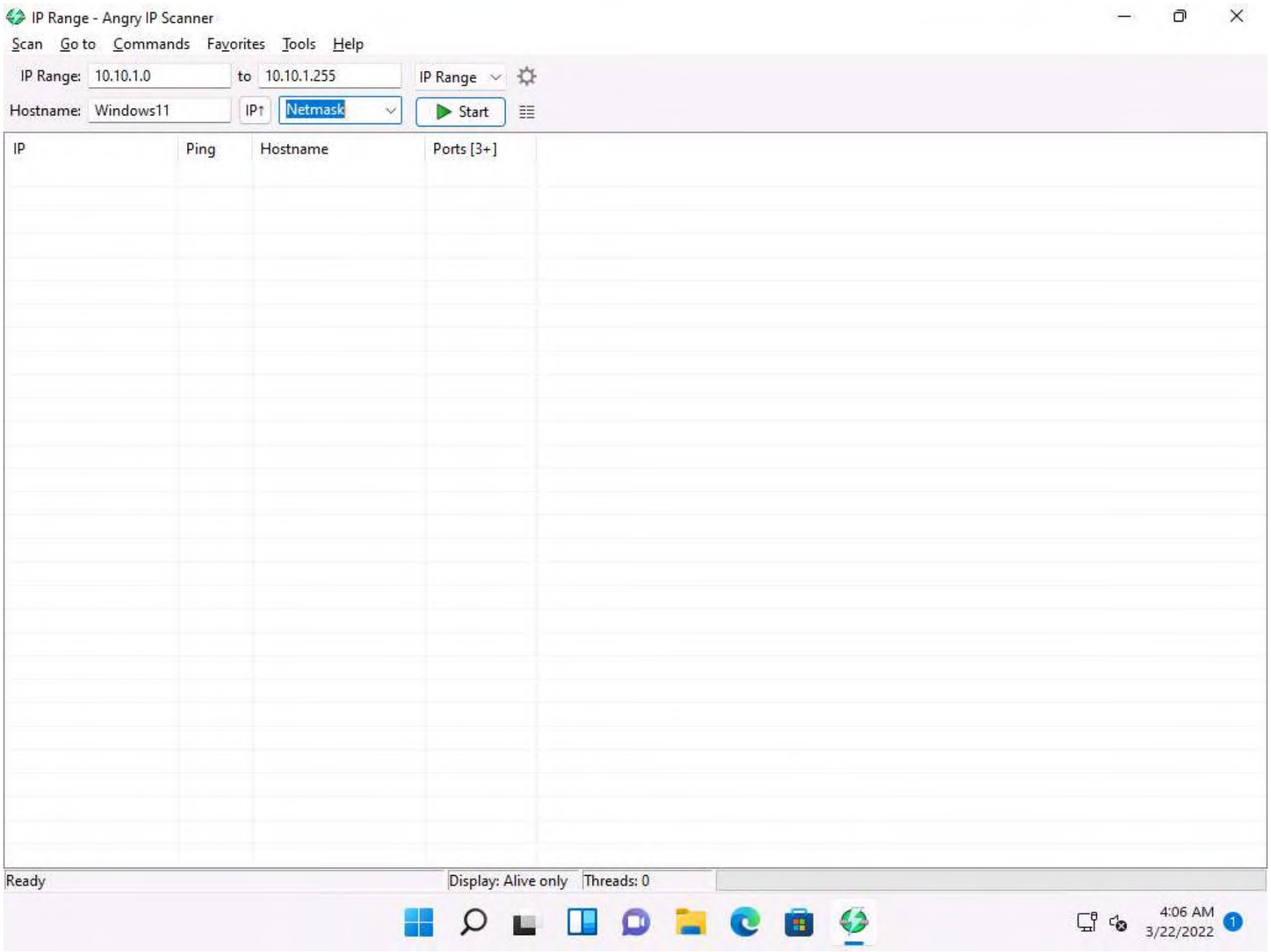
6. The **Preferences** window appears. In the **Scanning** tab, under the **Pinging** section, select the **Pinging method** as **Combined UDP+TCP** from the drop-down list.



7. Now, switch to the **Display** tab. Under the **Display in the results list** section, select the **Alive hosts (responding to pings) only** radio button and click **OK**.

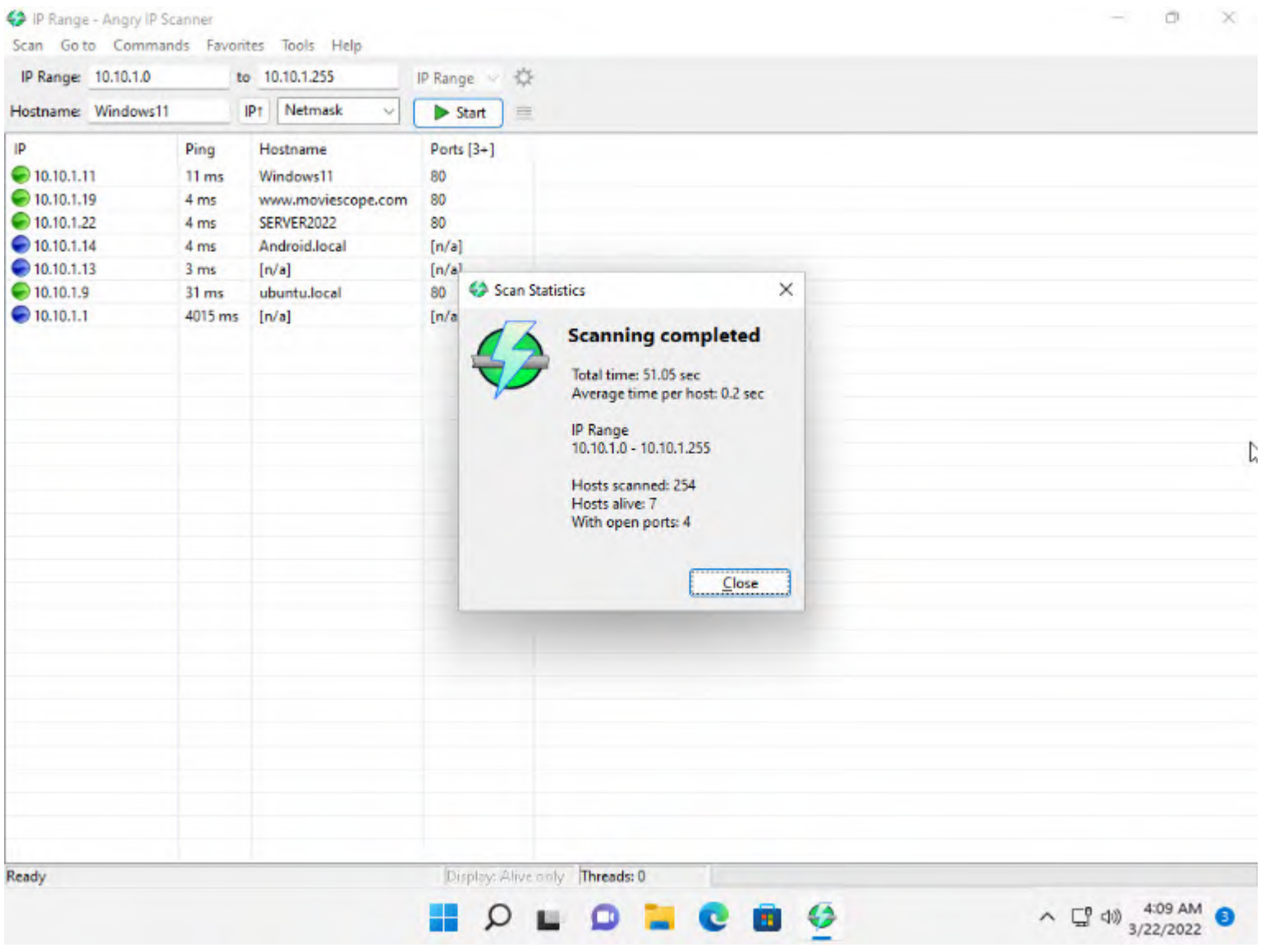


8. In the **IP Range - Angry IP Scanner** window, click the **Start** button to start scanning the IP range that you entered.

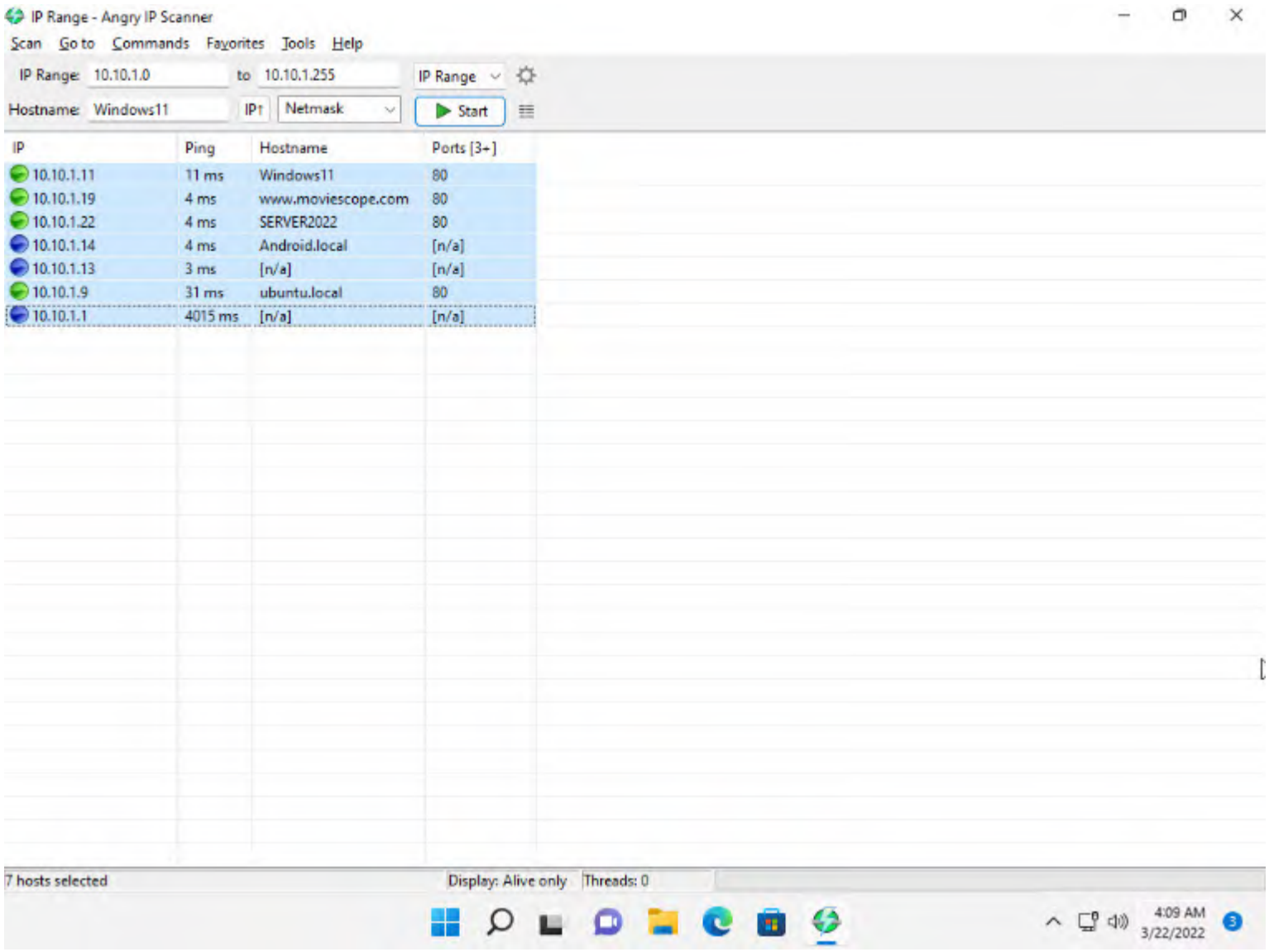


9. **Angry IP Scanner** starts scanning the IP range and begins to list out the alive hosts found along with their hostnames. Check the progress bar on the bottom-right corner to see the progress of the scanning.

10. After the scanning is completed, a **Scan Statistics** pop-up appears. Note the total number of **Hosts alive** (here, 7) and click **Close**.



11. The results of the scan appear in the **IP Range - Angry IP Scanner** window. You can see all active IP addresses with their hostnames listed in the main window.



12. This concludes the demonstration of discovering alive hosts in the target range of IP addresses using Angry IP Scanner.
13. You can also use other ping sweep tools such as **SolarWinds Engineer's Toolset** (https://www.solarwinds.com), **NetScanTools Pro** (https://www.netscantools.com), **Colasoft Ping Tool** (https://www.colasoft.com), **Visual Ping Tester** (http://www.pingtester.net), and **OpUtils** (https://www.manageengine.com) to discover active hosts in the target network.
14. Close all open windows and document all the acquired information.

Lab 2: Perform Port and Service Discovery

Lab Scenario

As a professional ethical hacker or a pen tester, the next step after discovering active hosts in the target network is to scan for open ports and services running on the target IP addresses in the target network. This discovery of open ports and services can be performed via various port scanning tools and techniques.

Lab Objectives

- Perform port and service discovery using MegaPing
- Perform port and service discovery using NetScanTools Pro
- Perform port scanning using sx tool
- Explore various network scanning techniques using Nmap
- Explore various network scanning techniques using Hping3

Overview of Port and Service Discovery

Port scanning techniques are categorized according to the type of protocol used for communication within the network.

- TCP Scanning
  - Open TCP scanning methods (TCP connect/full open scan)
  - Stealth TCP scanning methods (Half-open Scan, Inverse TCP Flag Scan, ACK flag probe scan, third party and spoofed TCP scanning methods)
- UDP Scanning
- SCTP Scanning
  - SCTP INIT Scanning
  - SCTP COOKIE/ECHO Scanning
- SSDP and List Scanning
- IPv6 Scanning

Task 1: Perform Port and Service Discovery using MegaPing

MegaPing is a toolkit that provides essential utilities for Information System specialists, system administrators, IT solution providers, and individuals. It is used to detect live hosts and open ports of the system in the network, and can scan your entire network and provide information such as open shared resources, open ports, services/drivers active on the computer, key registry entries, users and groups, trusted domains, printers, etc. You can also perform various network troubleshooting activities with the help of integrated network utilities such as DNS lookup name, DNS list hosts, Finger, host monitor, IP scanner, NetBIOS scanner, ping, port scanner, share scanner, traceroute, and Whois.

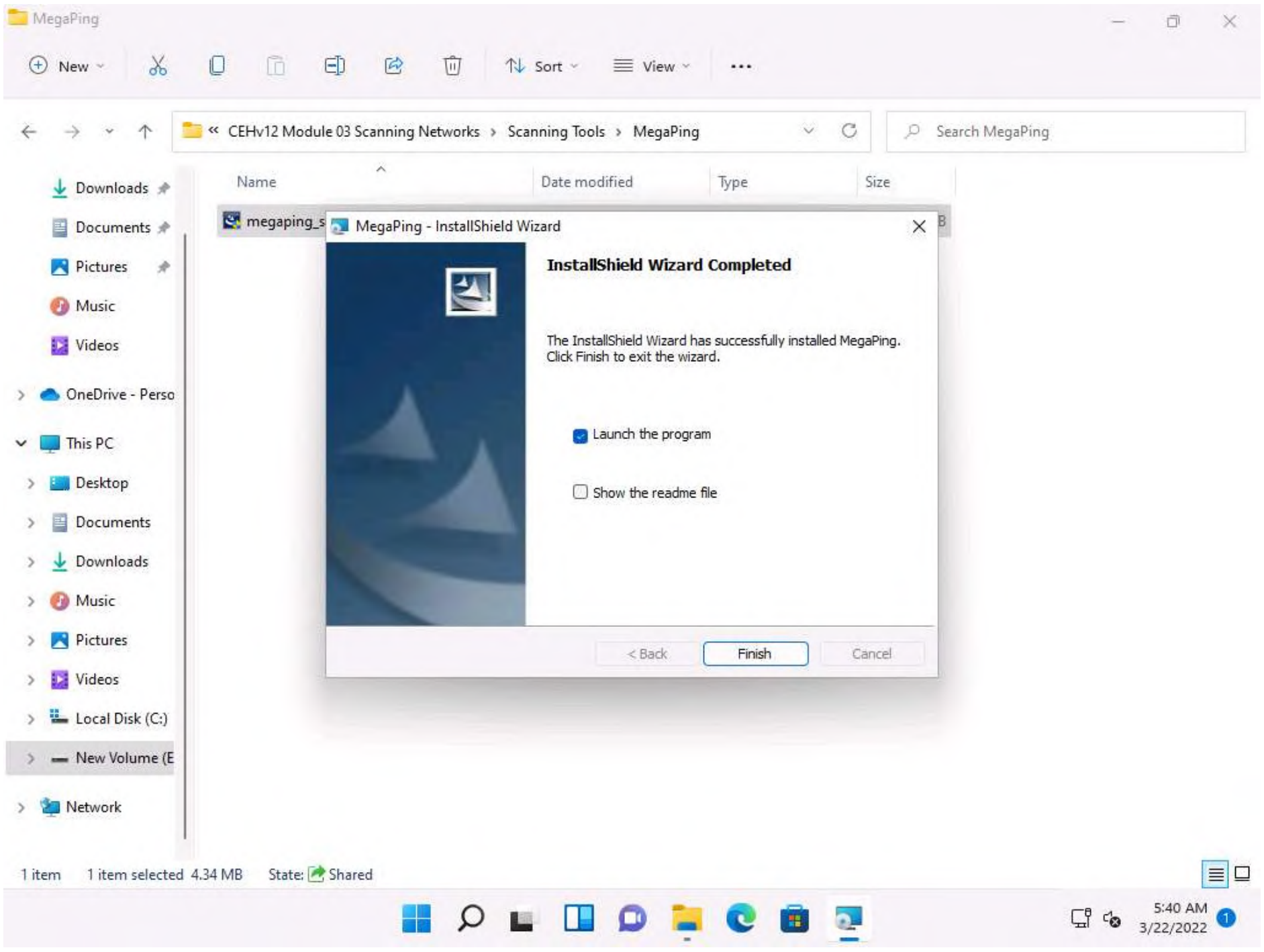
Here, we will use the MegaPing tool to scan for open ports and services running on the target range of IP addresses.

1. In the **Windows 11** machine, navigate to **E:\CEH-Tools\CEHv12 Module 03 Scanning Networks\Scanning Tools\MegaPing** and double-click **megaping_setup.exe**.

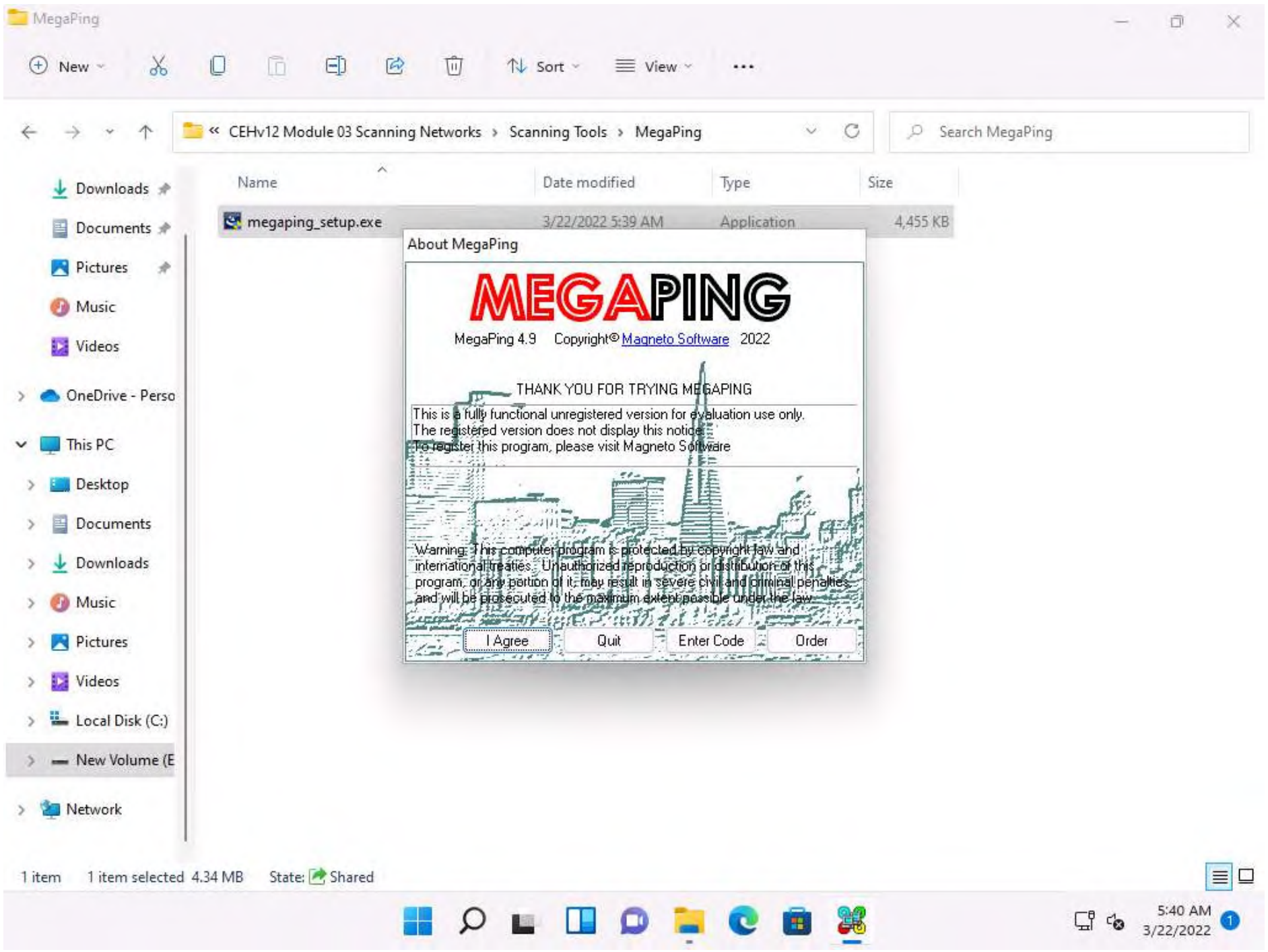
   Note: If a **User Account Control** pop-up appears, click **Yes**.

2. The **MegaPing - InstallShield Wizard** window appears; click **Next** and follow the wizard-driven installation steps to install **MegaPing**.

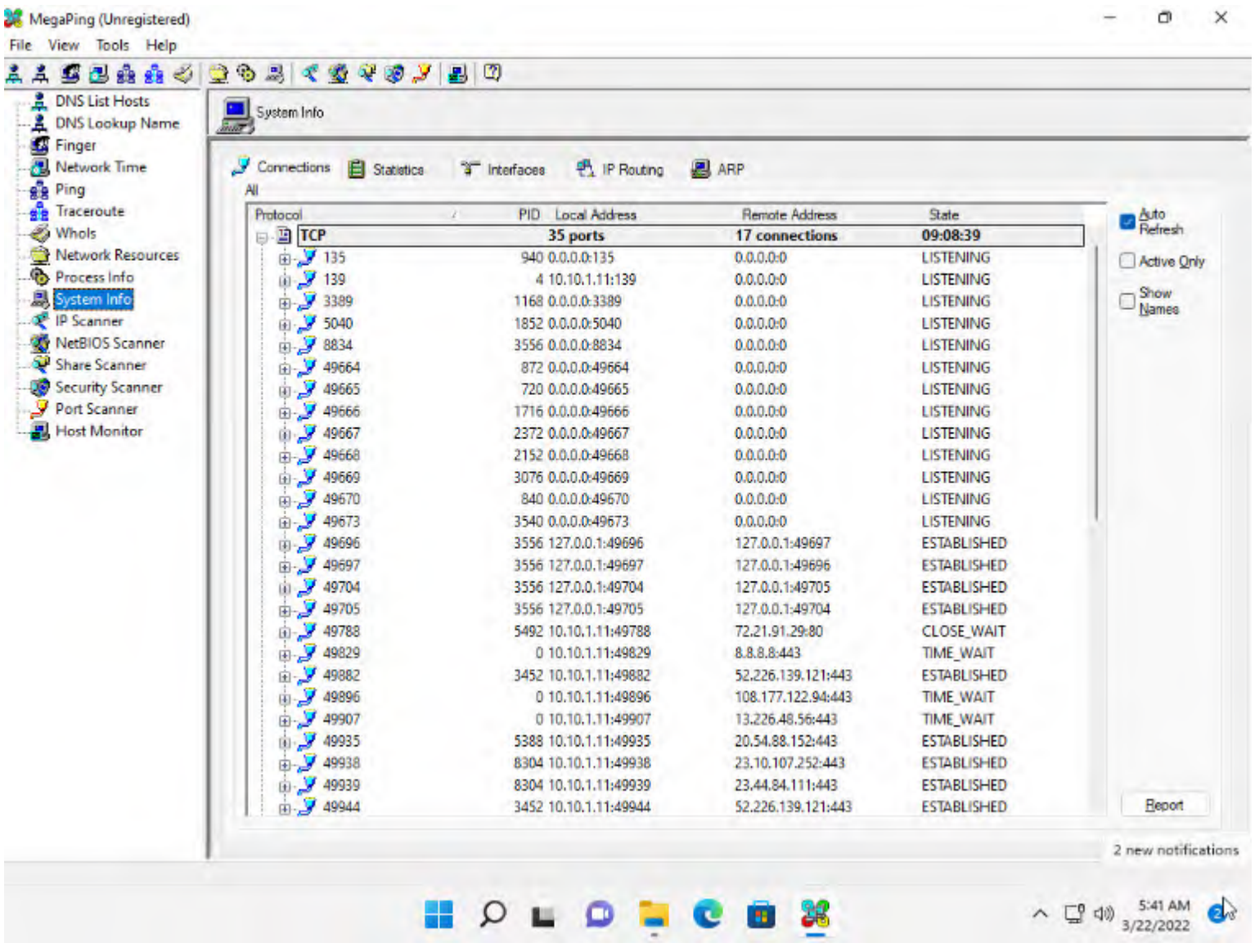
3. After the completion of the installation, click on the **Launch the program** checkbox and click **Finish**.



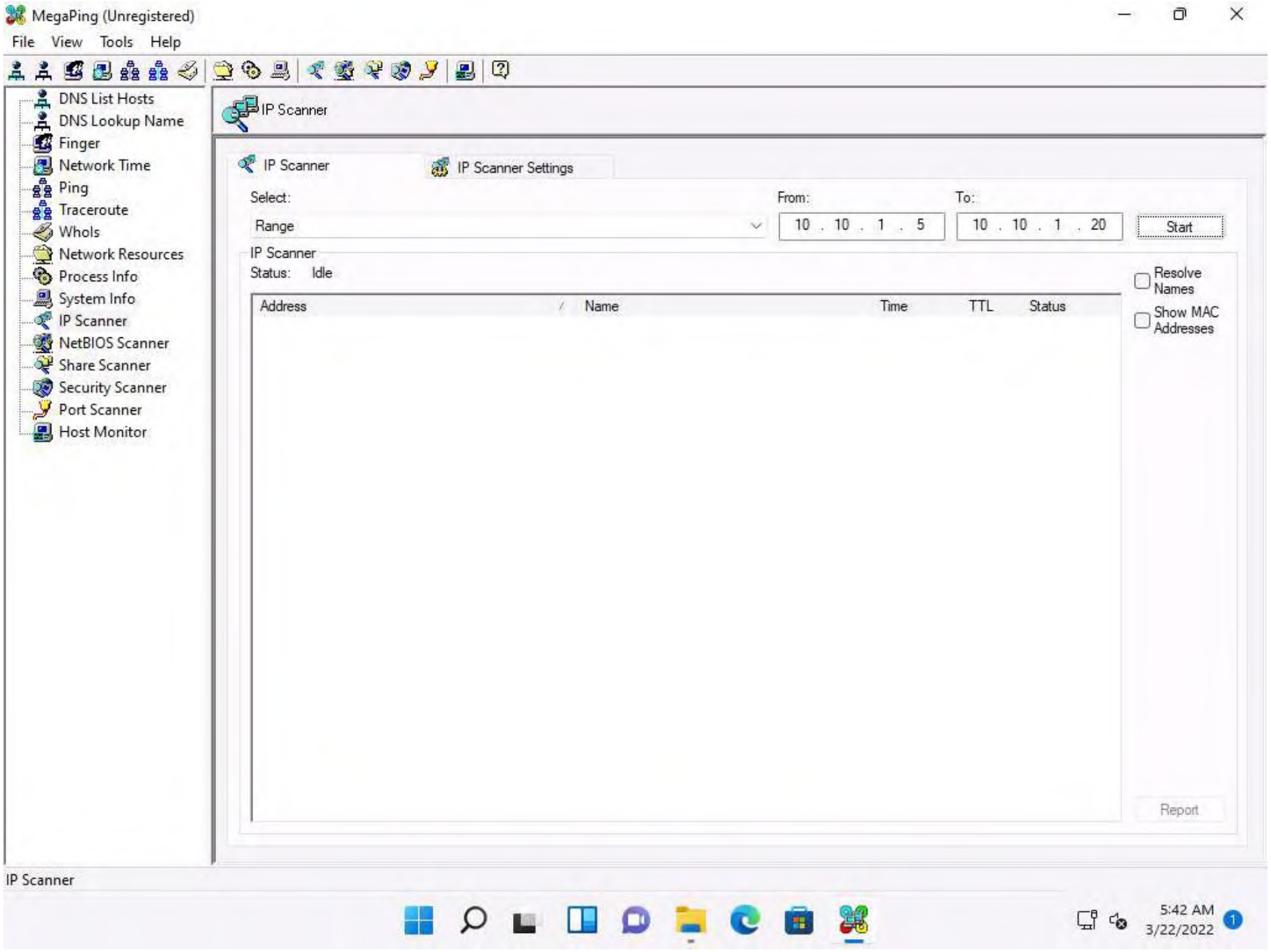
4. The **About MegaPing** window appears; click the **I Agree** button.



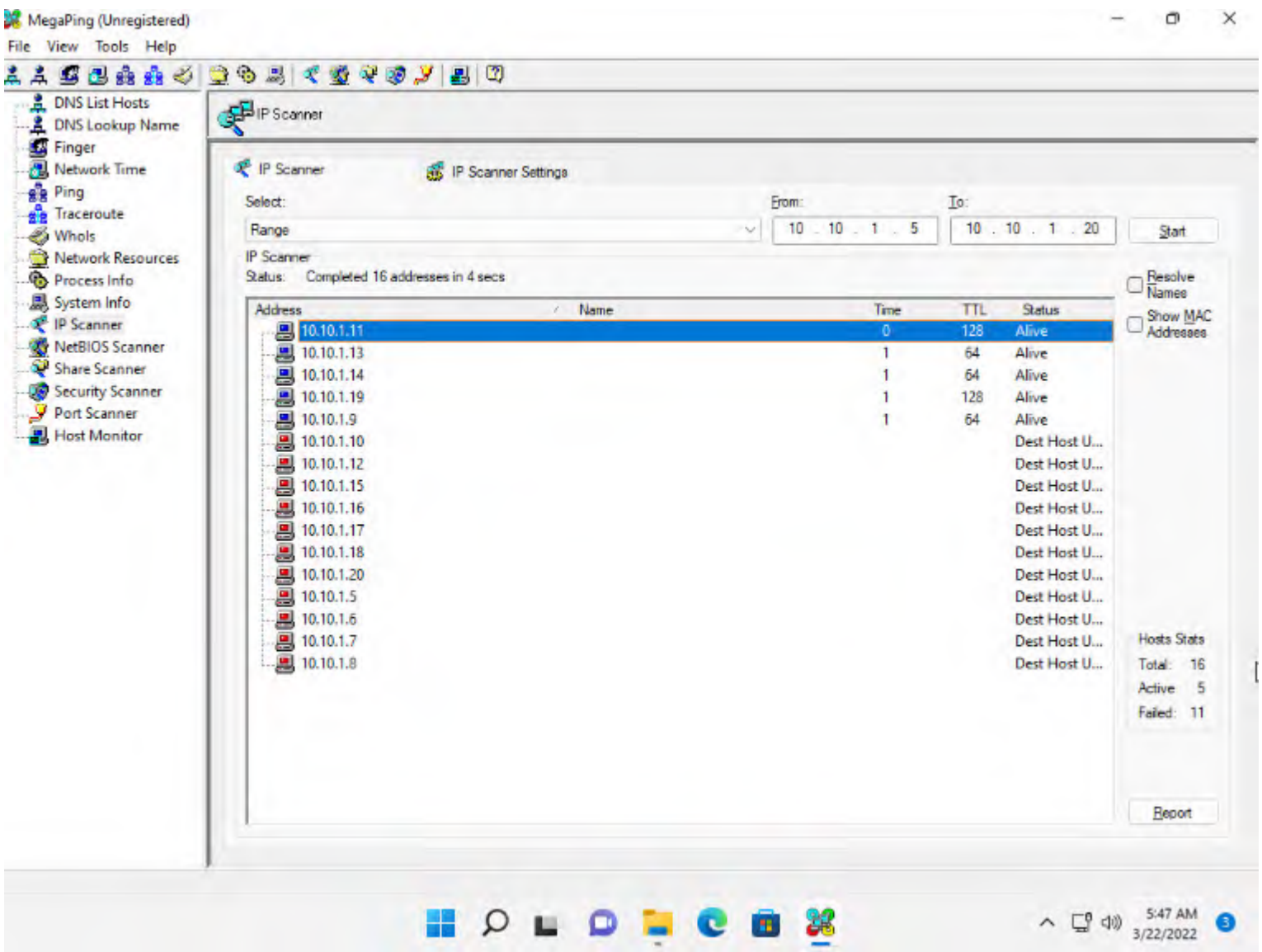
5. The **MegaPing (Unregistered)** GUI appears displaying the **System Info**, as shown in the screenshot.



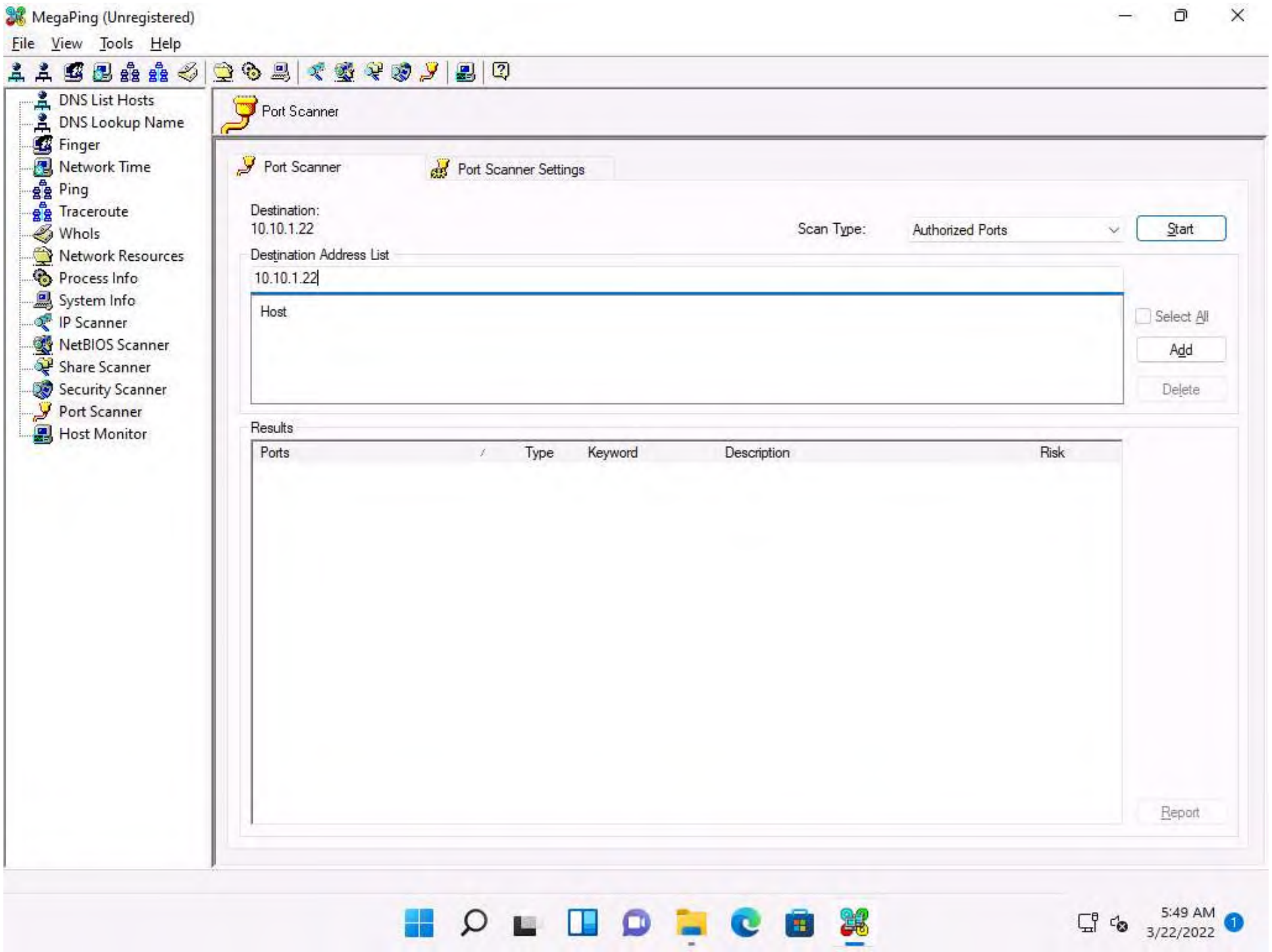
6. Select the **IP Scanner** option from the left pane. In the **IP Scanner** tab in the right-hand pane, enter the IP range in the **From** and **To** fields; in this lab, the IP range is **10.10.1.5** to **10.10.1.20**; then, click **Start**.



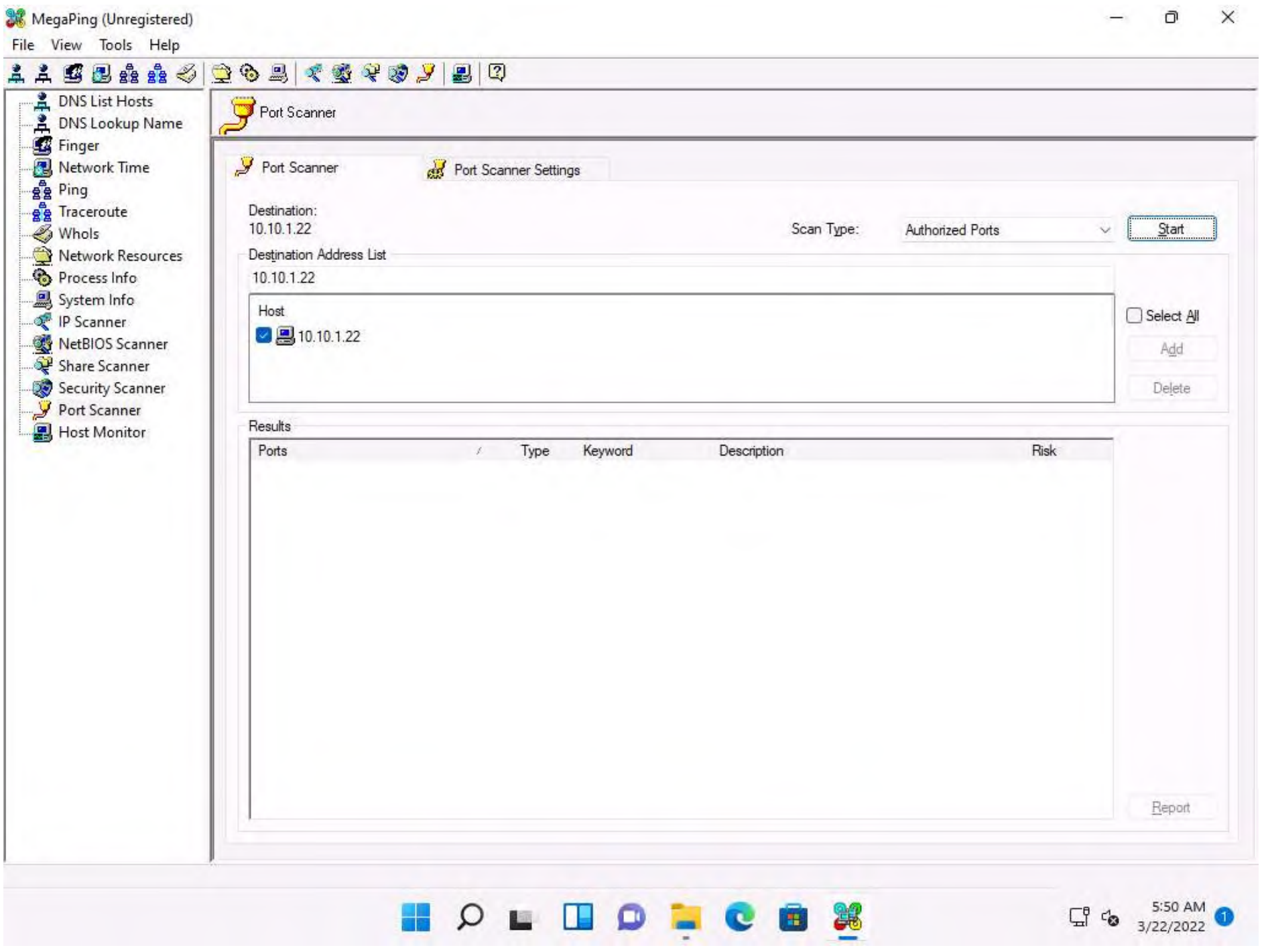
7. MegaPing lists all IP addresses under the specified target range with their TTL value, Status (dead or alive), and statistics of the dead and alive hosts, as shown in the screenshot.



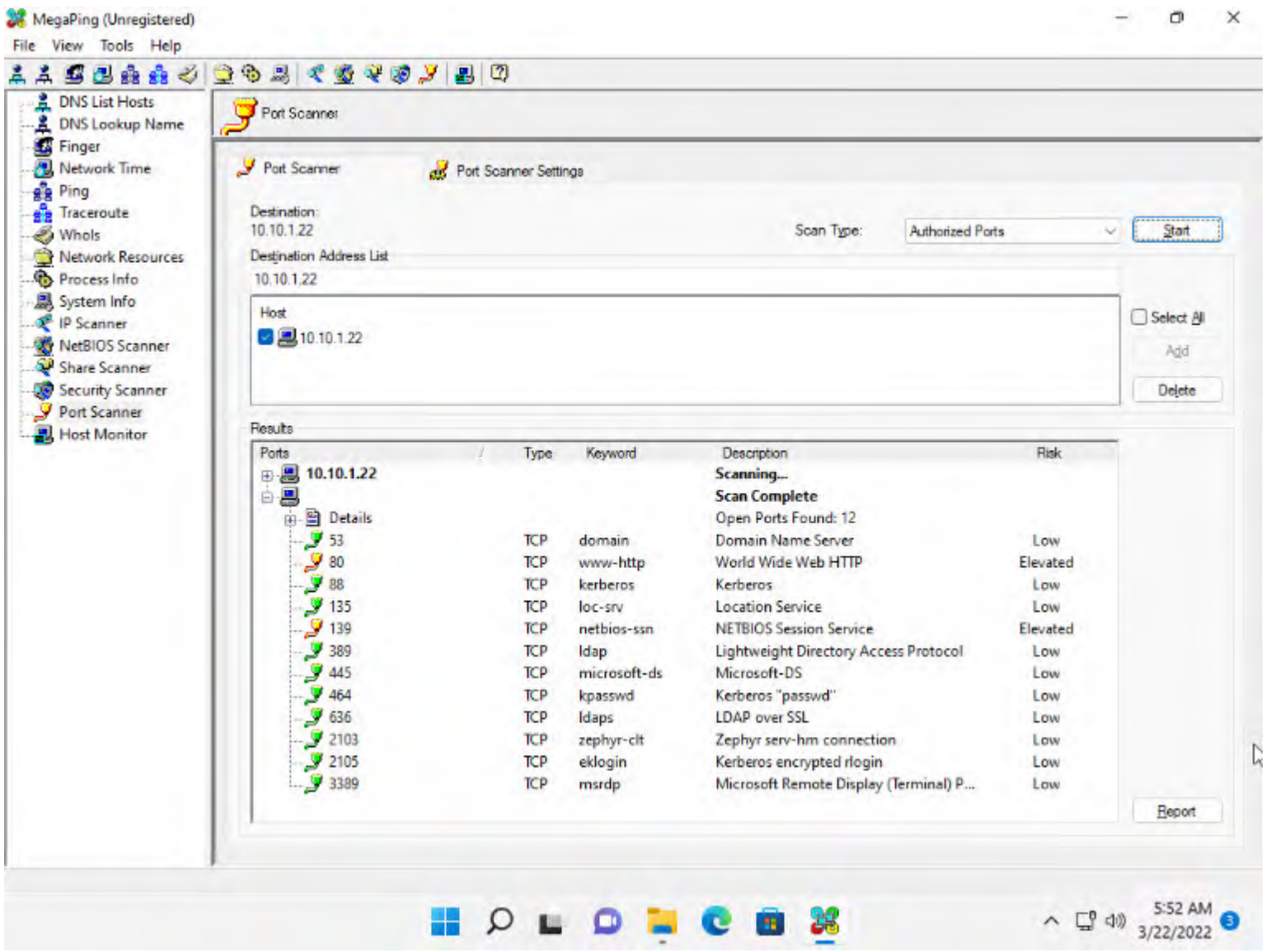
8. Select the **Port Scanner** option from the left-hand pane. In the **Port Scanner** tab in the right-hand pane, enter the IP address of the **Windows Server 2022** (**10.10.1.22**) machine into the **Destination Address List** field and click **Add**.



9. Select the **10.10.1.22** checkbox and click the **Start** button to start listening to the traffic on 10.10.1.22.



10. MegaPing lists the ports associated with **Windows Server 2022** (**10.10.1.22**), with detailed information on port number and type, service running on the port along with the description, and associated risk, as shown in the screenshot. Using this information attackers can penetrate the target network and compromise it, to launch attacks.



11. Similarly, you can perform port and service scanning on other target machines.
12. This concludes the demonstration of discovering open ports and services running on the target IP address using MegaPing.
13. Close all open windows and document all the acquired information.

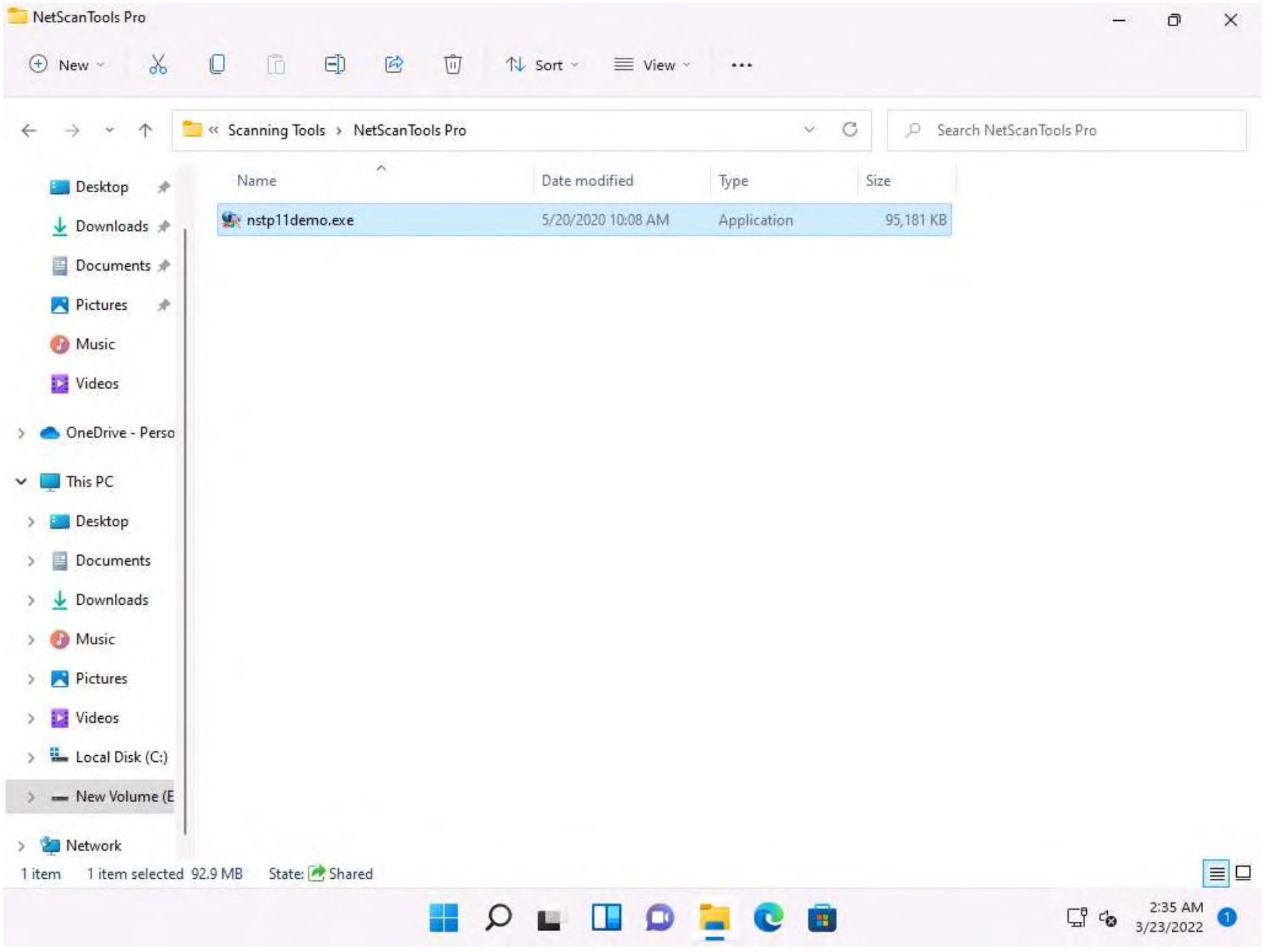
Task 2: Perform Port and Service Discovery using NetScanTools Pro

NetScanTools Pro is an integrated collection of utilities that gathers information on the Internet and troubleshoots networks for Network Professionals. With the available tools, you can research IPv4/IPv6 addresses, hostnames, domain names, e-mail addresses, and URLs on the target network.

Here, we will use the NetScanTools Pro tool to discover open ports and services running on the target range of IP addresses.

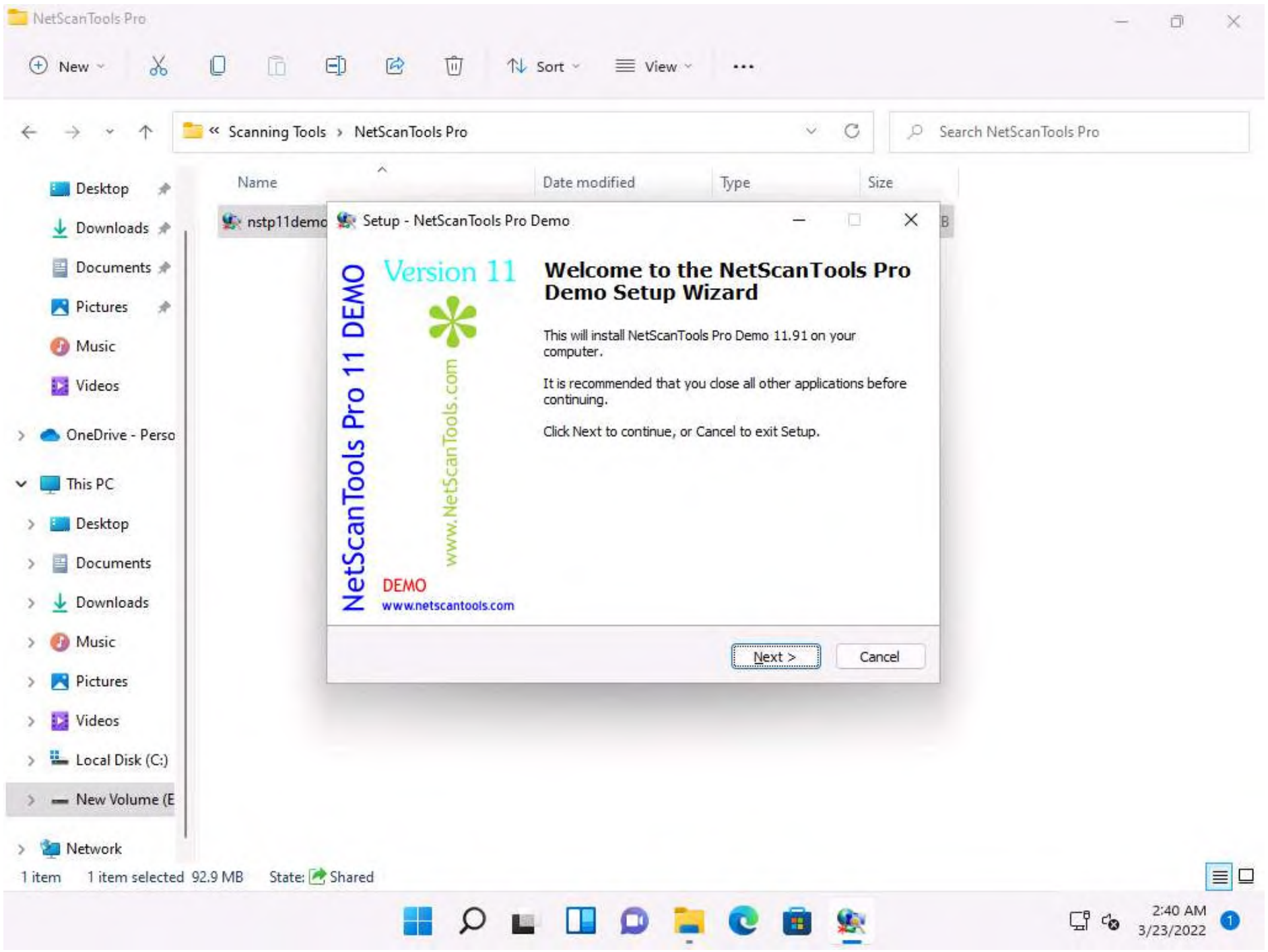
1. In the **Windows 11** machine, navigate to **E:\CEH-Tools\CEHv12 Module 03 Scanning Networks\Scanning Tools\NetScanTools Pro** and double-click **nstp11demo.exe**.

Note: If a **User Account Control** pop-up appears, click **Yes**.

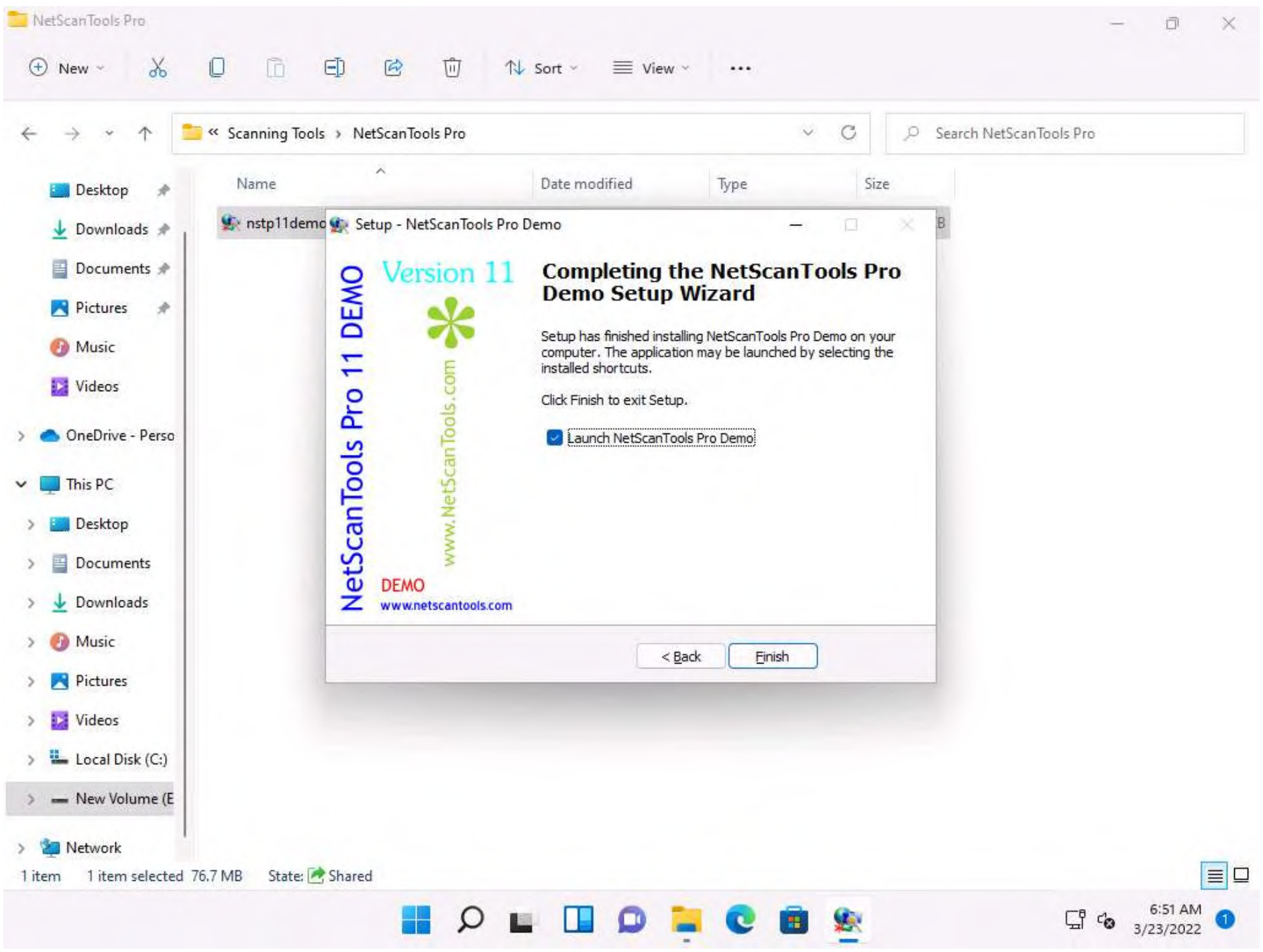


2. The **Setup - NetScanTools Pro Demo** window appears click **Next** and follow the wizard-driven installation steps to install **NetScanTools Pro**.

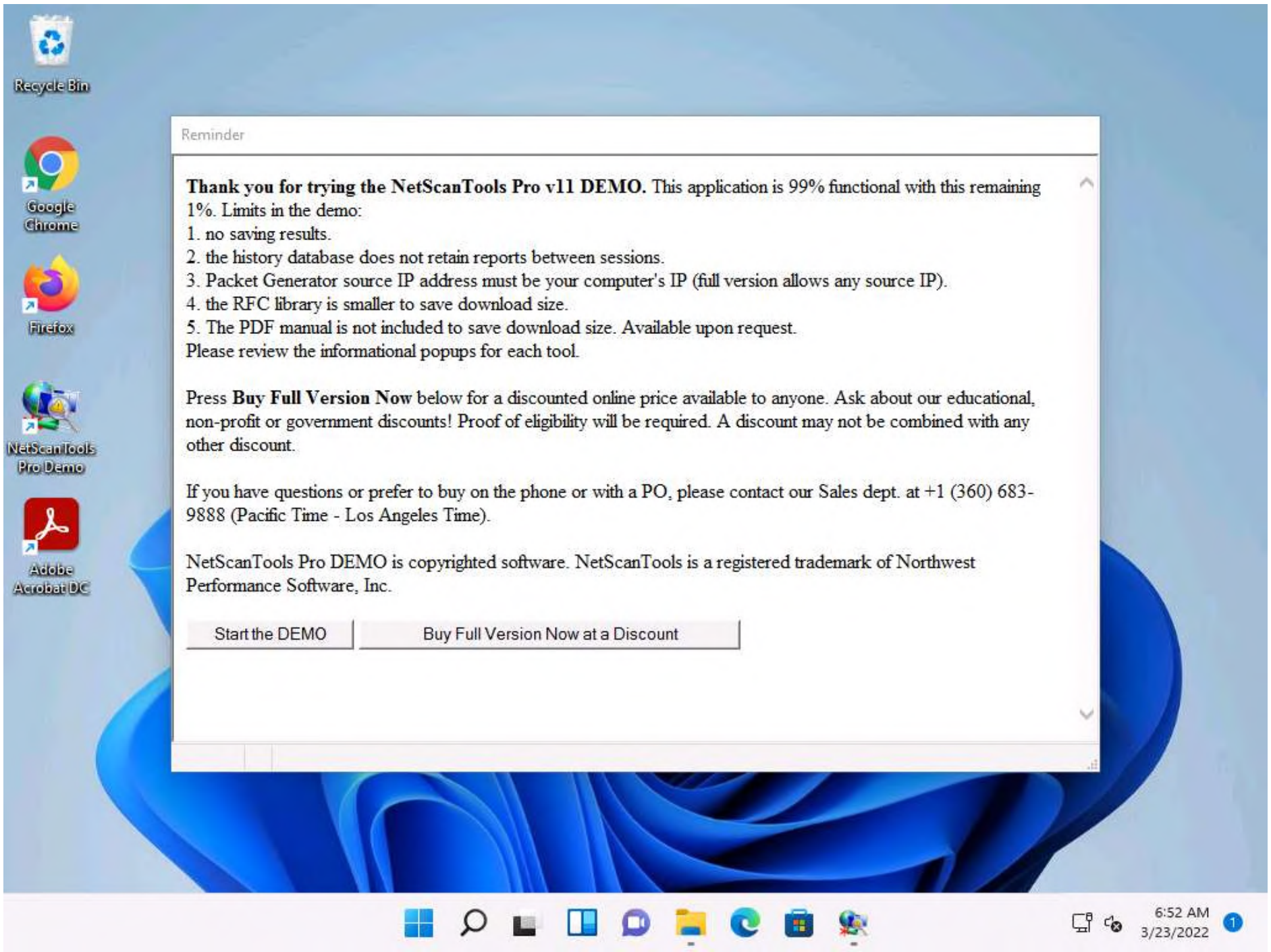
Note: If a **WinPcap 4.1.3 Setup** pop-up appears, click **Cancel**.



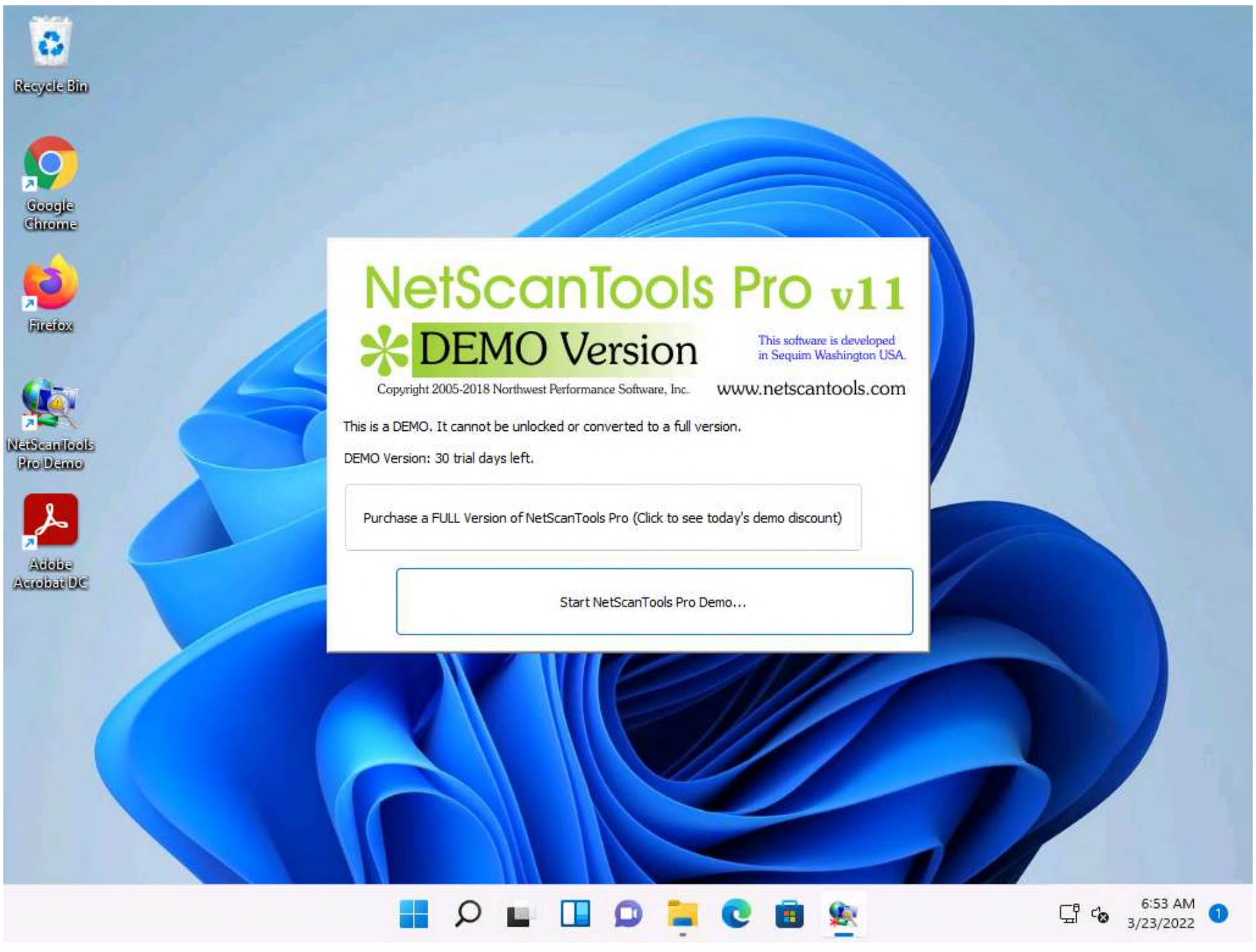
3. In the **Completing the NetScanTools Pro Demo Setup Wizard**, ensure that **Launch NetScanTools Pro Demo** is checked and click **Finish**.



4. The **Reminder** window appears; if you are using a demo version of NetScanTools Pro, click the **Start the DEMO** button.

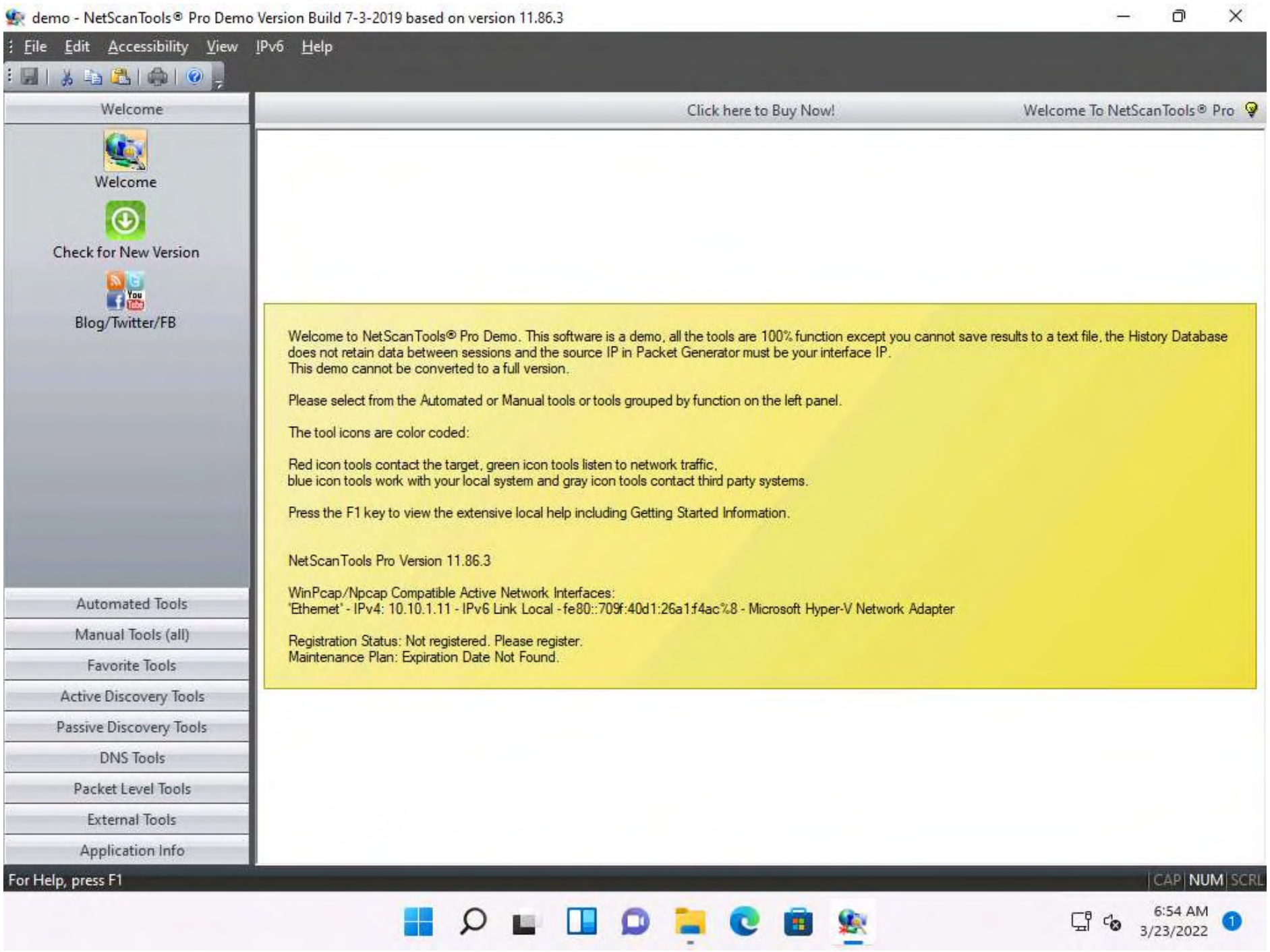


5. A **DEMO Version** pop-up appears; click the **Start NetScanTools Pro Demo...** button.



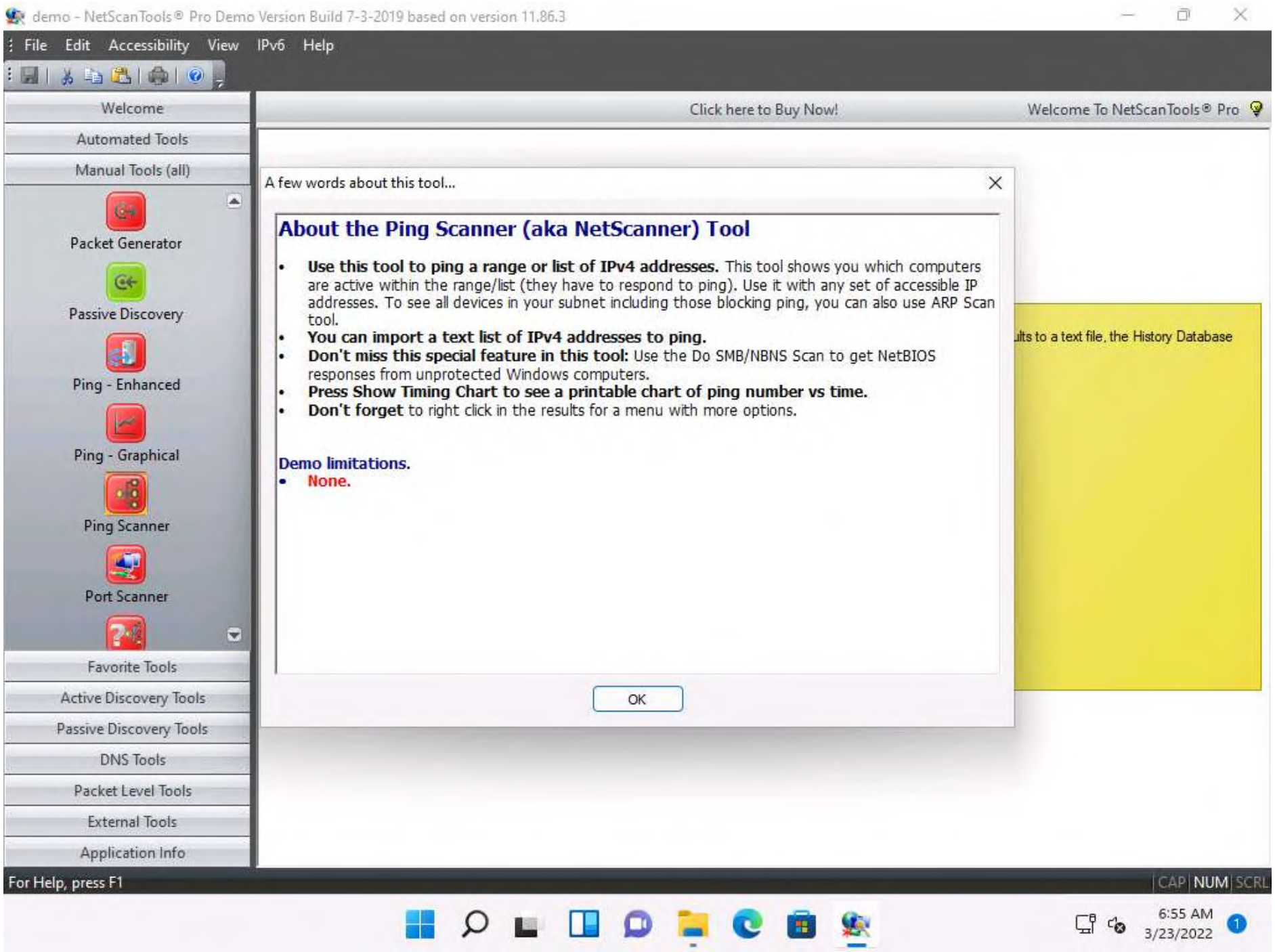
6. The **NetScanTools Pro** main window appears, as shown in the screenshot.

Note: The version of the **NetScanTools Pro** might differ when you perform the lab.



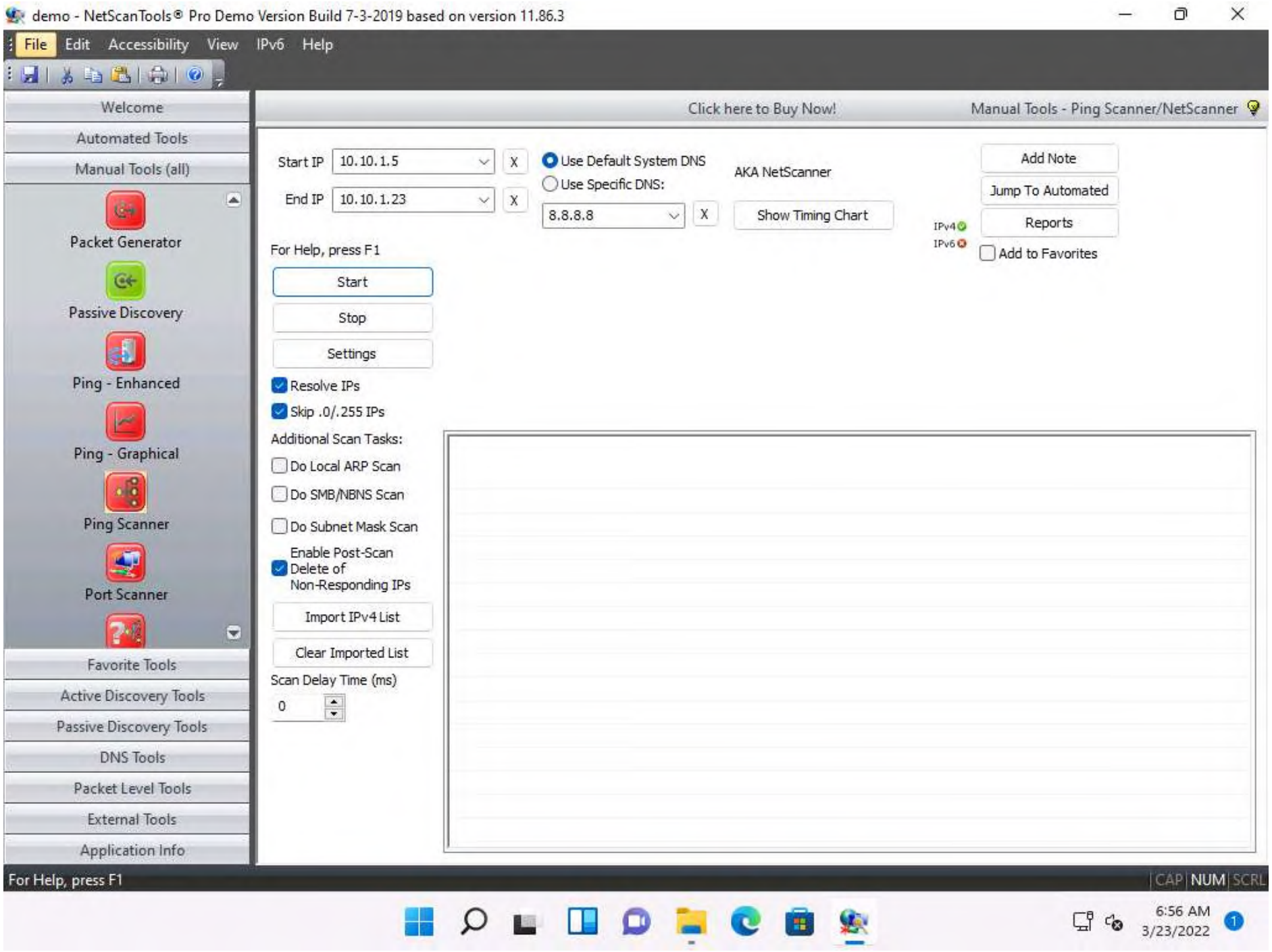
7. In the left-hand pane, under the **Manual Tools (all)** section, scroll down and click the **Ping Scanner** option, as shown in the screenshot.

8. A dialog box opens explaining the **Ping Scanner** tool; click **OK**.

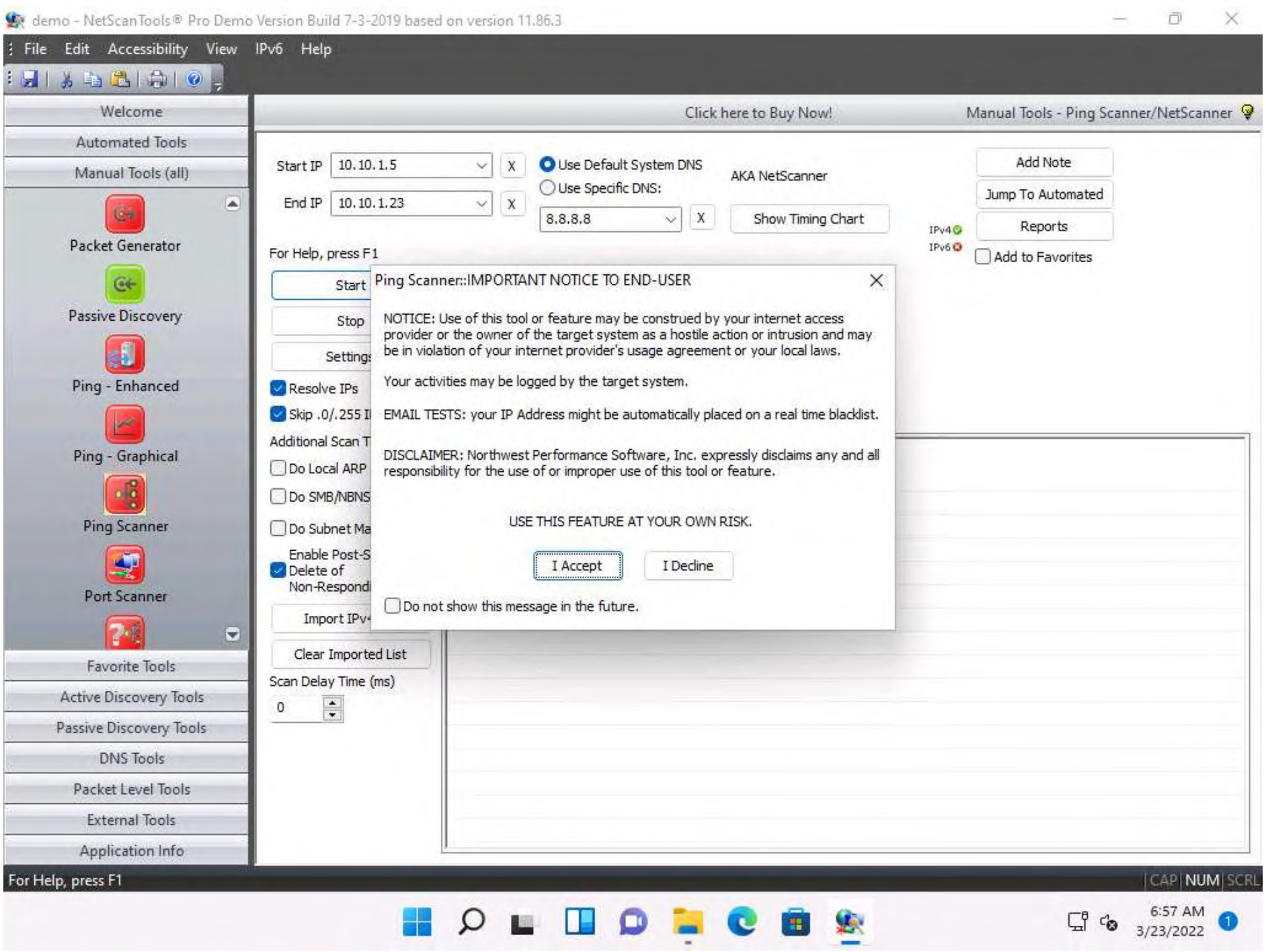


9. Ensure that **Use Default System DNS** is selected. Enter the range of IP addresses into the **Start IP** and **End IP** fields (here, **10.10.1.5 - 10.10.1.23**); then, click **Start**.

> Note: In this lab task, we are scanning **Parrot Machine**, **Windows Server 2022**, **Windows Server 2019**, and **Android** machines.

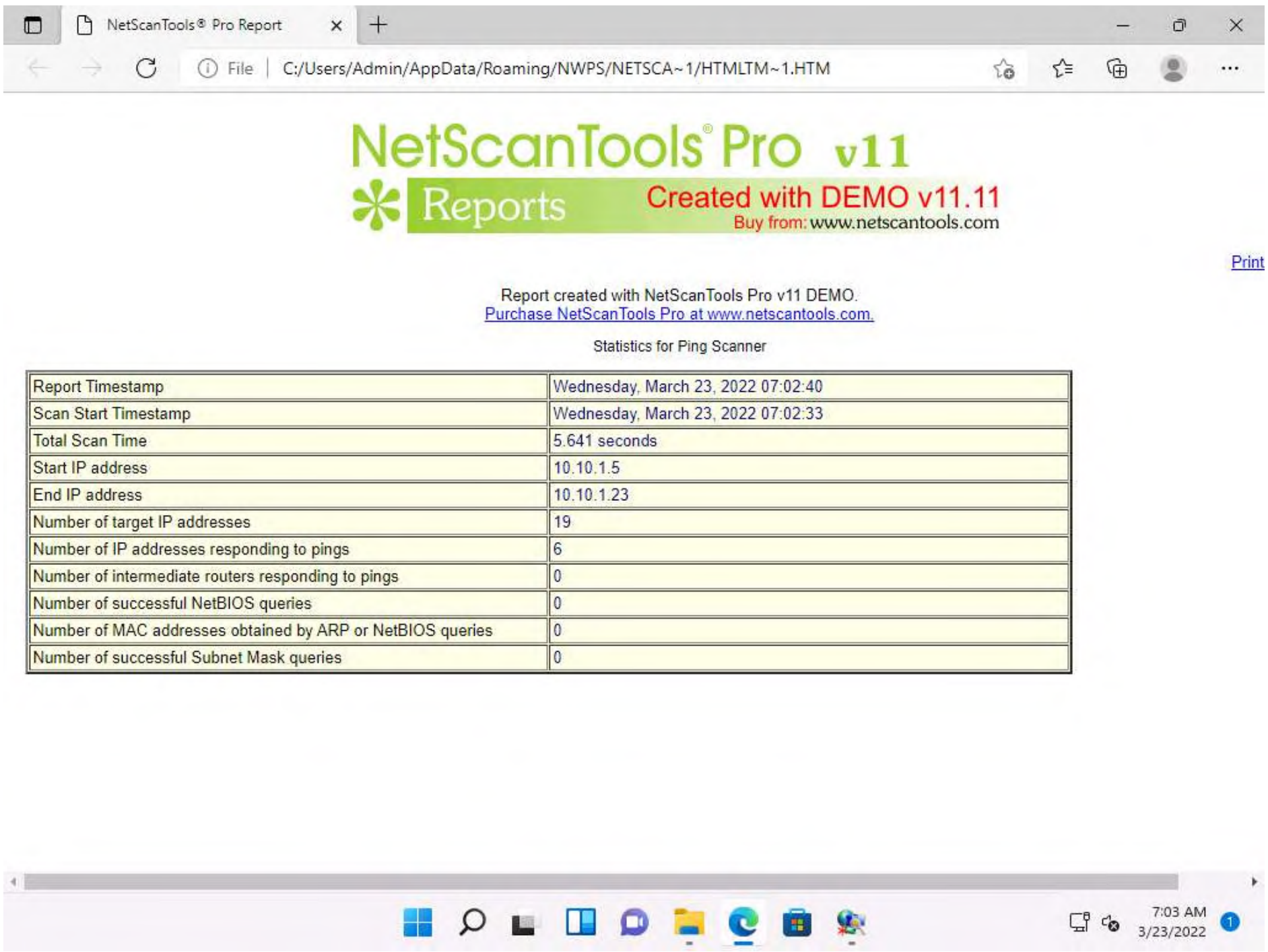


10. A **Ping Scanner** notice pop-up appears; click **I Accept**.



11. After the completion of the scan, a scan result appears in the web browser (here, **Google Chrome**).

Note: If **How do you want to open this file?** pop-up appears select **Google Chrome** from the list and click on **OK**.

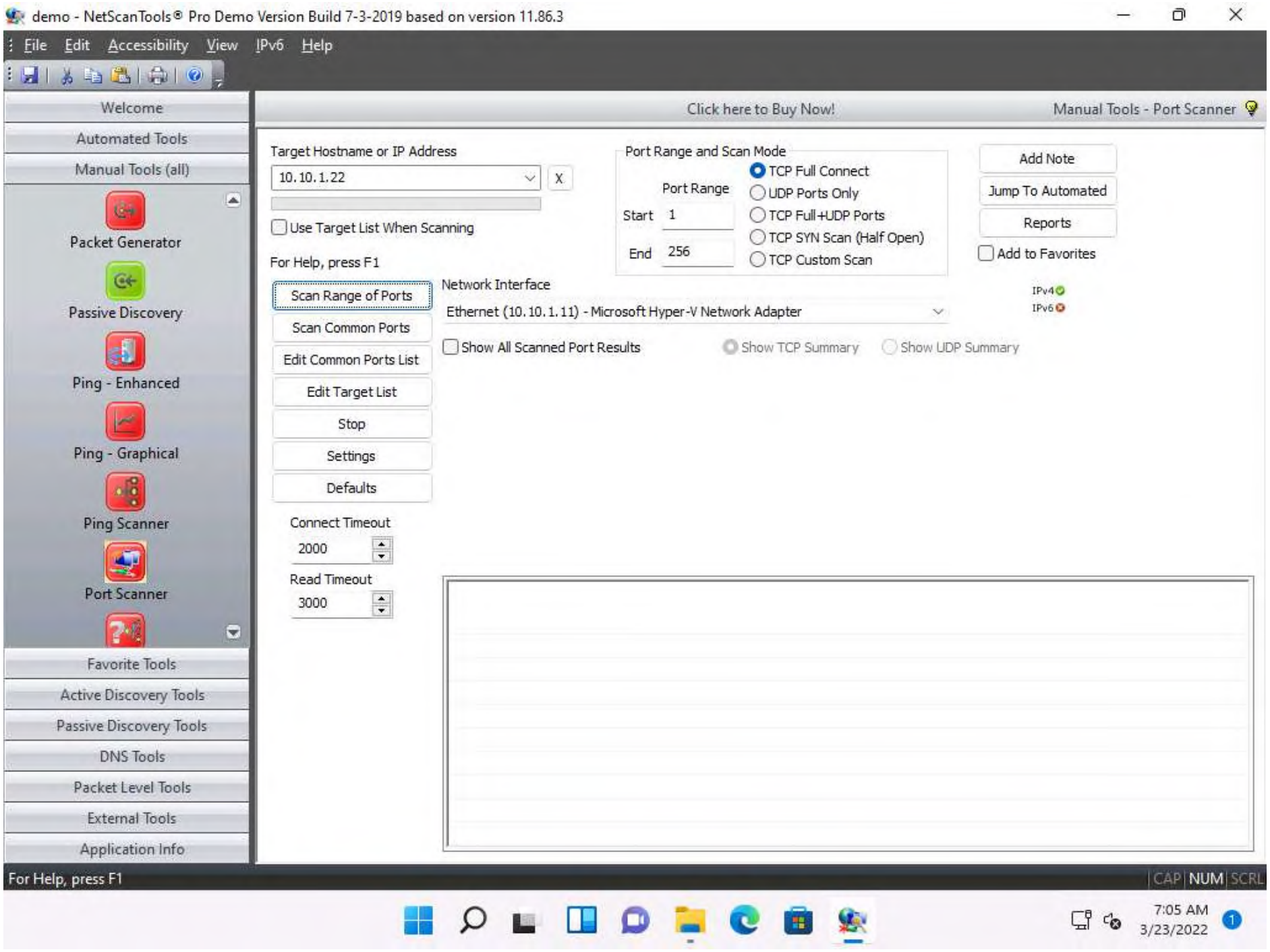


12. Close the browser and switch to the **NetScanTools Pro** window.

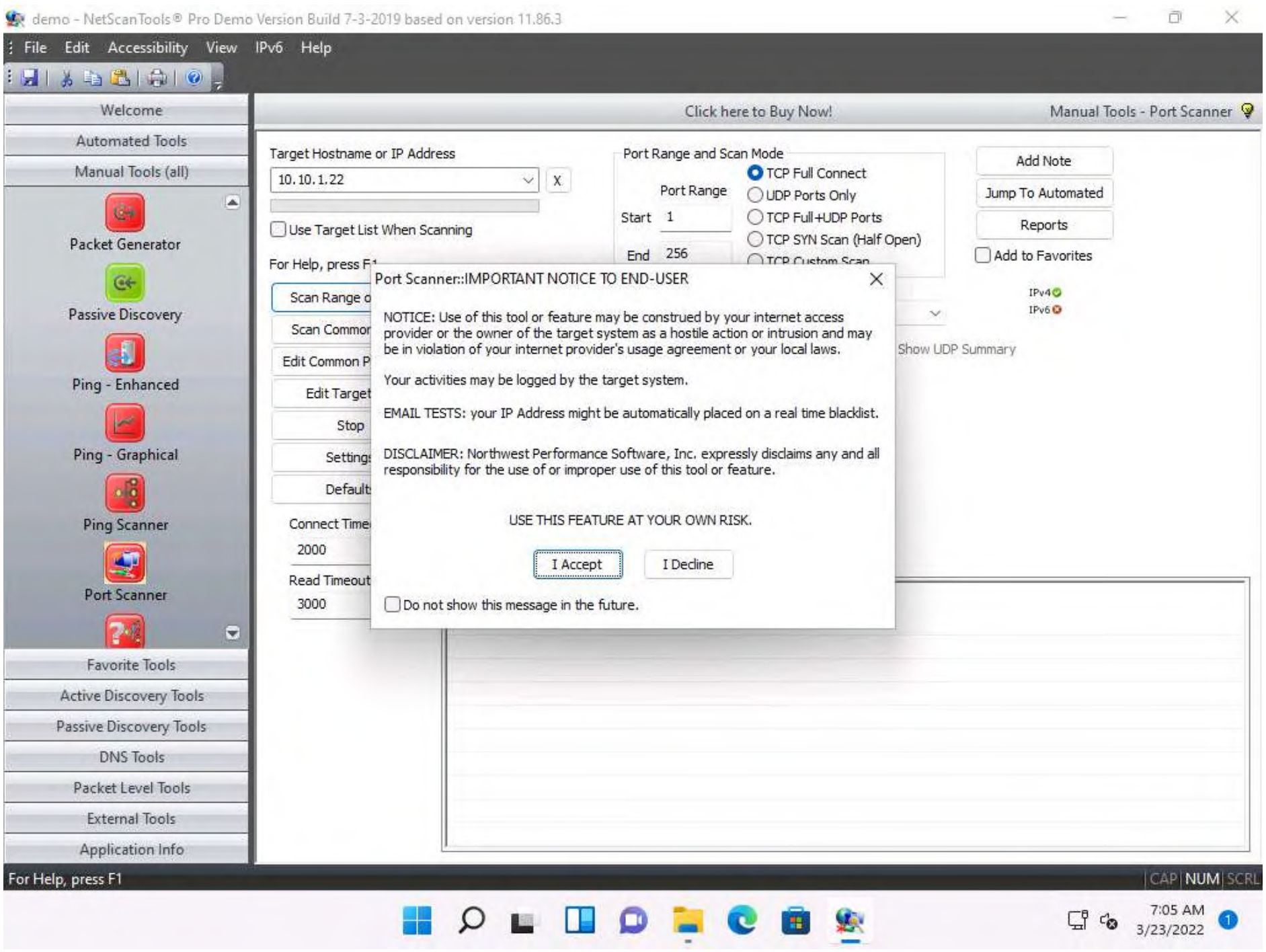
13. Now, click the **Port Scanner** option from the left-hand pane under the **Manual Tools (all)** section.

    Note: If a dialog box appears explaining the **Port Scanner** tool, click **OK**.

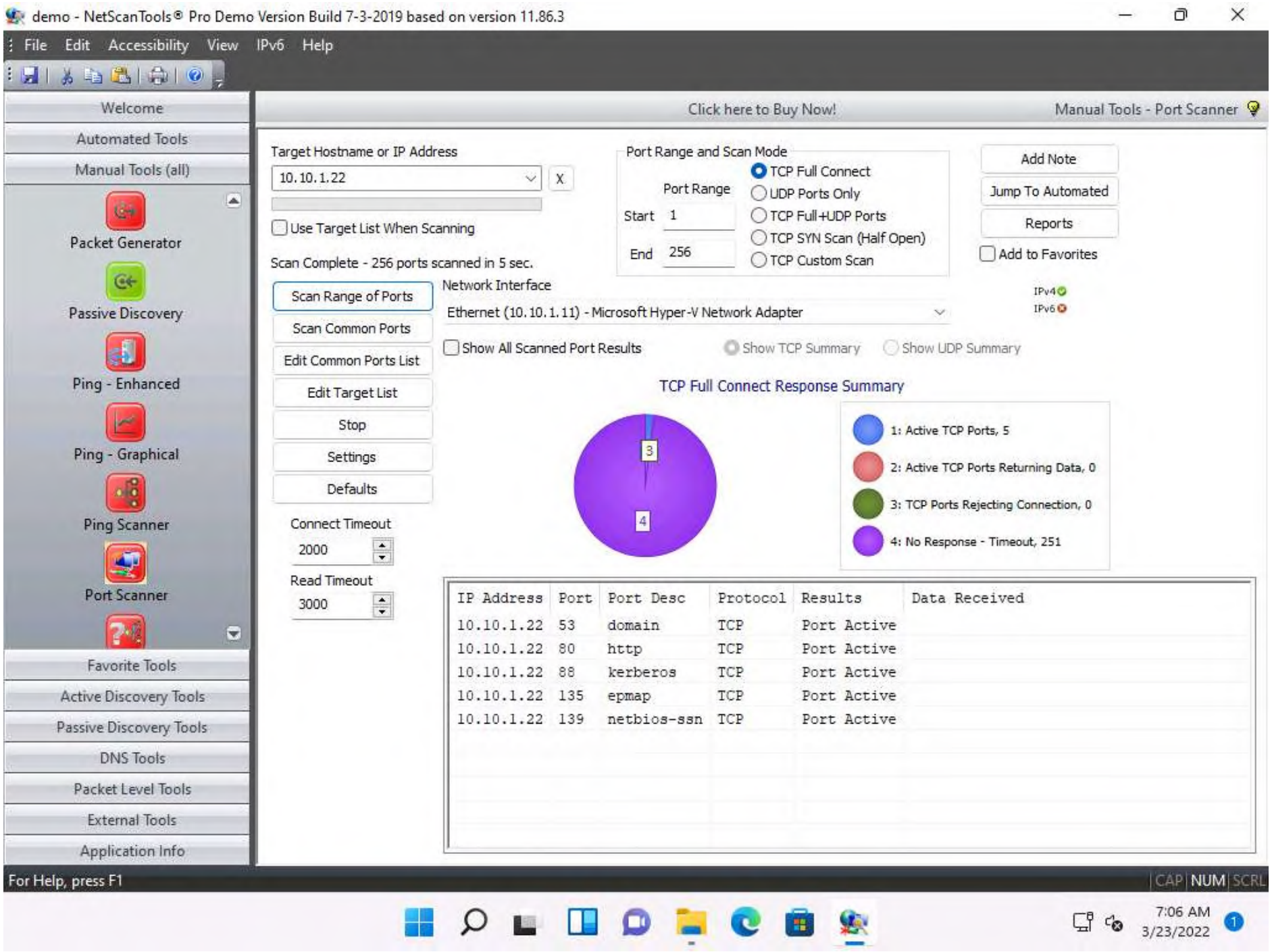
14. In the **Target Hostname or IP Address** field, enter the IP address of the target (here, **10.10.1.22**). Ensure that **TCP Full Connect** radio button is selected, and then click the **Scan Range of Ports** button.



15. A **Port Scanner** notice pop-up appears; click **I Accept**.



16. A result appears displaying the active ports and their descriptions, as shown in the screenshot.



17. By performing the above scans, you will be able to obtain a list of active machines in the network, their respective IP addresses and hostnames, and a list of all the open ports and services that will allow you to choose a target host in order to enter into its network and perform malicious activities such as ARP poisoning, sniffing, etc.

18. This concludes the demonstration of discovering open ports and services running on the target IP address using NetScanTools Pro.

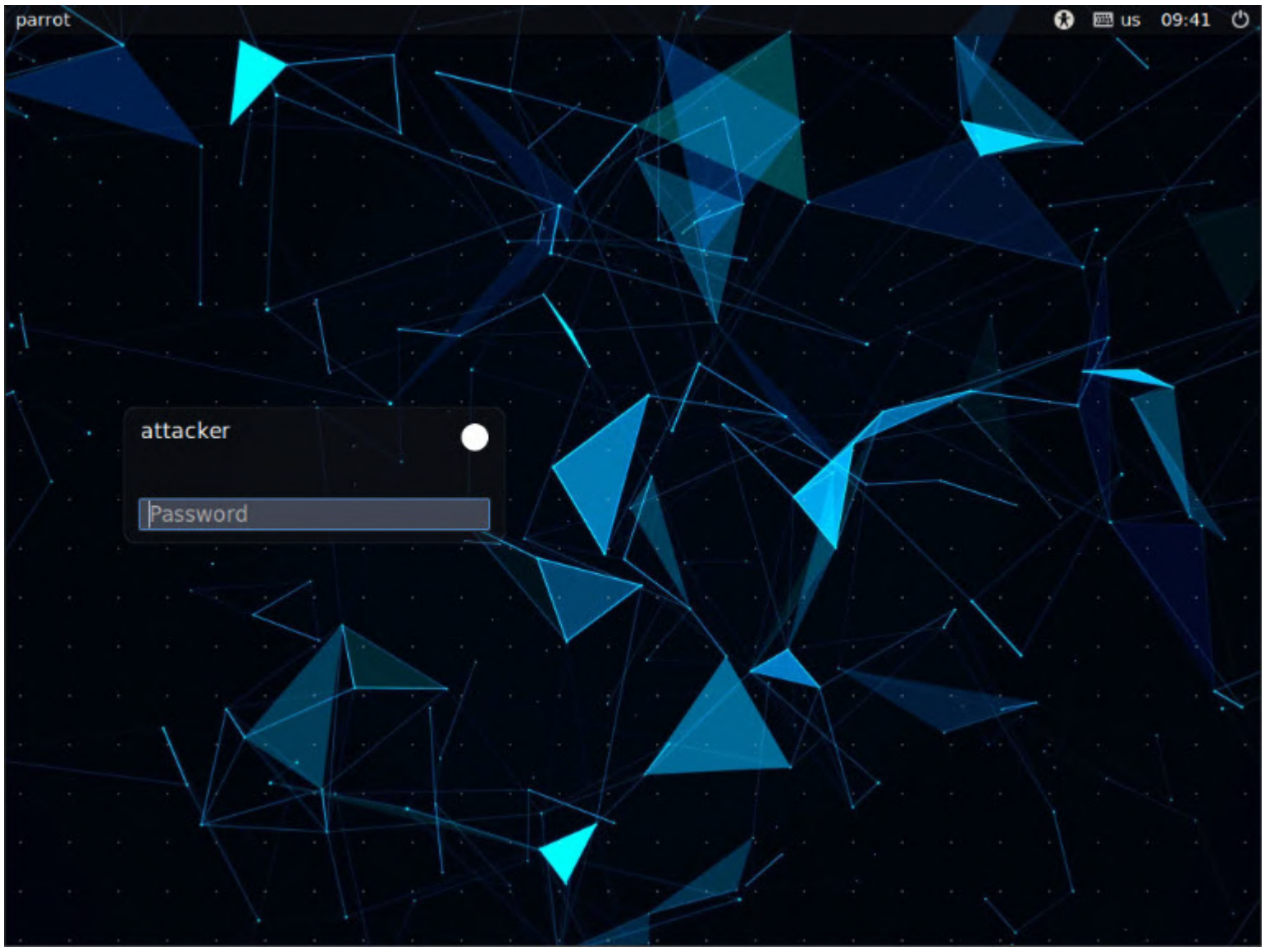
19. Close all open windows and document all the acquired information.

Task 3: Perform Port Scanning using sx Tool

The sx tool is a command-line network scanner that can be used to perform ARP scans, ICMP scans, TCP SYN scans, UDP scans and application scans such as SOCS5 scan, Docker scan and Elasticsearch scan.

Here, we will use sx to perform ARP scans, TCP scans and UDP scans to discover open ports in the target machine.

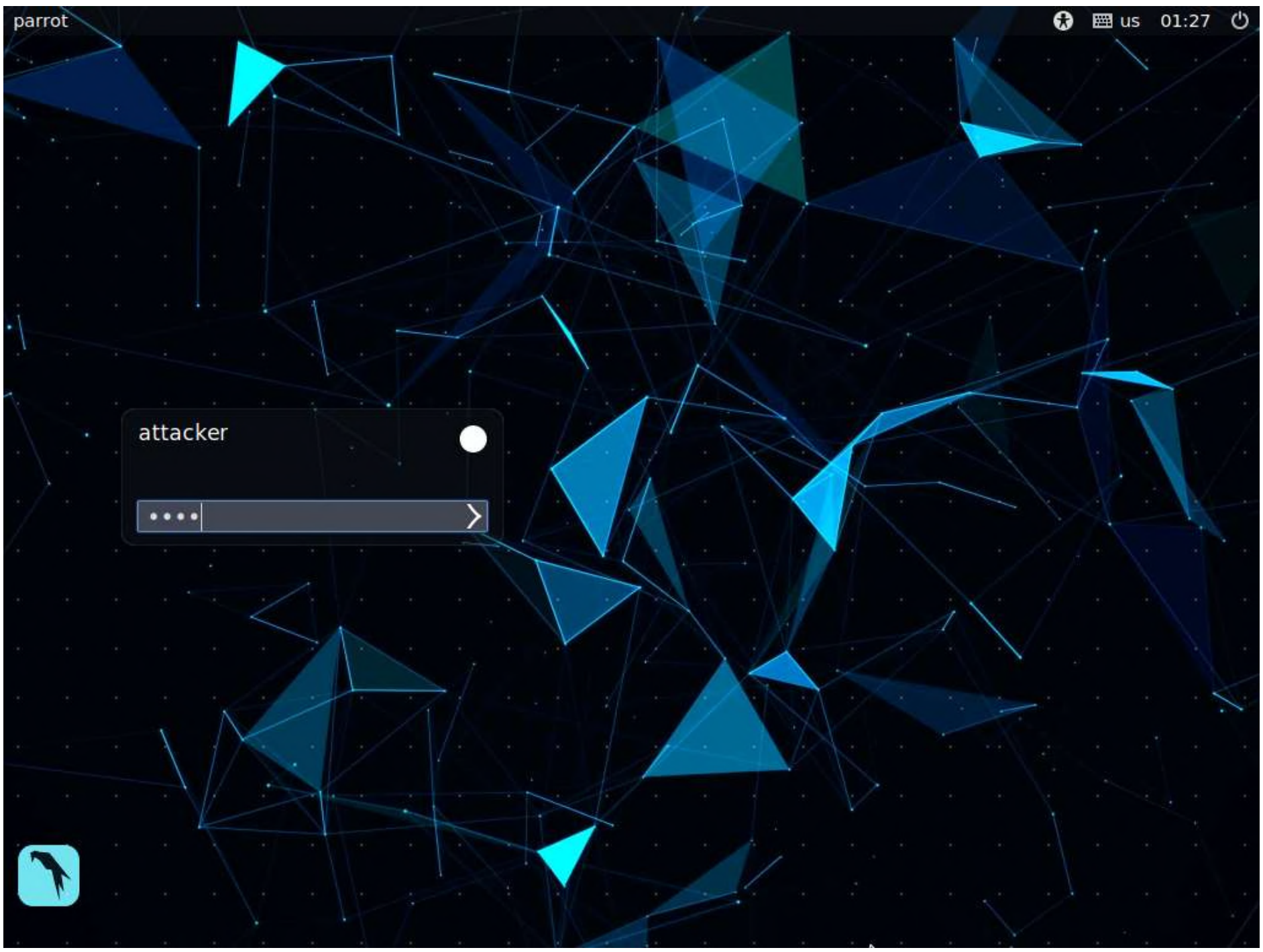
1. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.



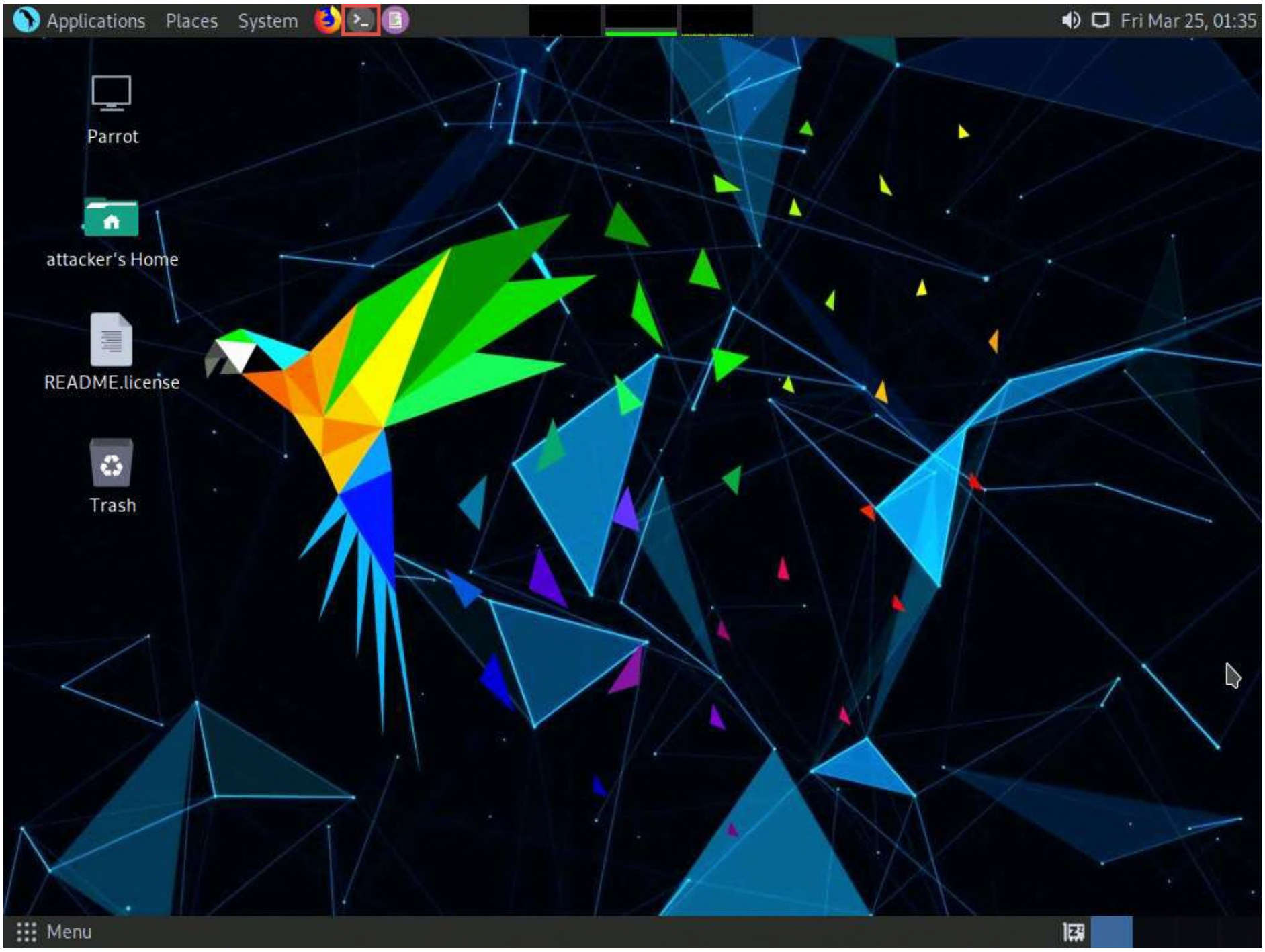
2. In the login page, the **attacker** username will be selected by default. Enter password as **toor** in the **Password** field and press **Enter** to log in to the machine.

Note: If a **Parrot Updater** pop-up appears at the top-right corner of **Desktop**, ignore and close it.

Note: If a **Question** pop-up window appears asking you to update the machine, click **No** to close the window.



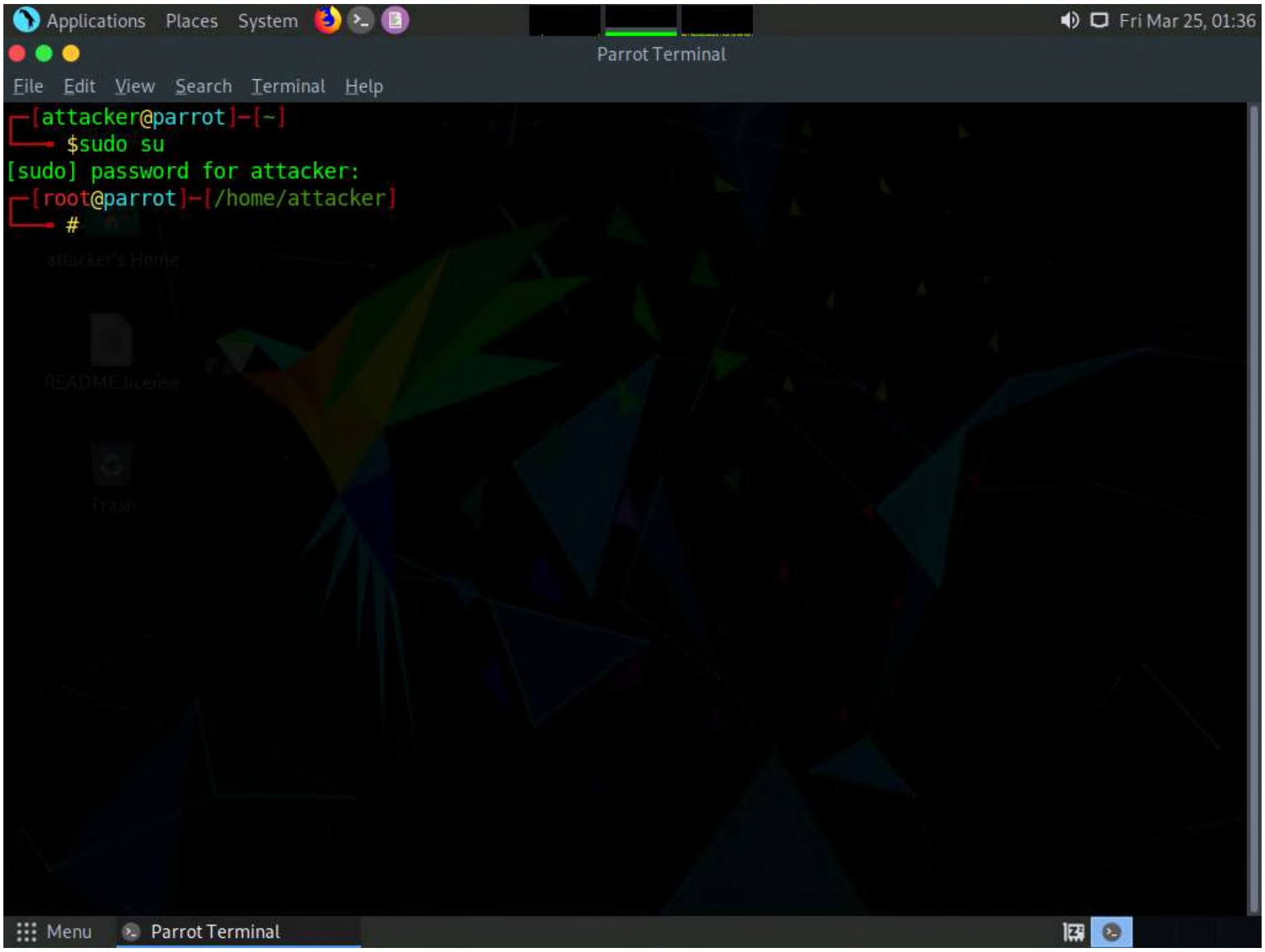
3. Click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.



4. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

5. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

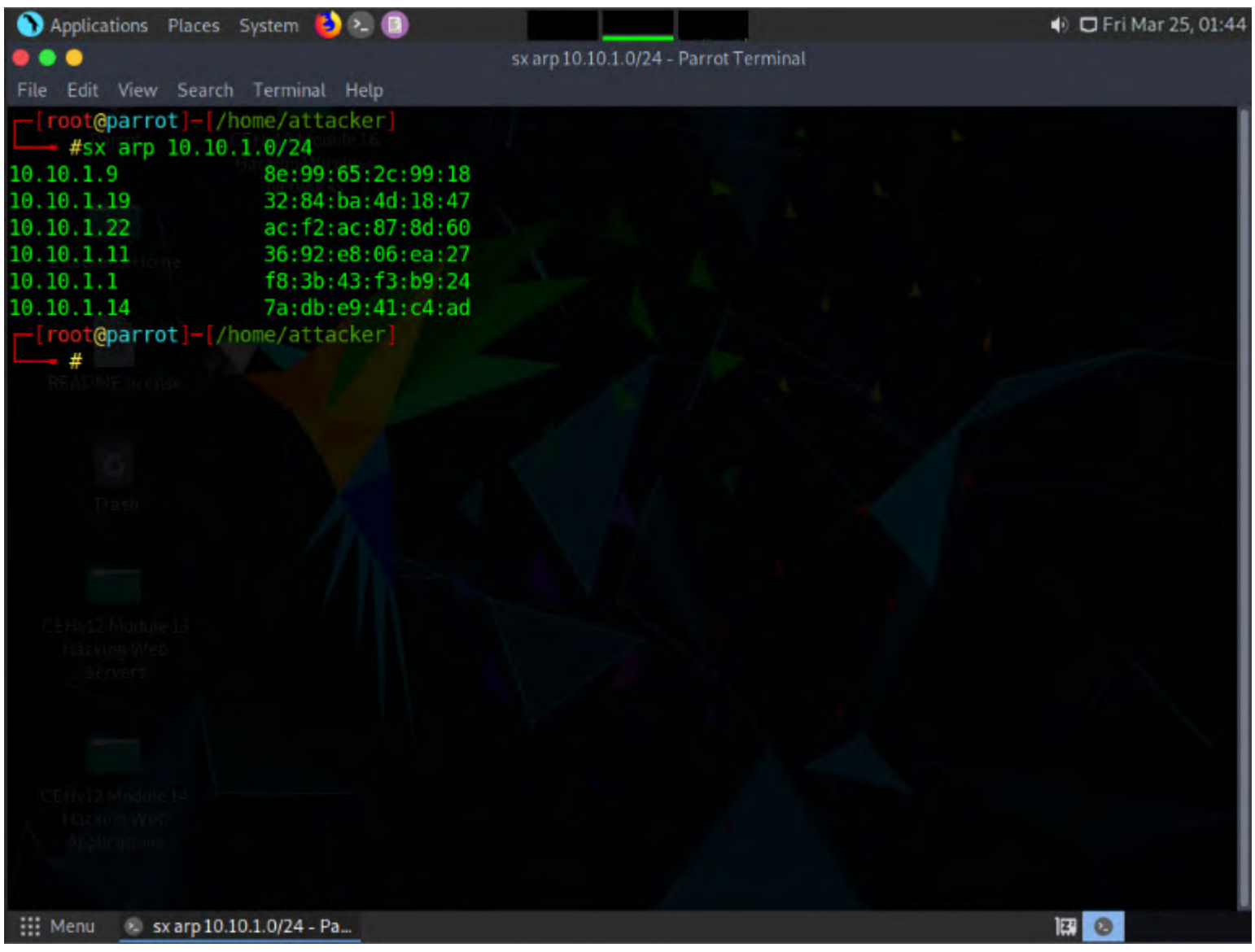
Note: The password that you type will not be visible.



6. In the terminal window, type **sx arp [Target subnet]** and press **Enter** (here, the target subnet is **10.10.1.0/24**) to scan all the IP addresses and MAC addresses associated with the connected devices in a local network).

Note: **arp**: performs an ARP scan.

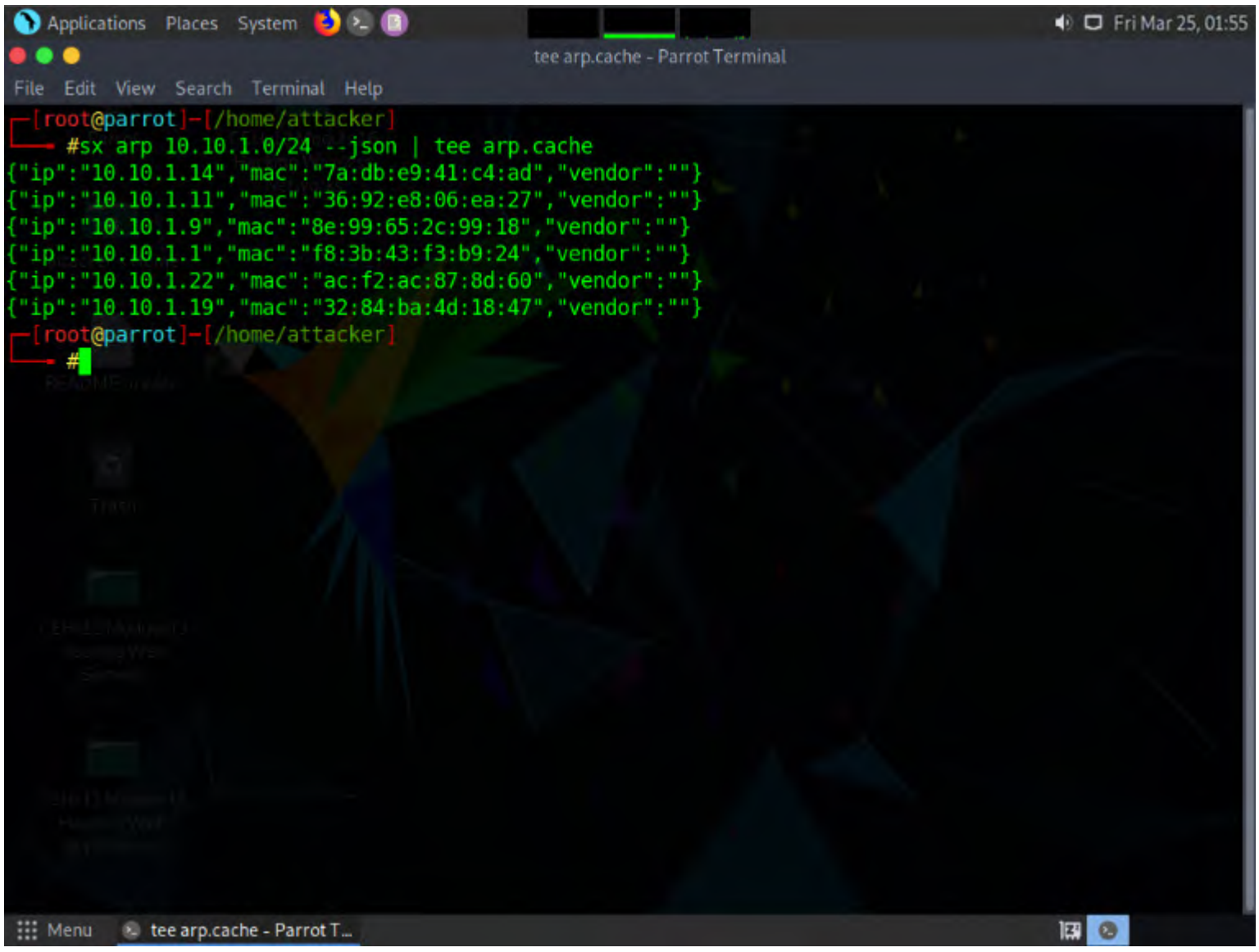
Note: The MAC addresses might vary when you perform this task.



7. Type **sx arp [Target subnet] --json | tee arp.cache** and press **Enter** to create arp.cache file (here, the target subnet is **10.10.1.0/24**).

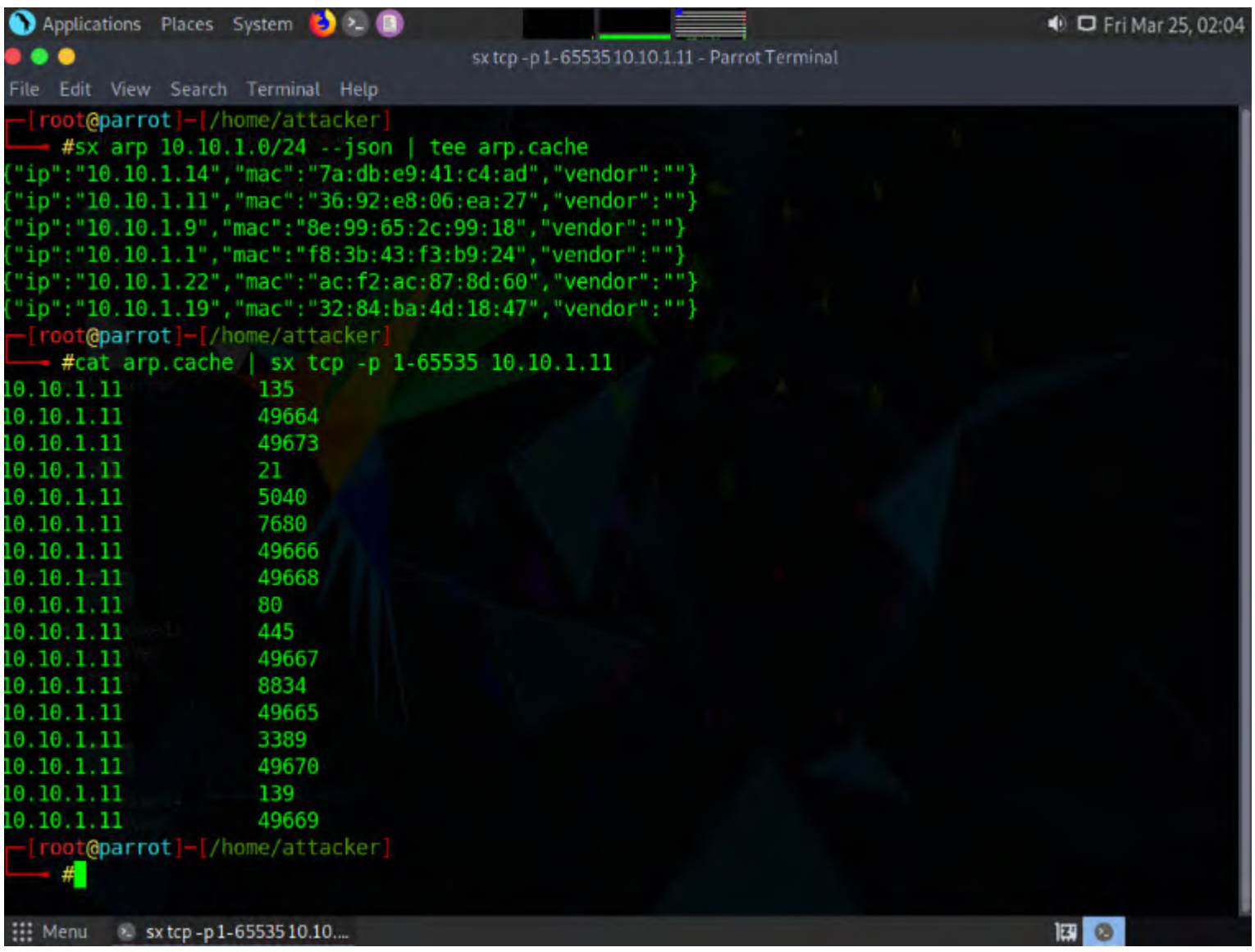
Note: **--json** converts a text file to the JSON format, **tee** writes the data to stdin.

Note: Before the actual scan, sx explicitly creates an ARP cache file which is a simple text file containing a JSON string on each line and has the same JSON fields as the ARP scan JSON output. The protocols such as TCP and UDP read the ARP cache file from stdin and then begin the scan.

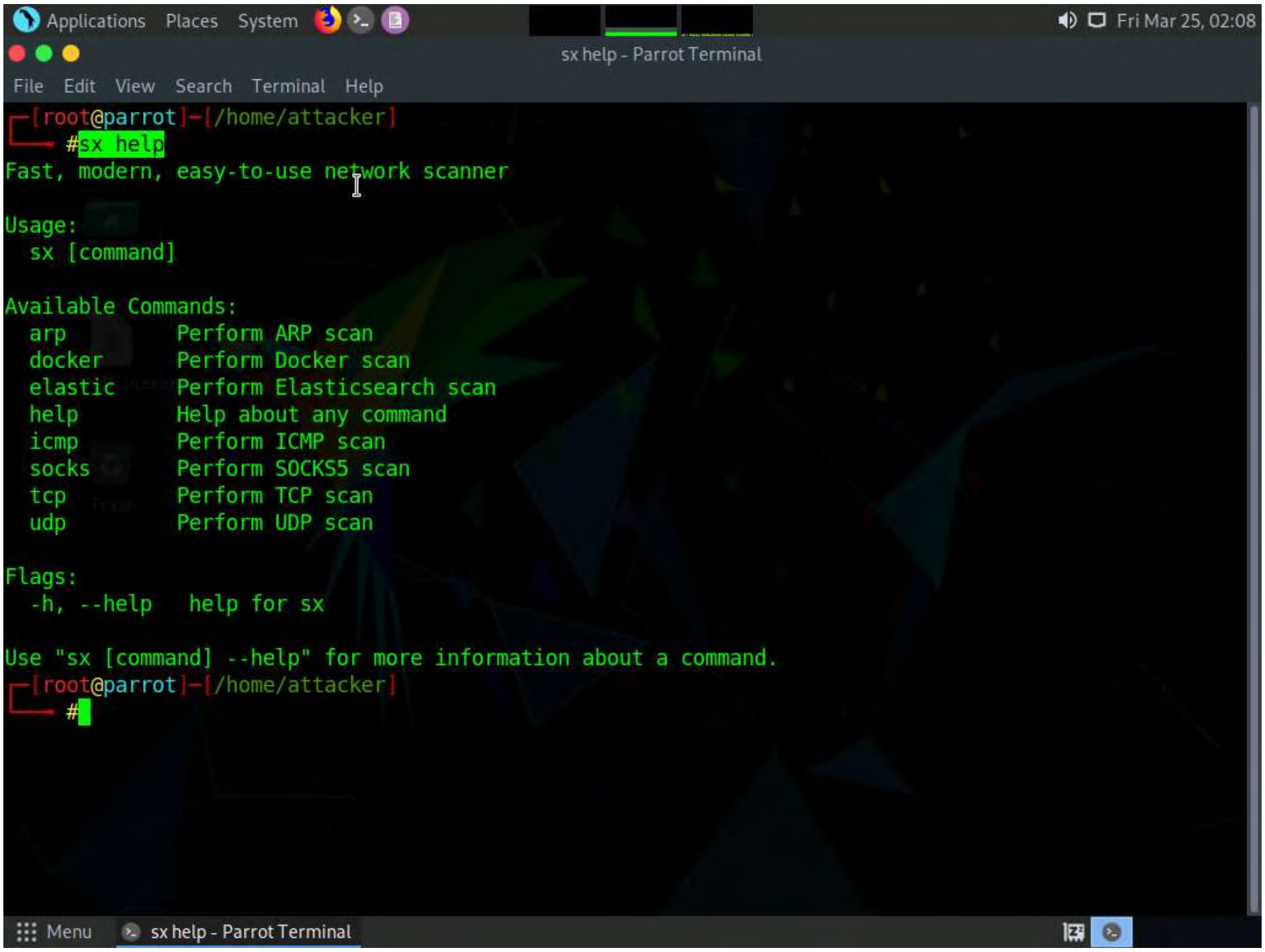


8. Type **cat arp.cache | sx tcp -p 1-65535 [Target IP address]** and press **Enter** to list all the open tcp ports on the target machine (here, the target IP address is **10.10.1.11**).

Note: **tcp**: performs a TCP scan, **-p**: specifies the range of ports to be scanned (here, the range is **1-65535**).



9. In the terminal, type **sx help** and press **Enter** to obtain the list of commands that can be used. For more information, you can further use **sx --help** command.



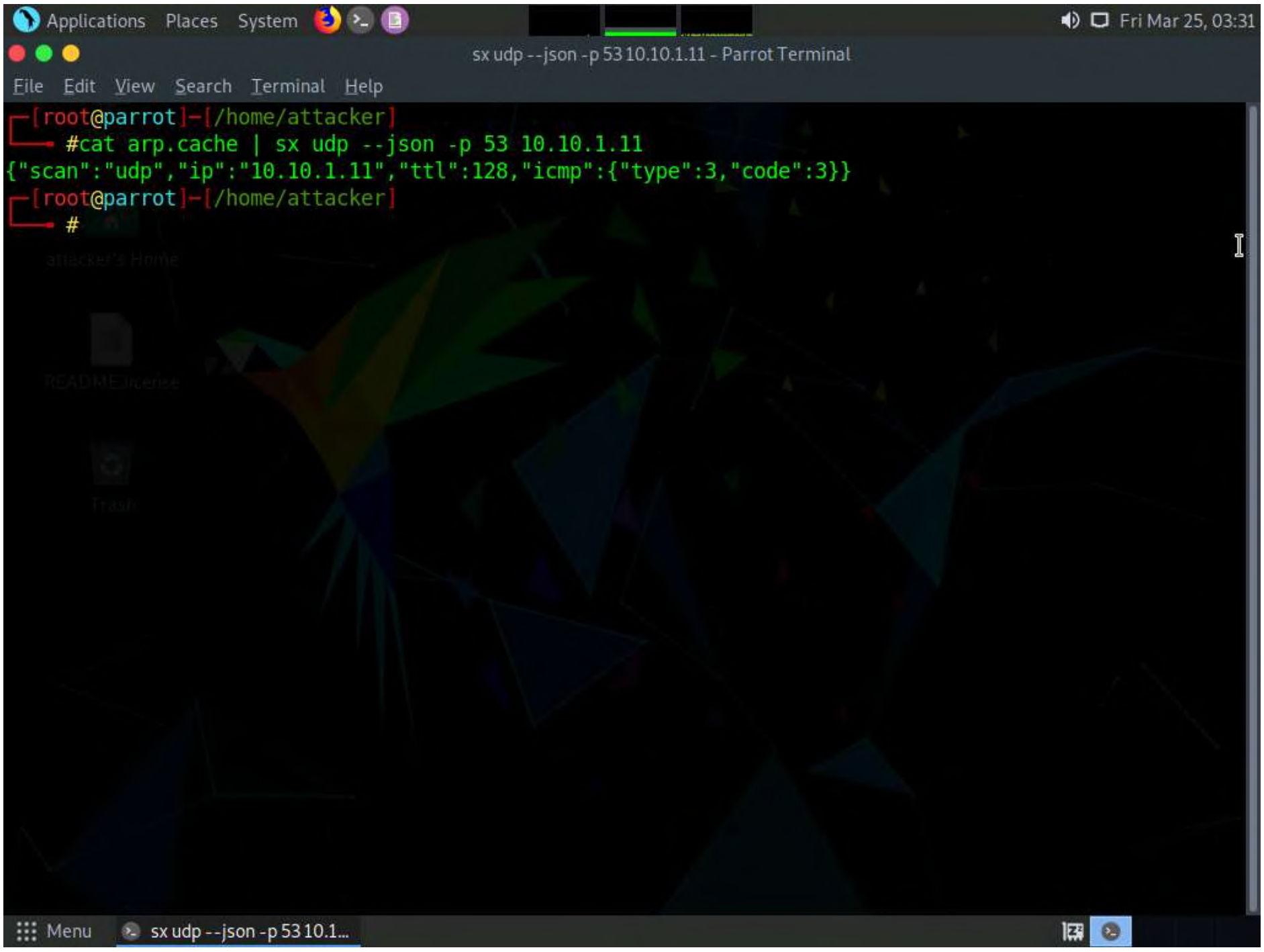
10. Now, let us perform UDP scan on the target machine to check if a port is open or closed.

11. In the terminal, type **cat arp.cache | sx udp --json -p [Target Port] 10.10.1.11** and press **Enter** (here, target port is **53**).

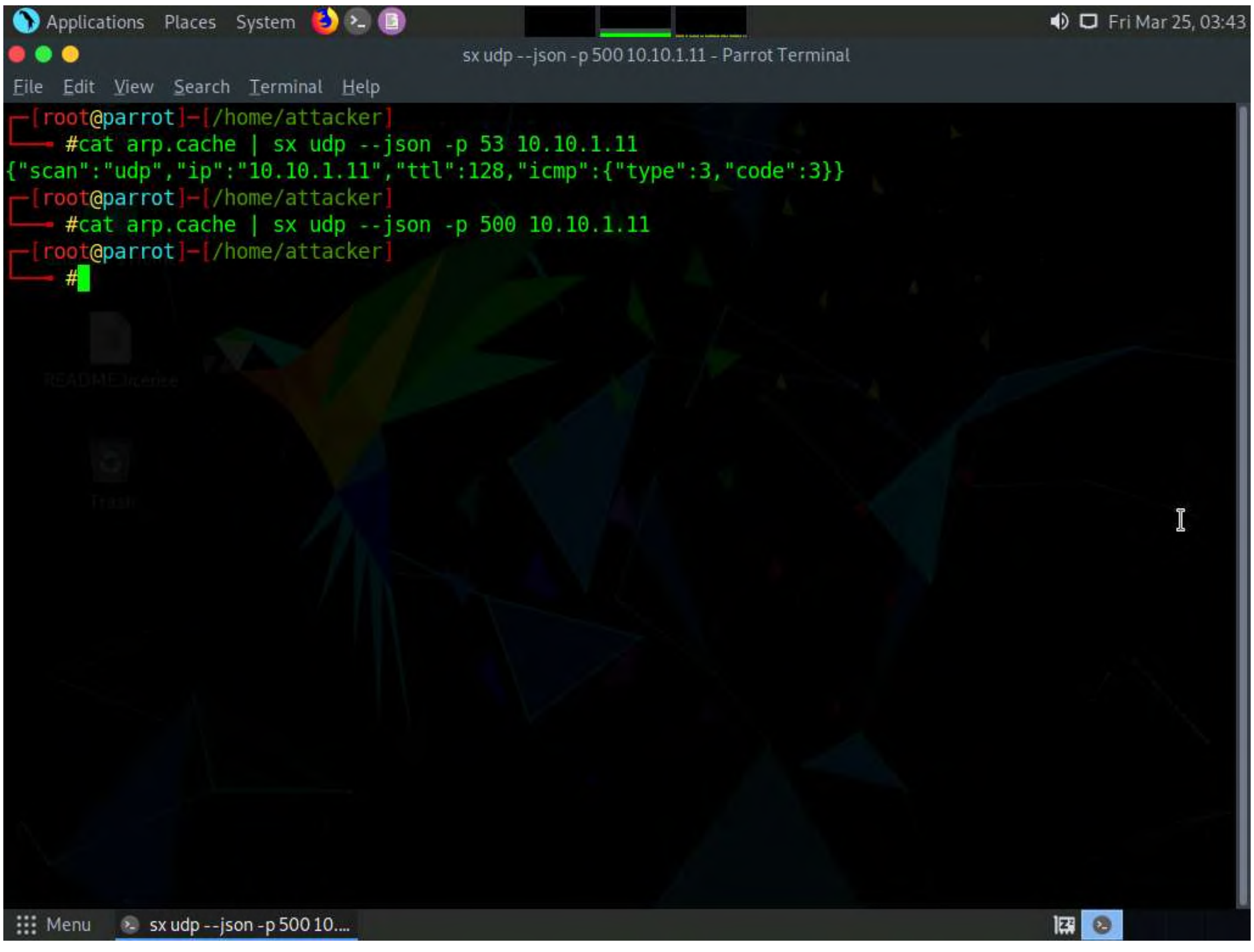
> Note: **udp**: performs a UDP scan, **-p** specifies the target port.
>
> Note: In a UDP scan **sx** returns the IP address, ICMP packet type and code set to the reply packet.

12. The result appears, with the reply packet from the host with **Destination Unreachable** type (**3**) and **Port Unreachable** code (**3**), which indicates that the target port is closed.

> Note: - According to **RFC1122**, a host should generate Destination Unreachable messages with code: 2 (Protocol Unreachable), when the designated transport protocol is not supported; or 3 (Port Unreachable), when the designated transport protocol (e.g., UDP) is unable to demultiplex the datagram but has no protocol mechanism to inform the sender.
>
> According to **RFC792**, network unreachable error is specified with code: 0, Host unreachable error with code: 1, Protocol unreachable error with code: 2, Port unreachable error with code 3.



13. Type **cat arp.cache | sx udp --json -p [Target Port] 10.10.1.11** and press **Enter** (here, the target port is **500**).



14. You can observe that sx does not return any code in the above command, which states that the target port is open.

15. This concludes the demonstration of port scanning using sx Tool.

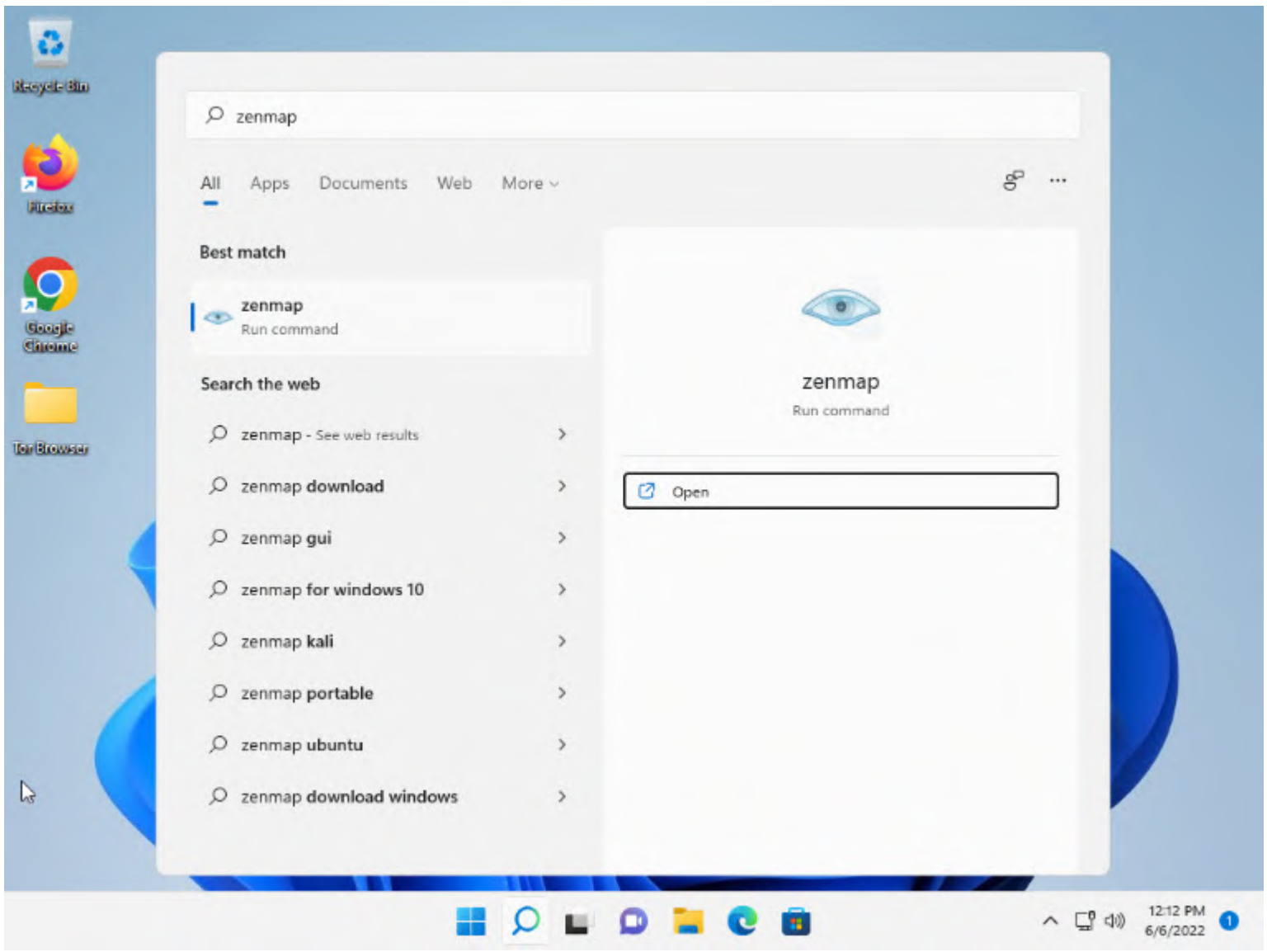
16. Close all open windows and document all acquired information.

Task 4: Explore Various Network Scanning Techniques using Nmap

Nmap comes with various inbuilt scripts that can be employed during a scanning process in an attempt to find the open ports and services running on the ports. It sends specially crafted packets to the target host, and then analyzes the responses to accomplish its goal. Nmap includes many port scanning mechanisms (TCP and UDP), OS detection, version detection, ping sweeps, etc.

Here, we will use Nmap to discover open ports and services running on the live hosts in the target network.

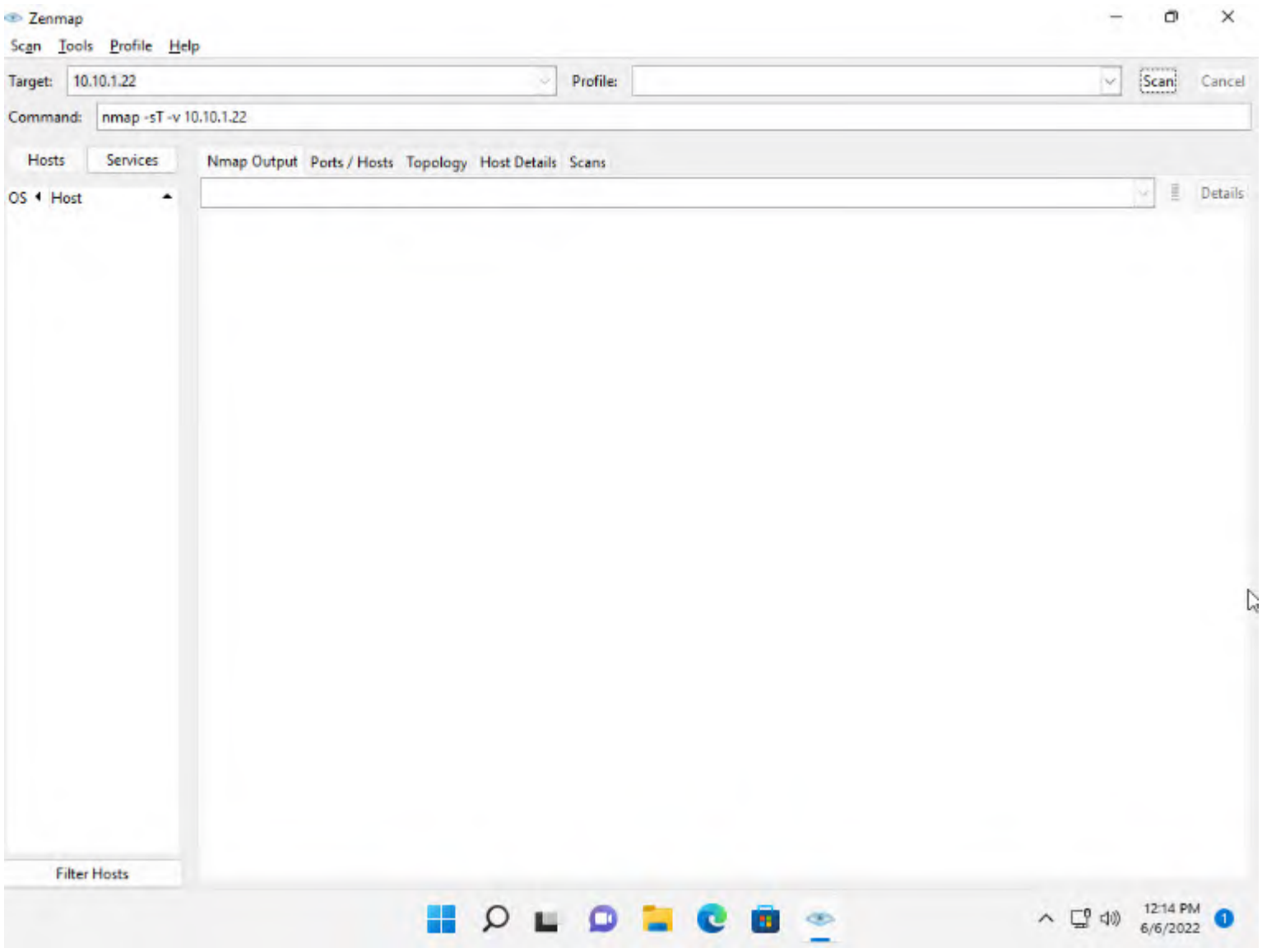
1. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine. In the **Windows 11** machine, click **Search** icon ( ) on the **Desktop**. Type **zenmap** in the search field, the **Zenmap** appears in the results, click **Open** to launch it.



2. The **Zenmap** appears; in the **Command** field, type the command **nmap -sT -v [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

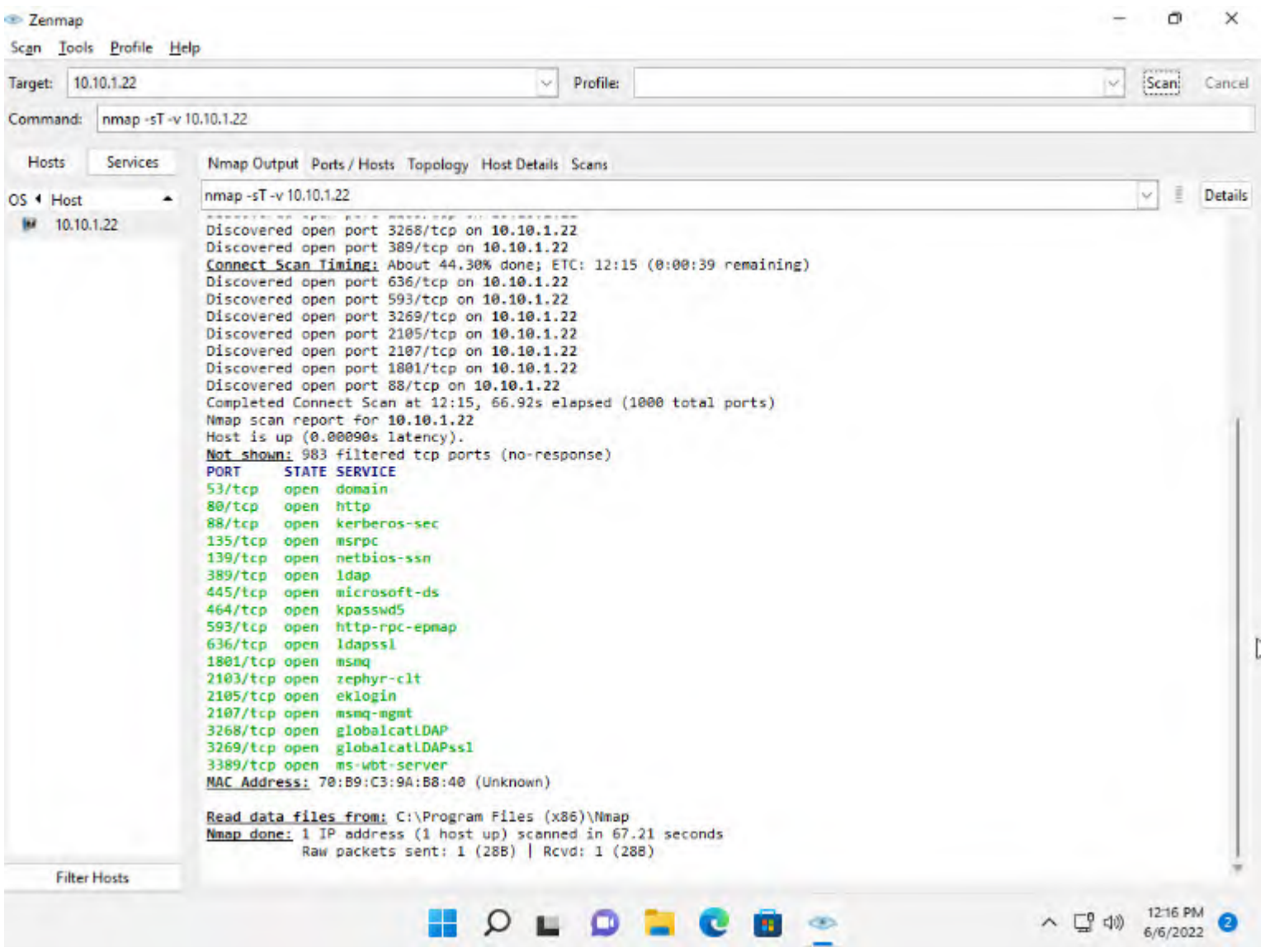
Note: **-sT**: performs the TCP connect/full open scan and **-v**: enables the verbose output (include all hosts and ports in the output).

Note: The MAC addresses might differ when you perform the task.

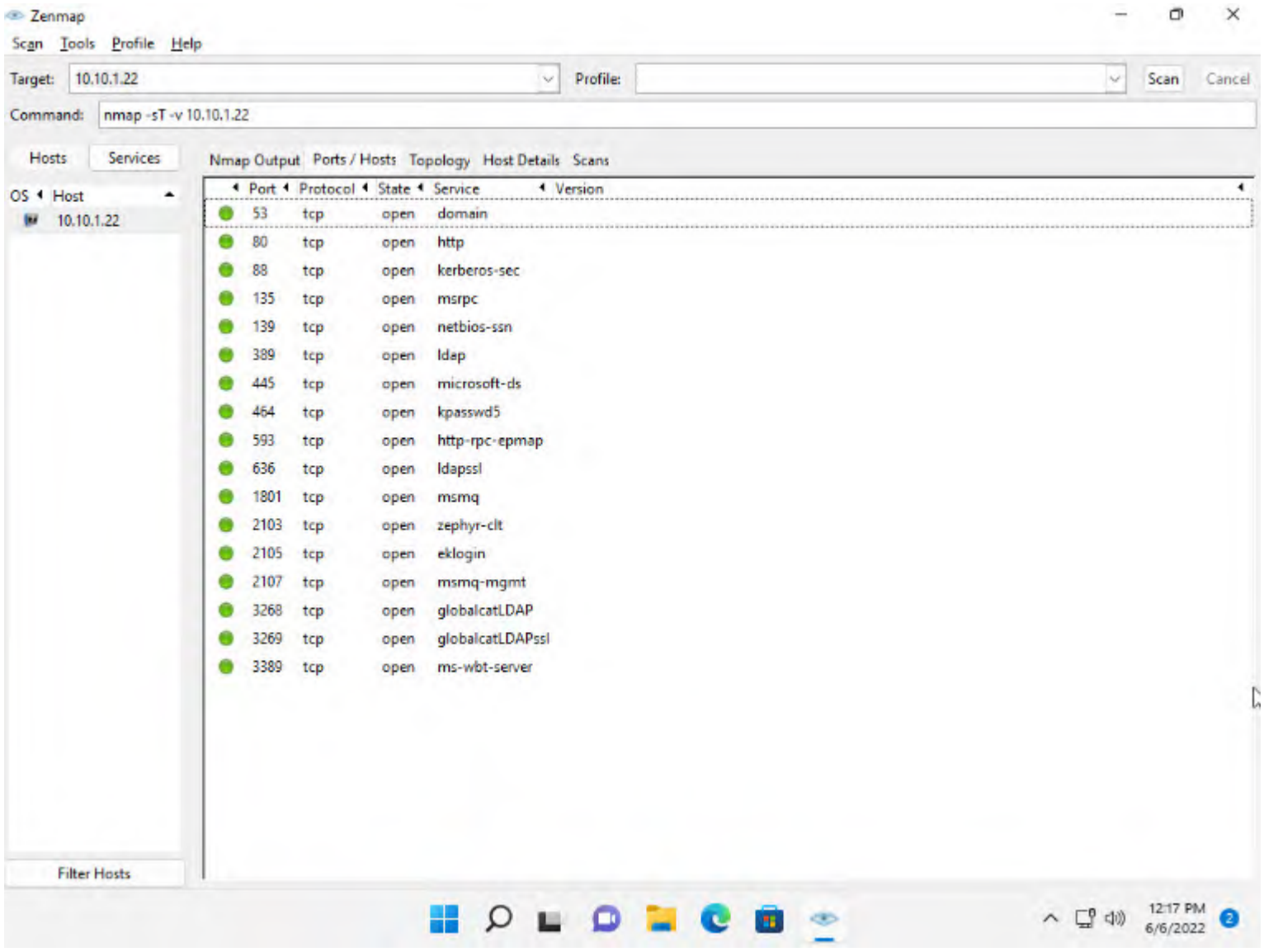


3. The scan results appear, displaying all the open TCP ports and services running on the target machine, as shown in the screenshot.

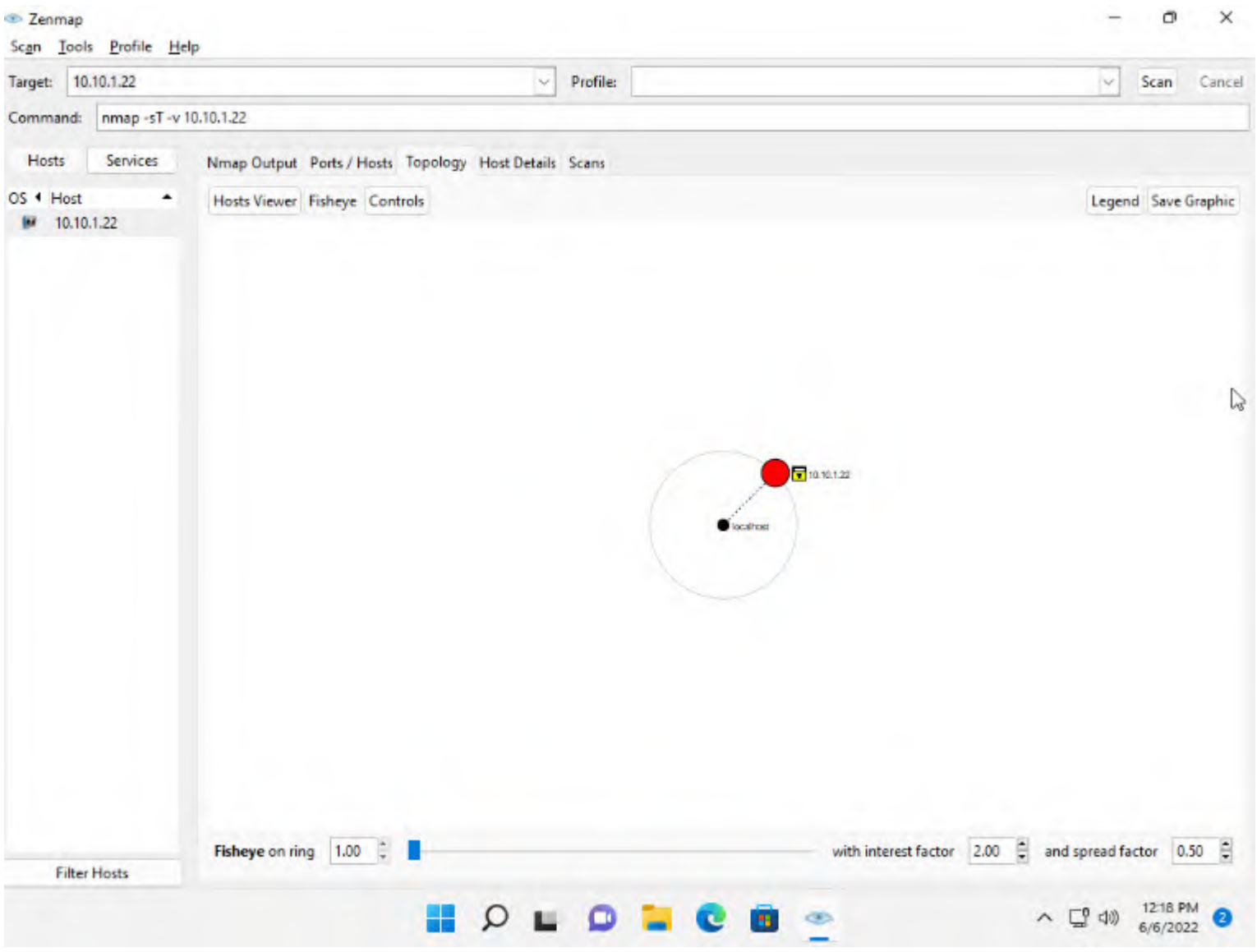
Note: TCP connect scan completes a three-way handshake with the target machine. In the TCP three-way handshake, the client sends a SYN packet, which the recipient acknowledges with the SYN+ACK packet. In turn, the client acknowledges the SYN+ACK packet with an ACK packet to complete the connection. Once the handshake is completed, the client sends an RST packet to end the connection.



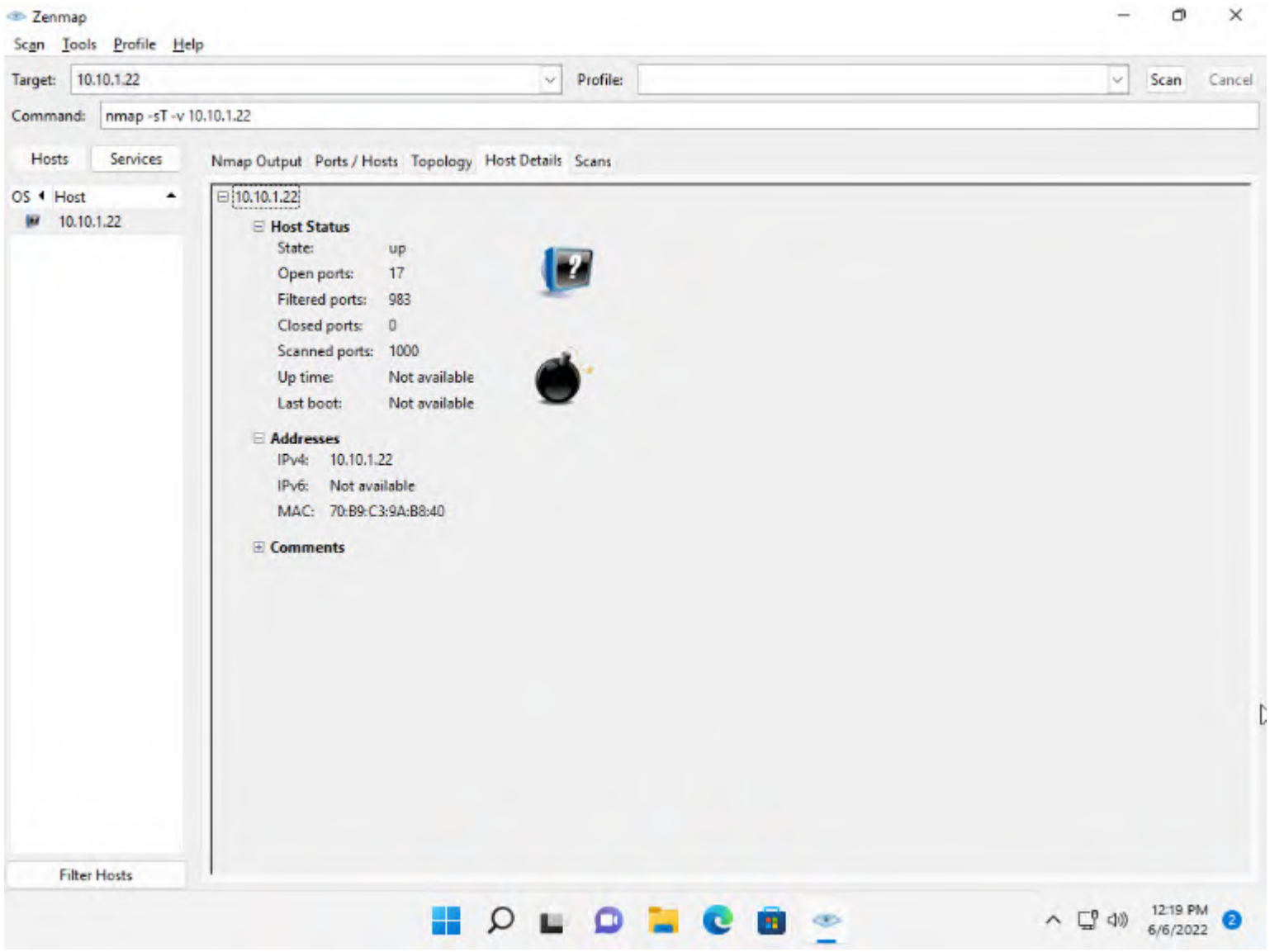
4. Click the **Ports/Hosts** tab to gather more information on the scan results. Nmap displays the Port, Protocol, State, Service, and Version of the scan.



5. Click the **Topology** tab to view the topology of the target network that contains the provided IP address and click the **Fisheye** option to view the topology clearly.

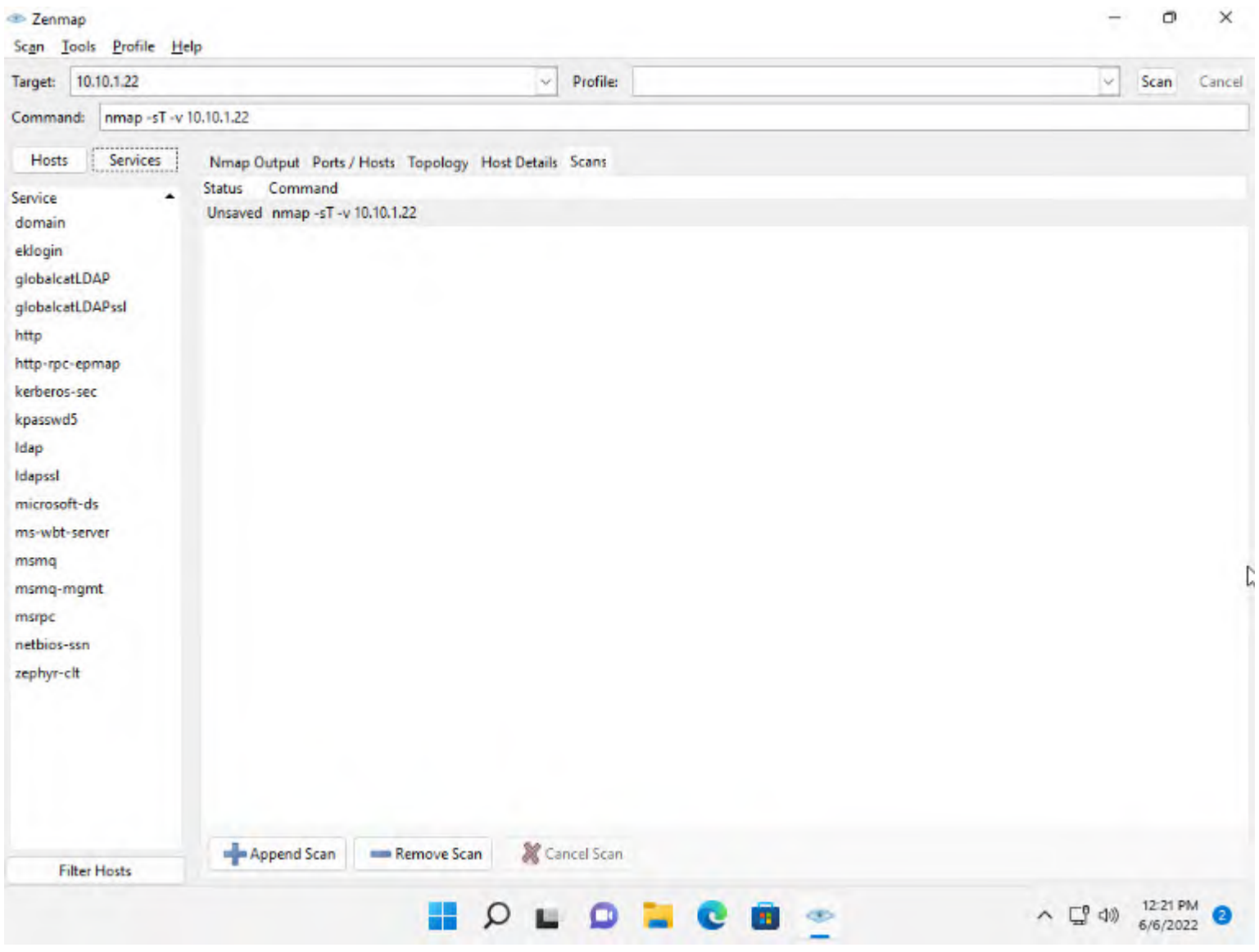


6. In the same way, click the **Host Details** tab to view the details of the TCP connect scan.



7. Click the **Scans** tab to view the command used to perform TCP connect/full open scan.

8. Click the **Services** tab located in the left pane of the window. This tab displays a list of services.

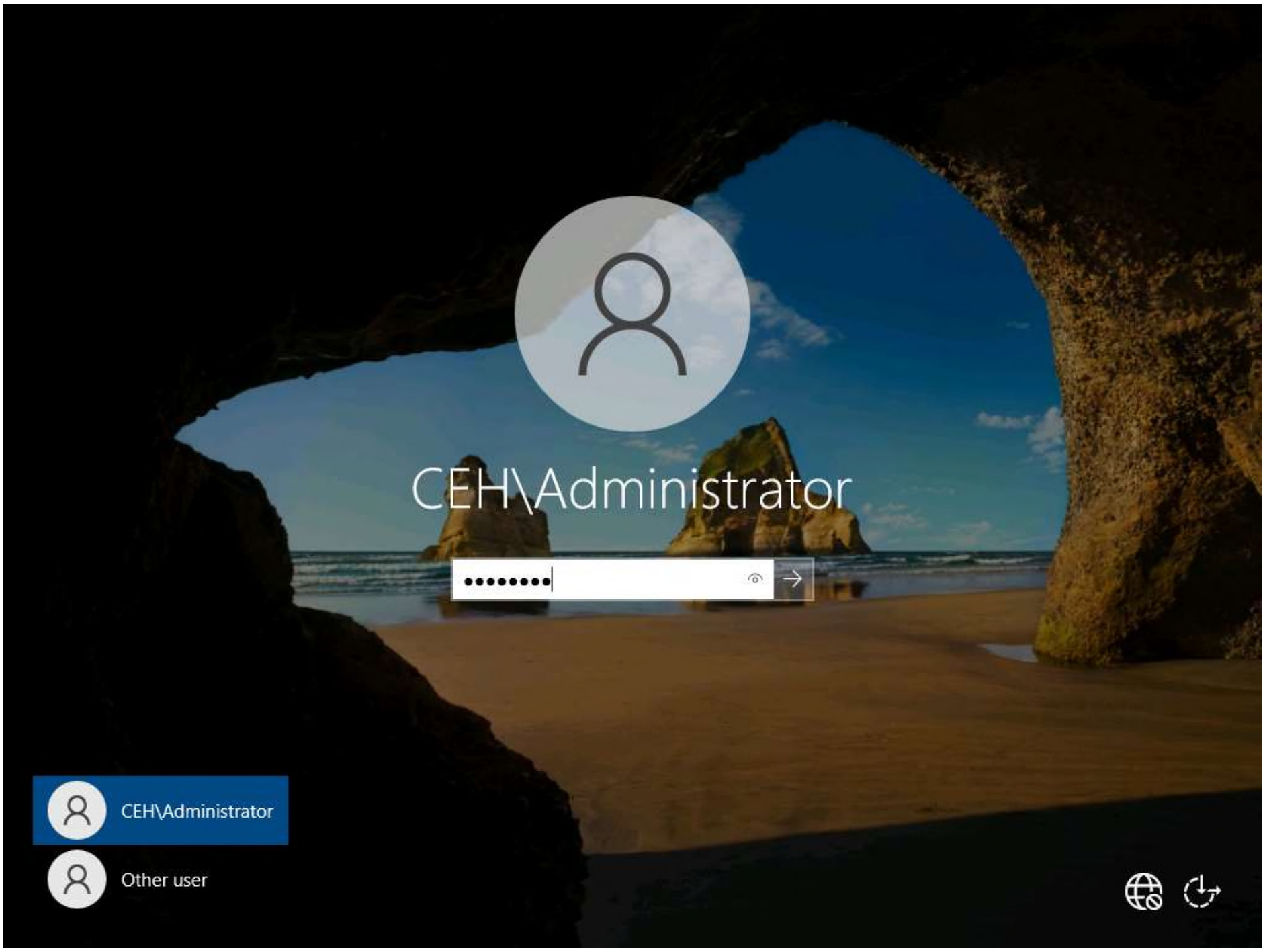


> Note: You can use any of these services and their open ports to enter into the target network/host and establish a connection.

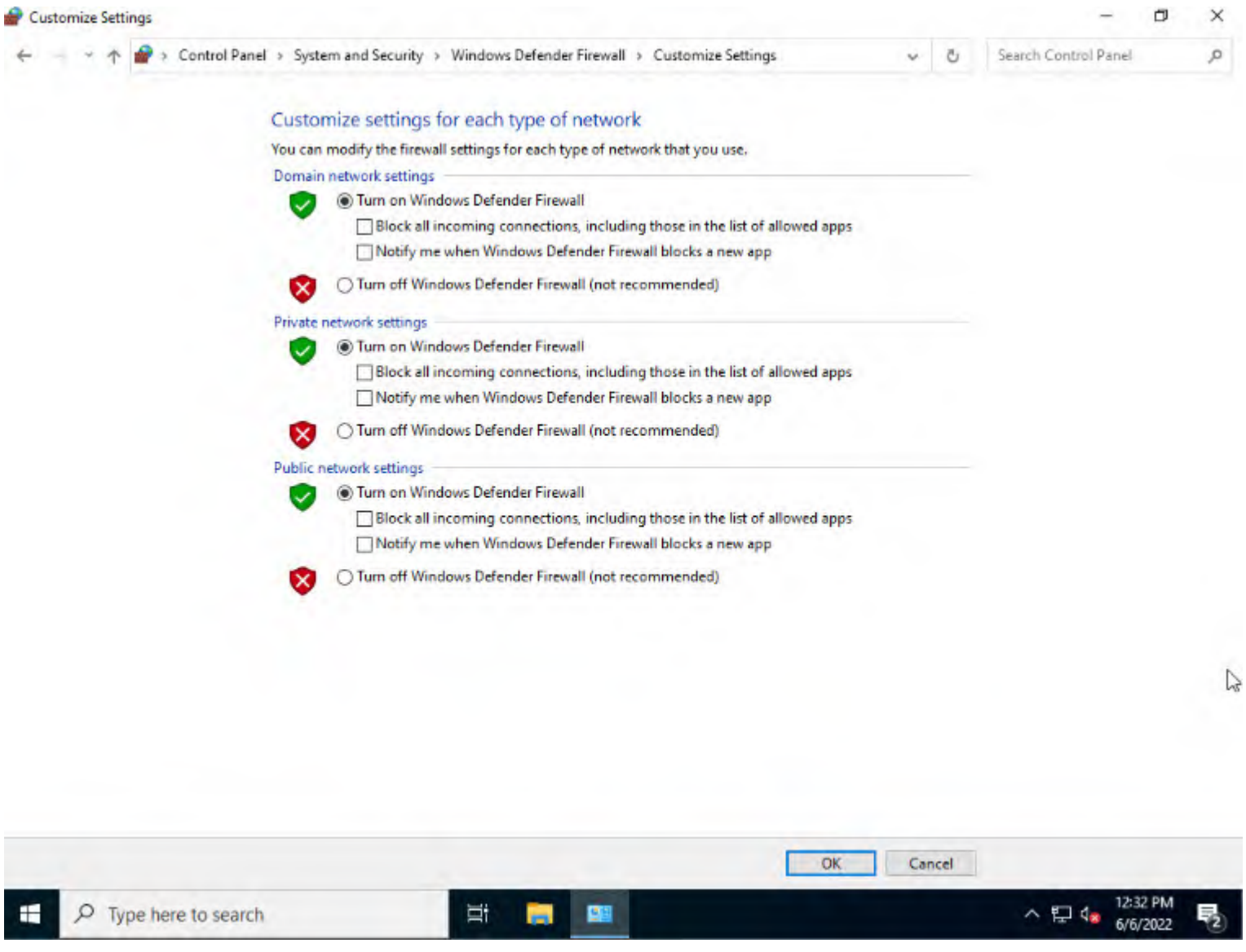
9. In this sub-task, we shall be performing a stealth scan/TCP half-open scan, Xmas scan, TCP Maimon scan, and ACK flag probe scan on a firewall-enabled machine (i.e., **Windows Server 2022**) in order to observe the result. To do this, we need to enable **Windows Firewall** in the **Windows Server 2022** machine.

10. Click **CEHv12 Windows Server 2022** to switch to the **Windows Server 2022** machine.

11. Click **Ctrl+Alt+Del** to activate the machine. By default, **CEH\Administrator** user profile is selected, type **Pa$$w0rd** in the **Password** field and press **Enter** to login.



12. Navigate to **Control Panel** --> **System and Security** --> **Windows Defender Firewall** --> **Turn Windows Defender Firewall on or off**, enable Windows Firewall and click **OK**, as shown in the screenshot.

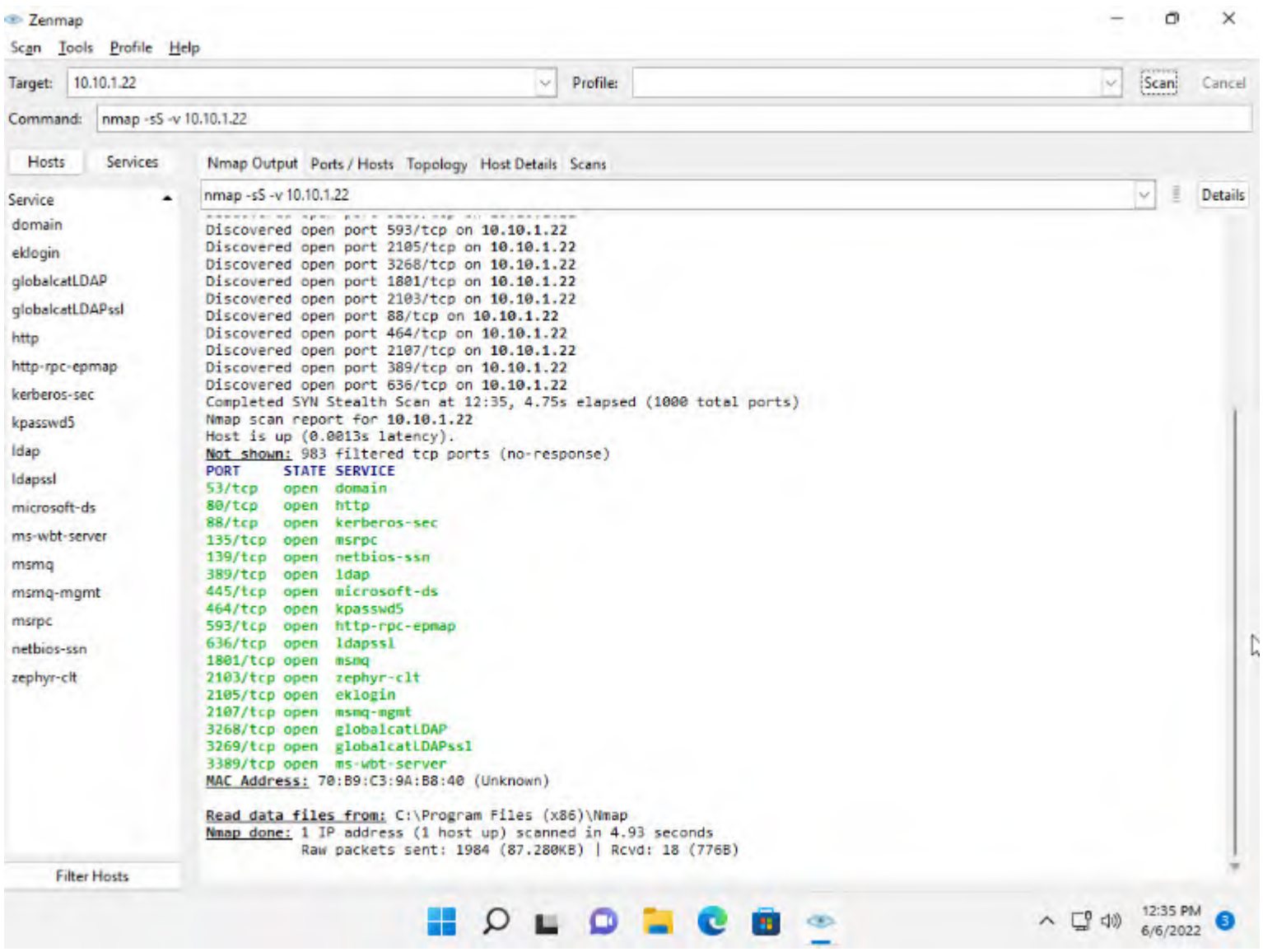


13. Now, click **CEHv12 Windows 11** switch to the **Windows 11** machine. In the **Command** field of **Zenmap**, type the command **nmap -sS -v [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

Note: **-sS**: performs the stealth scan/TCP half-open scan and **-v**: enables the verbose output (include all hosts and ports in the output).

14. The scan results appear, displaying all open TCP ports and services running on the target machine, as shown in the screenshot.

Note: The stealth scan involves resetting the TCP connection between the client and server abruptly before completion of three-way handshake signals, and hence leaving the connection half-open. This scanning technique can be used to bypass firewall rules, logging mechanisms, and hide under network traffic.



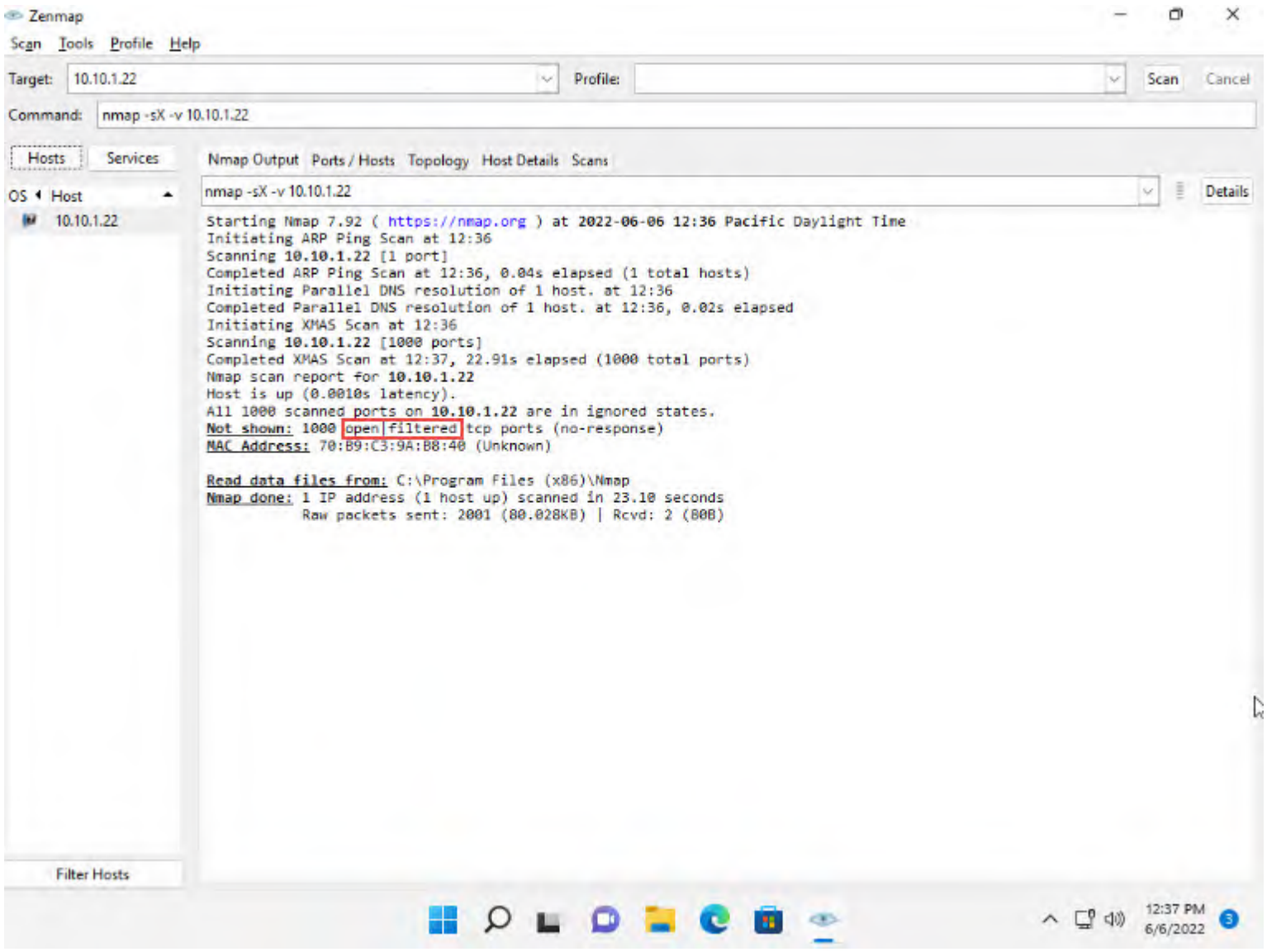
15. As shown in the last task, you can gather detailed information from the scan result in the **Ports/Hosts**, **Topology**, **Host Details**, and **Scan** tab.

16. In the **Command** field of **Zenmap**, type the command **nmap -sX -v [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

Note: **-sX**: performs the Xmas scan and **-v**: enables the verbose output (include all hosts and ports in the output).

17. The scan results appear, displaying that the ports are either open or filtered on the target machine, which means a firewall has been configured on the target machine.

Note: Xmas scan sends a TCP frame to a target system with FIN, URG, and PUSH flags set. If the target has opened the port, then you will receive no response from the target system. If the target has closed the port, then you will receive a target system reply with an RST.

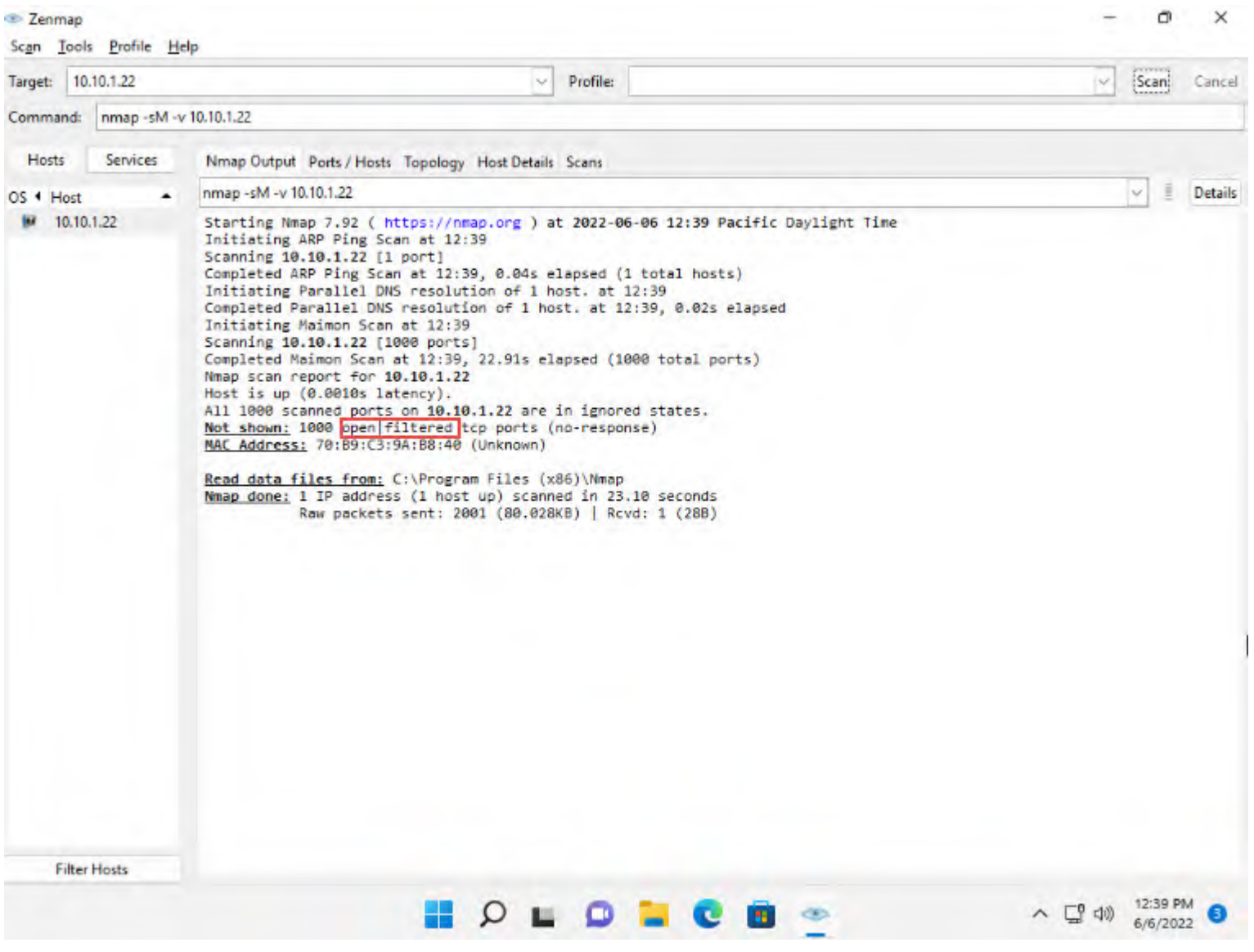


18. In the **Command** field, type the command **nmap -sM -v [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

Note: **-sM**: performs the TCP Maimon scan and **-v**: enables the verbose output (include all hosts and ports in the output).

19. The scan results appear, displaying either the ports are open/filtered on the target machine, which means a firewall has been configured on the target machine.

Note: In the TCP Maimon scan, a FIN/ACK probe is sent to the target; if there is no response, then the port is Open|Filtered, but if the RST packet is sent as a response, then the port is closed.

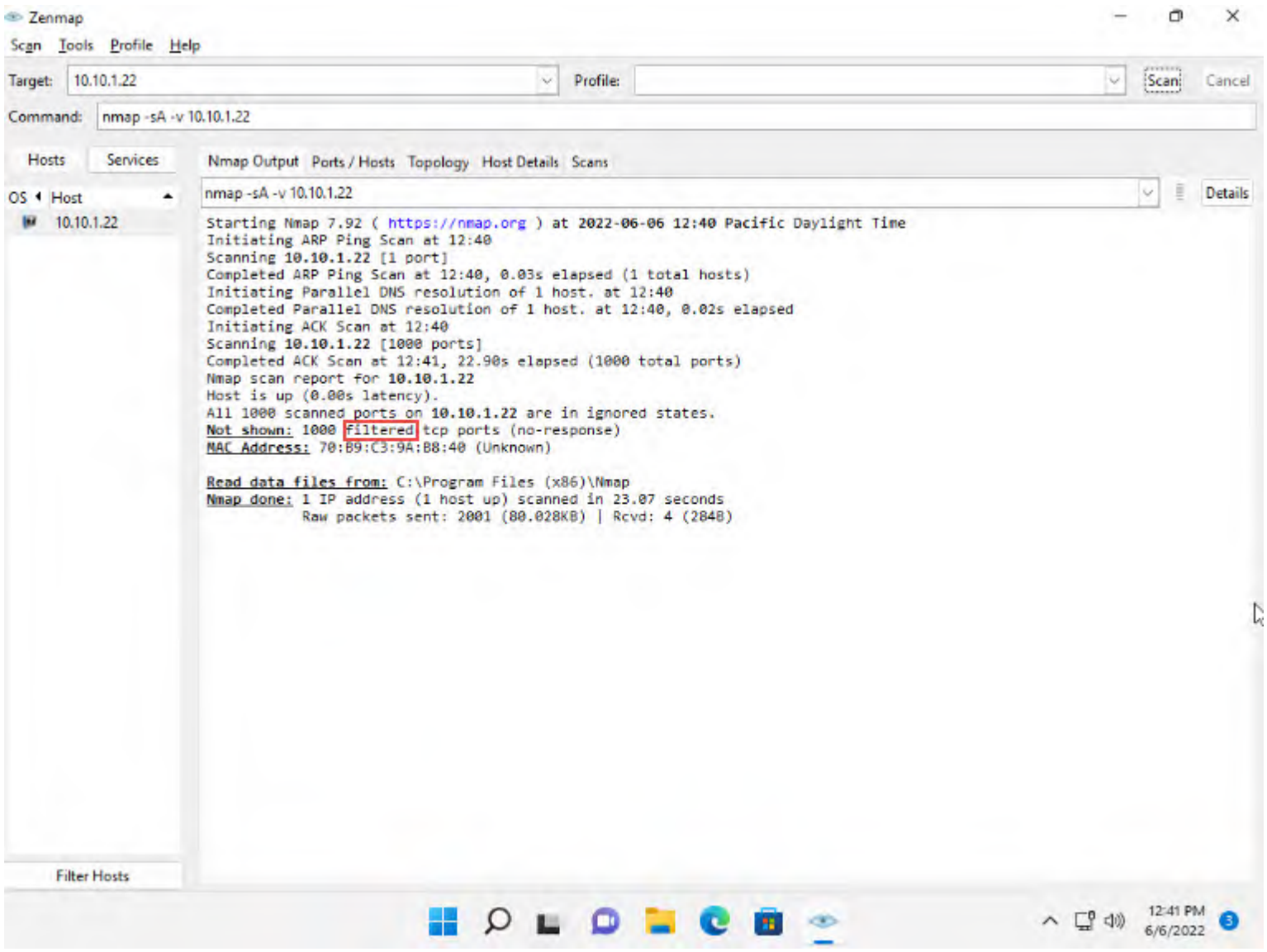


20. In the **Command** field, type the command **nmap -sA -v [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

Note: **-sA**: performs the ACK flag probe scan and **-v**: enables the verbose output (include all hosts and ports in the output).

21. The scan results appear, displaying that the ports are filtered on the target machine, as shown in the screenshot.

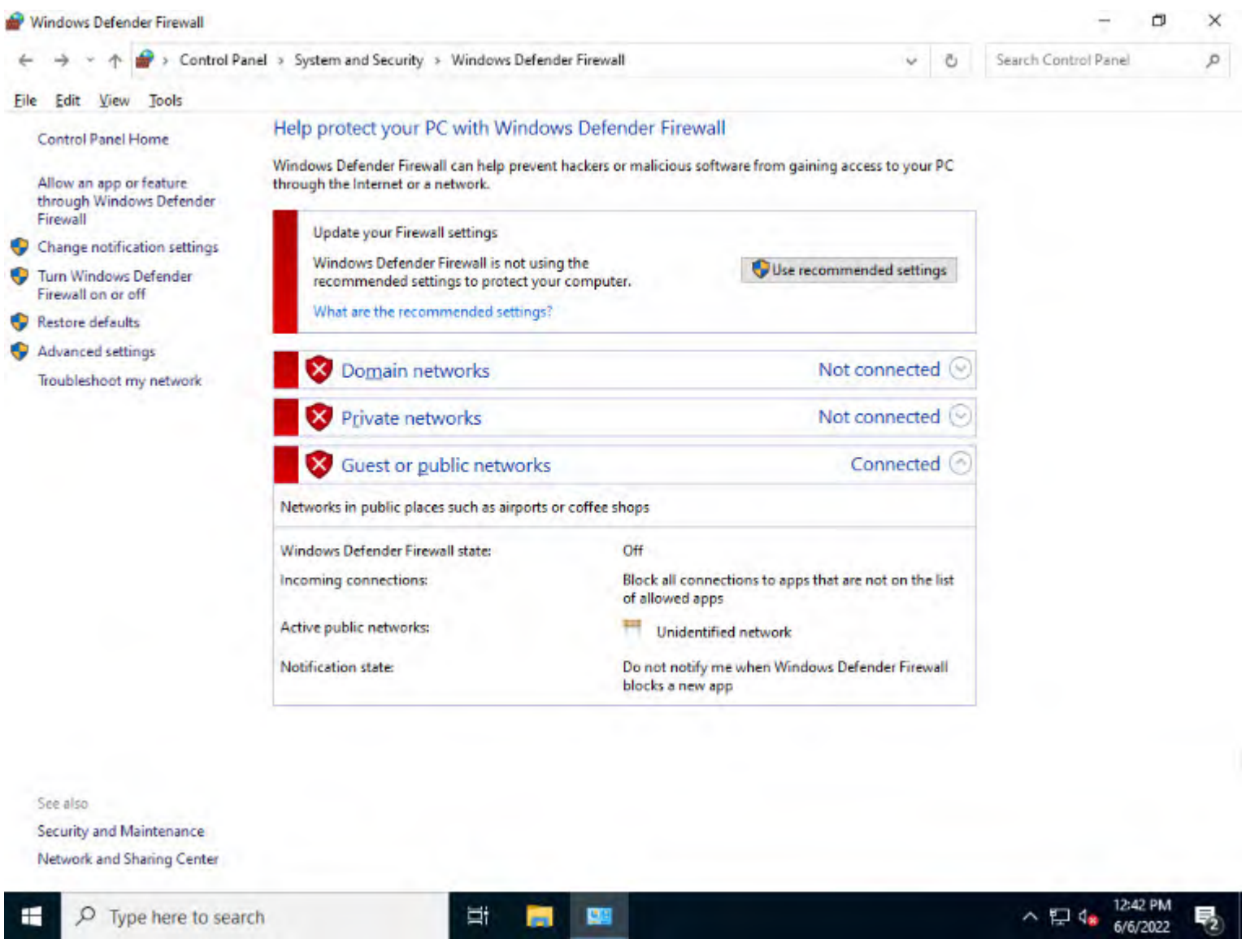
Note: The ACK flag probe scan sends an ACK probe packet with a random sequence number; no response implies that the port is filtered (stateful firewall is present), and an RST response means that the port is not filtered.



22. Now, click **CEHv12 Windows Server 2022** to switch to the **Windows Server 2022** machine.

23. If you are logged out of the **Windows Server 2022** machine, then click **Ctrl+Alt+Del** to activate the machine. By default, **CEH\Administrator** user profile is selected, type **Pa$$w0rd** in the **Password** field and press **Enter** to login.

24. Turn off the **Windows Defender Firewall** from **Control Panel**.

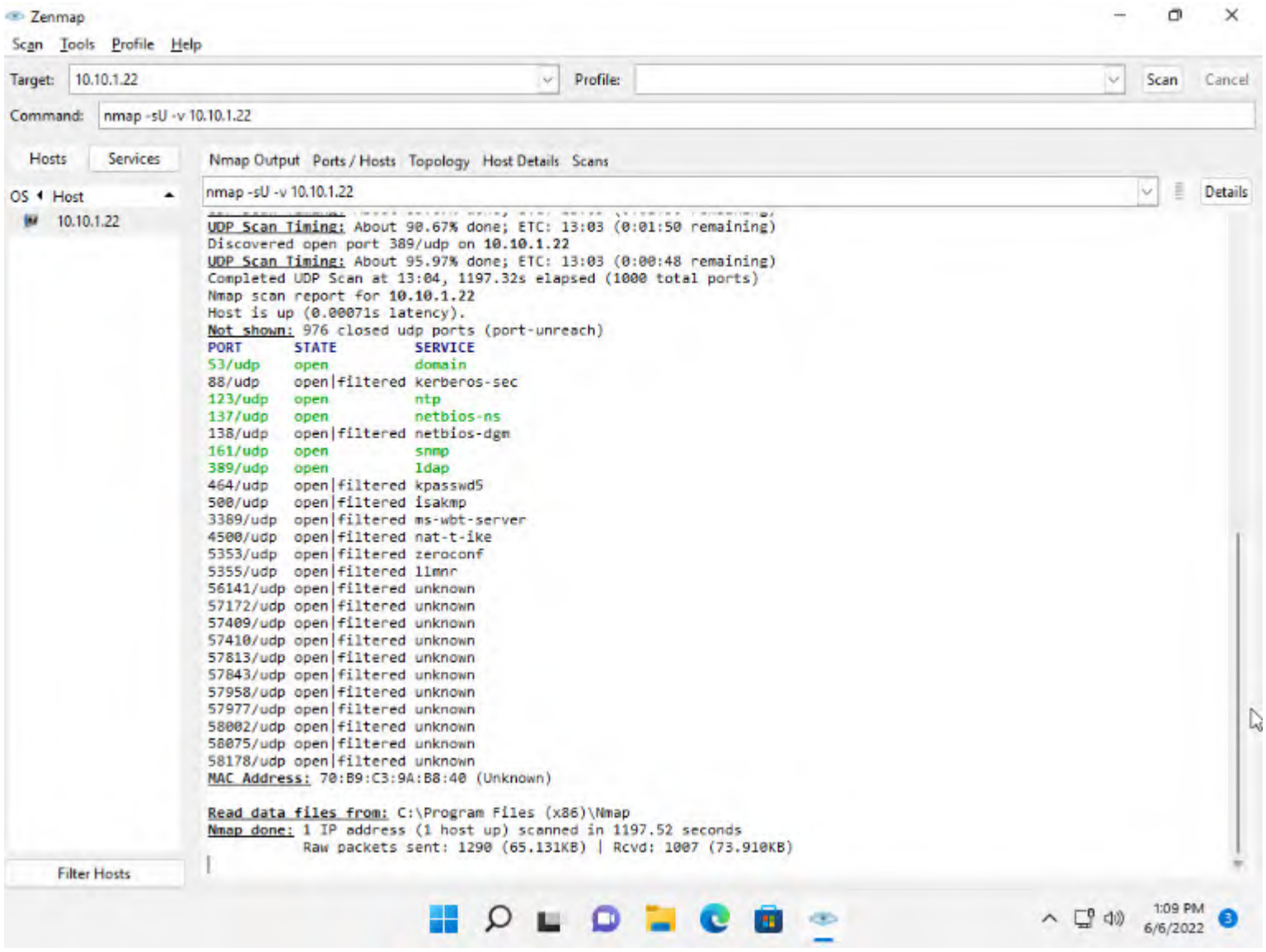


25. Now, click **CEHv12 Windows 11** to navigate back to the **Windows 11** machine. In the **Command** field of **Zenmap**, type the command **nmap -sU -v [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

> Note: **-sU**: performs the UDP scan and **-v**: enables the verbose output (include all hosts and ports in the output).

26. The scan results appear, displaying all open UDP ports and services running on the target machine, as shown in the screenshot.

> Note: This scan will take approximately 20 minutes to finish the scanning process and the results might differ in your lab environment.
>
> Note: The UDP scan uses UDP protocol instead of the TCP. There is no three-way handshake for the UDP scan. It sends UDP packets to the target host; no response means that the port is open. If the port is closed, an ICMP port unreachable message is received.

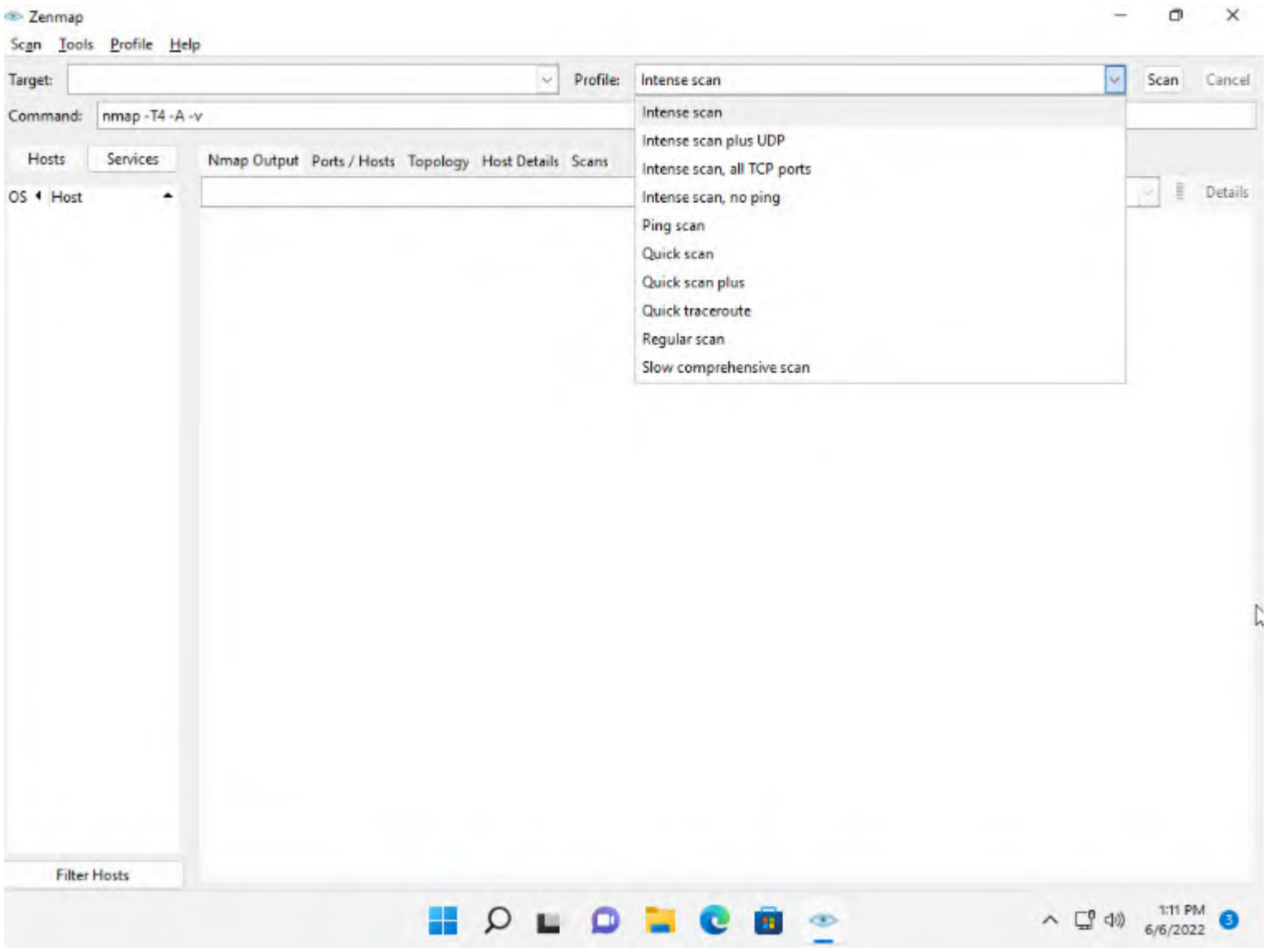


27. Close the **Zenmap** window.

28. You can create your scan profile, or you can also choose the default scan profiles available in Nmap to scan a network.

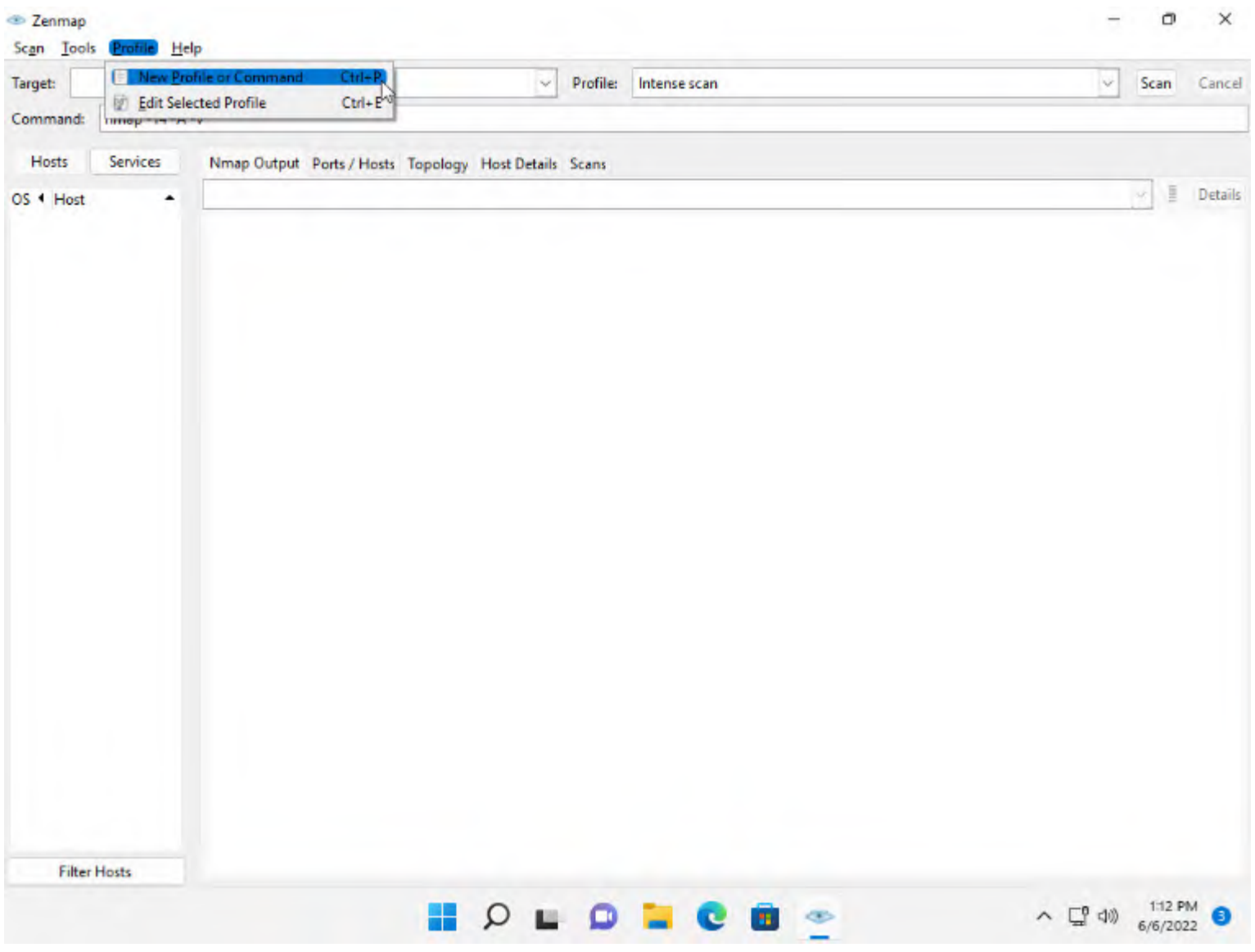
29. Click **Search** icon ( ) on the **Desktop**. Type **zenmap** in the search field, the **Nmap - Zenmap GUI** appears in the results, click **Open** to launch it.

30. To choose the default scan profiles available in Nmap, click on the drop-down icon in the **Profile** field and select the scanning technique you want to use.

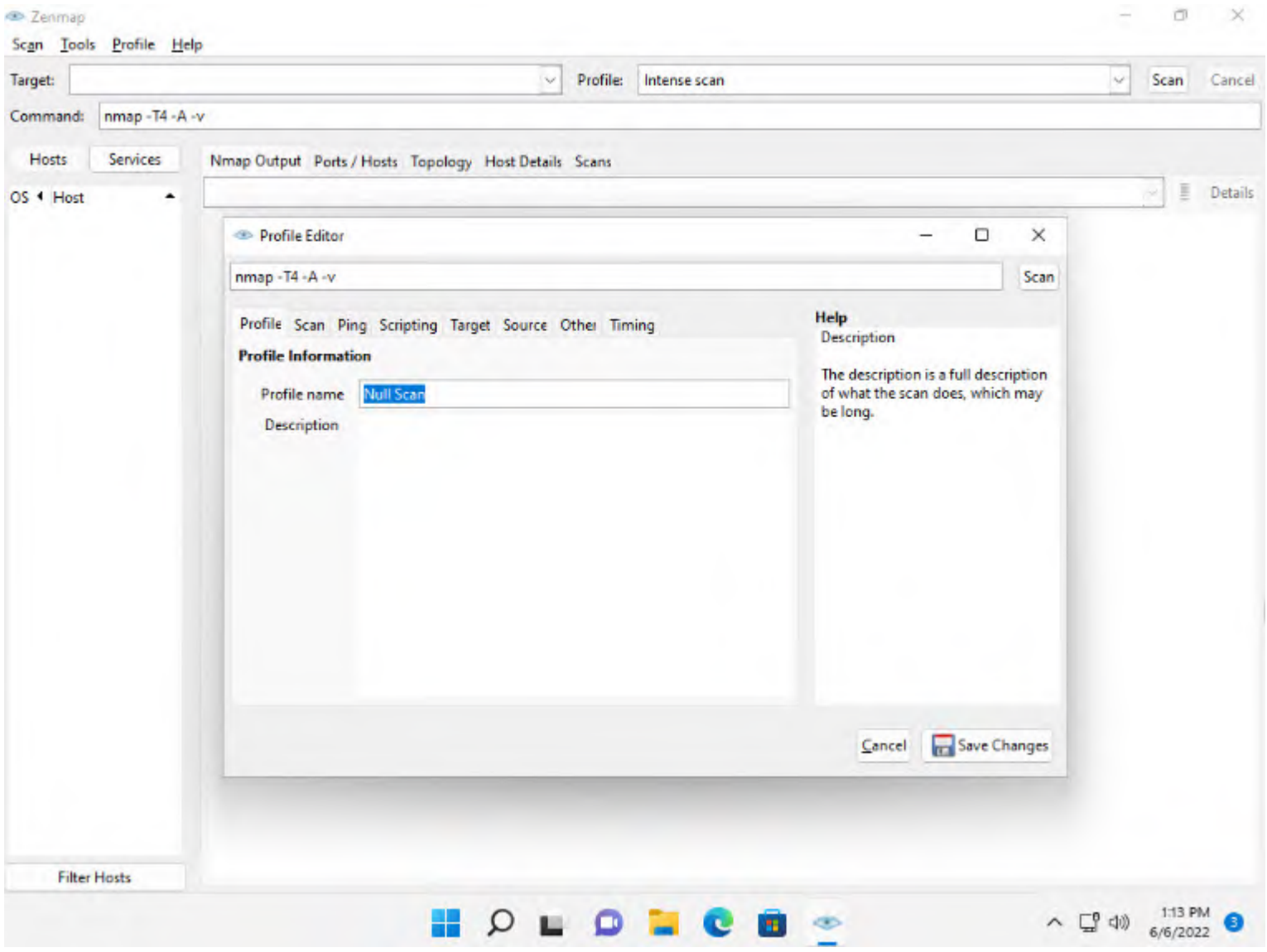


31. To create a scan profile; click **Profile** --> **New Profile or Command**.

Note: If a **User Account Control** pop-up appears, click **Yes**.



32. The **Profile Editor** window appears. In the **Profile** tab, under the **Profile Information** section, input a profile name (here, **Null Scan**) into the **Profile name** field.

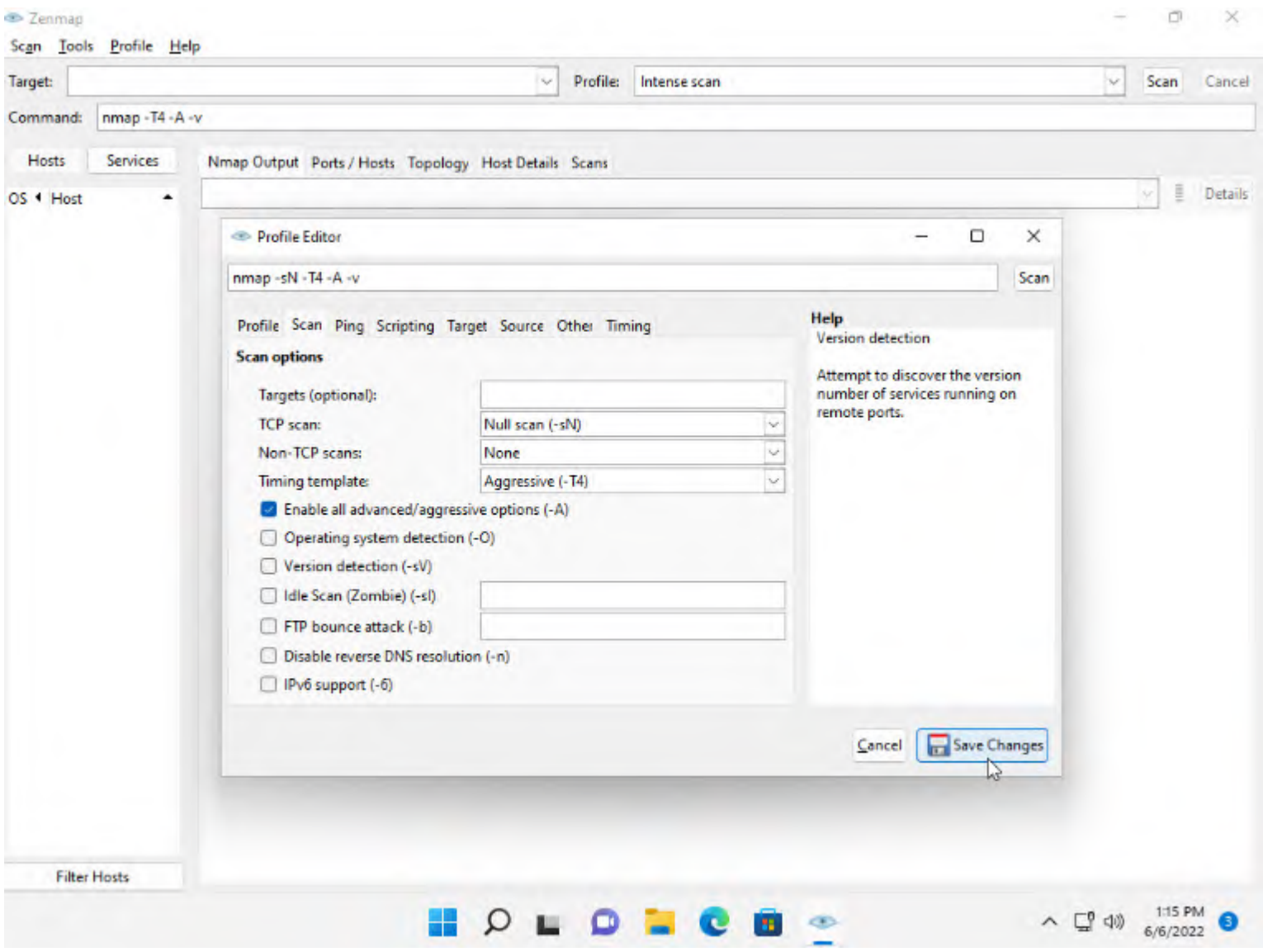


33. Now, click the **Scan** tab and select the scan option (here, **Null scan (-sN)**) from the **TCP scan** drop-down list.

34. Select **None** in the **Non-TCP scans** drop-down list and **Aggressive (-T4)** in the **Timing template** list. Ensure that the **Enable all advanced/aggressive options (-A)** checkbox is selected and click **Save Changes**, as shown in the screenshot.

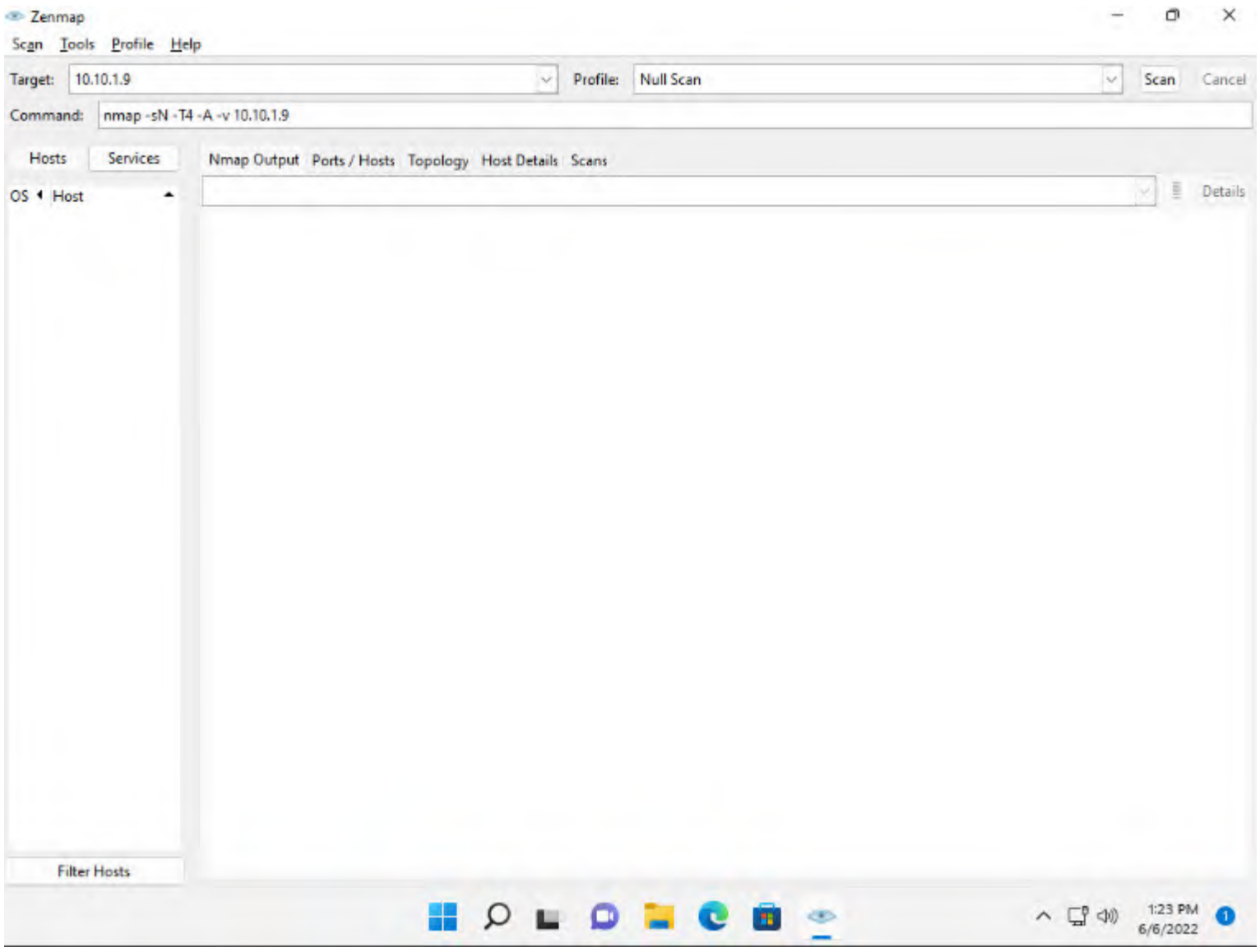
> Note: Using this configuration, you are setting Nmap to perform a null scan with the time template as **-T4** and all **aggressive** options enabled.

35. This will create a new profile, and will thus be added to the profile list.

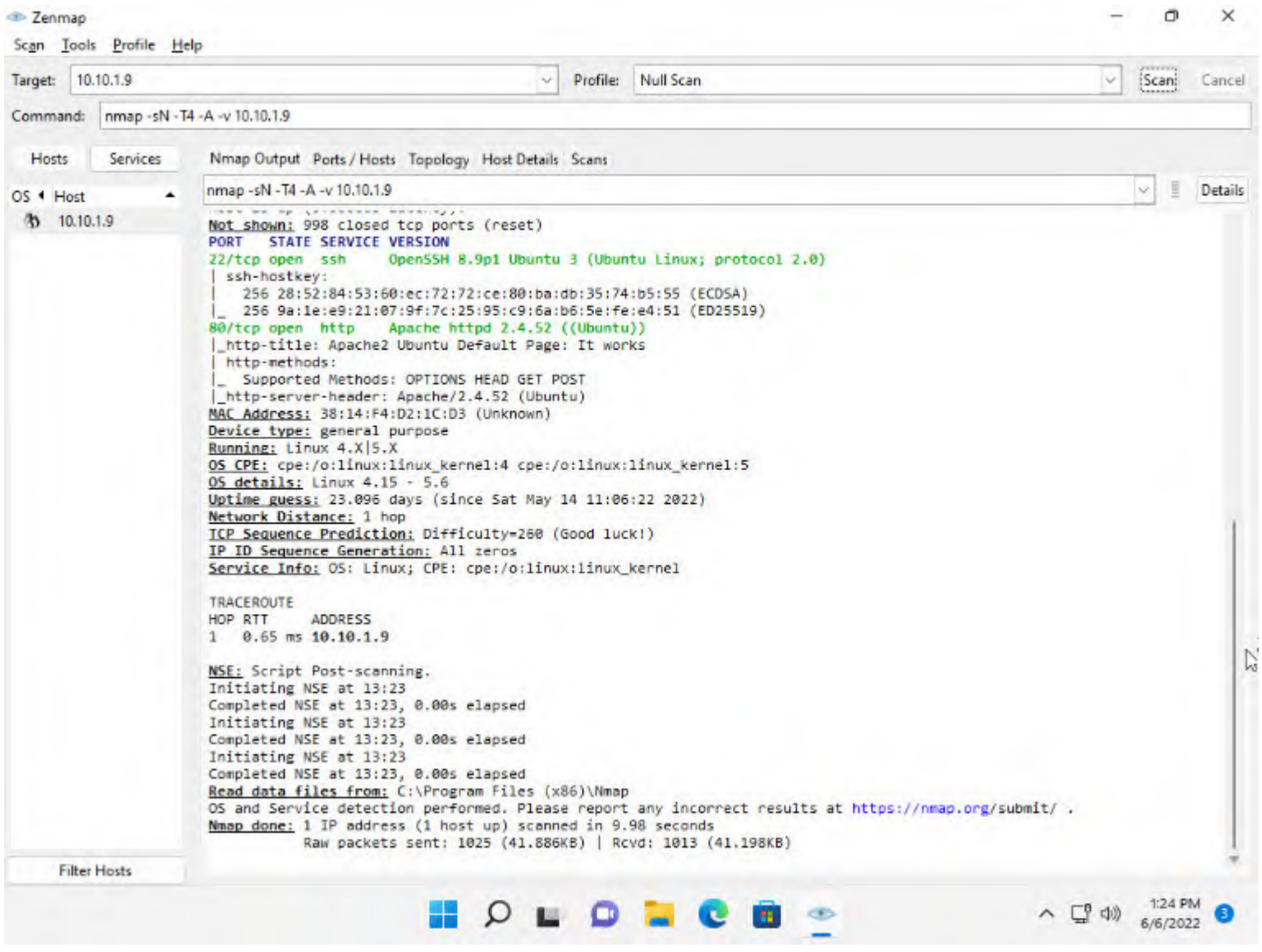


36. In this sub-task, we will be targeting the **Ubuntu** machine (**10.10.1.9**).

37. In the main window of **Zenmap**, enter the target IP address (here, **10.10.1.9**) in the **Target** field to scan. Select the **Null Scan** profile, which you created from the **Profile** drop-down list, and then click **Scan**.



38. Nmap scans the target and displays results in the **Nmap Output** tab, as shown in the screenshot.



39. Apart from the aforementioned port scanning and service discovery techniques, you can also use the following scanning techniques to perform a port and service discovery on a target network using Nmap.

    IDLE/IPID Header Scan: A TCP port scan method that can be used to send a spoofed source address to a computer to discover what services are available.

    # nmap -sI -v [target IP address]

    SCTP INIT Scan: An INIT chunk is sent to the target host; an INIT+ACK chunk response implies that the port is open, and an ABORT Chunk response means that the port is closed.

    # nmap -sY -v [target IP address]

SCTP COOKIE ECHO Scan: A COOKIE ECHO chunk is sent to the target host; no response implies that the port is open and ABORT Chunk response means that the port is closed.

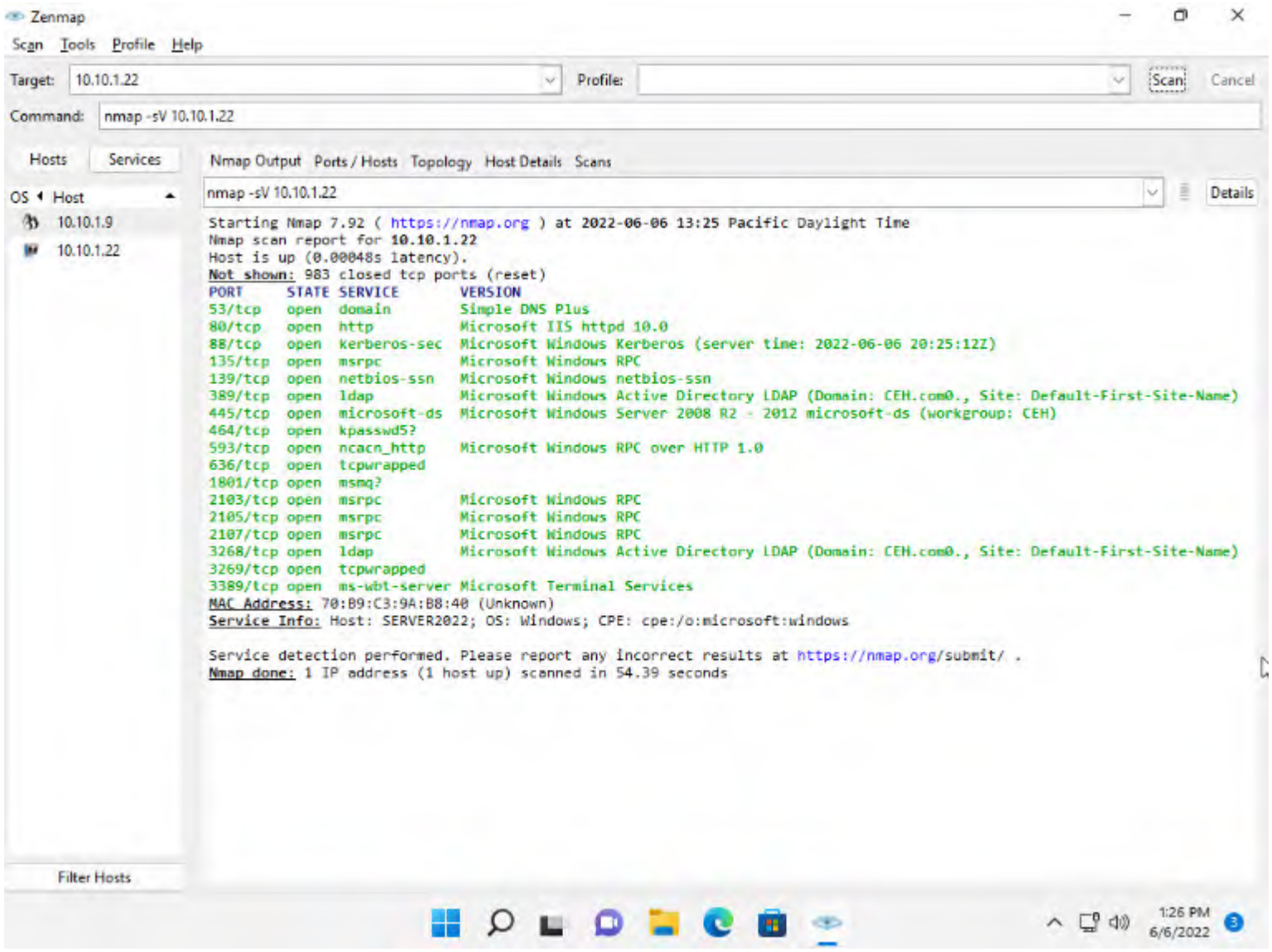
 # nmap -sZ -v [target IP address]

40. In the **Command** field, type the command **nmap -sV [Target IP Address]** (here, the target IP address is **10.10.1.22**) and click **Scan**.

Note: **-sV**: detects service versions.

41. The scan results appear, displaying that open ports and the version of services running on the ports, as shown in the screenshot.

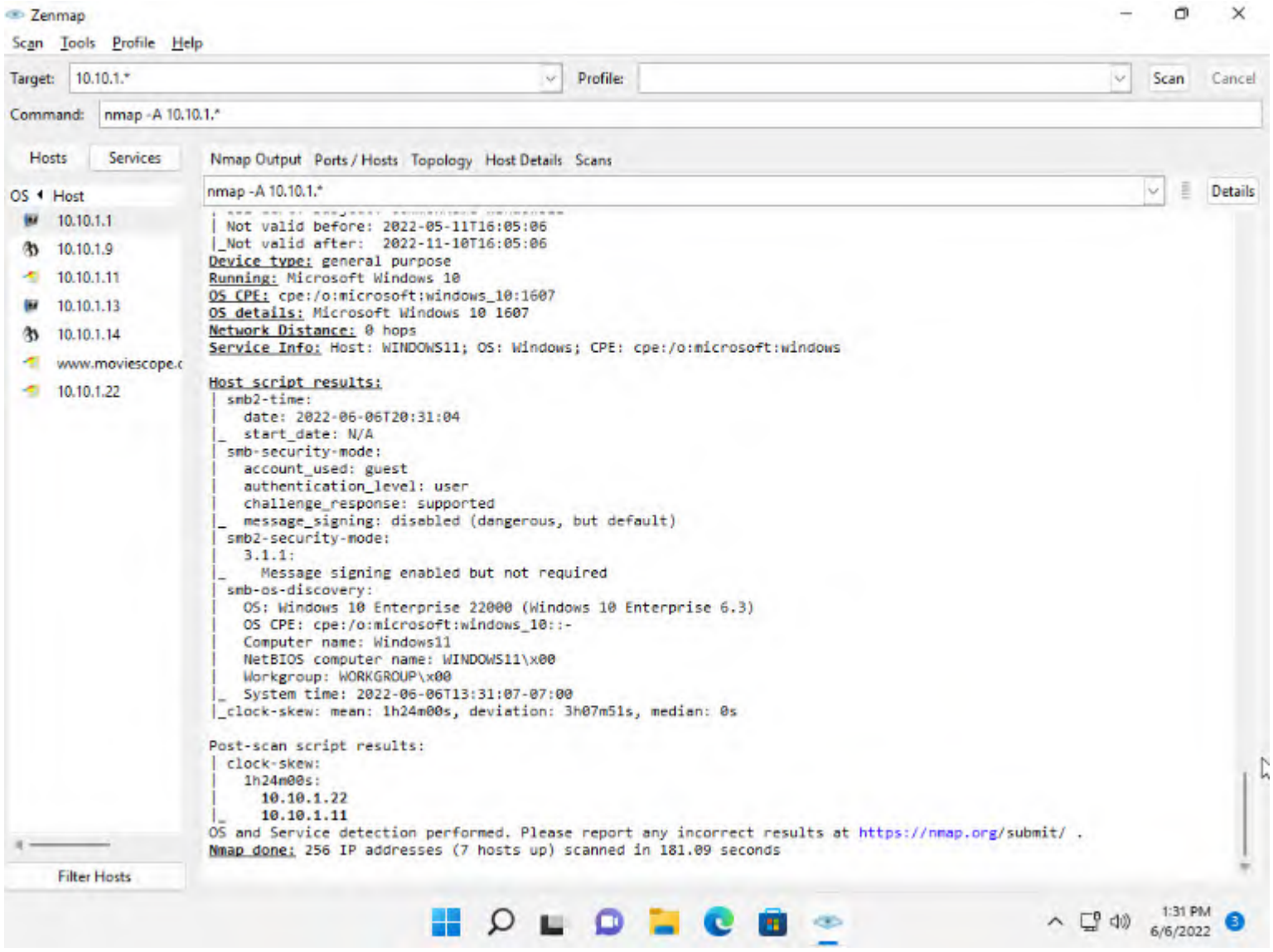
Note: Service version detection helps you to obtain information about the running services and their versions on a target system. Obtaining an accurate service version number allows you to determine which exploits the target system is vulnerable to.



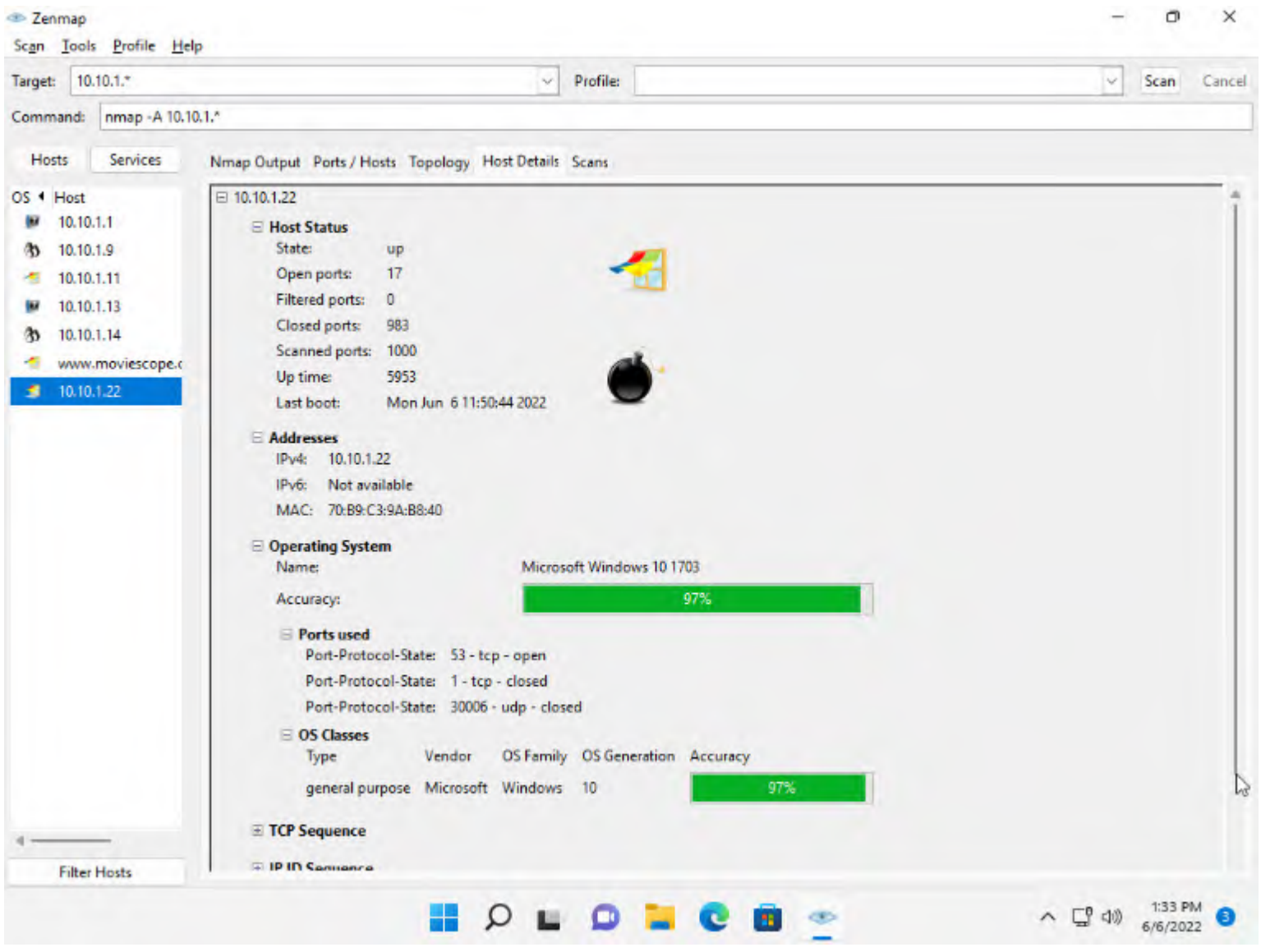
42. In the **Command** field, type the command **nmap -A [Target Subnet]** (here, target subnet is **10.10.1. *) and click **Scan**. By providing the "" (asterisk) wildcard, you can scan a whole subnet or IP range.

Note: **-A**: enables aggressive scan. The aggressive scan option supports OS detection (-O), version scanning (-sV), script scanning (-sC), and traceroute (--traceroute). You should not use -A against target networks without permission.

43. Nmap scans the entire network and displays information for all the hosts that were scanned, along with the open ports and services, device type, details of OS, etc. as shown in the screenshot.



44. Choose an IP address **10.10.1.22** from the list of hosts in the left-pane and click the **Host Details** tab. This tab displays information such as **Host Status**, **Addresses**, **Operating System**, **Ports used**, **OS Classes**, etc. associated with the selected host.



45. This concludes the demonstration of discovering target open ports, services, services versions, device type, OS details, etc. of the active hosts in the target network using various scanning techniques of Nmap.

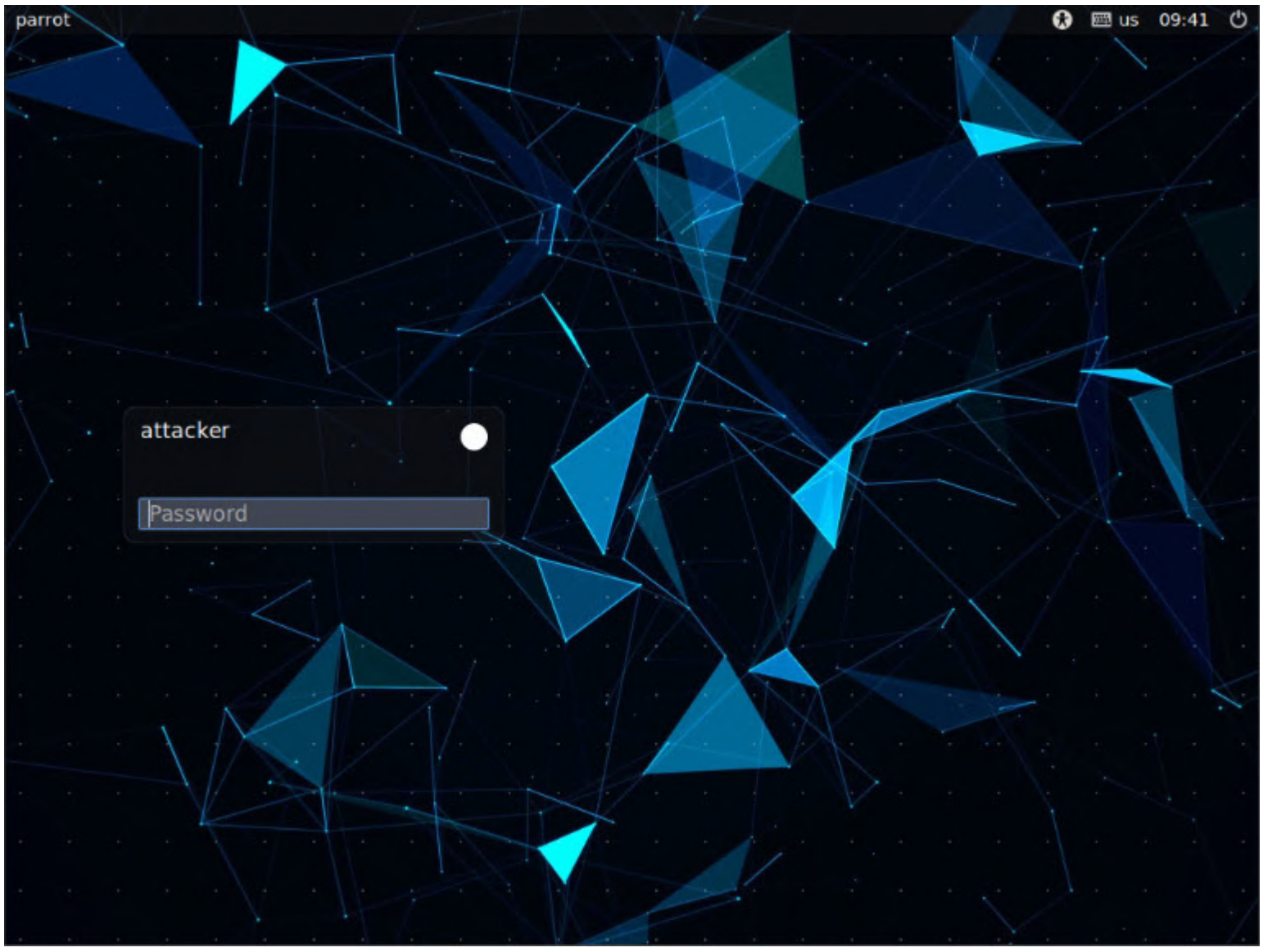
46. Close all open windows and document all the acquired information.

Task 5: Explore Various Network Scanning Techniques using Hping3

Hping2/Hping3 is a command-line-oriented network scanning and packet crafting tool for the TCP/IP protocol that sends ICMP echo requests and supports TCP, UDP, ICMP, and raw-IP protocols. Using Hping, you can study the behavior of an idle host and gain information about the target such as the services that the host offers, the ports supporting the services, and the OS of the target.

Here, we will use Hping3 to discover open ports and services running on the live hosts in the target network.

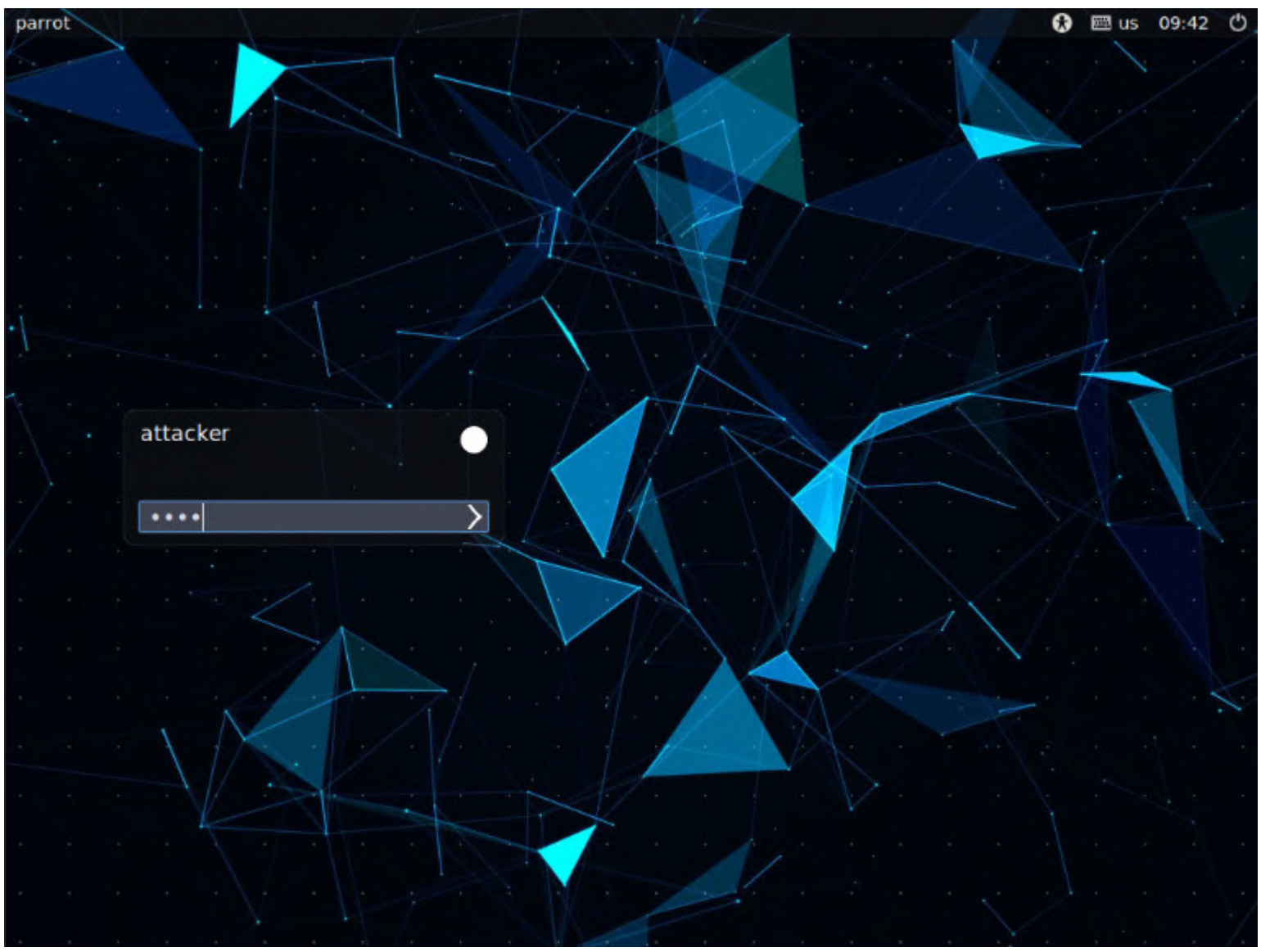
1. To launch **Parrot Security** machine, click **CEHv12 Parrot Security**.



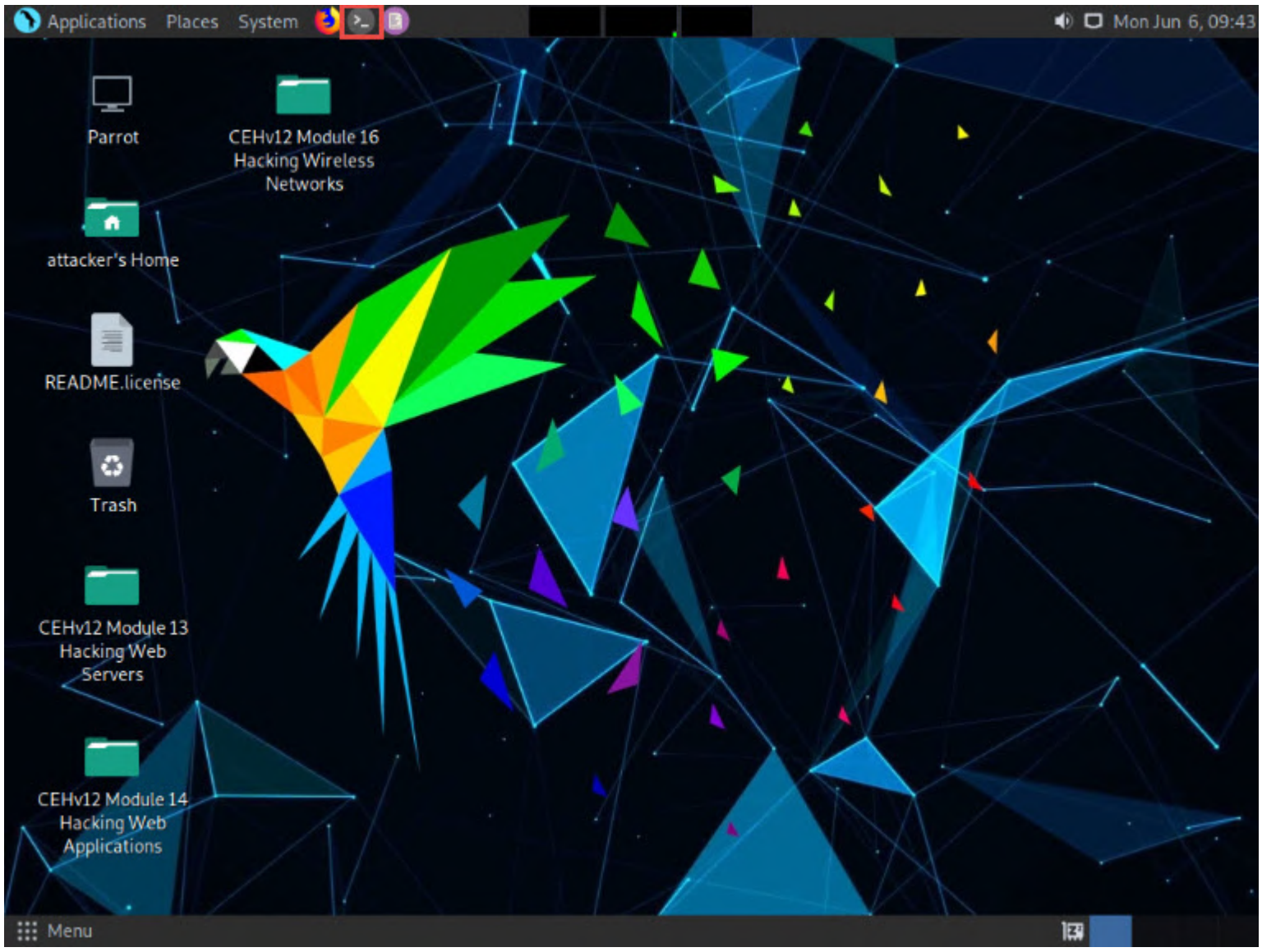
2. In the login page, the **attacker** username will be selected by default. Enter password as **toor** in the **Password** field and press **Enter** to log in to the machine.

Note: If a **Parrot Updater** pop-up appears at the top-right corner of **Desktop**, ignore and close it.

Note: If a **Question** pop-up window appears asking you to update the machine, click **No** to close the window.



3. Click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.

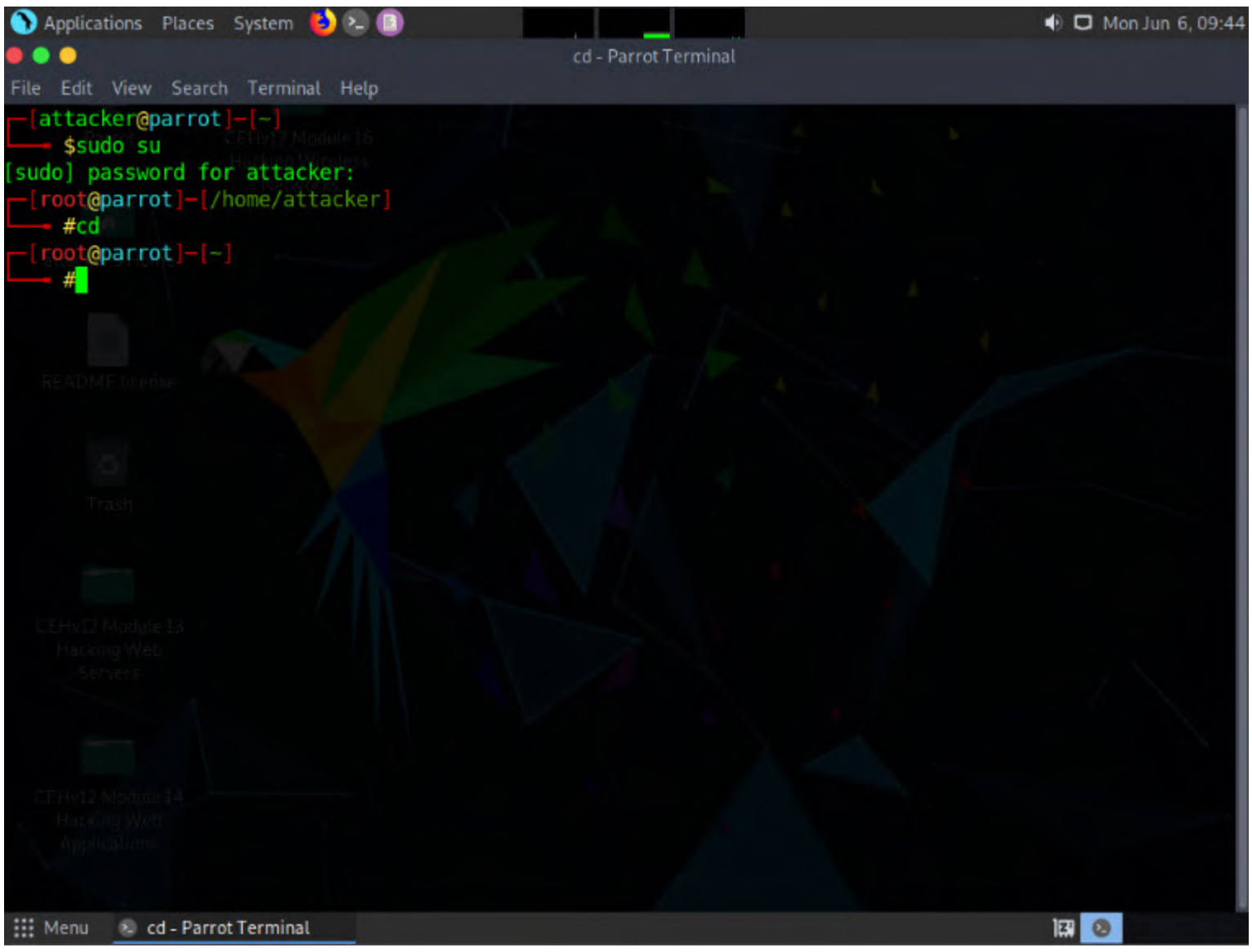


4. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

5. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

> Note: The password that you type will not be visible.

6. Now, type **cd** and press **Enter** to jump to the root directory.

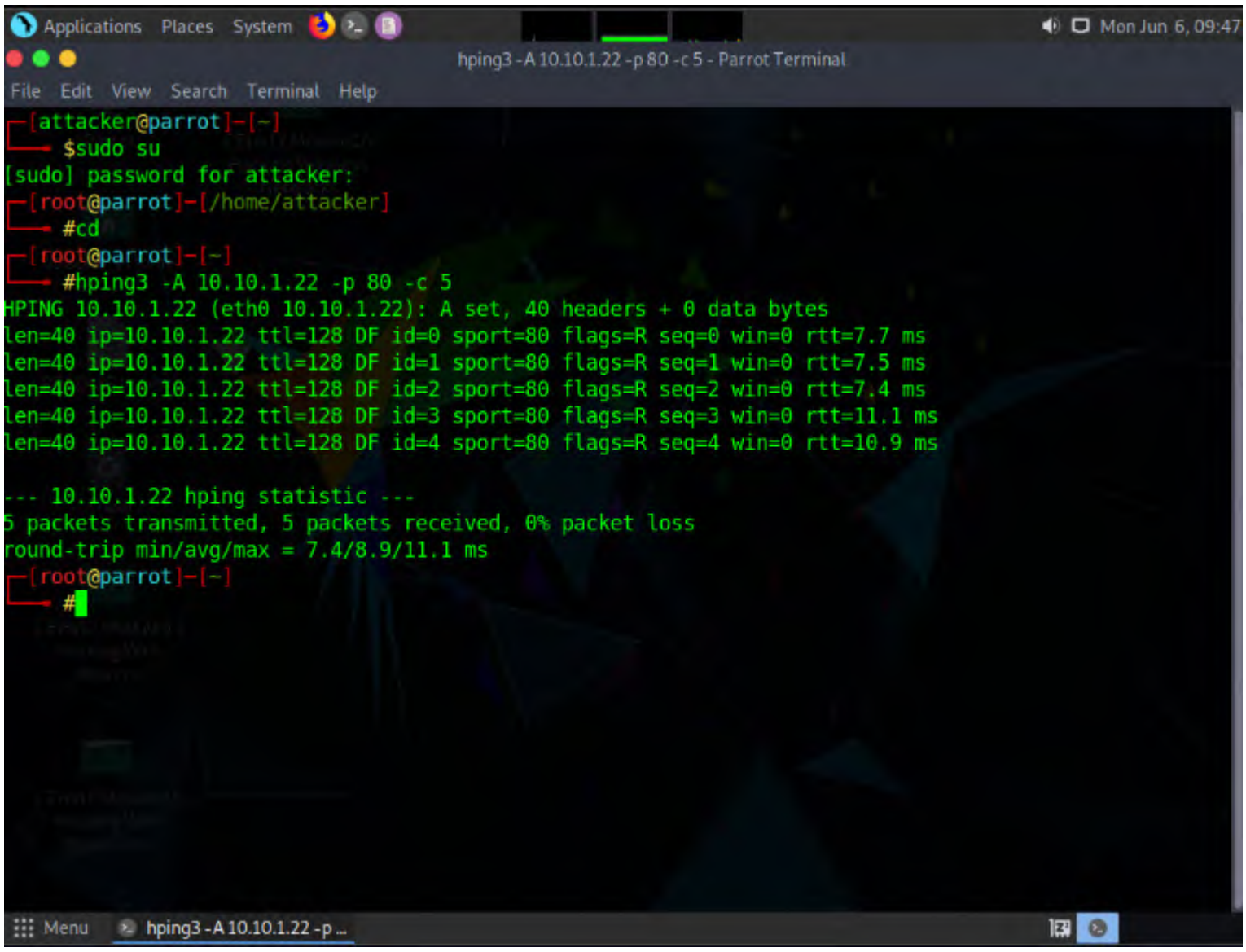


7. A **Parrot Terminal** window appears. In the terminal window, type **hping3 -A [Target IP Address] -p 80 -c 5** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

> Note: In this command, **-A** specifies setting the ACK flag, **-p** specifies the port to be scanned (here, **80**), and **-c** specifies the packet count (here, **5**).

8. In a result, the number of packets sent and received is equal, indicating that the respective port is open, as shown in the screensho

Note: The ACK scan sends an ACK probe packet to the target host; no response means that the port is filtered. If an RST response returns, this means that the port is closed.

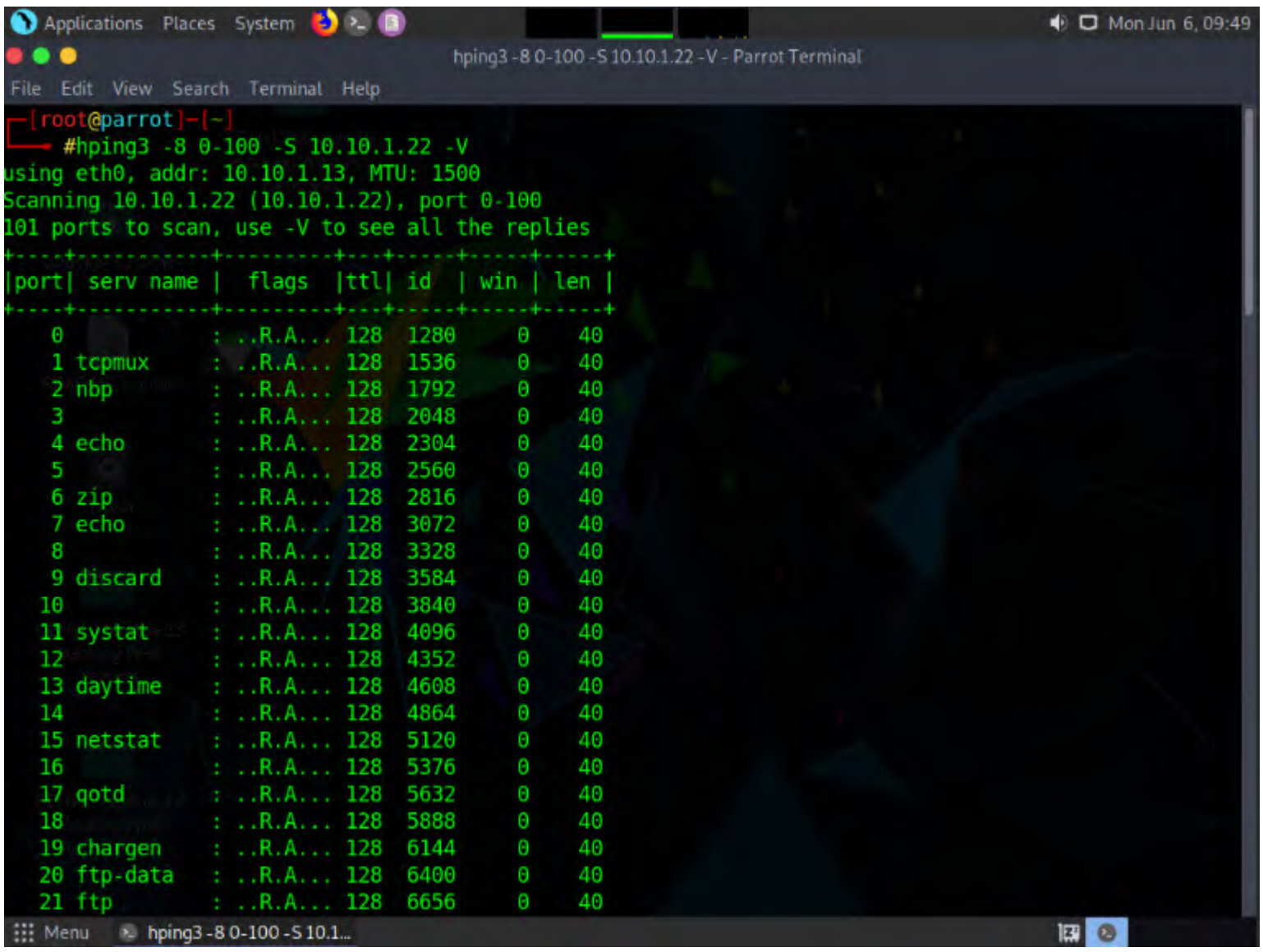


9. In the terminal window, type **hping3 -8 0-100 -S [Target IP Address] -V** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

Note: In this command, **-8** specifies a scan mode, **-p** specifies the range of ports to be scanned (here, **0-100**), and **-V** specifies the verbose mode.

10. The result appears, displaying the open ports along with the name of service running on each open port, as shown in the screenshot.

Note: The SYN scan principally deals with three of the flags: SYN, ACK, and RST. You can use these three flags for gathering illegal information from servers during the enumeration process.

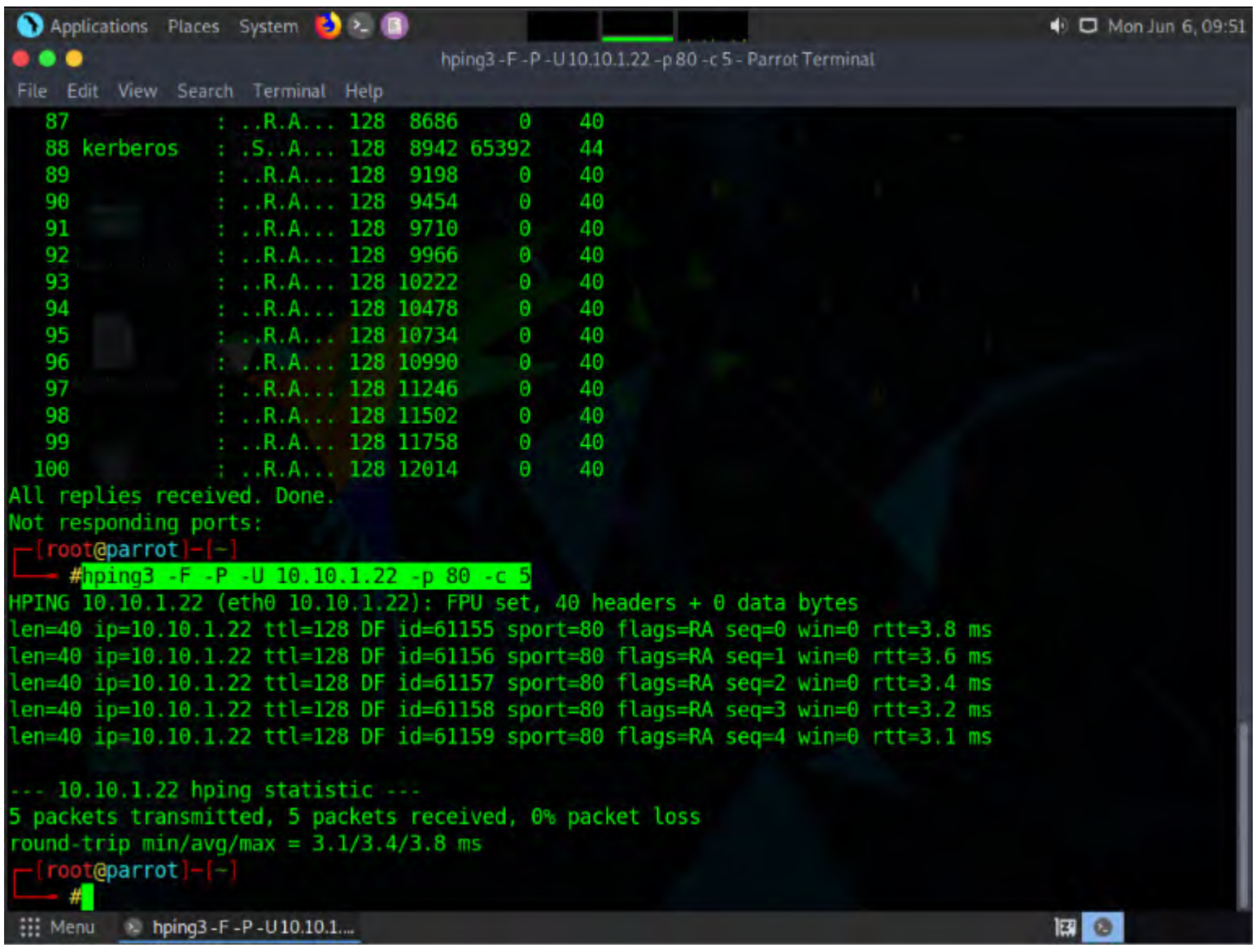


11. In the **terminal** window, type **hping3 -F -P -U [Target IP Address] -p 80 -c 5** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

> Note: In this command, **-F** specifies setting the FIN flag, **-P** specifies setting the PUSH flag, **-U** specifies setting the URG flag, **-c** specifies the packet count (here, **5**), and **-p** specifies the port to be scanned (here, **80**).

12. The results demonstrate that the number of packets sent and received is equal, thereby indicating that the respective port is open, as shown in the screenshot.

> Note: FIN, PUSH, and URG scan the port on the target IP address. If a port is open on the target, you will receive a response. If the port is closed, Hping will return an RST response.

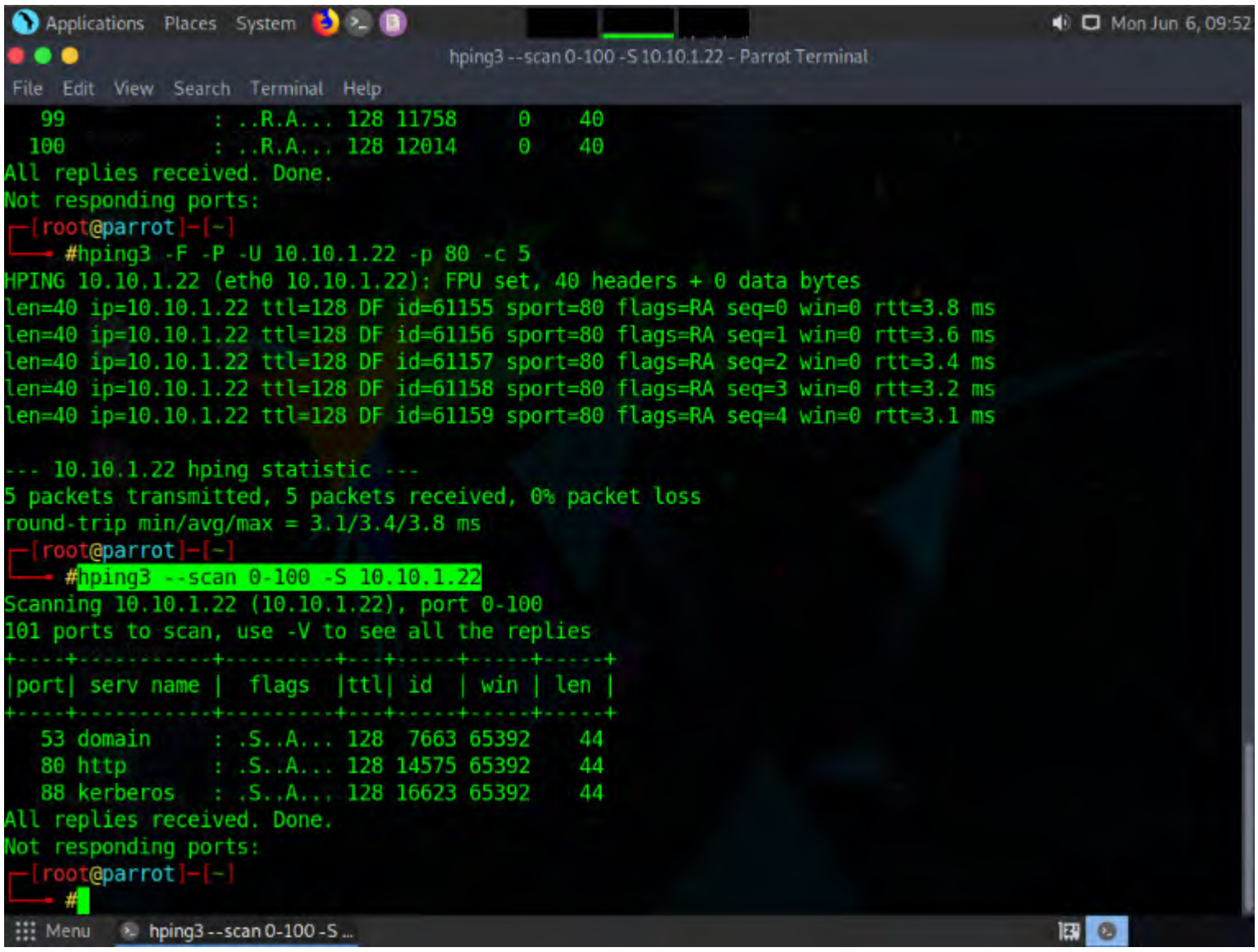


13. In the **terminal** window, type **hping3 --scan 0-100 -S [Target IP Address]** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

> Note: In this command, **--scan** specifies the port range to scan, **0-100** specifies the range of ports to be scanned, and **-S** specifies setting the SYN flag.

14. The result appears displaying the open ports and names of the services running on the target IP address, as shown in the screenshot.

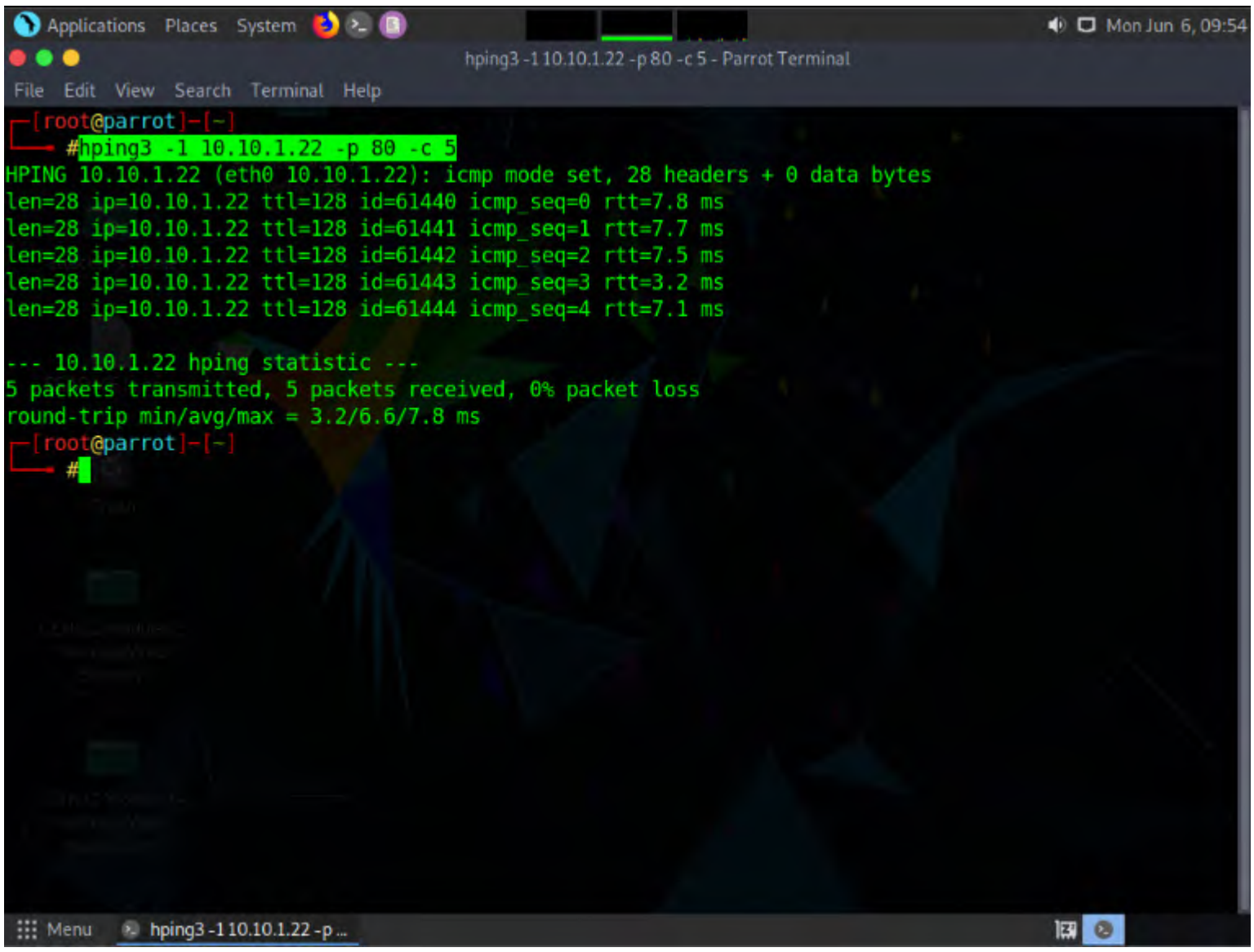
> Note: In the TCP stealth scan, the TCP packets are sent to the target host; if a SYN+ACK response is received, it indicates that the ports are open.



15. In the **terminal** window, type **hping3 -1 [Target IP Address] -p 80 -c 5** to perform ICMP scan (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**

> Note: In this command, **-1** specifies ICMP ping scan, **-c** specifies the packet count (here, **5**), and **-p** specifies the port to be scanned (here, **80**).

16. The results demonstrate that hping has sent ICMP echo requests to 10.10.1.22 and received ICMP replies which determines that the host is up.



17. Apart from the aforementioned port scanning and service discovery techniques, you can also use the following scanning techniques to perform a port and service discovery on a target network using Hping3.

    Entire subnet scan for live host: **hping3 -1 [Target Subnet] --rand-dest -I eth0**

    UDP scan: **hping3 -2 [Target IP Address] -p 80 -c 5**

18. This concludes the demonstration of discovering open ports and services running on the live hosts in the target network using Hping3.

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

Lab 3: Perform OS Discovery

Lab Scenario

As a professional ethical hacker or a pen tester, the next step after discovering the open ports and services running on the target range of IP addresses is to perform OS discovery. Identifying the OS used on the target system allows you to assess the system's vulnerabilities and the exploits that might work on the system to perform additional attacks.

Lab Objectives

Identify the target system's OS with Time-to-Live (TTL) and TCP window sizes using Wireshark
Perform OS discovery using Nmap Script Engine (NSE)
Perform OS discovery using Unicornscan

Overview of OS Discovery/ Banner Grabbing

Banner grabbing, or OS fingerprinting, is a method used to determine the OS that is running on a remote target system.

There are two types of OS discovery or banner grabbing techniques:

Active Banner Grabbing Specially crafted packets are sent to the remote OS, and the responses are noted, which are then compared with a database to determine the OS. Responses from different OSes vary, because of differences in the TCP/IP stack implementation.

Passive Banner Grabbing This depends on the differential implementation of the stack and the various ways an OS responds to packets. Passive banner grabbing includes banner grabbing from error messages, sniffing the network traffic, and banner grabbing from page extensions.

Parameters such as TTL and TCP window size in the IP header of the first packet in a TCP session plays an important role in identifying the OS running on the target machine. The TTL field determines the maximum time a packet can remain in a network, and the TCP window size determines the length of the packet reported. These values differ for different OSes: you can refer to the following table to learn the TTL values and TCP window size associated with various OSes.

Operating System	Time To Live	TCP Window Size
Linux	64	5840
FreeBSD	64	65535
OpenBSD	255	16384
Windows	128	65,535 bytes to 1 Gigabyte
Cisco Routers	255	4128
Solaris	255	8760
AIX	255	16384

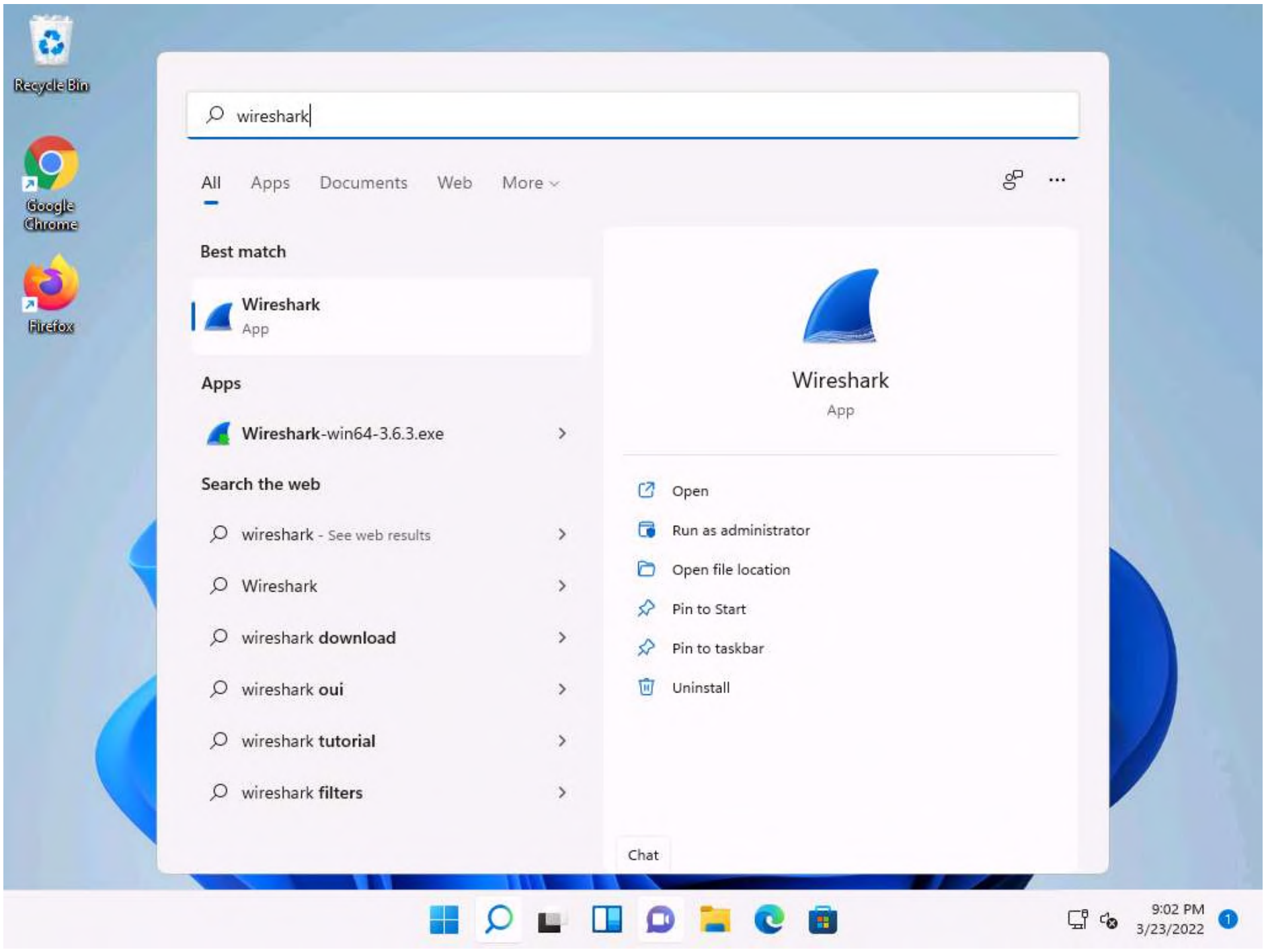
Task 1: Identify the Target System's OS with Time-to-Live (TTL) and TCP Window Sizes using Wireshark

Wireshark is a network protocol analyzer that allows capturing and interactively browsing the traffic running on a computer network. It is used to identify the target OS through sniffing/capturing the response generated from the target machine to the request-originated machine. Further, you can observe the TTL and TCP window size fields in the captured TCP packet. Using these values, the target OS can be determined.

Here, we will use the Wireshark tool to perform OS discovery on the target host(s).

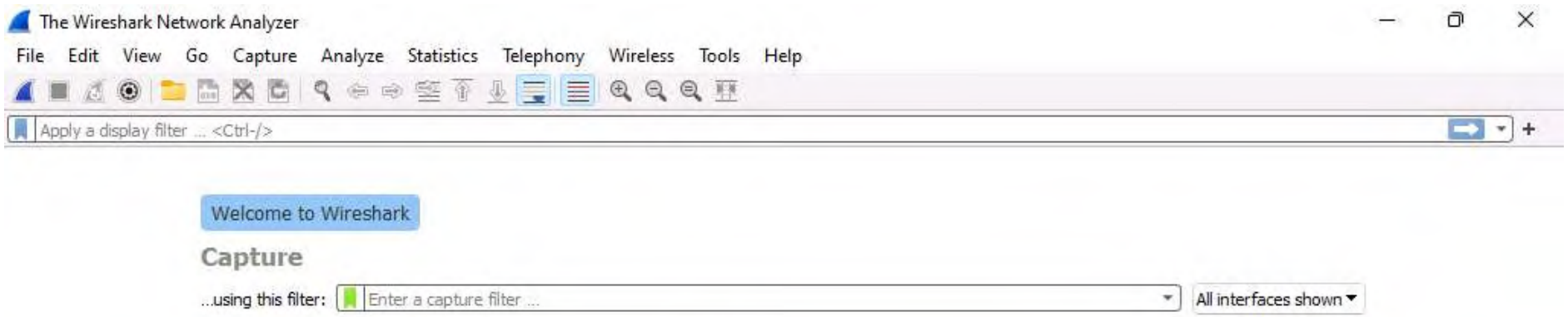
1. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine.

2. Click **Search** icon ( ) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Open** to launch it.



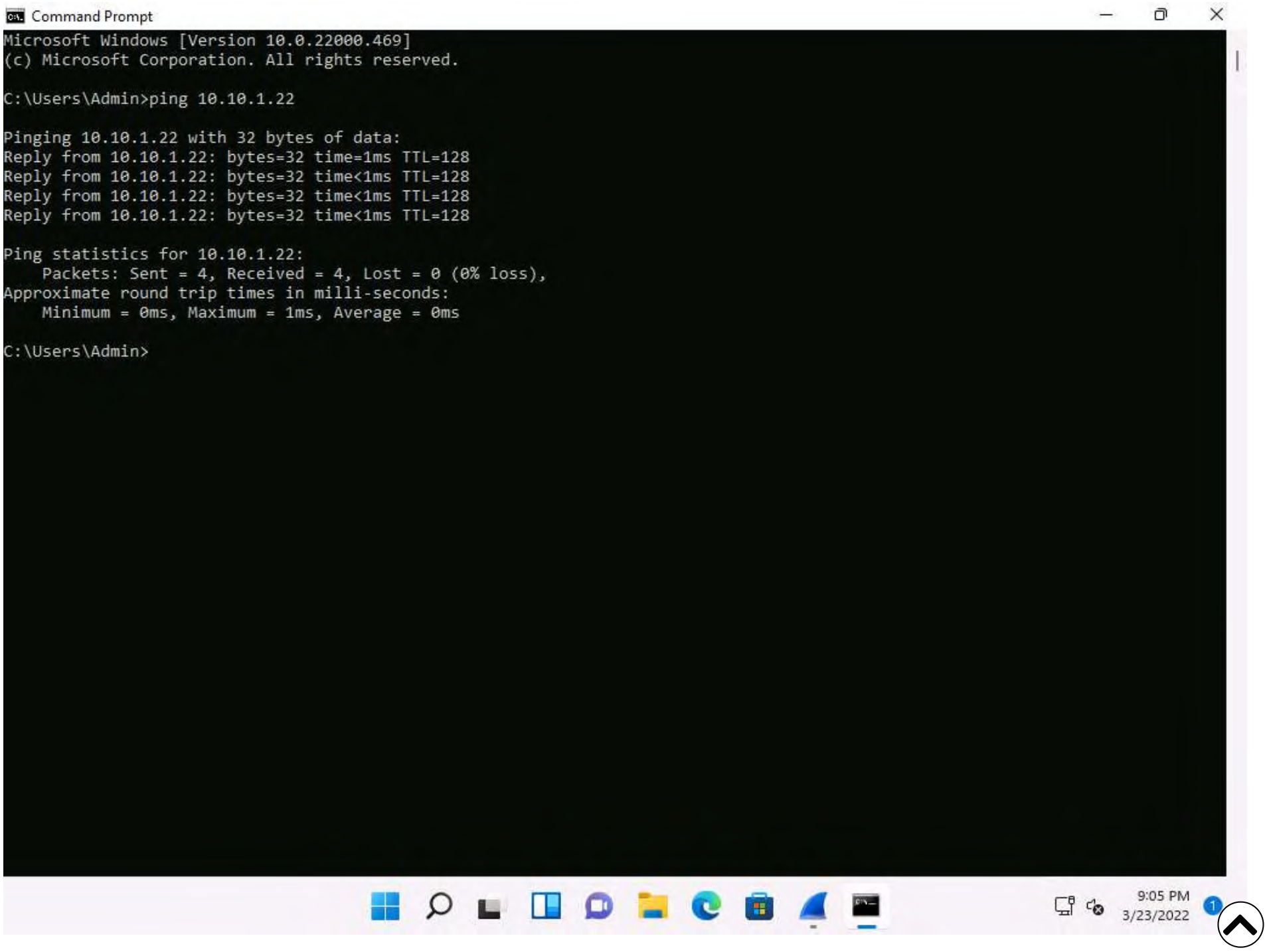
3. The **Wireshark Network Analyzer** main window appears; double-click the available ethernet or interface (here, **Ethernet**) to start the packet capture, as shown in the screenshot.

Note: If **Software Update** window appears, click **Remind me later**.

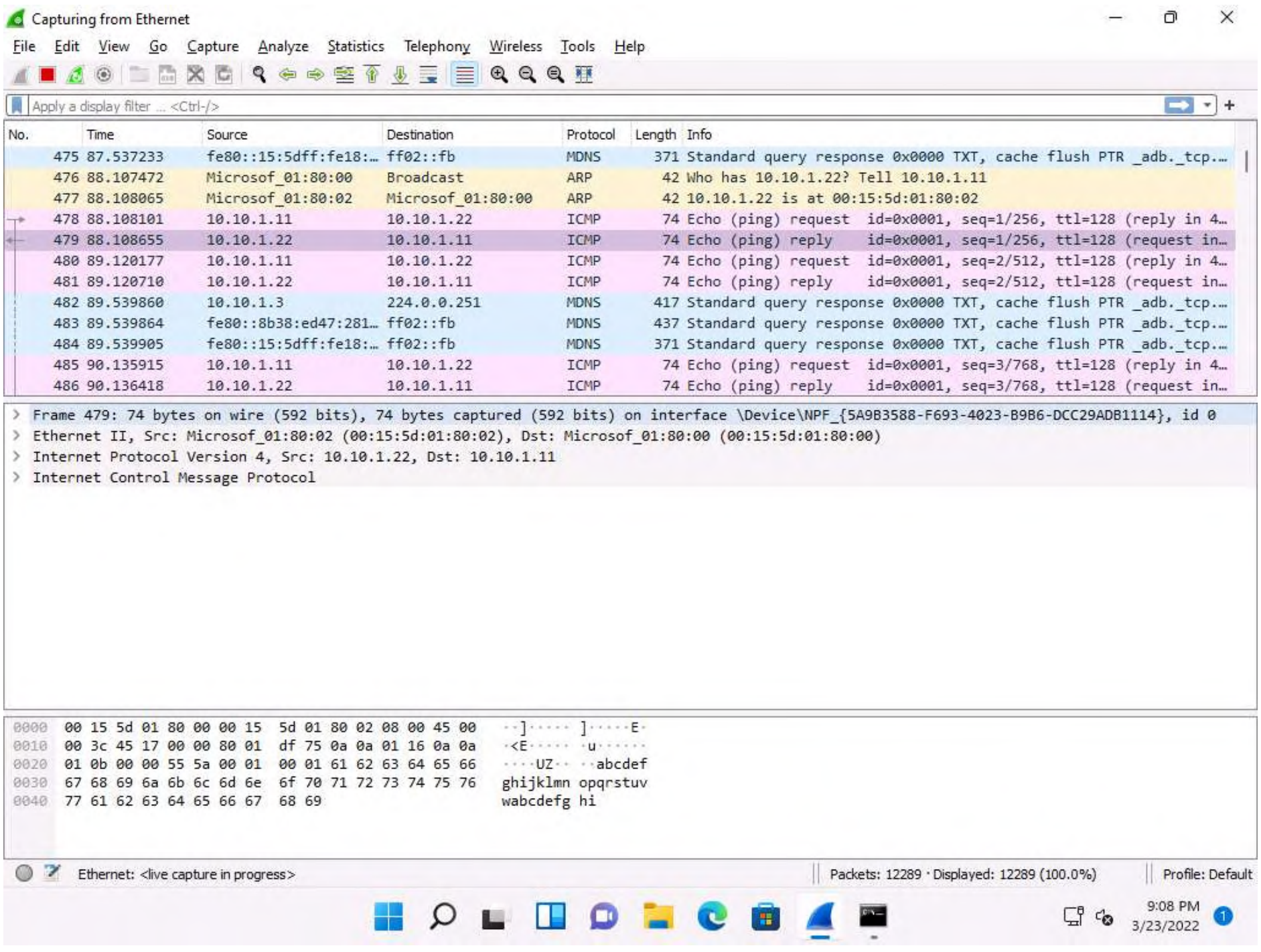


4. Open the **Command Prompt**, type **ping 10.10.1.22** and press **Enter**.

Note: **10.10.1.22** is the IP address of the **Windows Server 2022** machine.

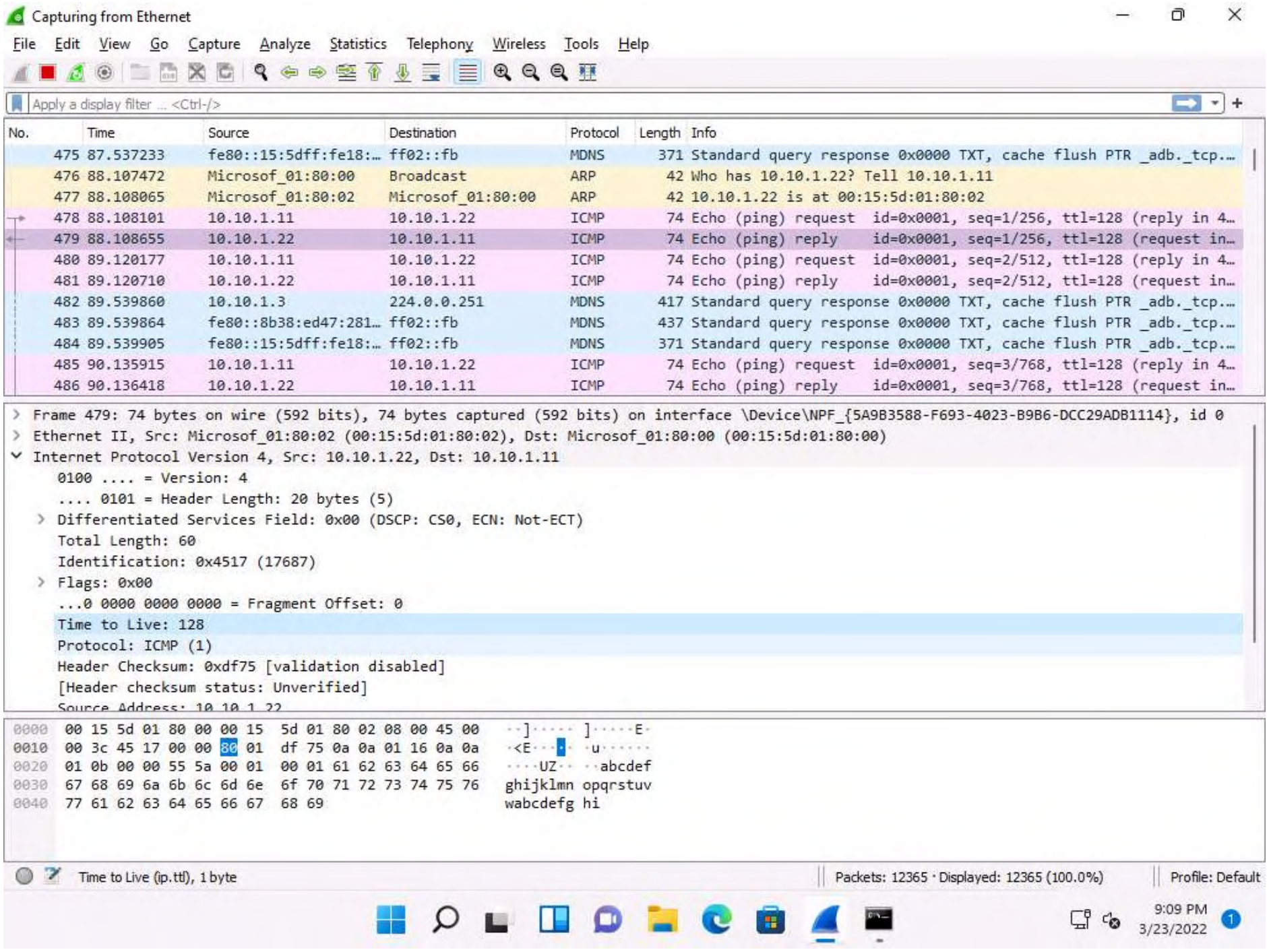


5. Observe the packets captured by **Wireshark**.

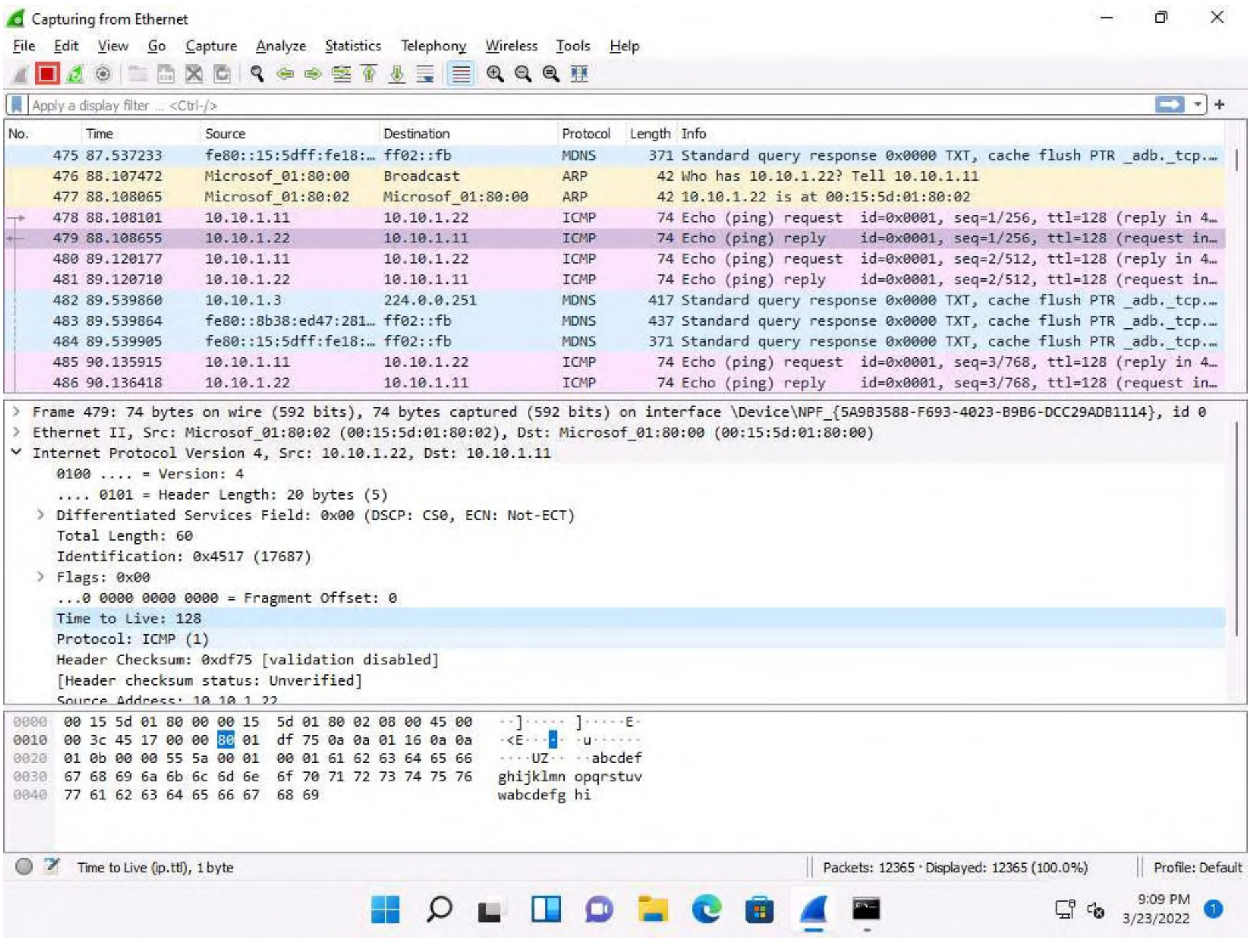


6. Choose any packet of the ICMP reply from the **Windows Server 2022** (**10.10.1.22**) to **Windows 11** (**10.10.1.11**) machines and expand the **Internet Protocol Version 4** node in the **Packet Details** pane.

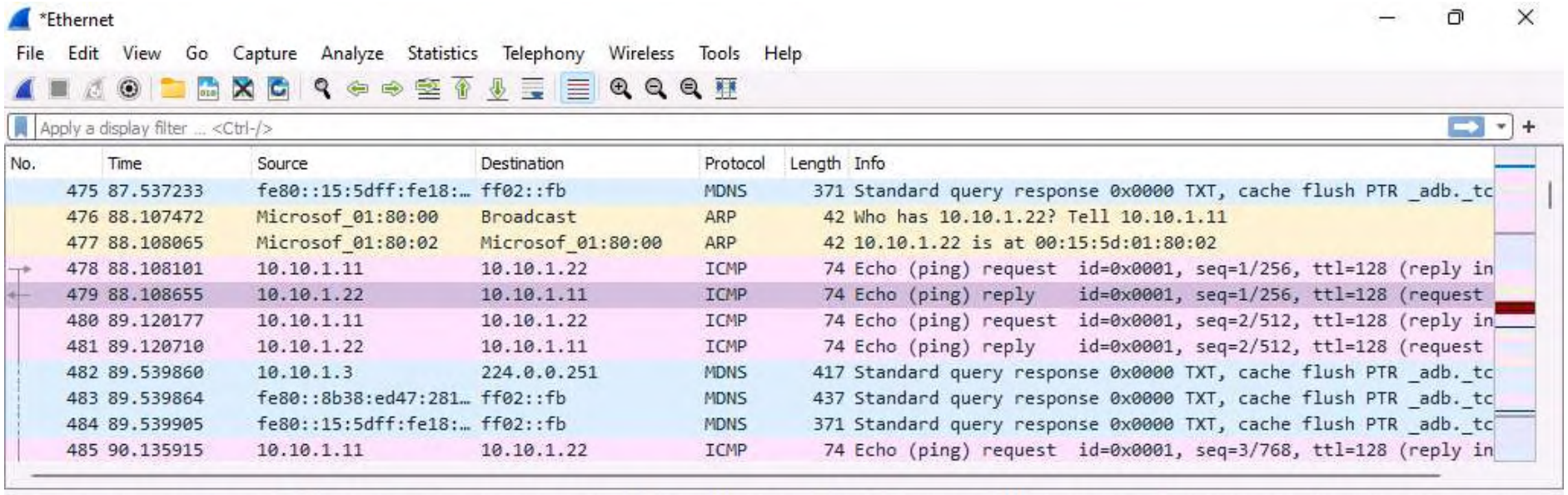
7. The TTL value is recorded as **128**, which means that the ICMP reply possibly came from a Windows-based machine.

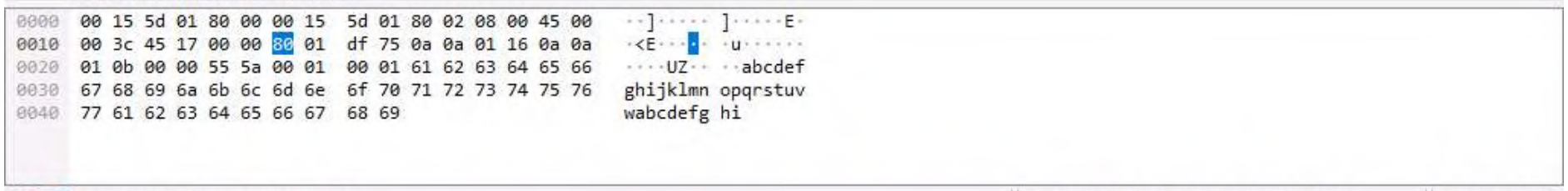


8. Now, stop the capture in the **Wireshark** window by clicking on the **Stop** button from the toolbar.



9. Now, click the **Start capturing packets** button from the toolbar. If an **Unsaved packets...** pop-up appears, click **Continue without Saving**.

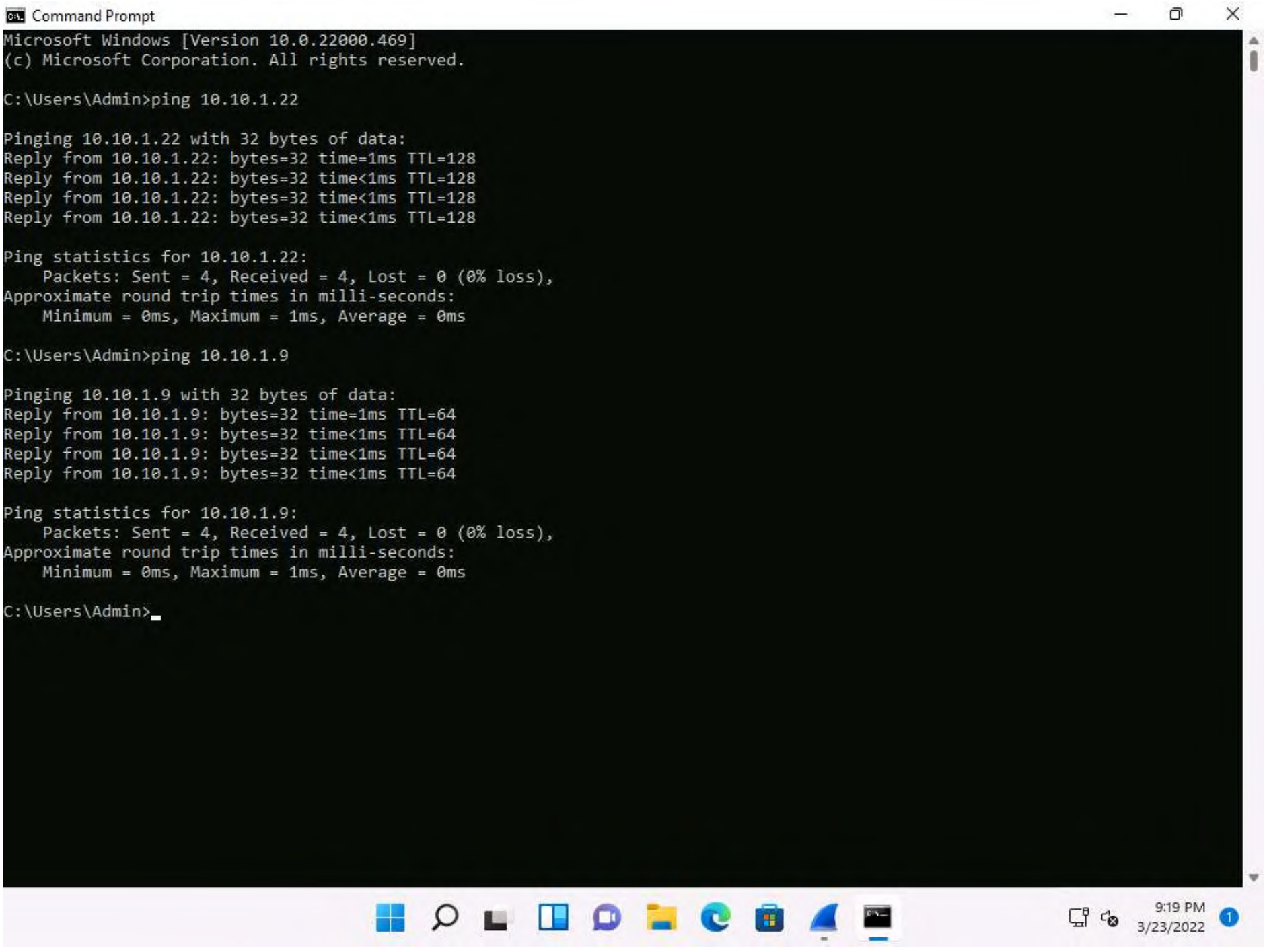




10. Wireshark will start capturing the new packets.

11. In the **Command Prompt** window, type **ping 10.10.1.9** and press **Enter**.

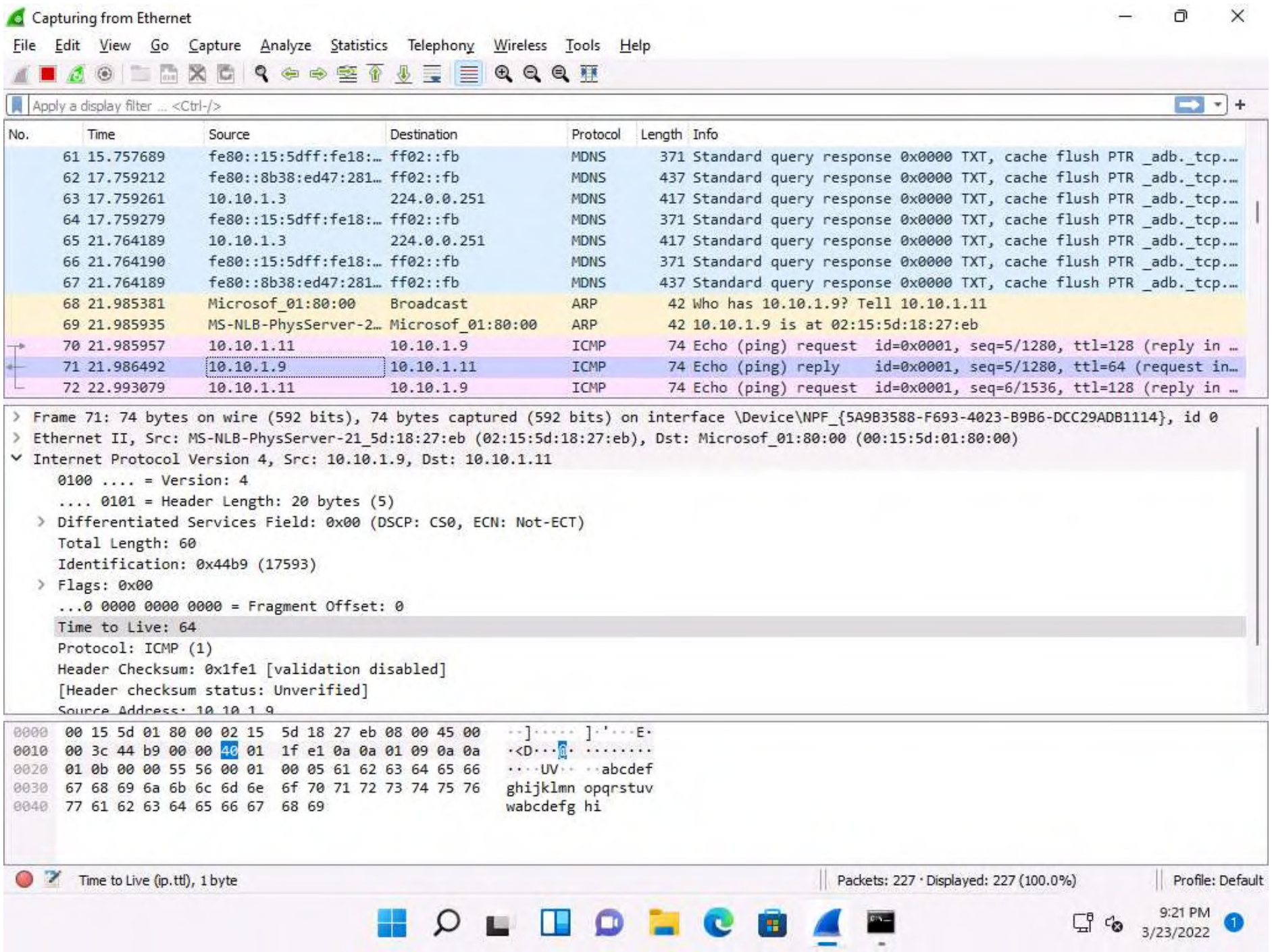
> Note: **10.10.1.9** is the IP address of the **Ubuntu** machine.



12. Observe the packets captured by **Wireshark**.

13. Choose any packet of ICMP reply from the **Ubuntu** (**10.10.1.9**) to **Windows 11** (**10.10.1.11**) machine and expand the **Internet Protocol Version 4** node in the **Packet Details** pane.

14. The TTL value is recorded as **64**, which means the ICMP reply possibly came from a Linux-based machine.



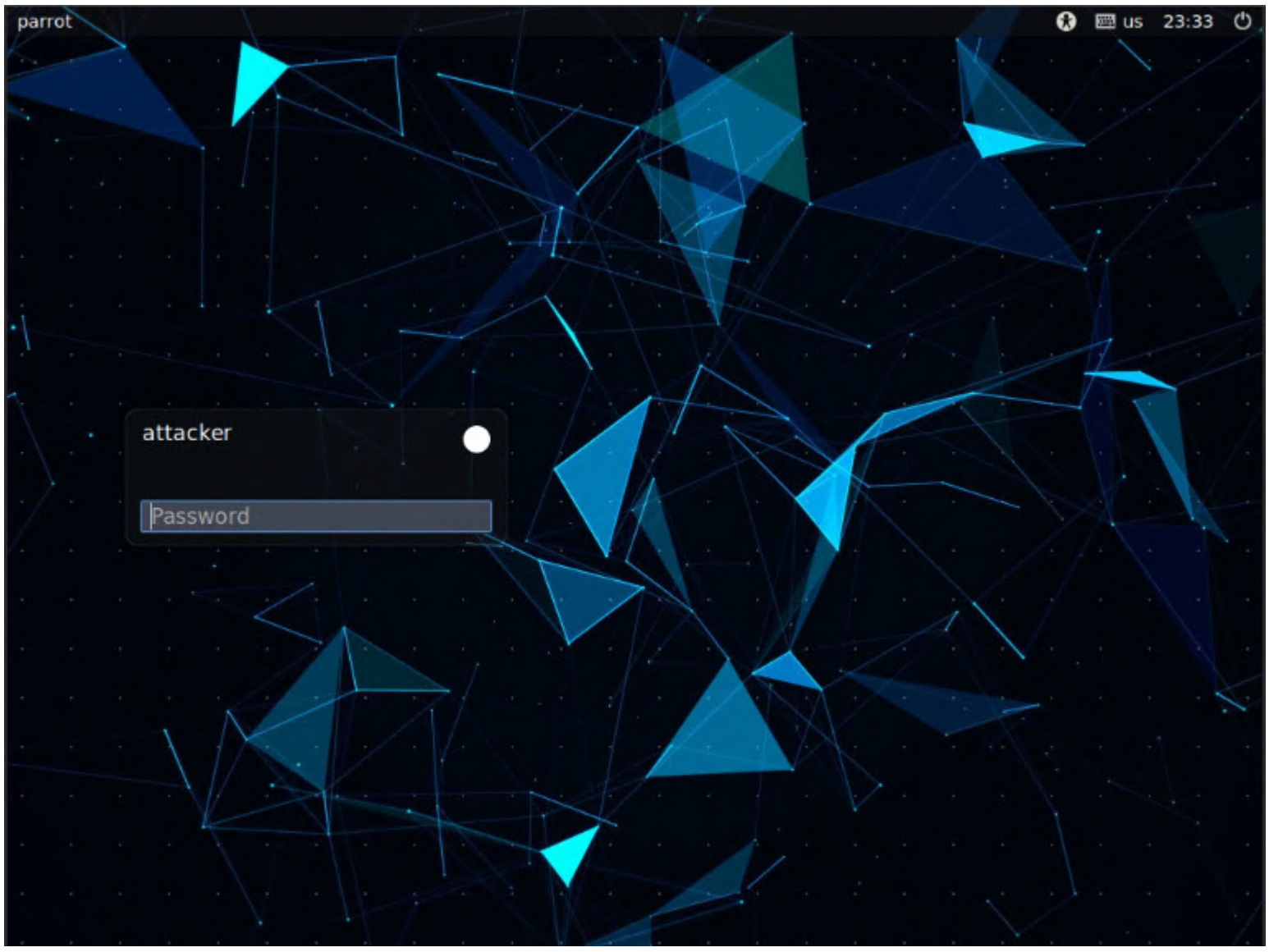
15. Stop the capture in the **Wireshark** window by clicking on the Stop button.
16. This concludes the demonstration of identifying the OS of the target system using Wireshark.
17. Close all open windows and document all the acquired information.

Task 2: Perform OS Discovery using Nmap Script Engine (NSE)

Nmap, along with Nmap Script Engine (NSE), can extract considerable valuable information from the target system. In addition to Nmap commands, NSE provides scripts that reveal all sorts of useful information from the target system. Using NSE, you may obtain information such as OS, computer name, domain name, forest name, NetBIOS computer name, NetBIOS domain name, workgroup, system time of a target system, etc.

Here, we will use Nmap to perform OS discovery using -A parameter, -O parameter, and NSE.

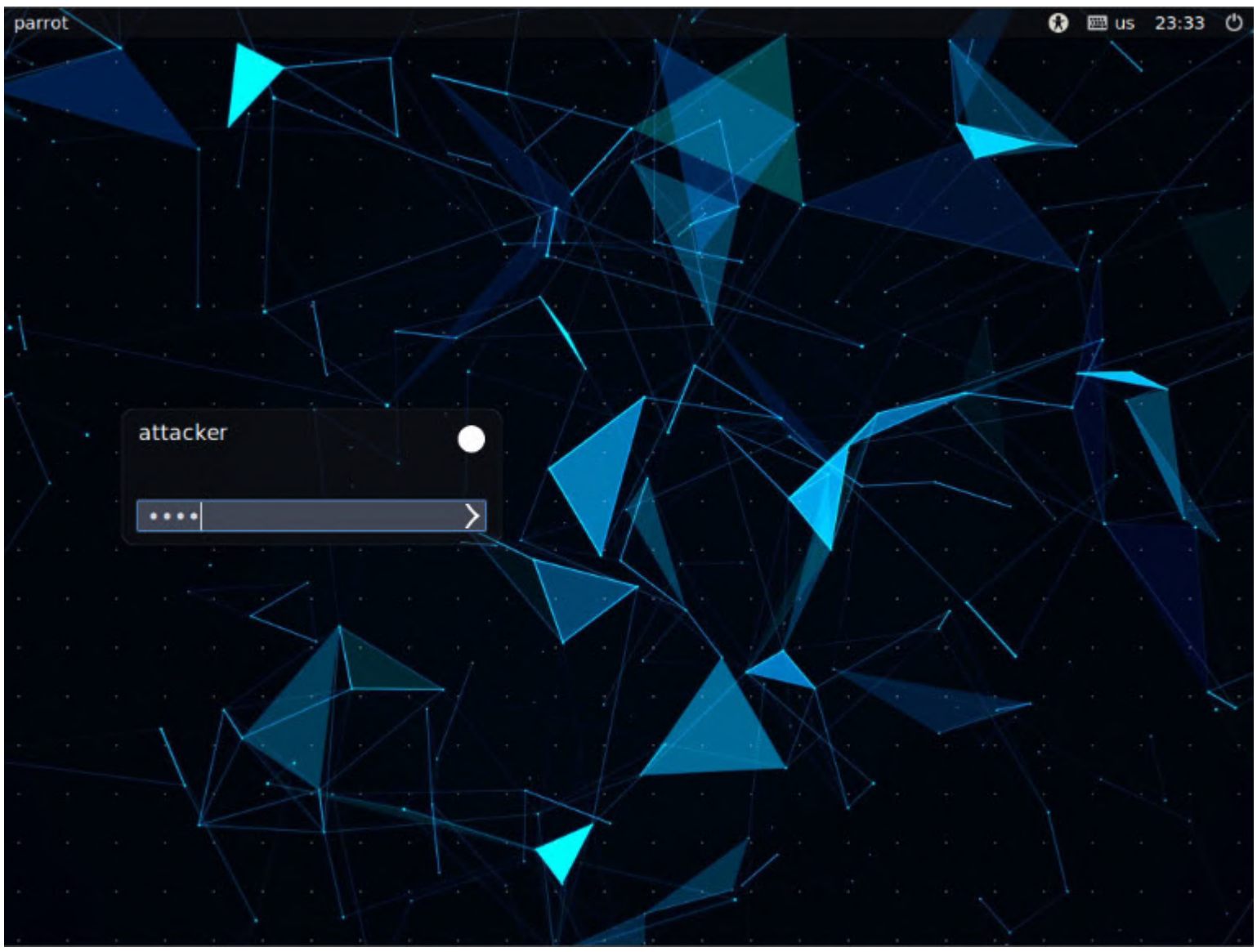
1. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.



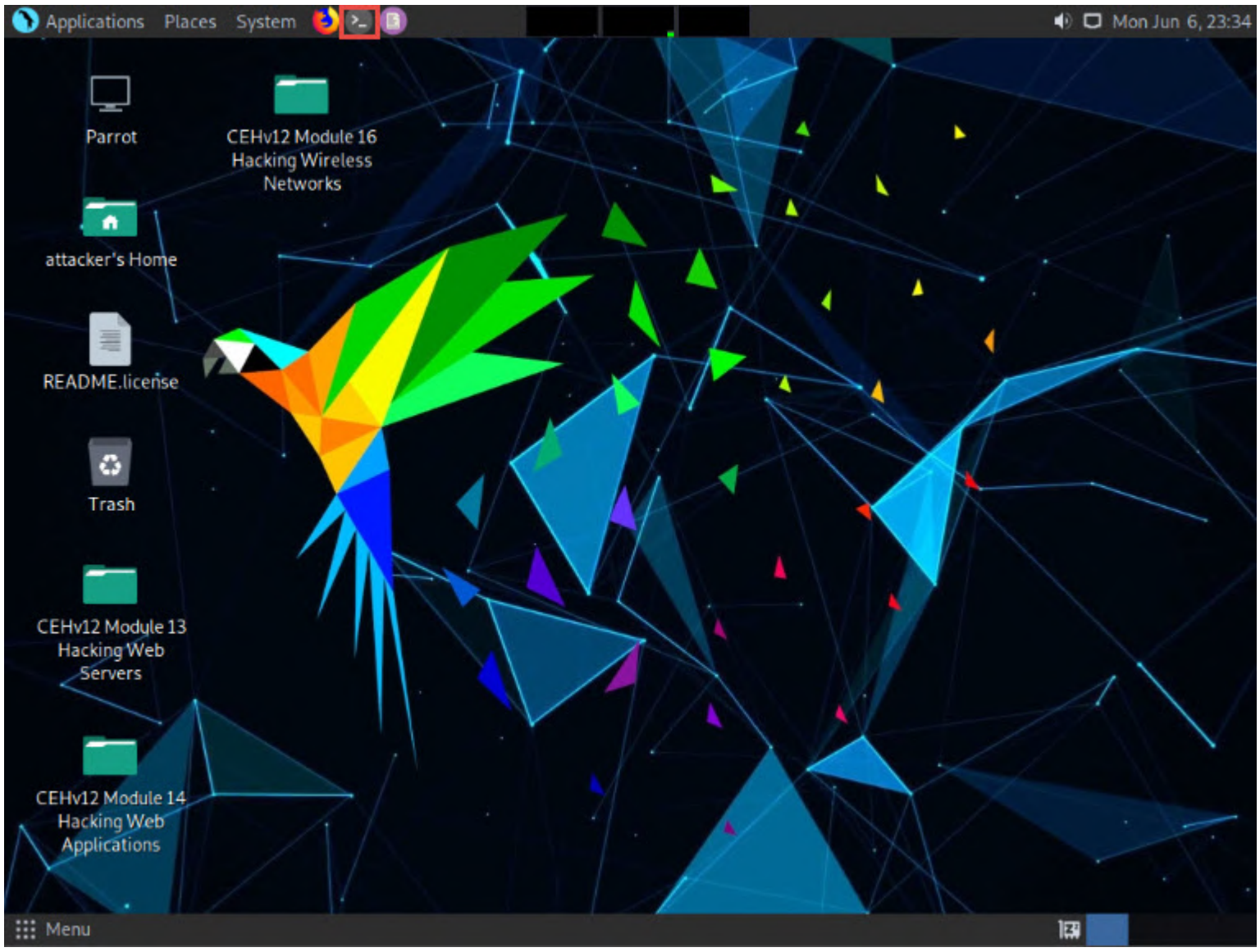
2. In the login page, the **attacker** username will be selected by default. Enter password as **toor** in the **Password** field and press **Enter** to log in to the machine.

Note: If a **Parrot Updater** pop-up appears at the top-right corner of **Desktop**, ignore and close it.

Note: If a **Question** pop-up window appears asking you to update the machine, click **No** to close the window.



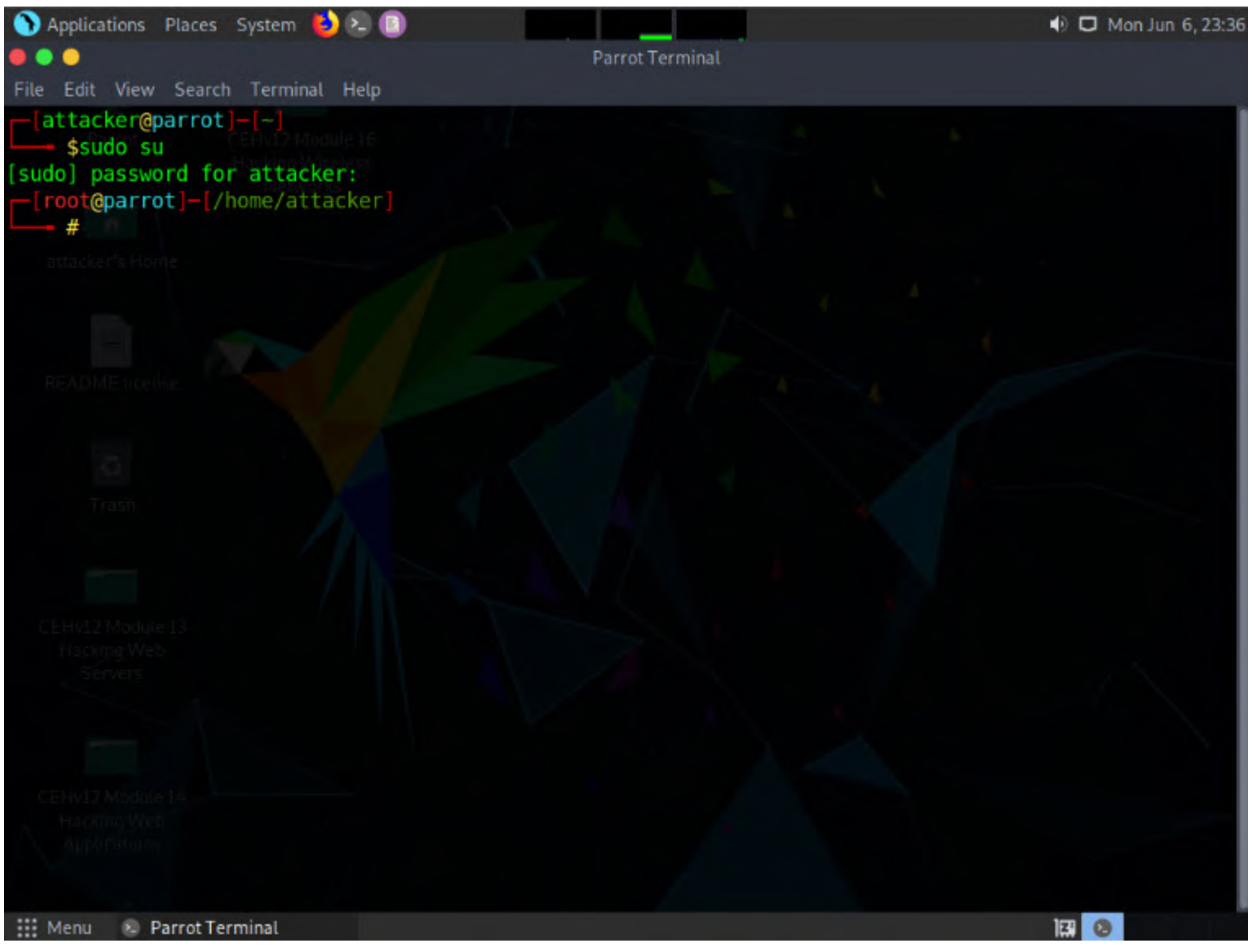
3. Click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.



4. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

5. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

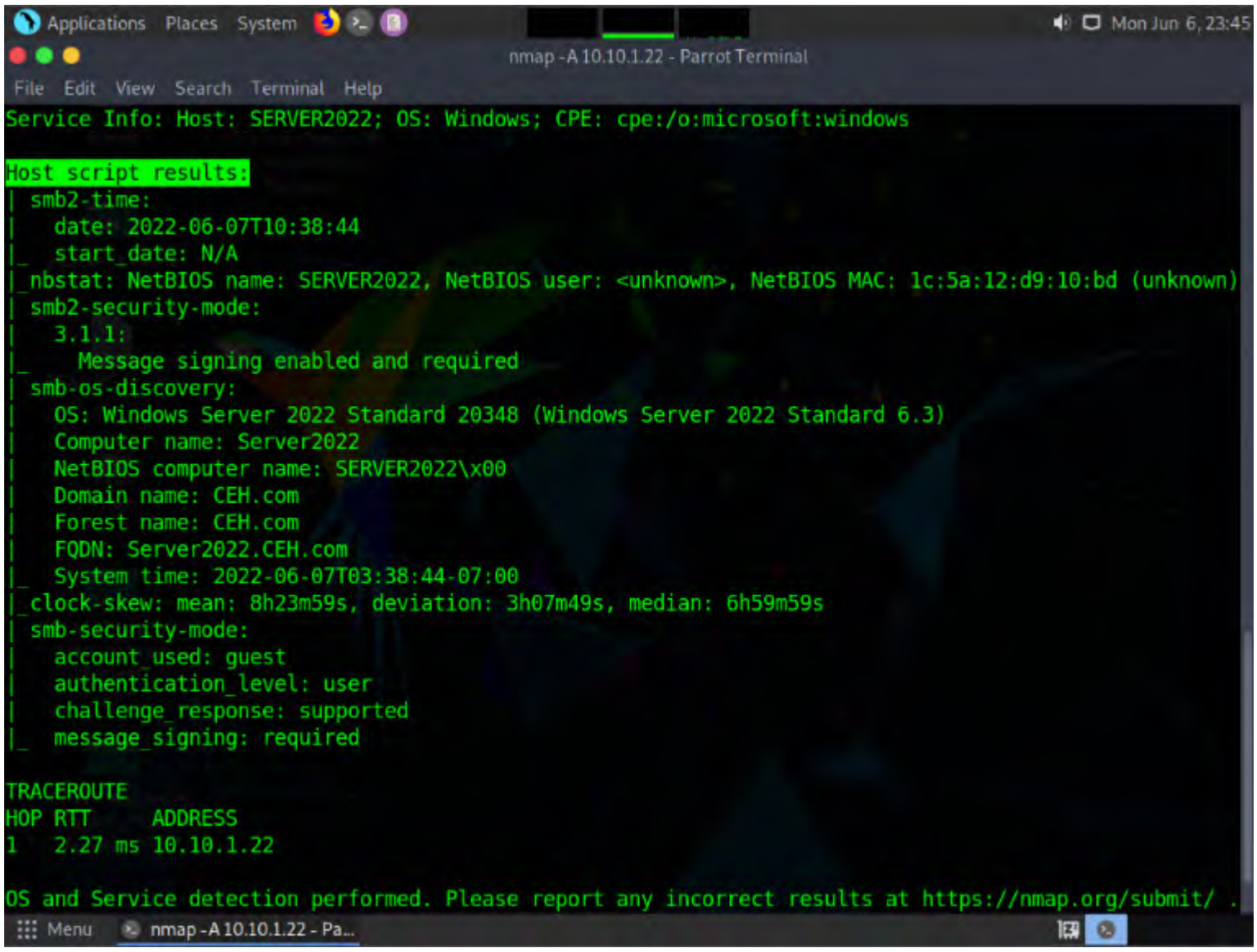
> Note: The password that you type will not be visible.



6. In the terminal window, type the command **nmap -A [Target IP Address]** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

> Note: **-A**: to perform an aggressive scan.
>
> Note: The scan takes approximately 10 minutes to complete.

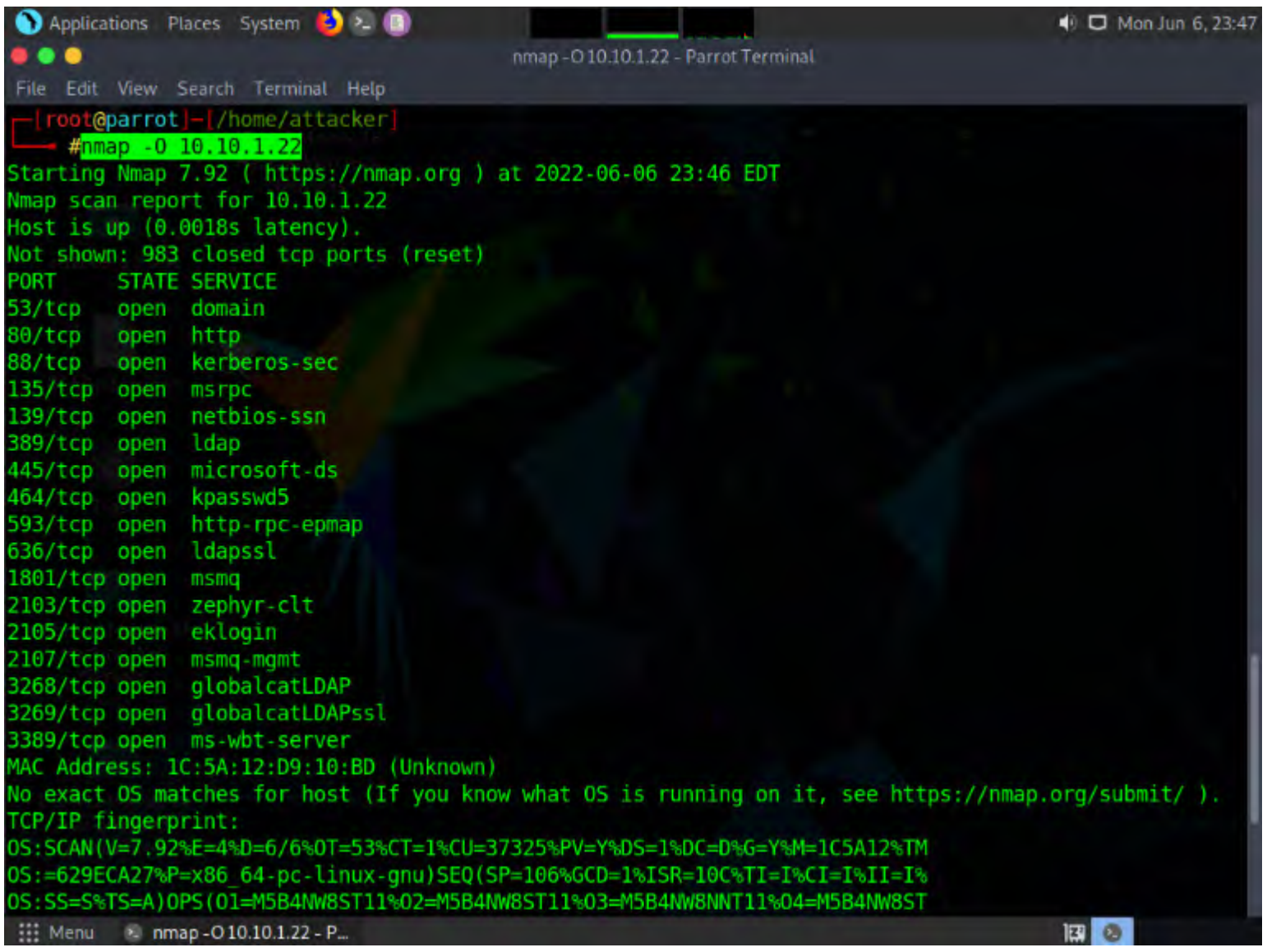
7. The scan results appear, displaying the open ports and running services along with their versions and target details such as OS, computer name, NetBIOS computer name, etc. under the **Host script results** section.



8. In the terminal window, type the command **nmap -O [Target IP Address]** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

Note: **-O**: performs the OS discovery.

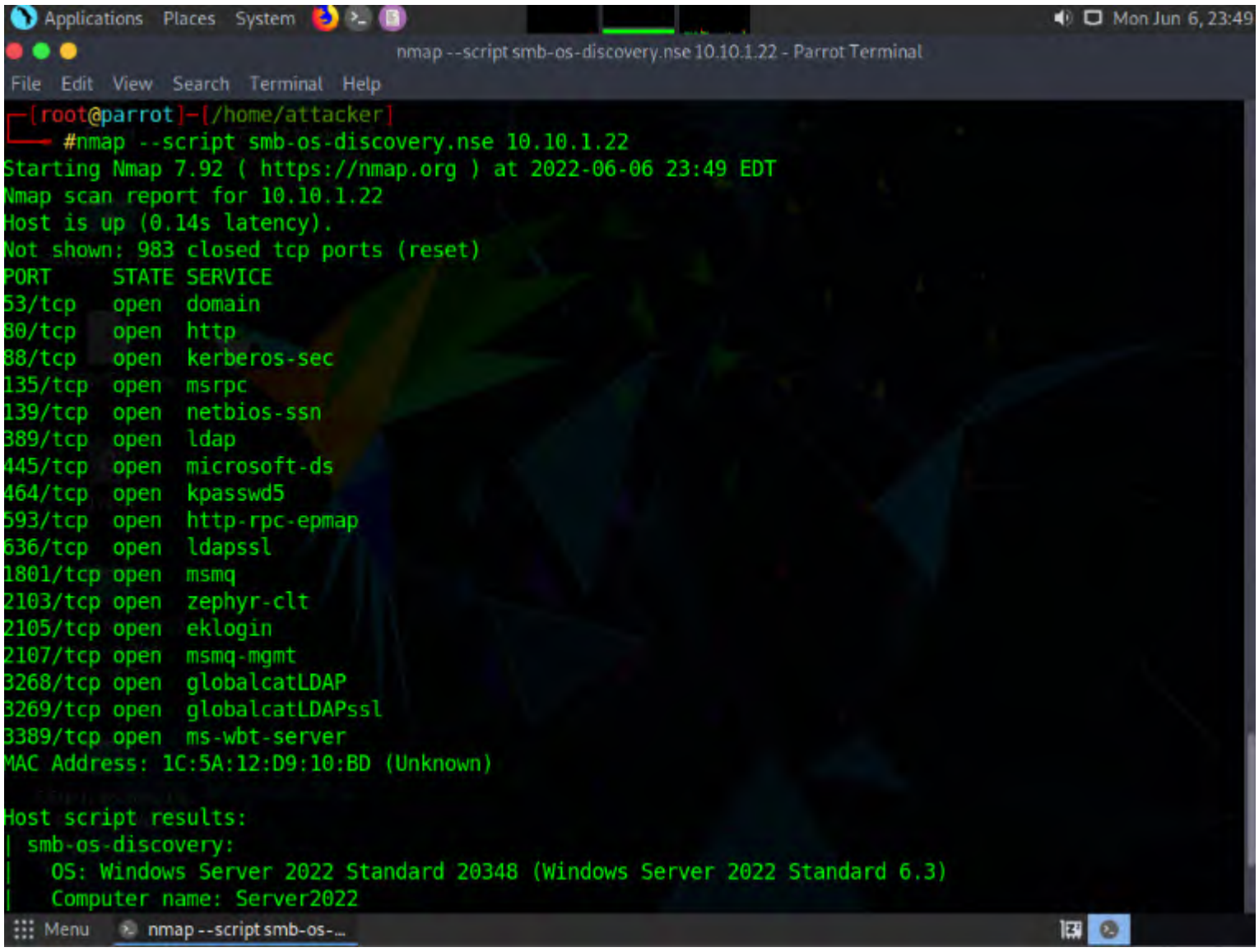
9. The scan results appear, displaying information about open ports, respective services running on the open ports, and the name of the OS running on the target system.

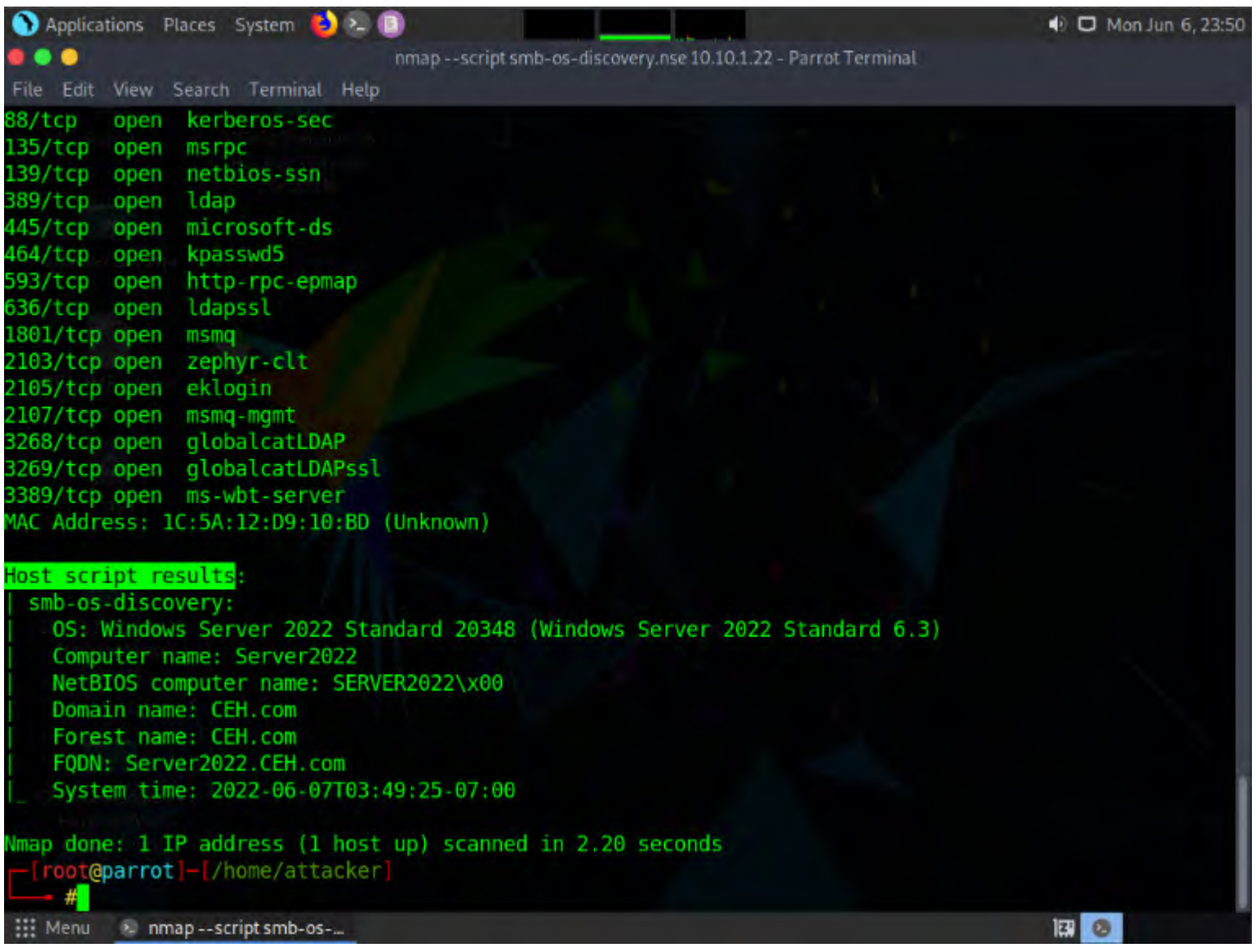


10. In the terminal window, type the command **nmap --script smb-os-discovery.nse [Target IP Address]** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

Note: **--script**: specifies the customized script and **smb-os-discovery.nse**: attempts to determine the OS, computer name, domain, workgroup, and current time over the SMB protocol (ports 445 or 139).

11. The scan results appear, displaying the target OS, computer name, NetBIOS computer name, etc. details under the **Host script results** section.





12. This concludes the demonstration of discovering the OS running on the target system using Nmap.

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

Task 3: Perform OS Discovery using Unicornscan

Unicornscan is a Linux-based command line-oriented network information-gathering and reconnaissance tool. It is an asynchronous TCP and UDP port scanner and banner grabber that enables you to discover open ports, services, TTL values, etc. running on the target machine. In Unicornscan, the OS of the target machine can be identified by observing the TTL values in the acquired scan result.

Here, we will use the Unicornscan tool to perform OS discovery on the target system.

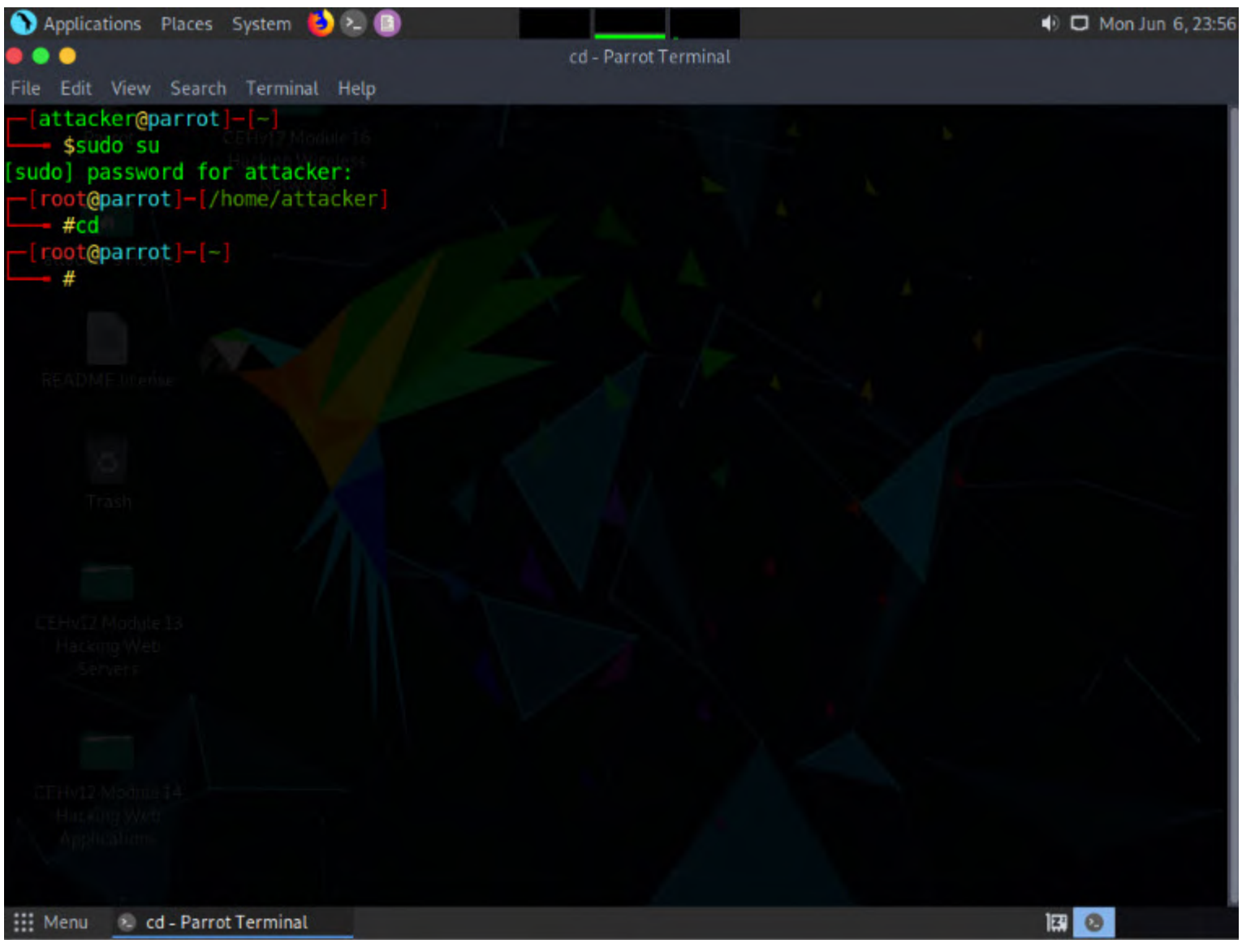
1. In the **Parrot Security** machine, click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.

2. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

3. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

Note: The password that you type will not be visible.

4. Now, type **cd** and press **Enter** to jump to the root directory.

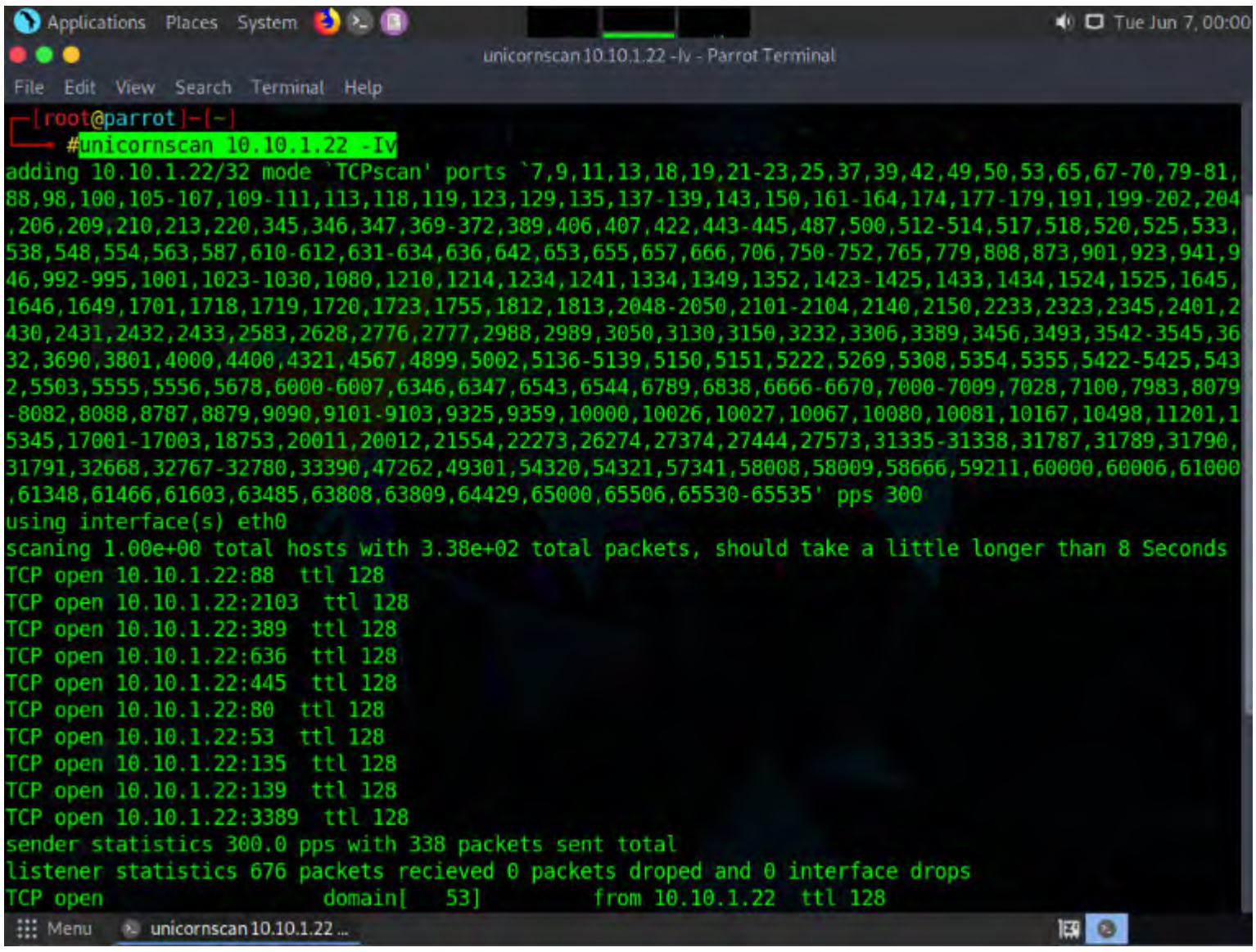


5. In the terminal window, type **unicornscan [Target IP Address] -Iv** (here, the target machine is **Windows Server 2022** [**10.10.1.22**]) and press **Enter**.

Note: In this command, **-I** specifies an immediate mode and **v** specifies a verbose mode.

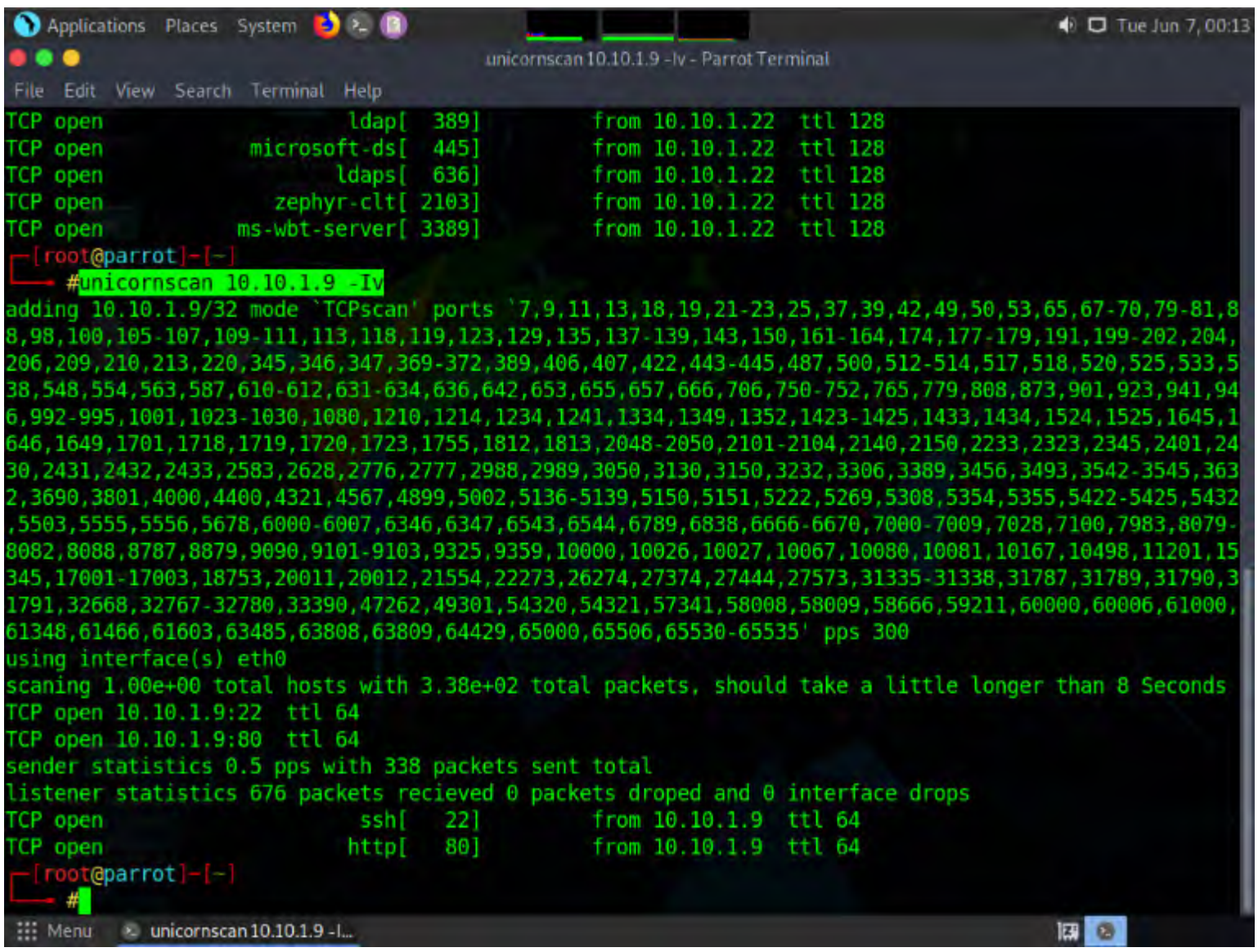
6. The scan results appear, displaying the open TCP ports along with the obtained TTL value of **128**. As shown in the screenshot, the **ttl** values acquired after the scan are **128**; hence, the OS is possibly Microsoft Windows (Windows 8/8.1/10/11 or Windows Server 16/19/22).

> Note: Here, the target machine is **Windows Server 2022** (**10.10.1.22**).



7. In the **Parrot Terminal** window, type **unicornscan [Target IP Address] -Iv** (here, the target machine is **Ubuntu** [**10.10.1.9**]) and press **Enter**.

8. The scan results appear, displaying the open TCP ports along with a TTL value of **64**. As shown in the screenshot, the **ttl** value acquired after the scan is **64**; hence, the OS is possibly a Linux-based machine (Google Linux, Ubuntu, Parrot, or Kali). Using this information, attackers can formulate an attack strategy based on the OS of the target system.



9. This concludes the demonstration of discovering the OS of the target machine using Unicornscan.

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

Lab 4: Scan beyond IDS and Firewall

Lab Scenario

As a professional ethical hacker or a pen tester, the next step after discovering the OS of the target IP address(es) is to perform network scanning without being detected by the network security perimeters such as the firewall and IDS. IDSs and firewalls are efficient security mechanisms; however, they still have some security limitations. You may be required to launch attacks to exploit these limitations using various IDS/firewall evasion techniques such as packet fragmentation, source routing, IP address spoofing, etc. Scanning beyond the IDS and firewall allows you to evaluate the target network's IDS and firewall security.

Lab Objectives

- Scan beyond IDS/firewall using various evasion techniques
- Create custom packets using Colasoft Packet Builder to scan beyond the IDS/firewall
- Create custom UDP and TCP packets using Hping3 to scan beyond the IDS/firewall

Overview of Scanning beyond IDS and Firewall

An Intrusion Detection System (IDS) and firewall are the security mechanisms intended to prevent an unauthorized person from accessing a network. However, even IDSs and firewalls have some security limitations. Firewalls and IDSs intend to avoid malicious traffic (packets) from entering into a network, but certain techniques can be used to send intended packets to the target and evade IDSs/firewalls.

Techniques to evade IDS/firewall:

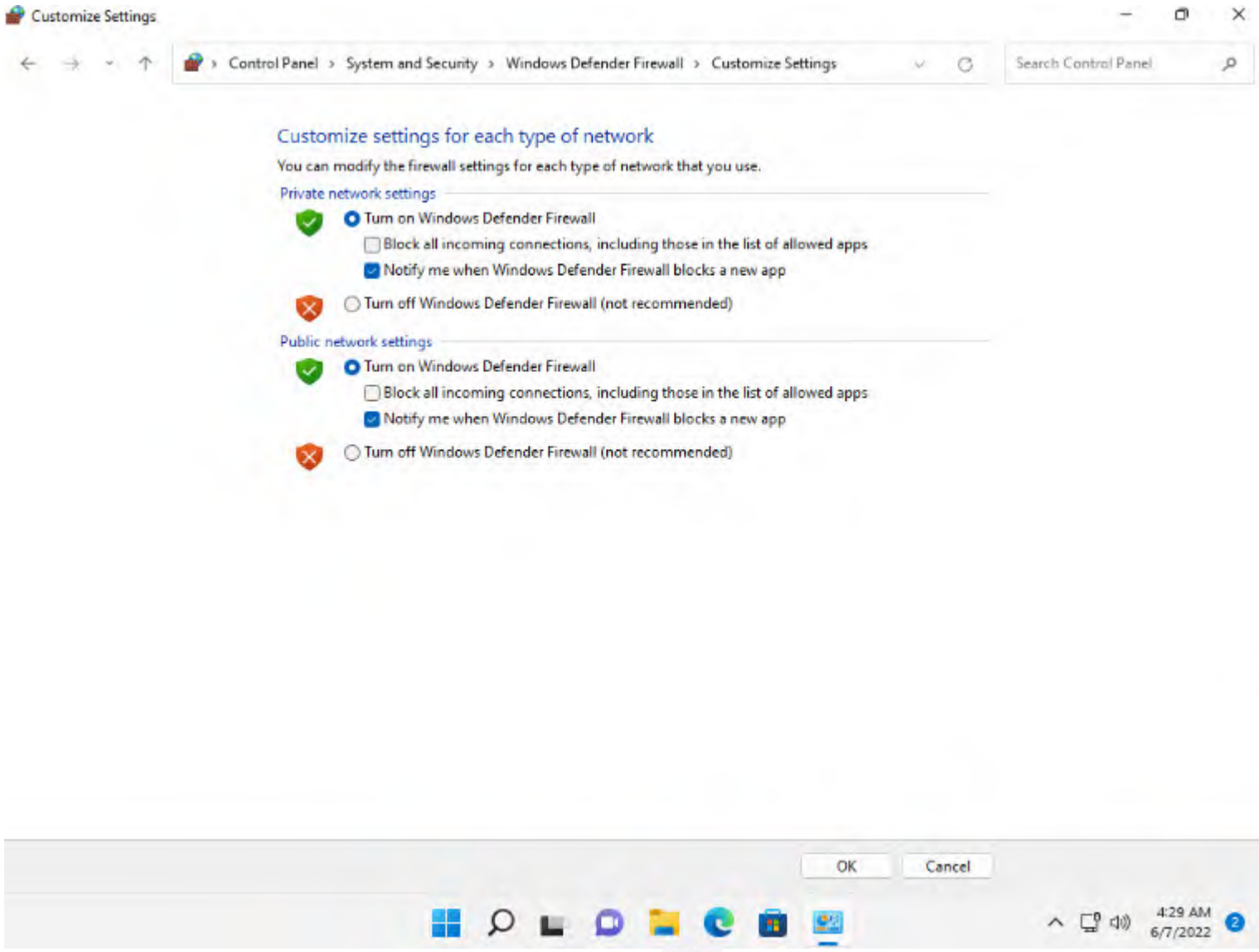
- **Packet Fragmentation**: Send fragmented probe packets to the intended target, which re-assembles it after receiving all the fragments
- **Source Routing**: Specifies the routing path for the malformed packet to reach the intended target
- **Source Port Manipulation**: Manipulate the actual source port with the common source port to evade IDS/firewall
- **IP Address Decoy**: Generate or manually specify IP addresses of the decoys so that the IDS/firewall cannot determine the actual IP address
- **IP Address Spoofing**: Change source IP addresses so that the attack appears to be coming in as someone else
- **Creating Custom Packets**: Send custom packets to scan the intended target beyond the firewalls
- **Randomizing Host Order**: Scan the number of hosts in the target network in a random order to scan the intended target that is lying beyond the firewall
- **Sending Bad Checksums**: Send the packets with bad or bogus TCP/UPD checksums to the intended target
- **Proxy Servers**: Use a chain of proxy servers to hide the actual source of a scan and evade certain IDS/firewall restrictions
- **Anonymizers**: Use anonymizers that allow them to bypass Internet censors and evade certain IDS and firewall rules

Task 1: Scan beyond IDS/Firewall using Various Evasion Techniques

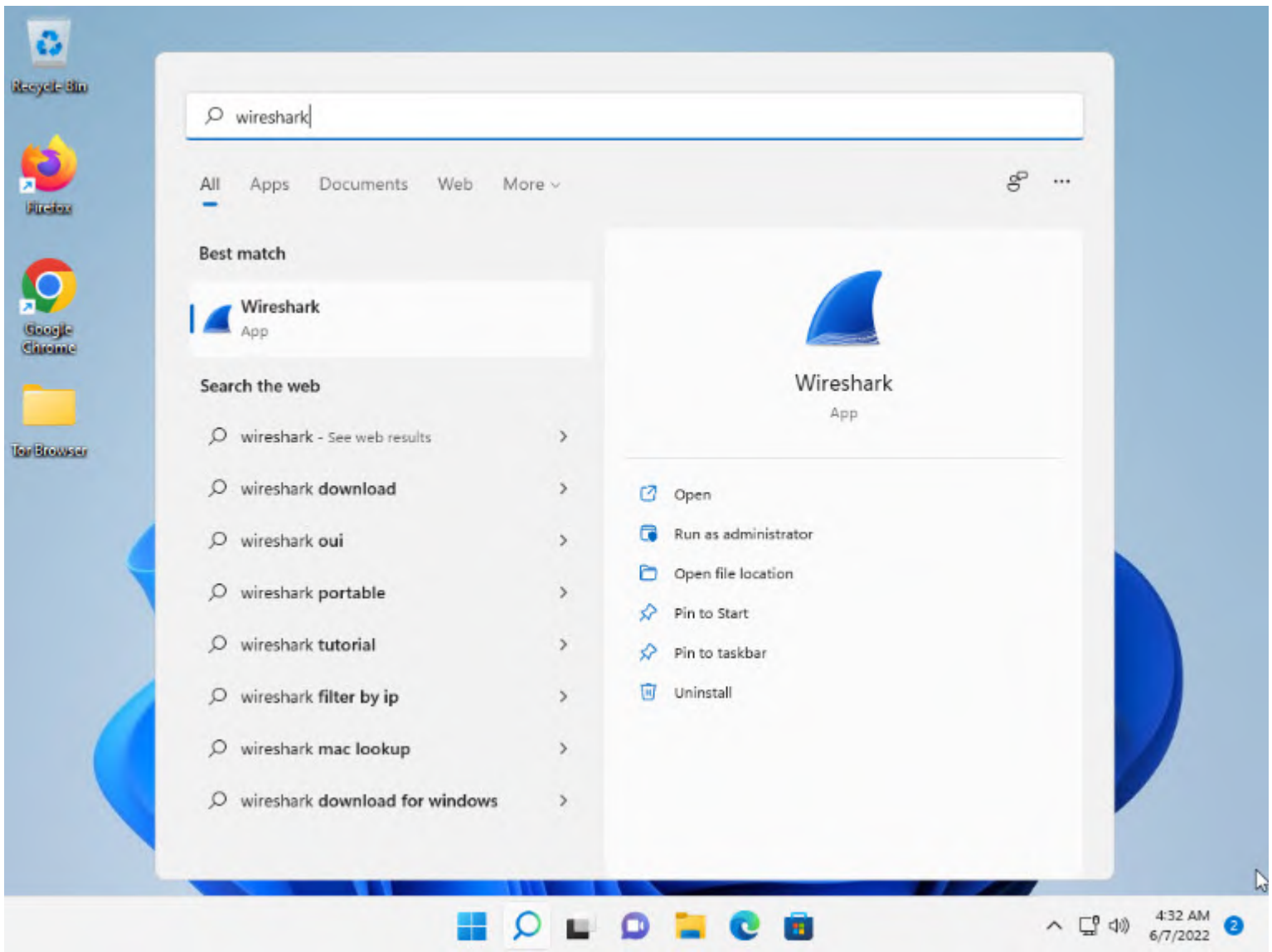
Nmap offers many features to help understand complex networks with enabled security mechanisms and supports mechanisms for bypassing poorly implemented defenses. Using Nmap, various techniques can be implemented, which can bypass the IDS/firewall security mechanisms.

Here, we will use Nmap to evade IDS/firewall using various techniques such as packet fragmentation, source port manipulation, MTU, and IP address decoy.

1. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine.
2. Navigate to **Control Panel** --> **System and Security** --> **Windows Defender Firewall** --> **Turn Windows Defender Firewall on or off**, enable Windows Defender Firewall and click **OK**, as shown in the screenshot.

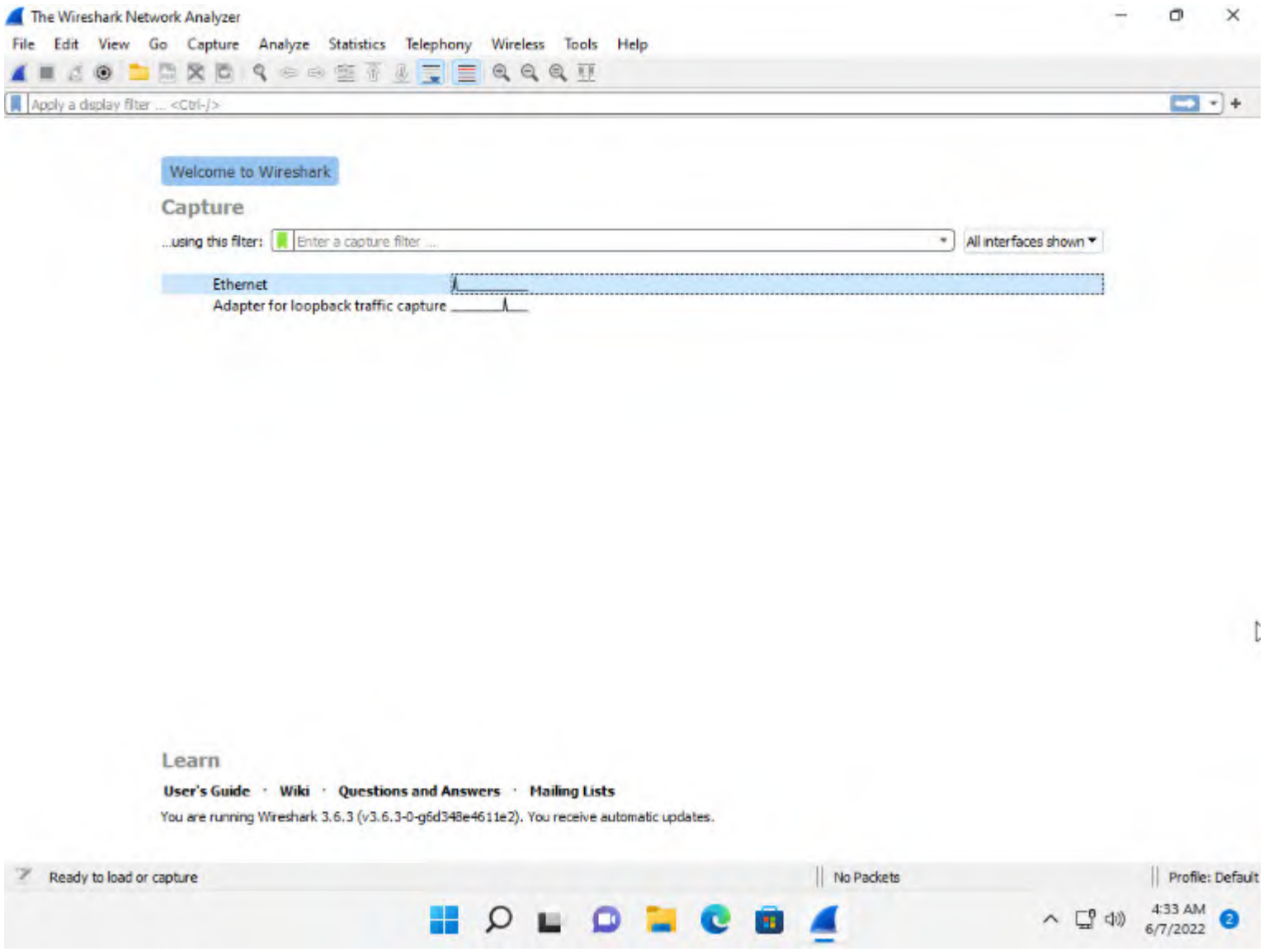


3. Minimize the **Control Panel** window, click **Search** icon ( ) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Open** to launch it.



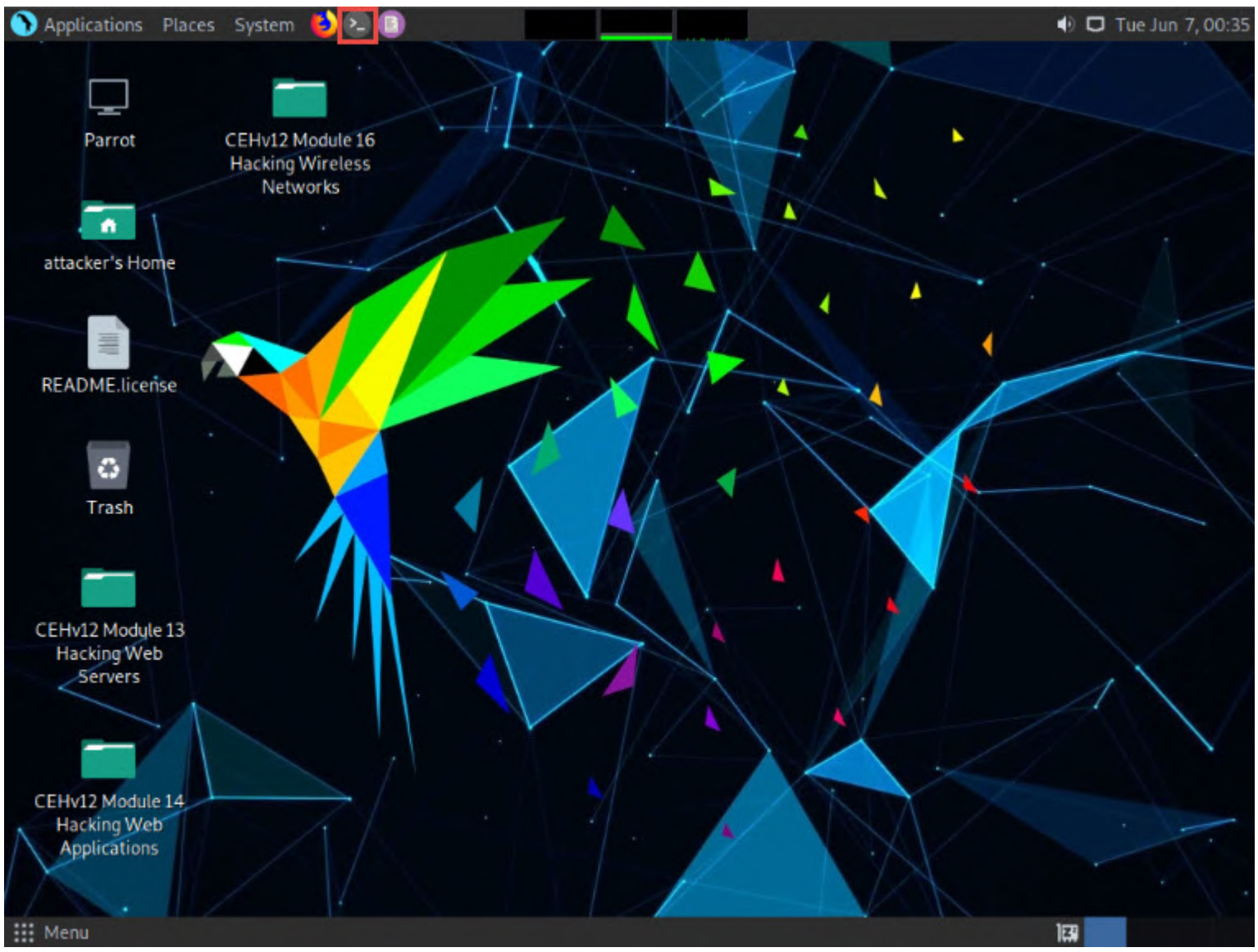
4. The **Wireshark Network Analyzer** window appears, Start capturing packets by double-clicking the available ethernet or interface (here, **Ethernet**).

Note: If **Software Update** window appears, click **Remind me later**.



5. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.

6. Click the **MATE Terminal** icon in the top-left corner of the **Desktop** to open a **Terminal** window.

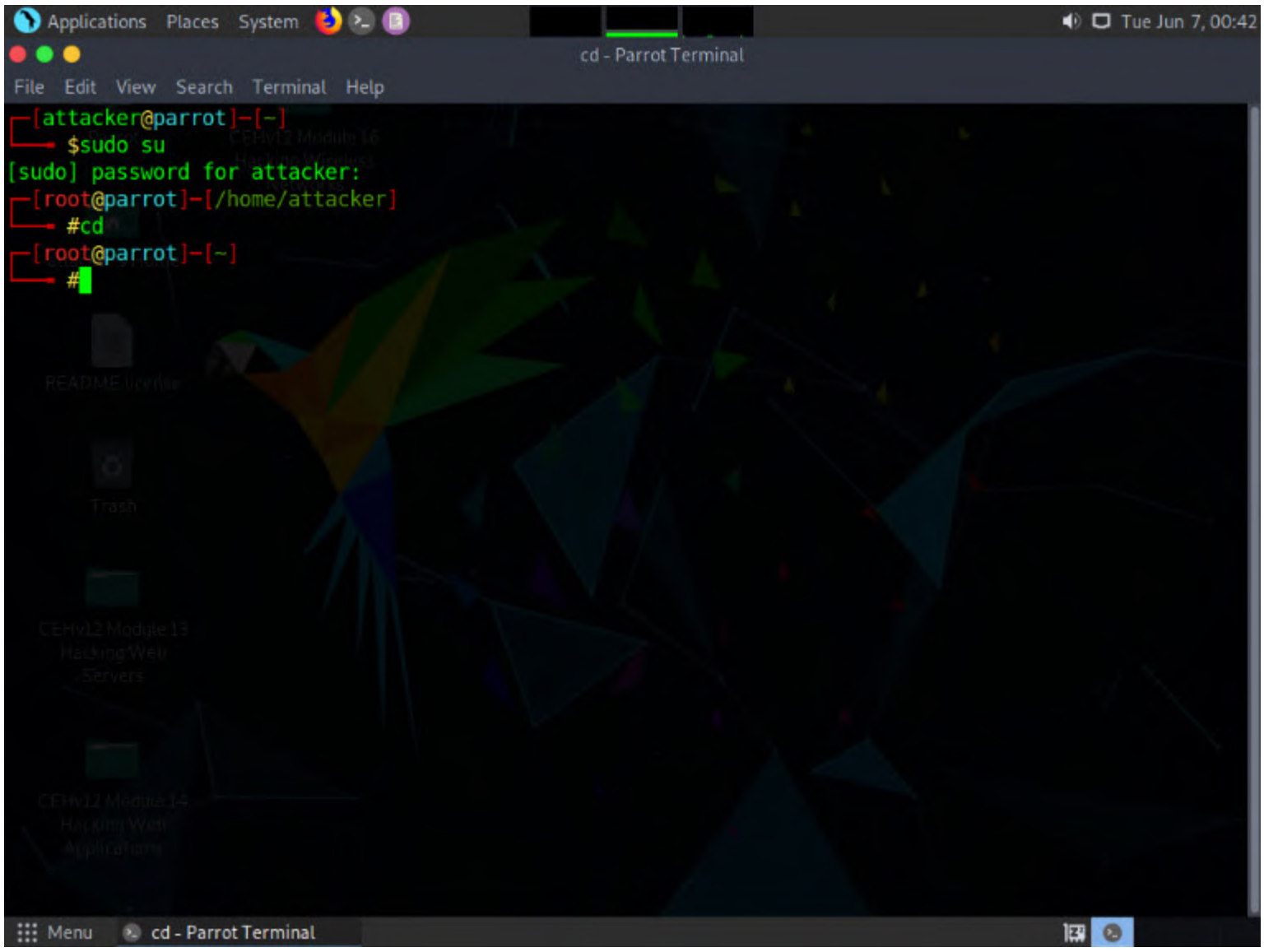


7. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

8. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

> Note: The password that you type will not be visible.

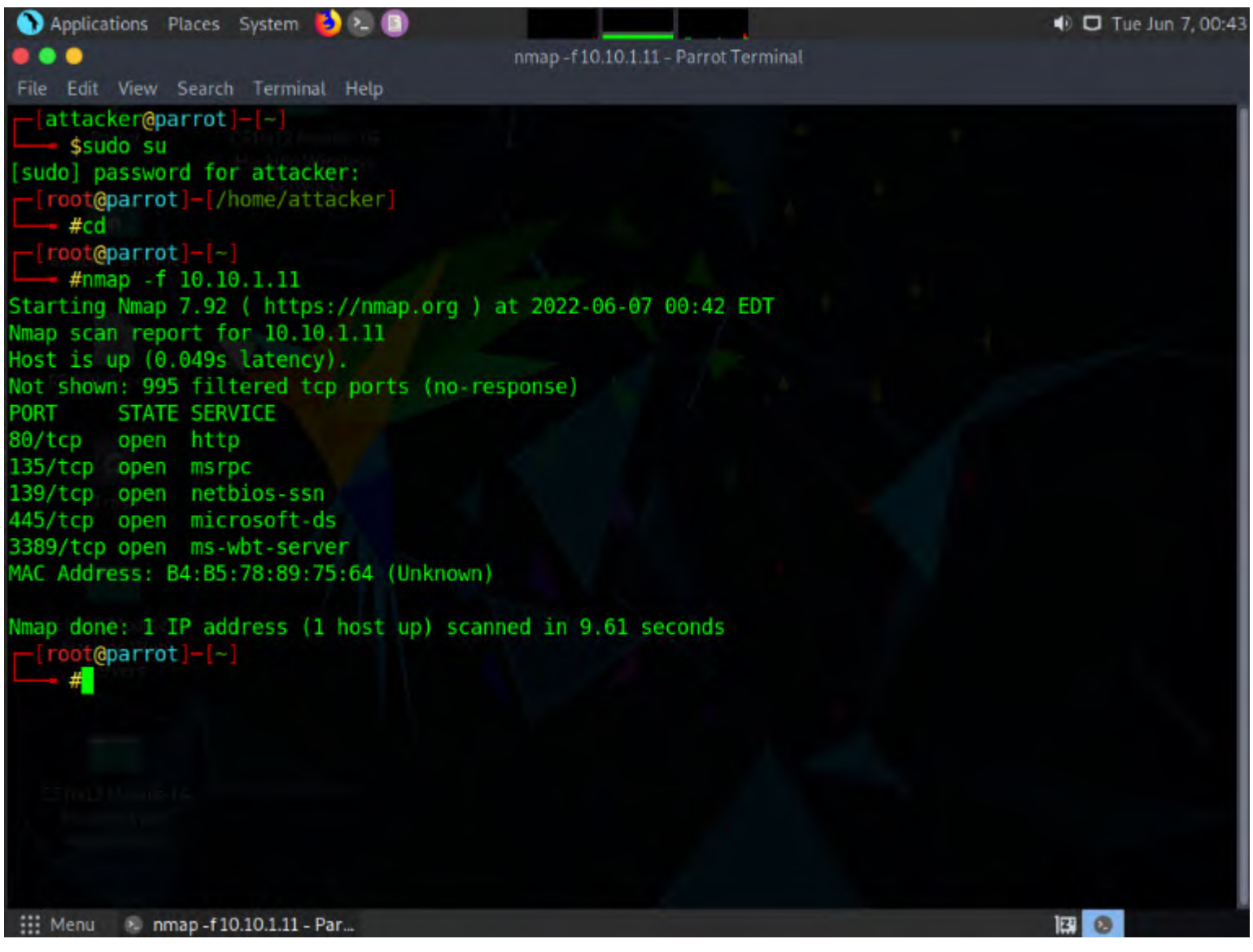
9. Now, type **cd** and press **Enter** to jump to the root directory.



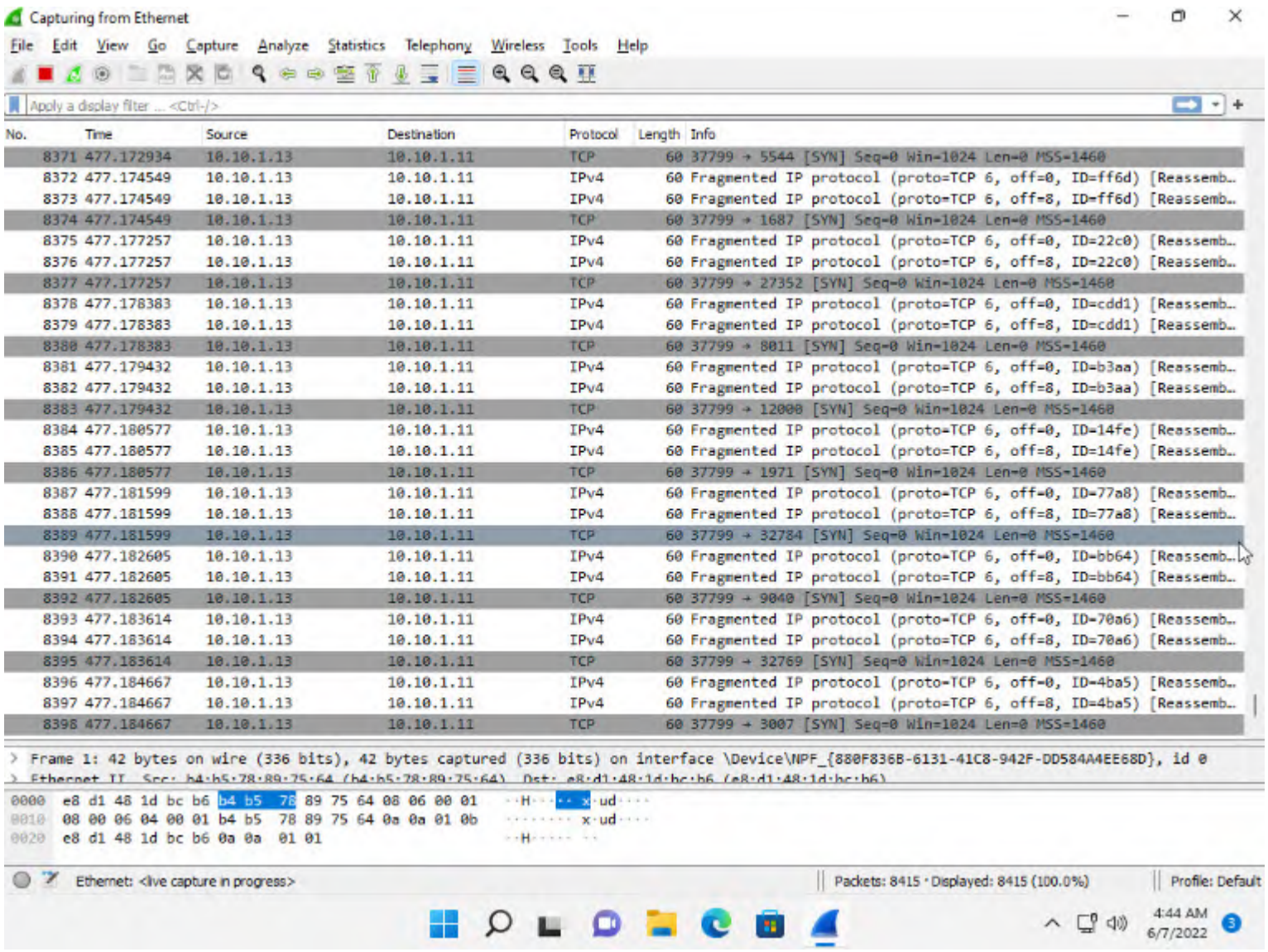
10. In the terminal window, type **nmap -f [Target IP Address]**, (here, the target machine is **Windows 11** [**10.10.1.11**]) and press **Enter**.

> Note: **-f** switch is used to split the IP packet into tiny fragment packets.
>
> Note: Packet fragmentation refers to the splitting of a probe packet into several smaller packets (fragments) while sending it to a network. When these packets reach a host, IDSs and firewalls behind the host generally queue all of them and process them one by one. However, since this method of processing involves greater CPU consumption as well as network resources, the configuration of most of IDSs makes it skip fragmented packets during port scans.

11. Although **Windows Defender Firewall** is turned on in the target system (here, **Windows 11**), you can still obtain the results displaying all open TCP ports along with the name of services running on the ports, as shown in the screenshot.



12. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine (target machine). You can observe the fragmented packets captured by the Wireshark, as shown in the screenshot.

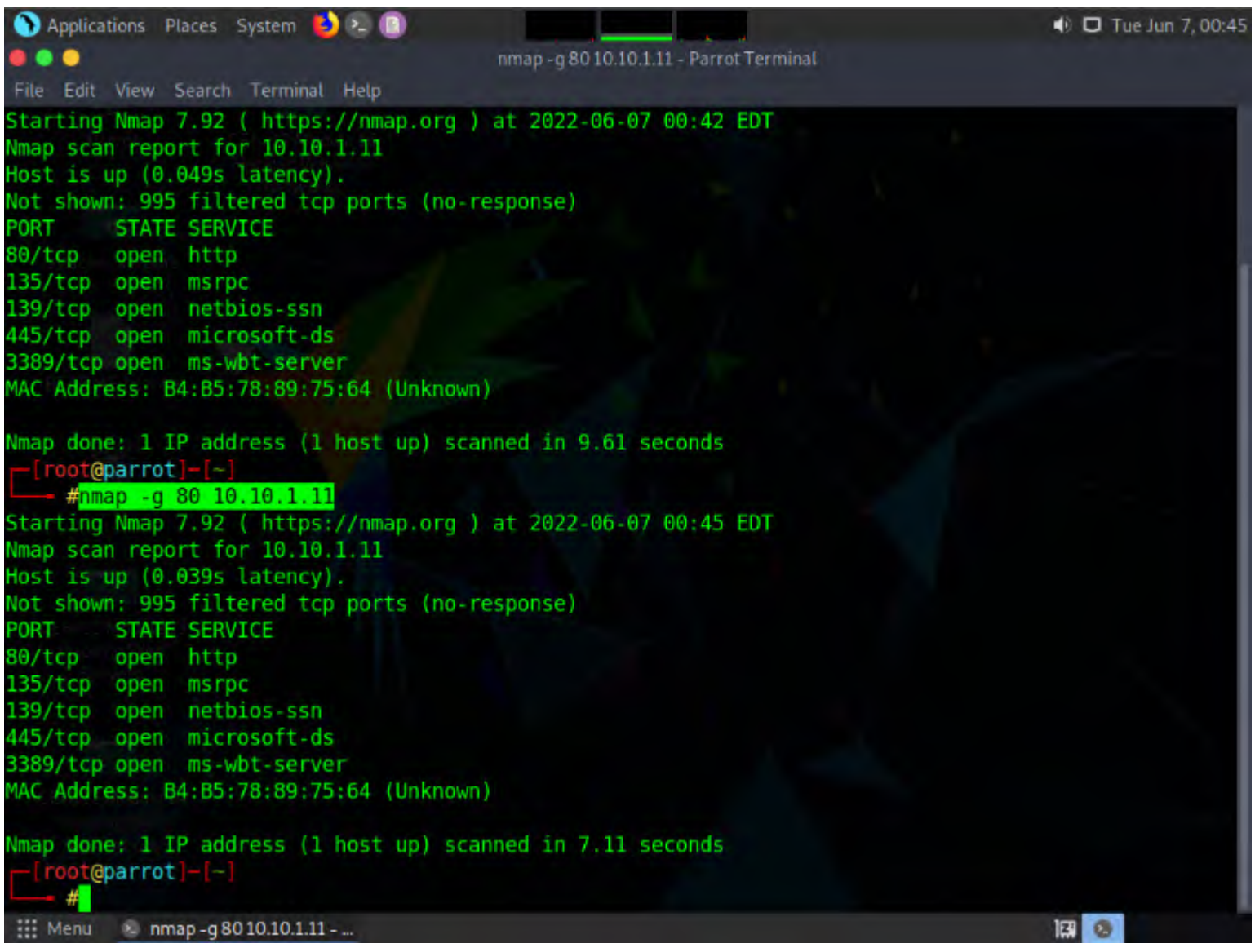


13. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.

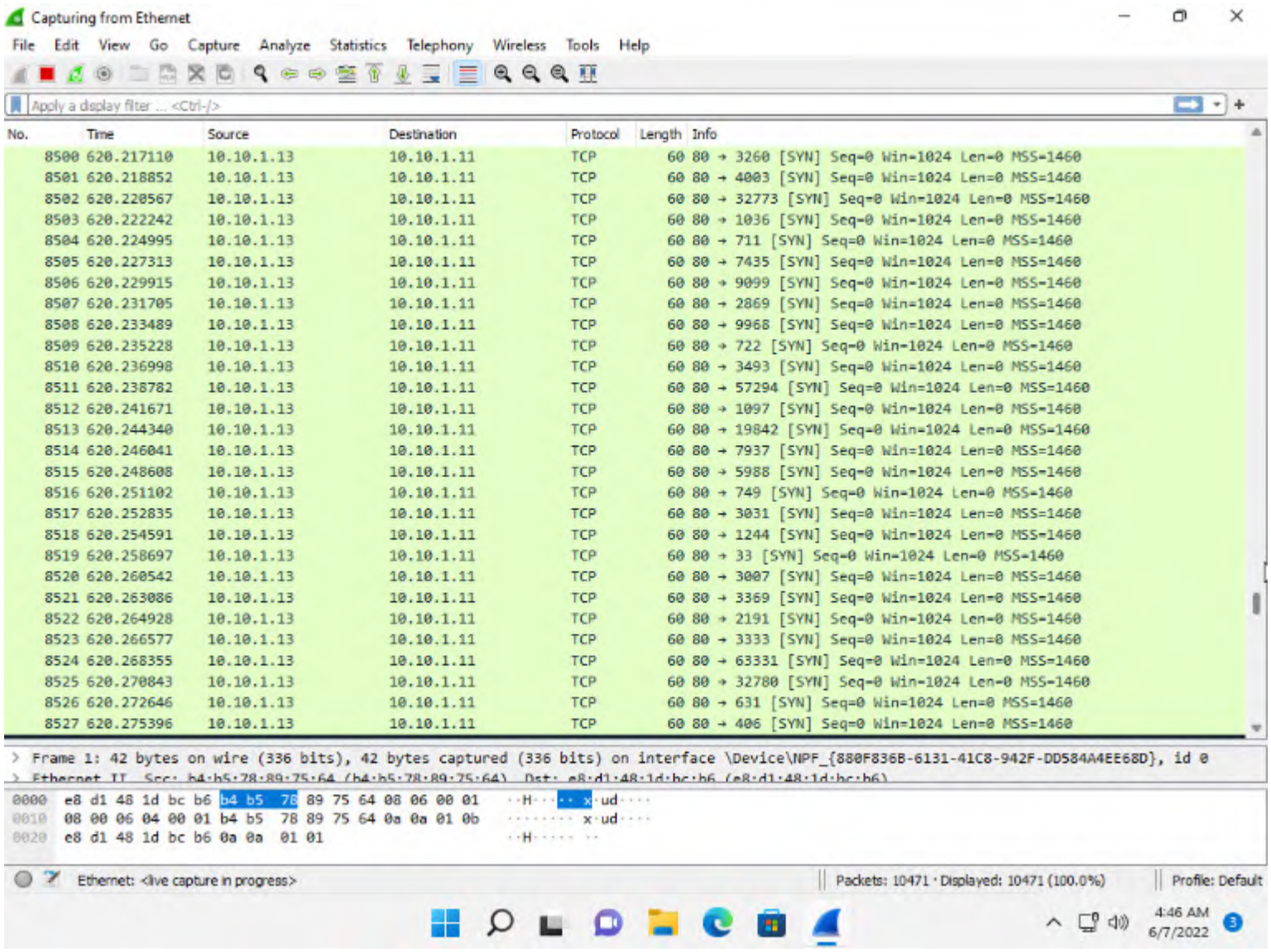
14. In the **Parrot Terminal** window, type **nmap -g 80 [Target IP Address]**, (here, target IP address is **10.10.1.11**) and press **Enter**.

> Note: In this command, you can use the **-g** or **--source-port** option to perform source port manipulation.
>
> Note: Source port manipulation refers to manipulating actual port numbers with common port numbers to evade IDS/firewall: this is useful when the firewall is configured to allow packets from well-known ports like HTTP, DNS, FTP, etc.

15. The results appear, displaying all open TCP ports along with the name of services running on the ports, as shown in the screenshot.



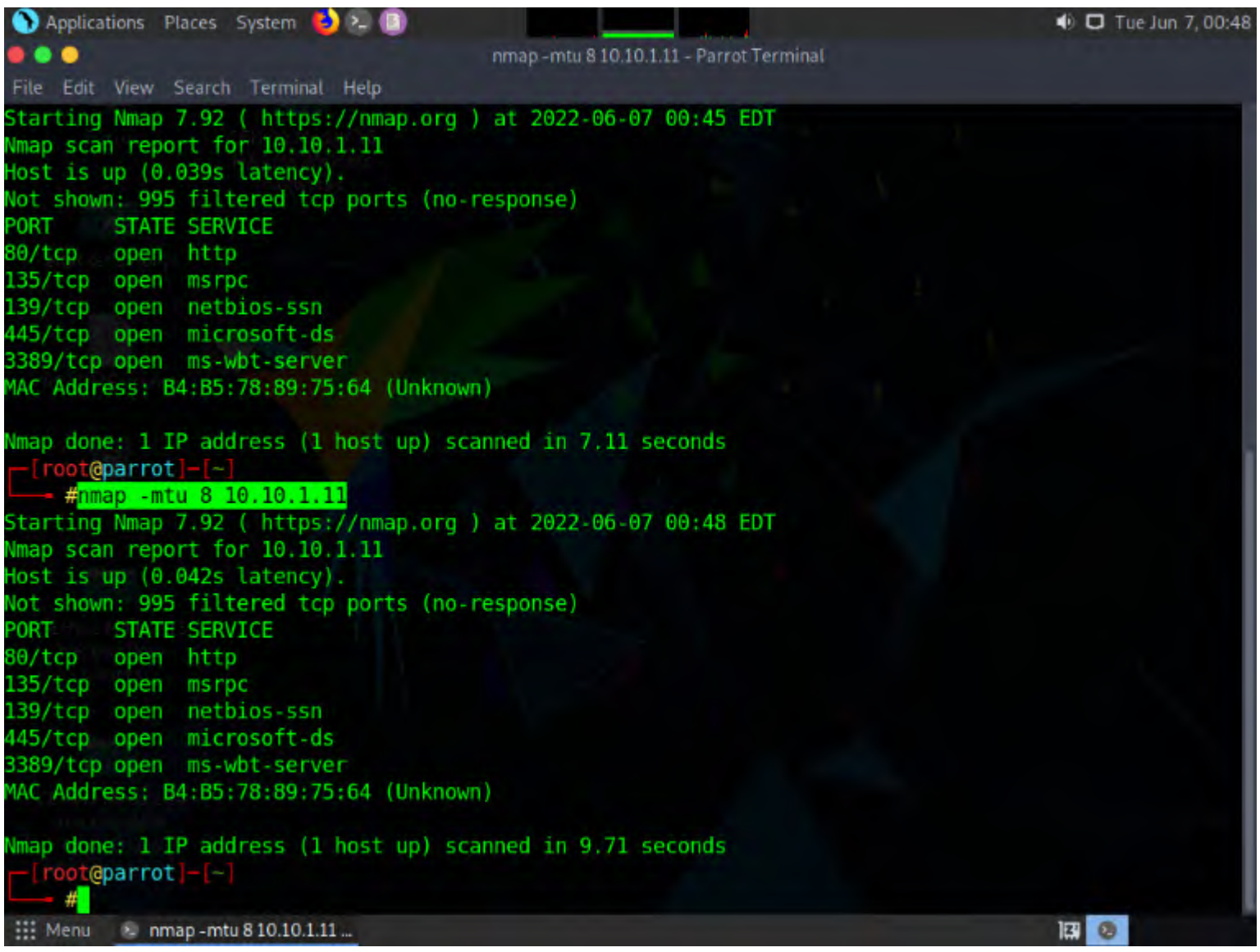
16. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine (target machine). In the Wireshark window, scroll-down and you can observe the TCP packets indicating that the port number 80 is used to scan other ports of the target host, as shown in the screenshot.



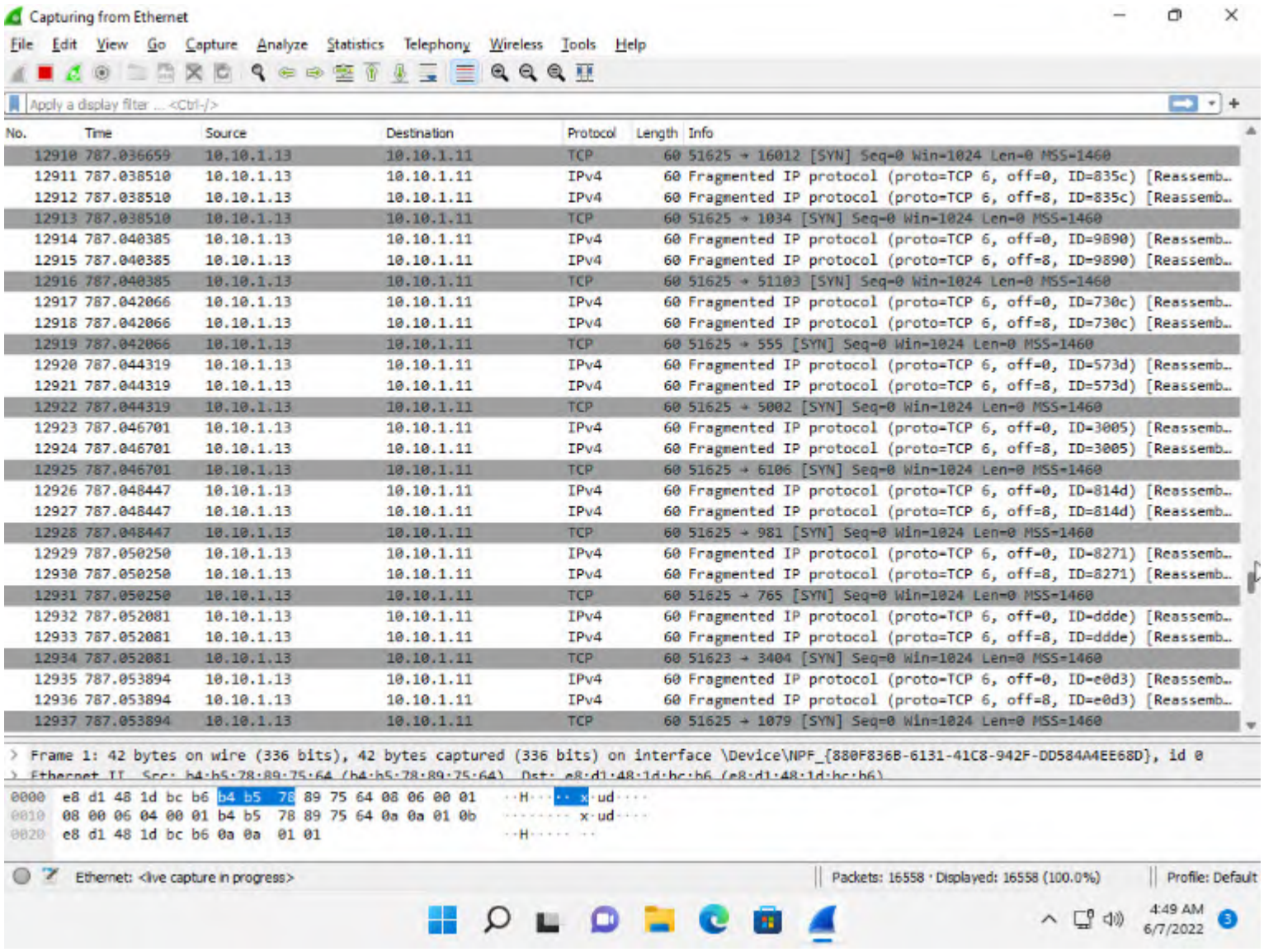
17. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.

18. Now, type **nmap -mtu 8 [Target IP Address]** (here, target IP address is **10.10.1.11**) and press **Enter**.

> Note: In this command, **-mtu**: specifies the number of Maximum Transmission Unit (MTU) (here, **8** bytes of packets).
>
> Note: Using MTU, smaller packets are transmitted instead of sending one complete packet at a time. This technique evades the filtering and detection mechanism enabled in the target machine.



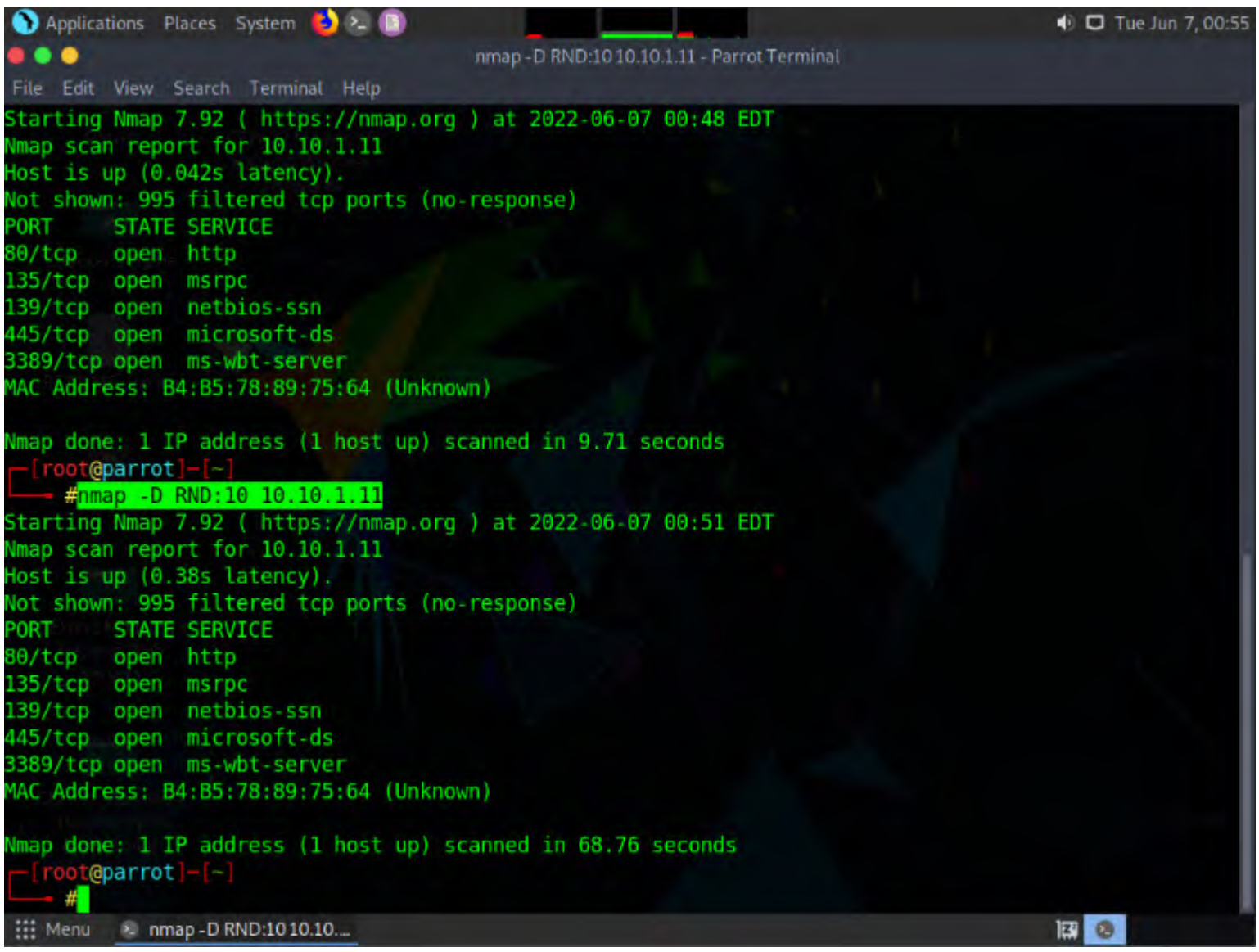
19. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine (target machine). In the Wireshark window, scroll-down and you can observe the fragmented packets having maximum length as 8 bytes, as shown in the screenshot.



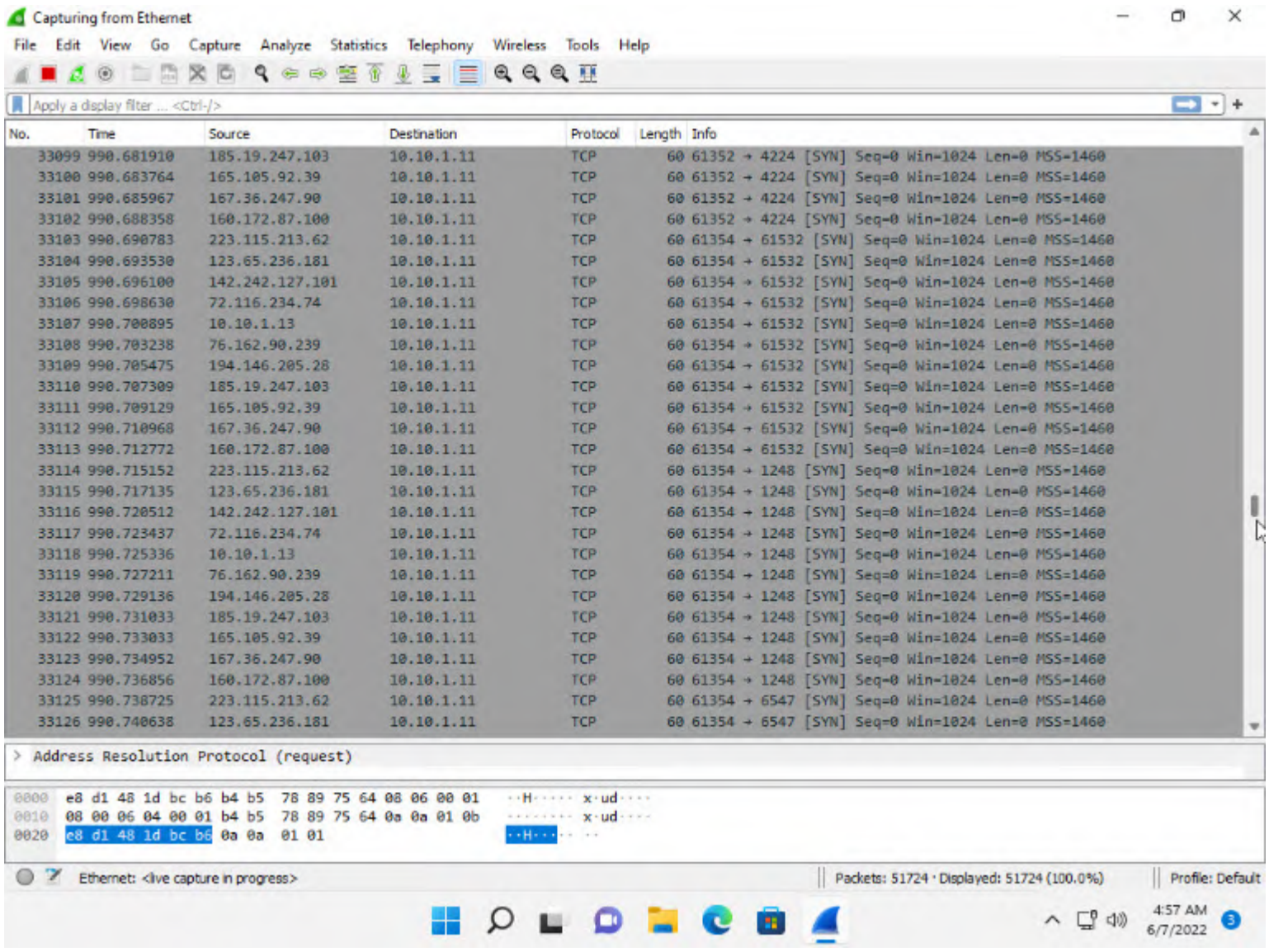
20. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.

21. Now, type **nmap -D RND:10 [Target IP Address]** (here, target IP address is **10.10.1.11**) and press **Enter**.

> Note: In this command, **-D**: performs a decoy scan and **RND**: generates a random and non-reserved IP addresses (here, **10**).
>
> Note: The IP address decoy technique refers to generating or manually specifying IP addresses of the decoys to evade IDS/firewall. This technique makes it difficult for the IDS/firewall to determine which IP address was actually scanning the network and which IP addresses were decoys. By using this command, Nmap automatically generates a random number of decoys for the scan and randomly positions the real IP address between the decoy IP addresses.



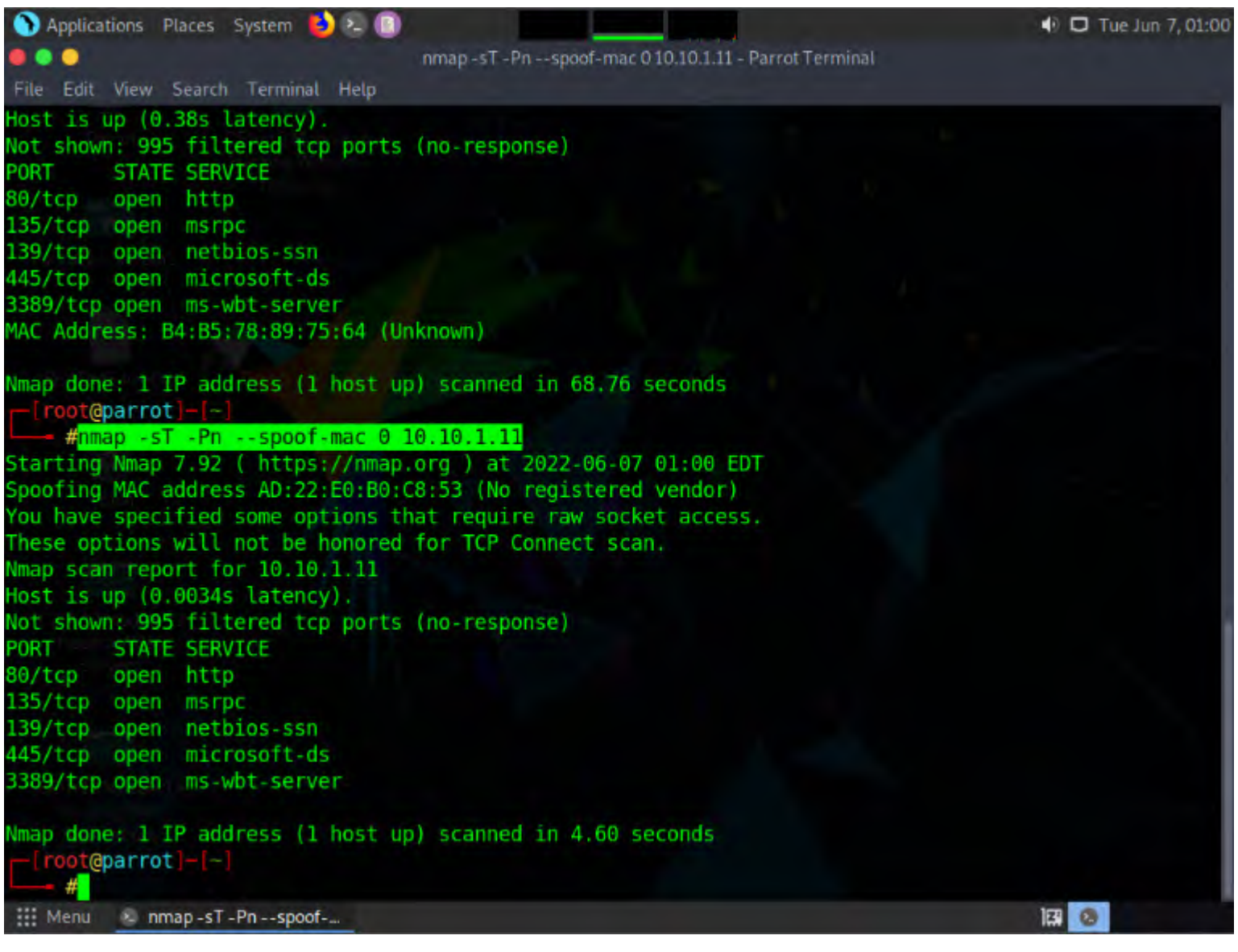
22. Now, click **CEHv12 Windows 11** to switch to the **Windows 11** machine (target machine). In the Wireshark window, scroll-down and you can observe the packets displaying the multiple IP addresses in the source section, as shown in the screenshot.



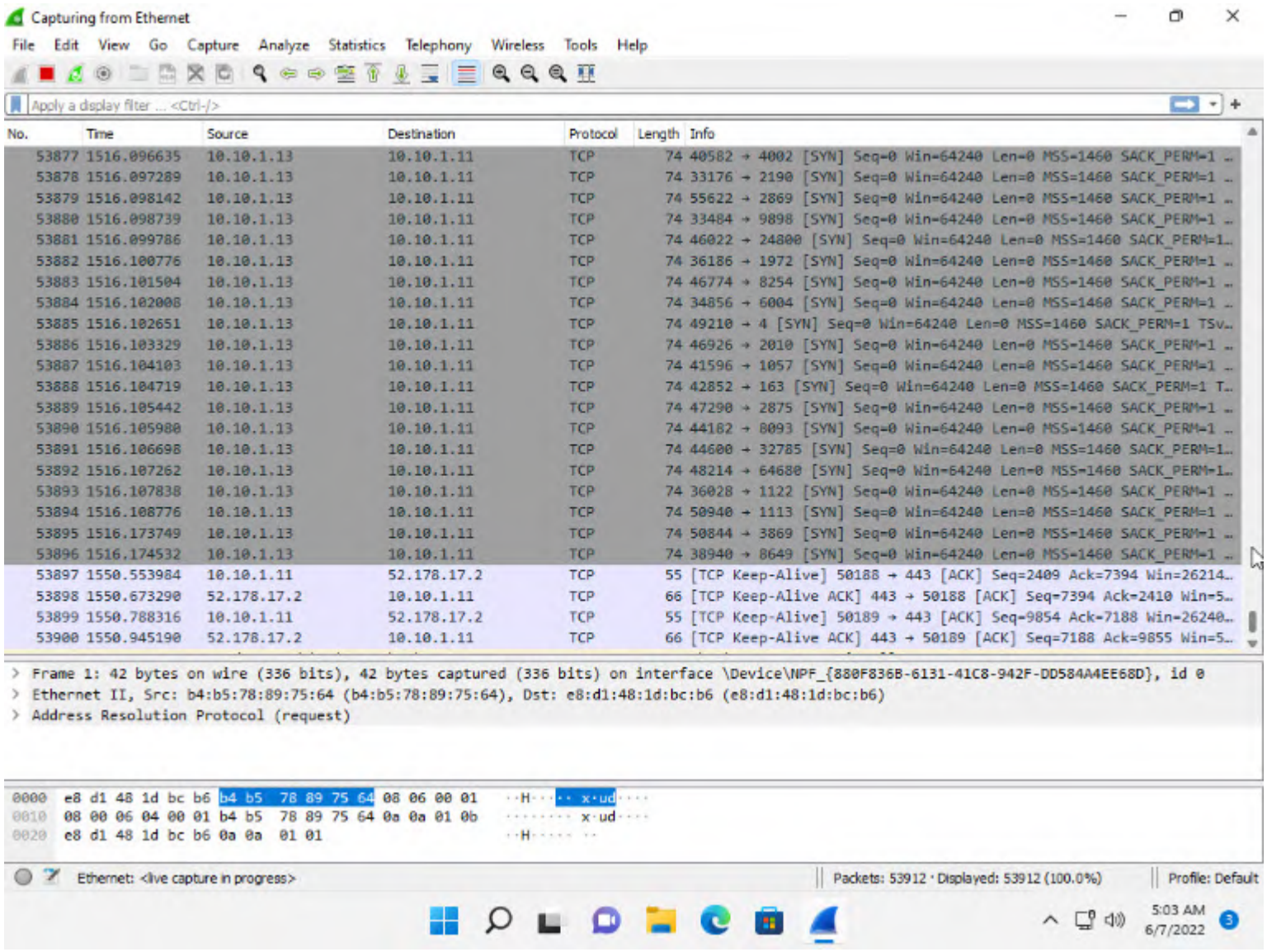
23. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.

24. In the terminal window type **nmap -sT -Pn --spoof-mac 0 [Target IP Address]** (here, target IP address is **10.10.1.11**) and press **Enter**.

> Note: In this command **--spoof-mac 0** represents randomizing the MAC address, **-sT**: performs the TCP connect/full open scan, **-Pn** is used to skip the host discovery.
>
> Note: MAC address spoofing technique involves spoofing a MAC address with the MAC address of a legitimate user on the network. This technique allows you to send request packets to the targeted machine/network pretending to be a legitimate host.



25. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine (target machine). In the Wireshark window, scroll-down and you can observe the captured TCP, as shown in the screenshot.



26. This concludes the demonstration of evading IDS and firewall using various evasion techniques in Nmap.
27. Close all open windows and document all the acquired information.

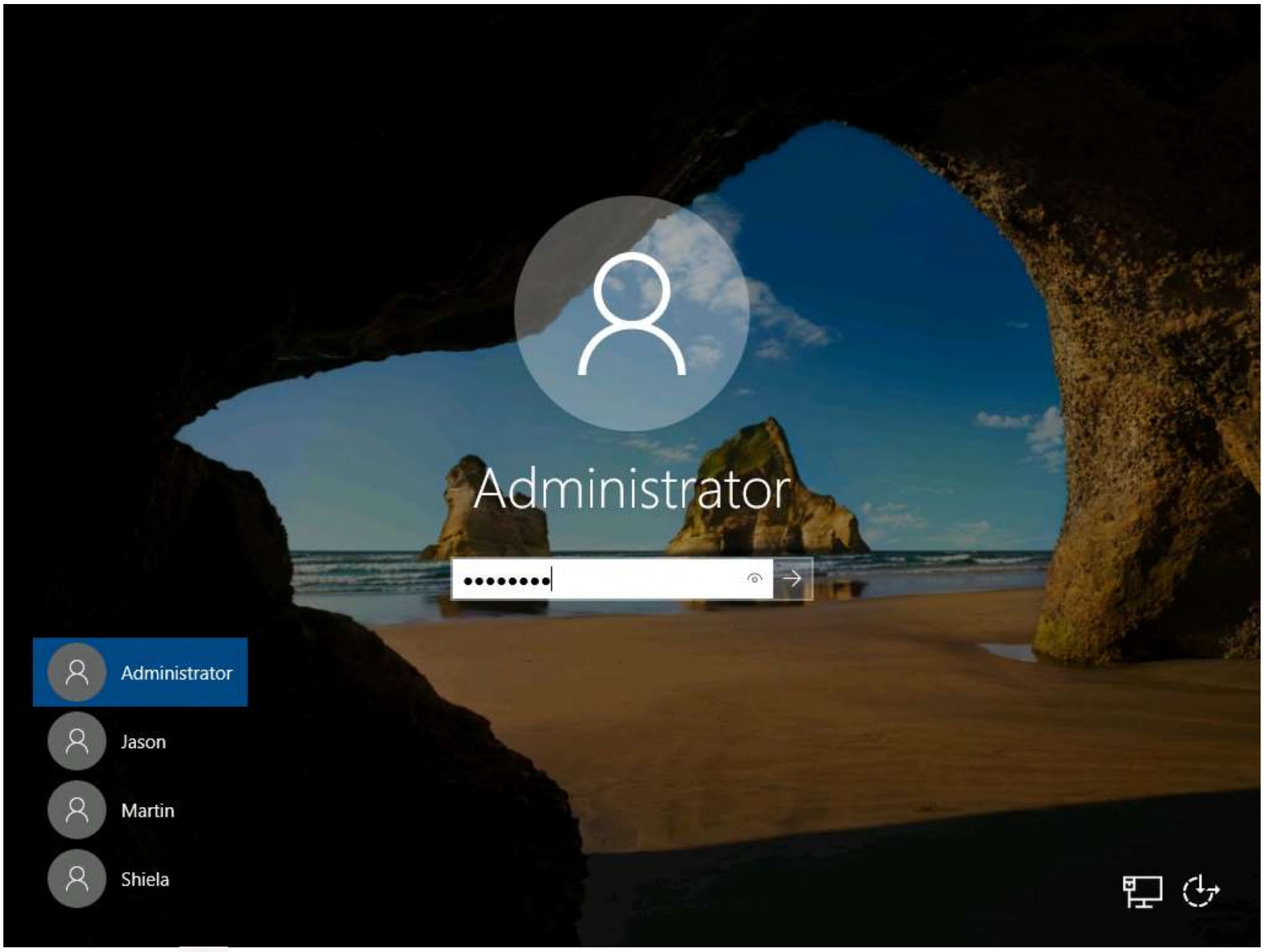
Task 2: Create Custom Packets using Colasoft Packet Builder to Scan beyond the IDS/Firewall

Colasoft Packet Builder is a tool that allows you to create custom network packets to assess network security. You can also select a TCP packet from the provided templates and change the parameters in the decoder editor, hexadecimal editor, or ASCII editor to create a packet. In addition to building packets, the Colasoft Packet Builder supports saving packets to packet files and sending packets to the network.

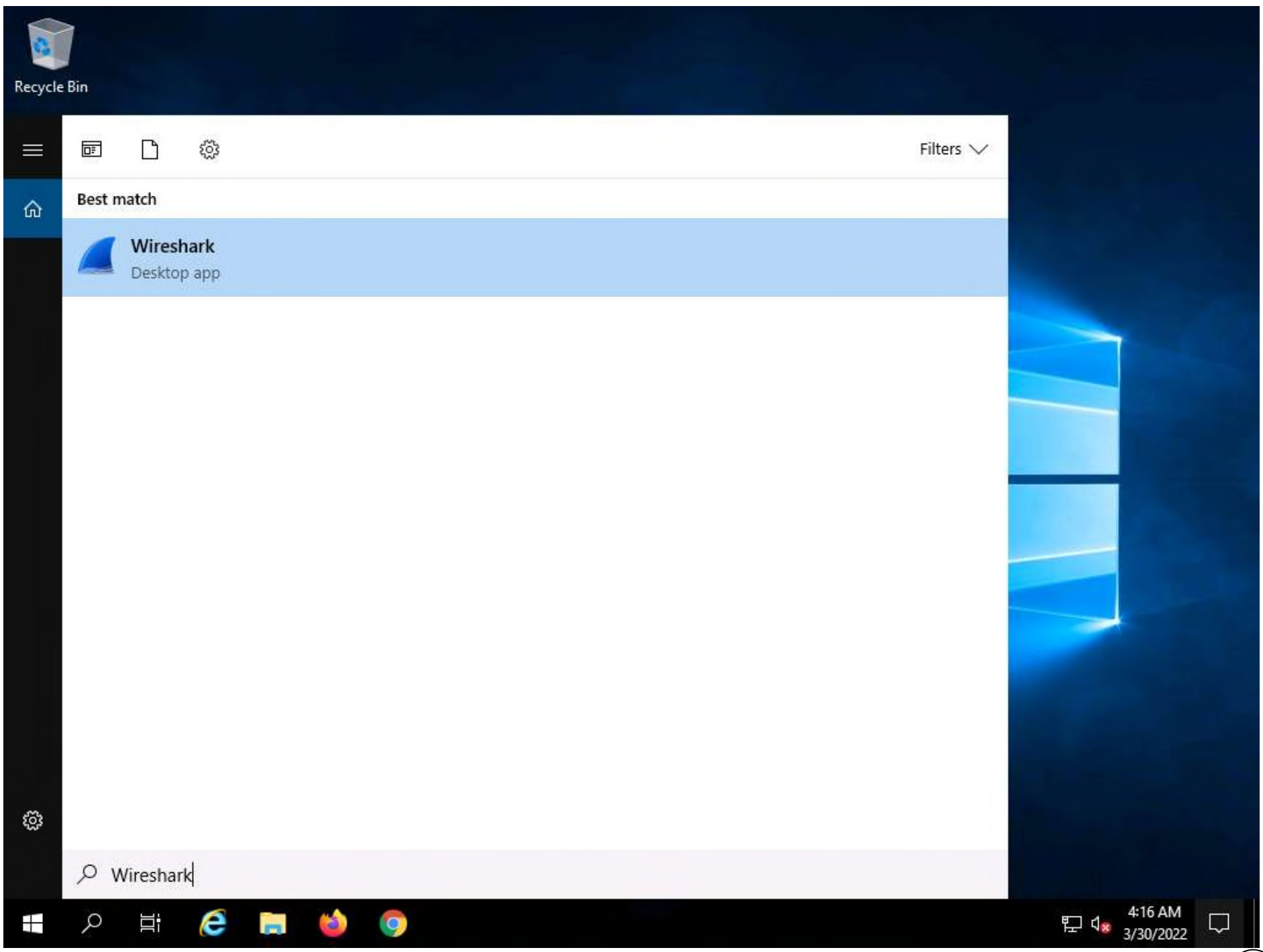
Here, we will use the Colasoft Packet Builder tool to create custom TCP packets to scan the target host by bypassing the IDS/firewall.

1. Click **CEHv12 Windows Server 2019** to switch to the **Windows Server 2019** machine.
2. Click **Ctrl+Alt+Del** to activate the machine. By default, **Administrator** user profile is selected, type **Pa$$w0rd** in the **Password** field and press **Enter** to login.

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

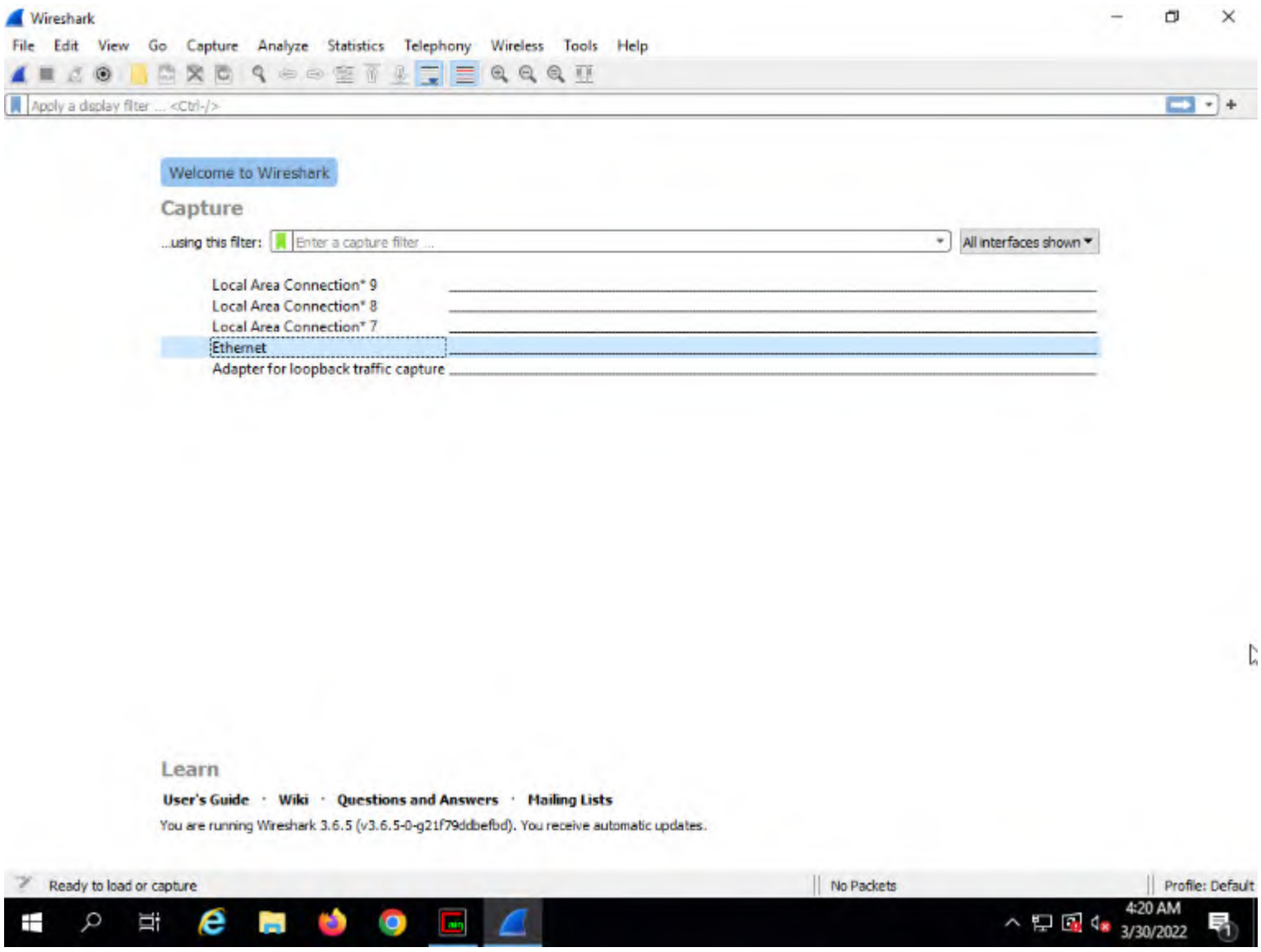


3. Click **Search** icon ( ) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Wireshark** to launch it.

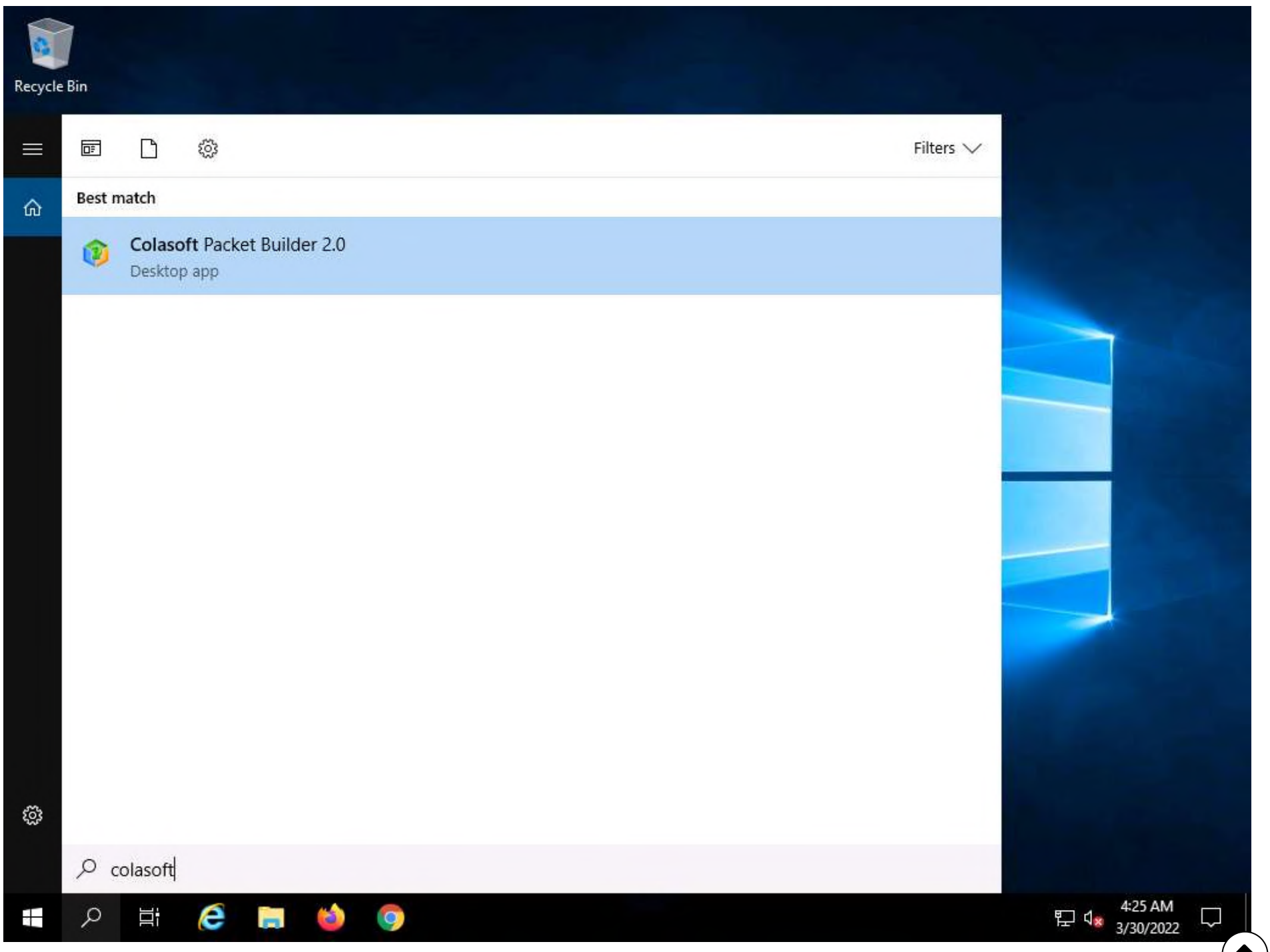


4. The **Wireshark Network Analyzer** main window appears; double-click the available ethernet or interface (here, **Ethernet**) to start the packet capture.

> Note: If a **Software Update** pop-up appears click on **Remind me later**.

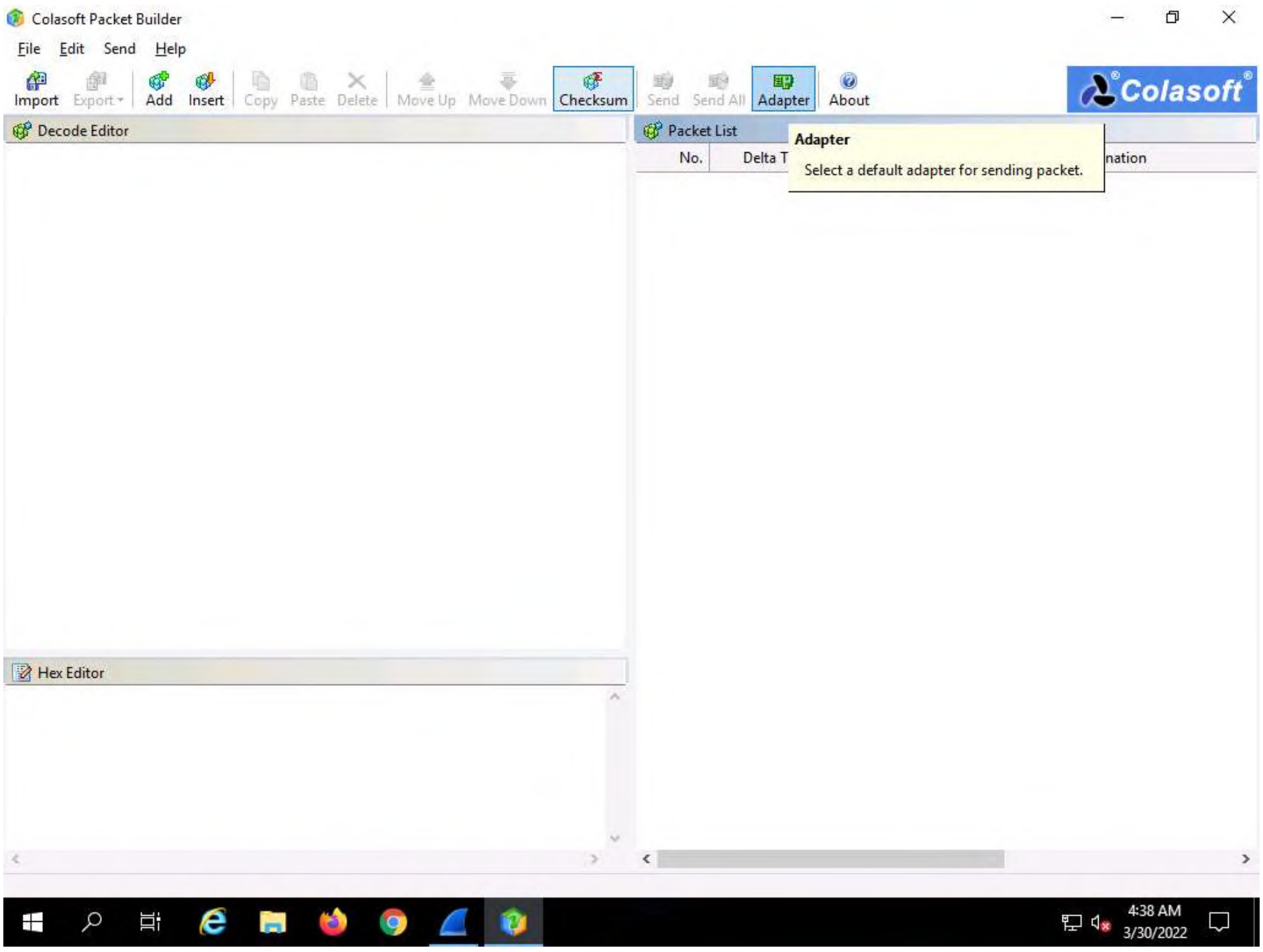


5. Minimize the **Wireshark** window, click **Search** icon ( ) on the **Desktop**. Type **colasoft** in the search field, the **Colasoft Packet Builder 2.0** appears in the results, click **Colasoft Packet Builder 2.0** to launch it.

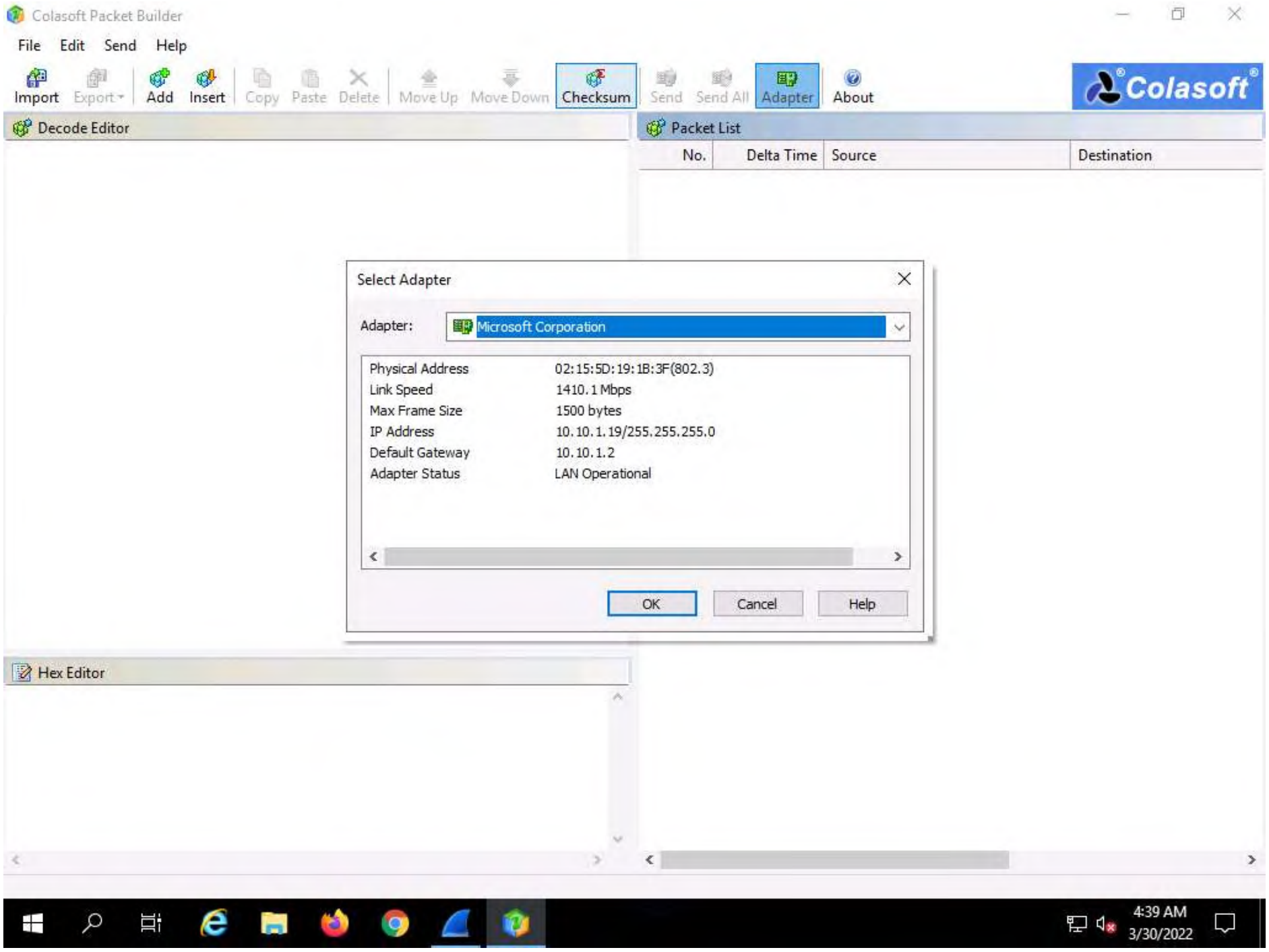


6. The **Colasoft Packet Builder** GUI appears; click on the **Adapter** icon, as shown in the screenshot.

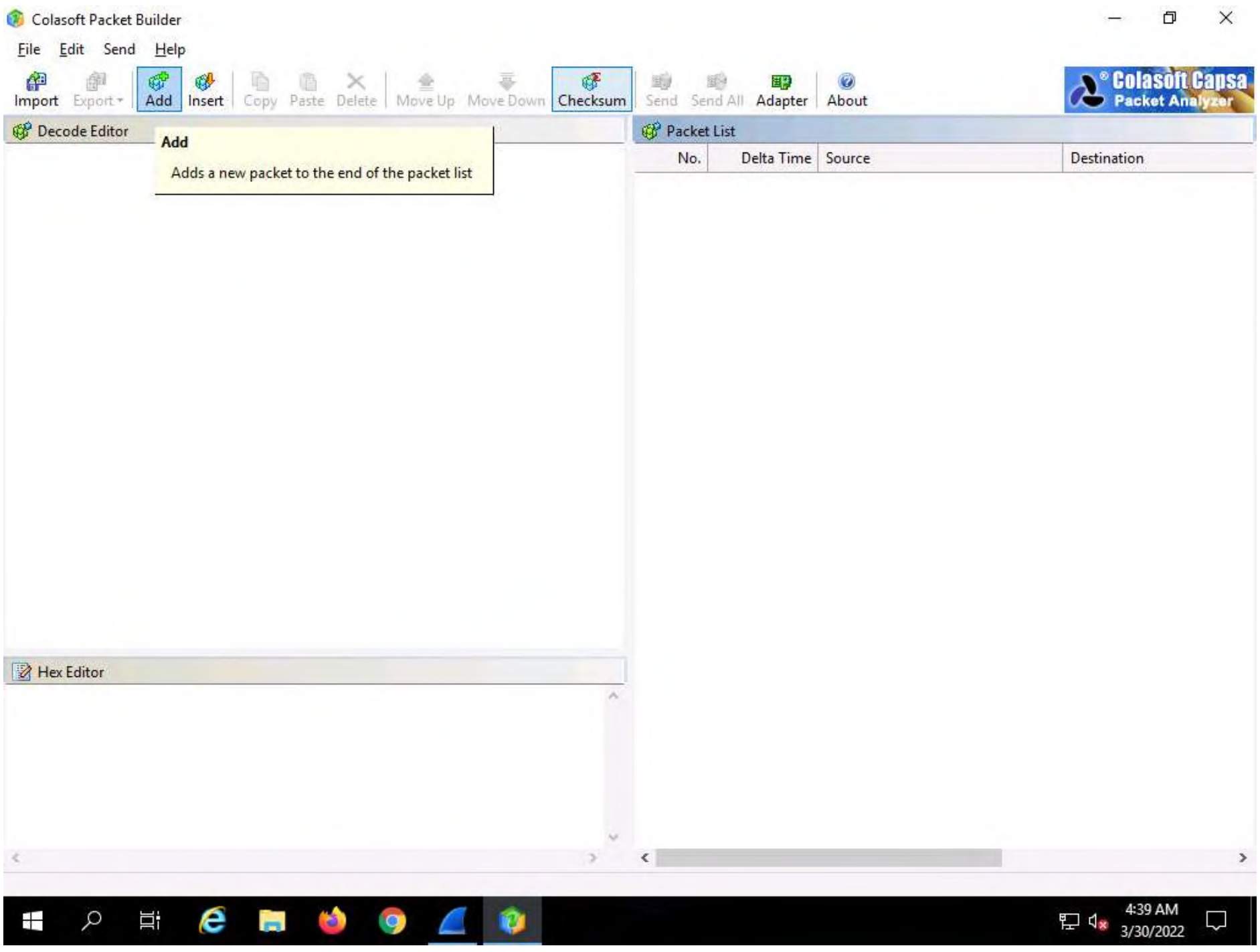
Note: If a pop-up appears, close the window.



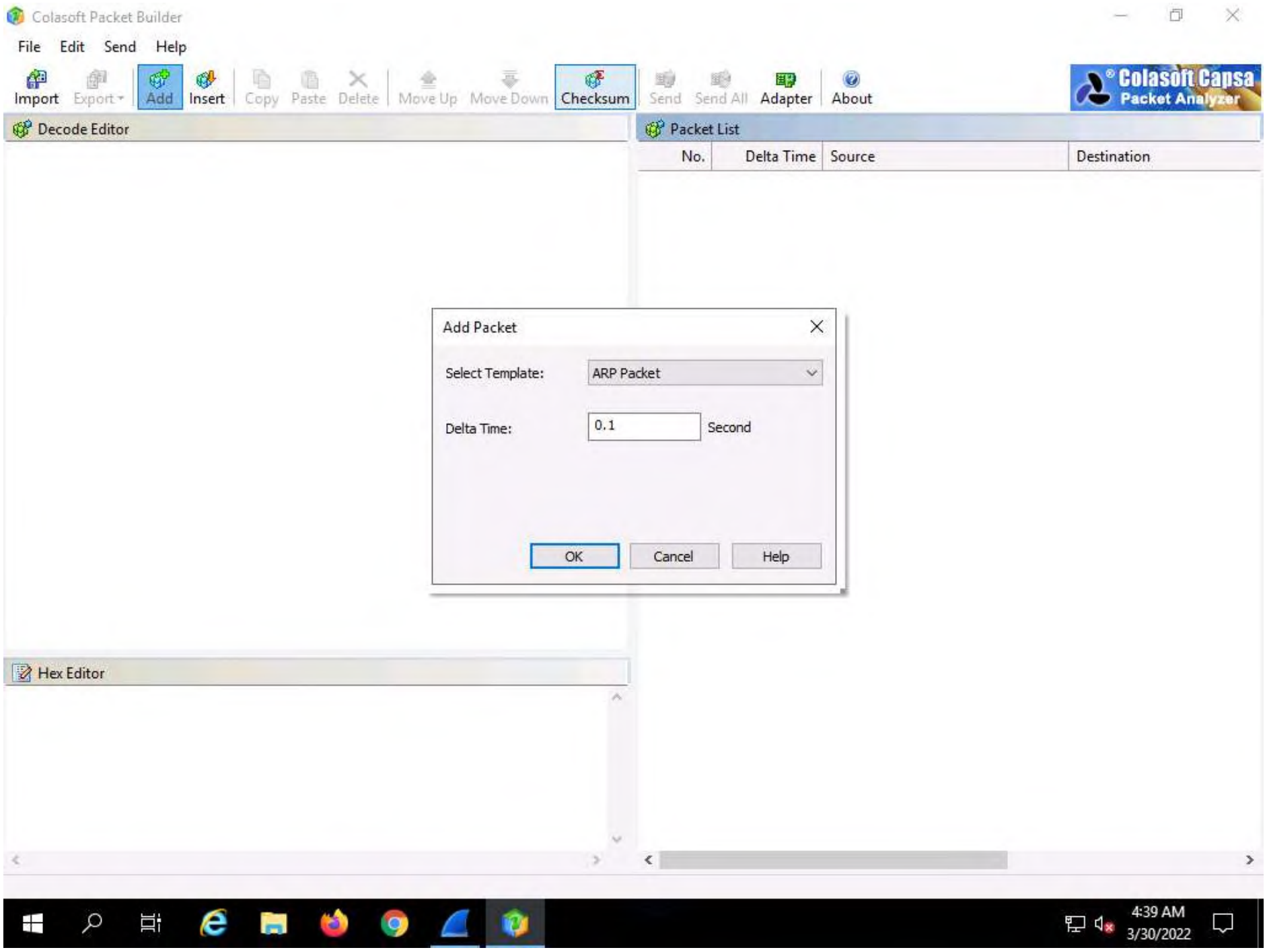
7. When the **Select Adapter** window appears, check the **Adapter** settings and click **OK**.



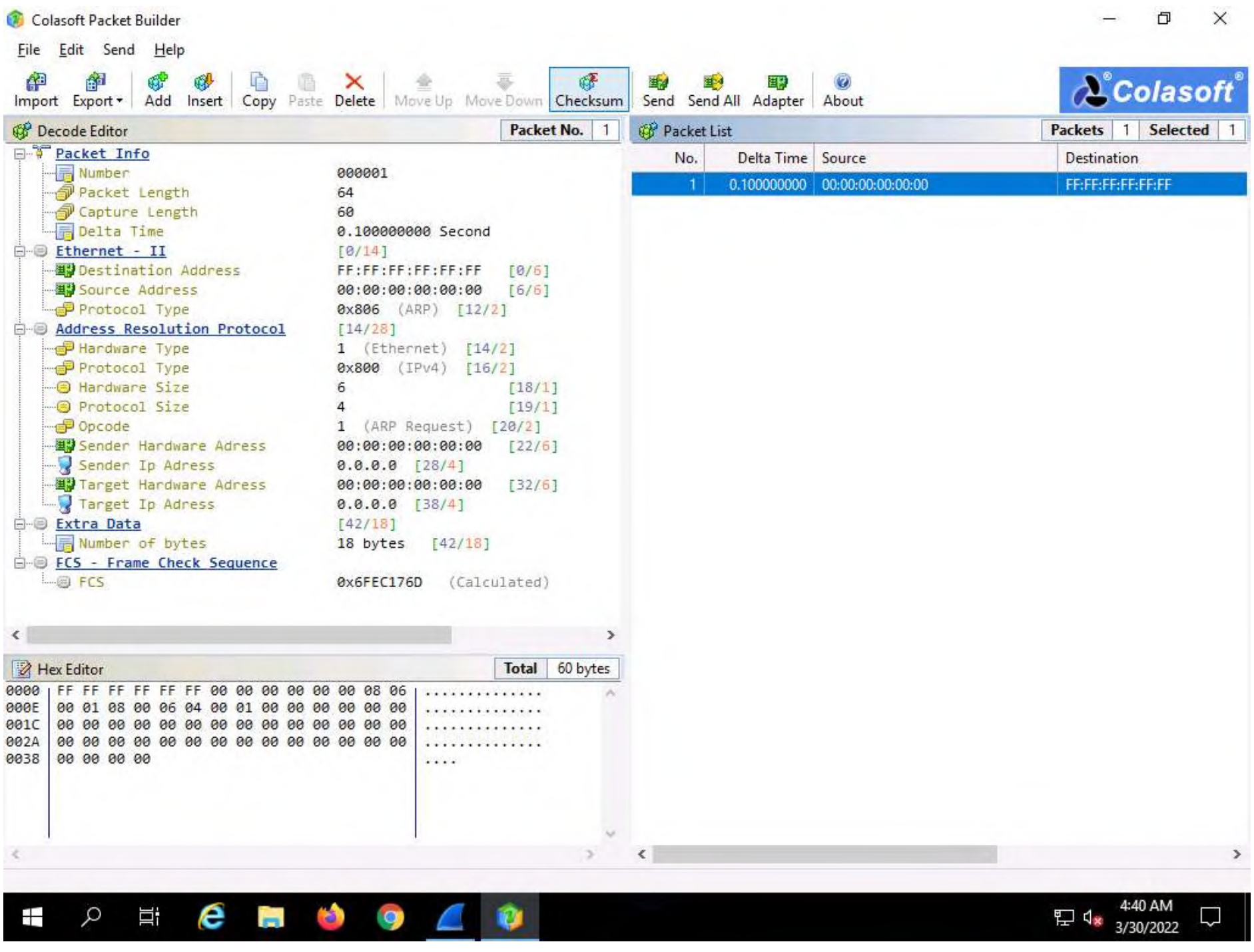
8. To add or create a packet, click the **Add** icon in the **Menu** bar.



9. In the **Add Packet** dialog box, select the **ARP Packet** template, set **Delta Time** as **0.1** seconds, and click **OK**.



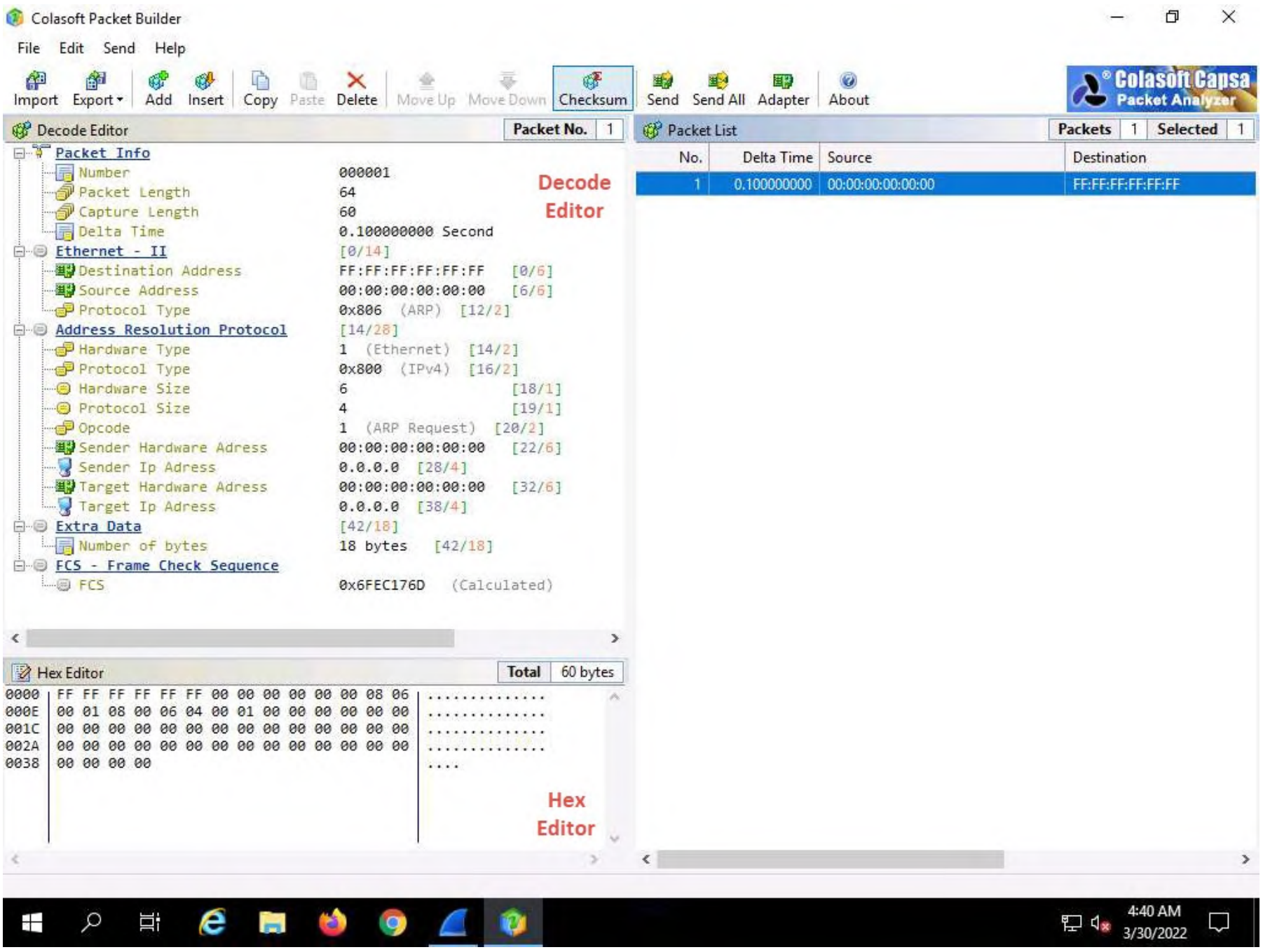
10. You can view the added packets list on the right-hand side of the window, under **Packet List**.



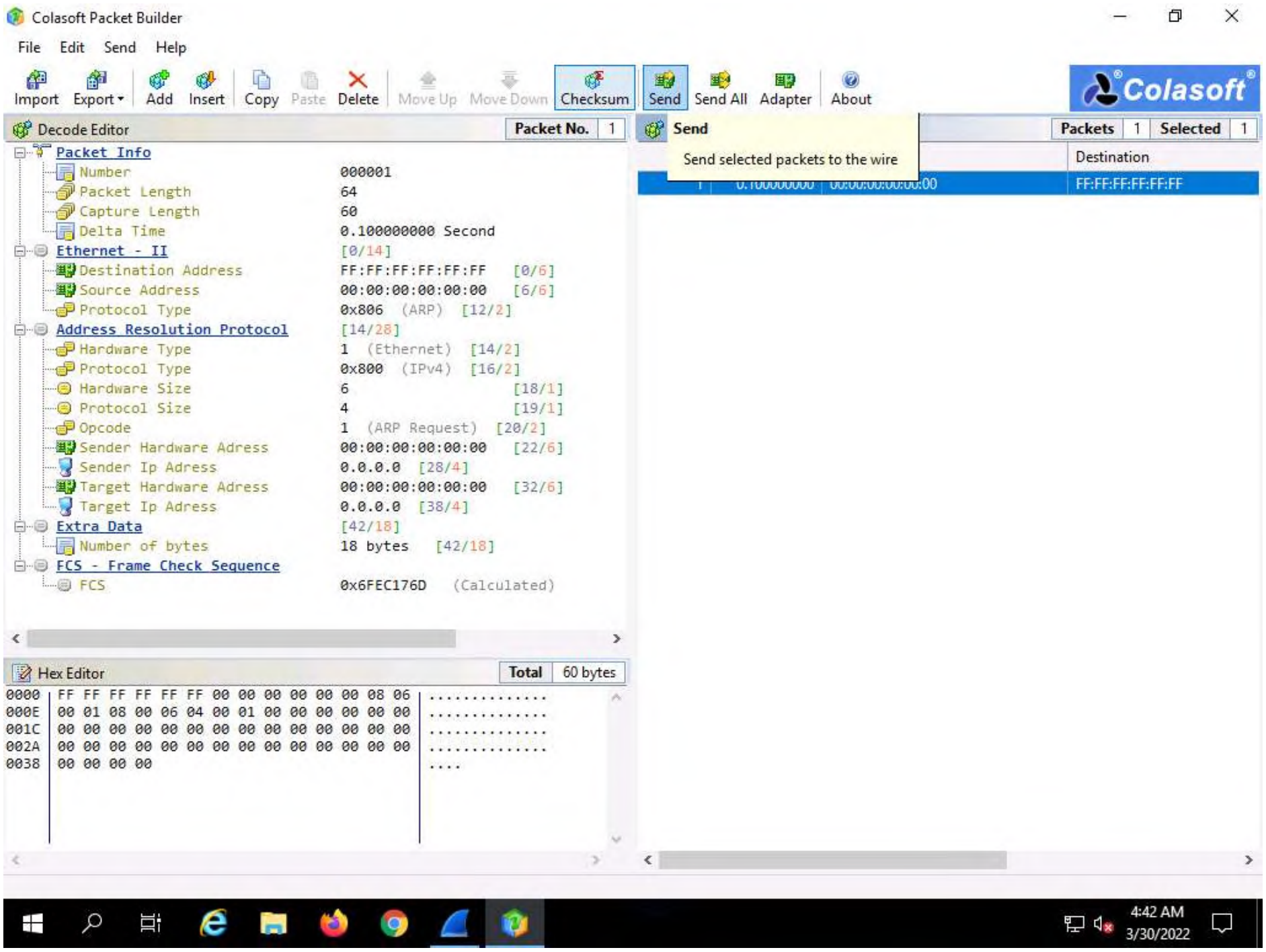
11. **Colasoft Packet Builder** allows you to edit the decoding information in the two editors, **Decode Editor** and **Hex Editor**, located in the left pane of the window.

The **Decode Editor** section allows you to edit the packet decoding information by double-clicking the item that you wish to decode.

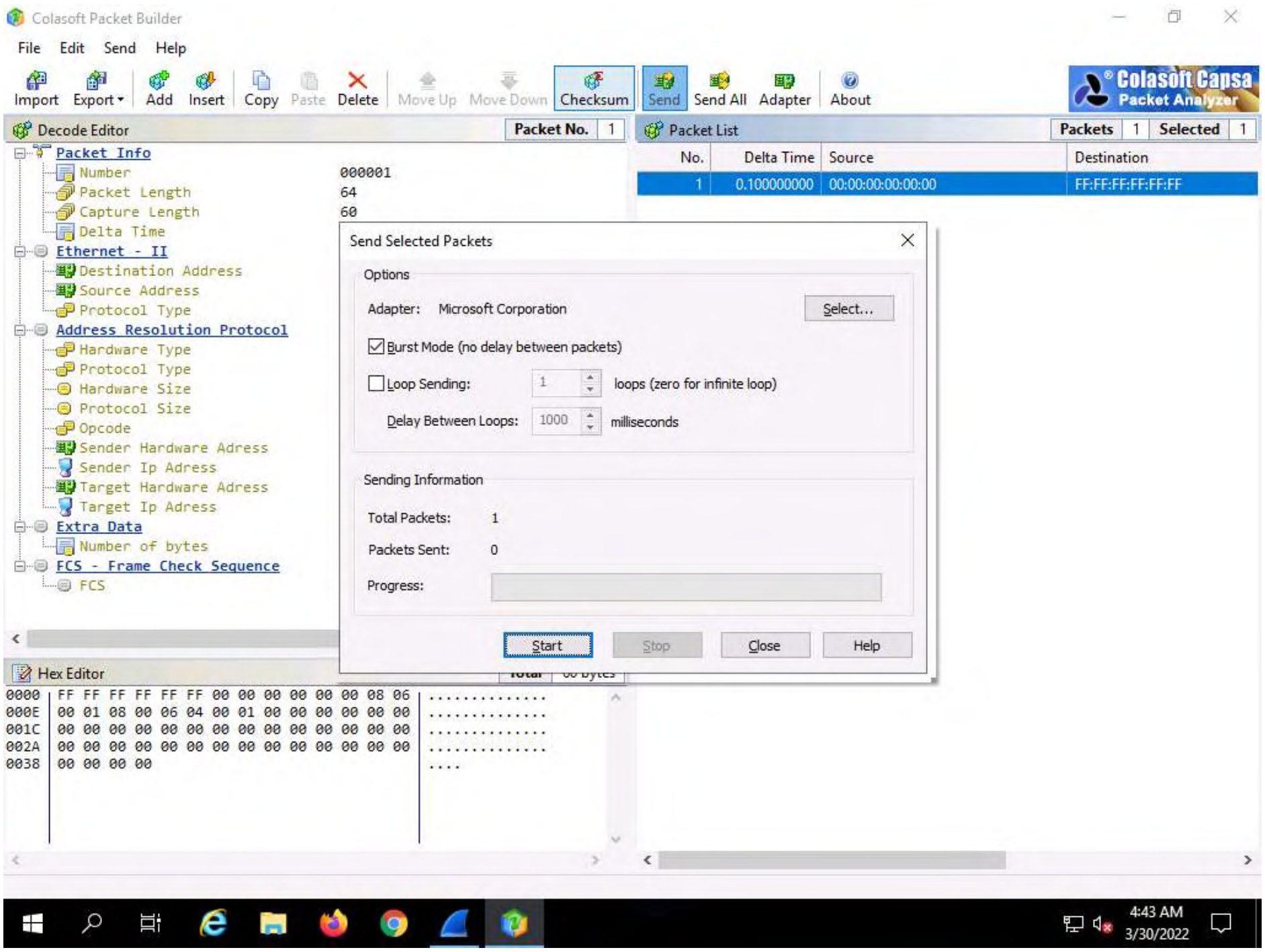
Hex Editor displays the actual packet contents in raw hexadecimal value on the left and its ASCII equivalent on the right.



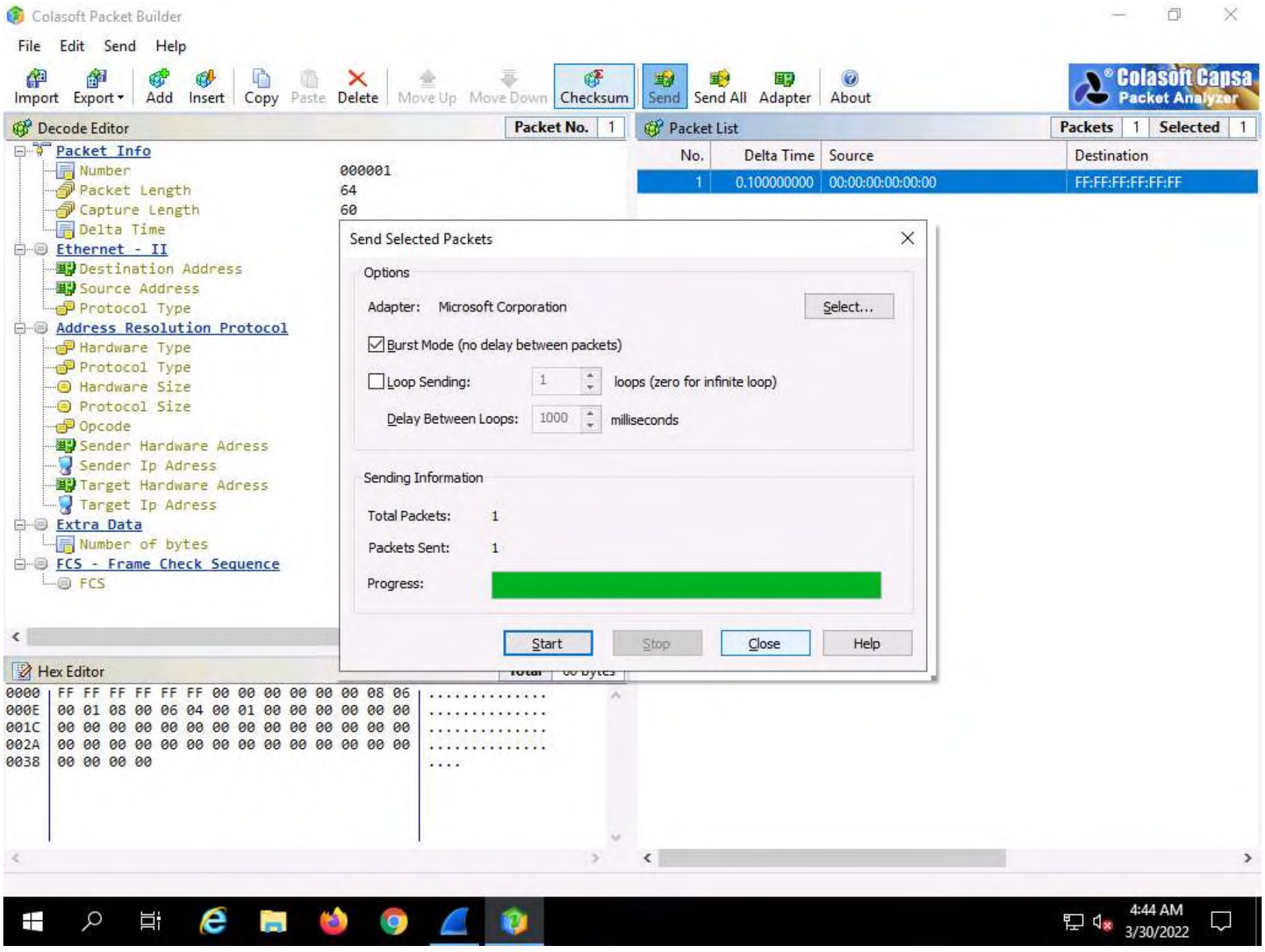
12. To send the packet, click **Send** from the **Menu** bar.



13. In the **Send Selected Packets** window, select the **Burst Mode (no delay between packets)** option, and then click **Start**.



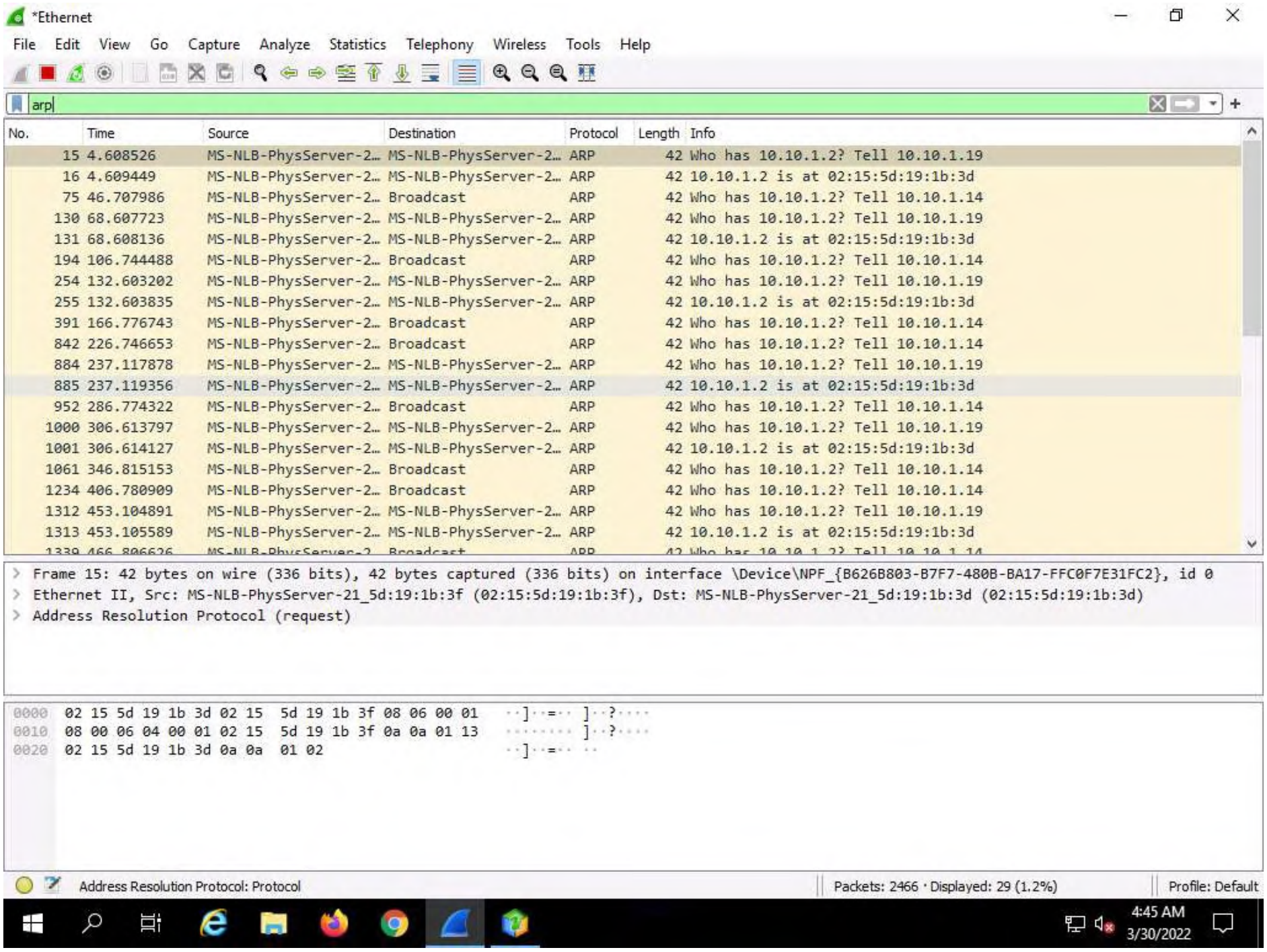
14. After the **Progress** bar completes, click **Close**.



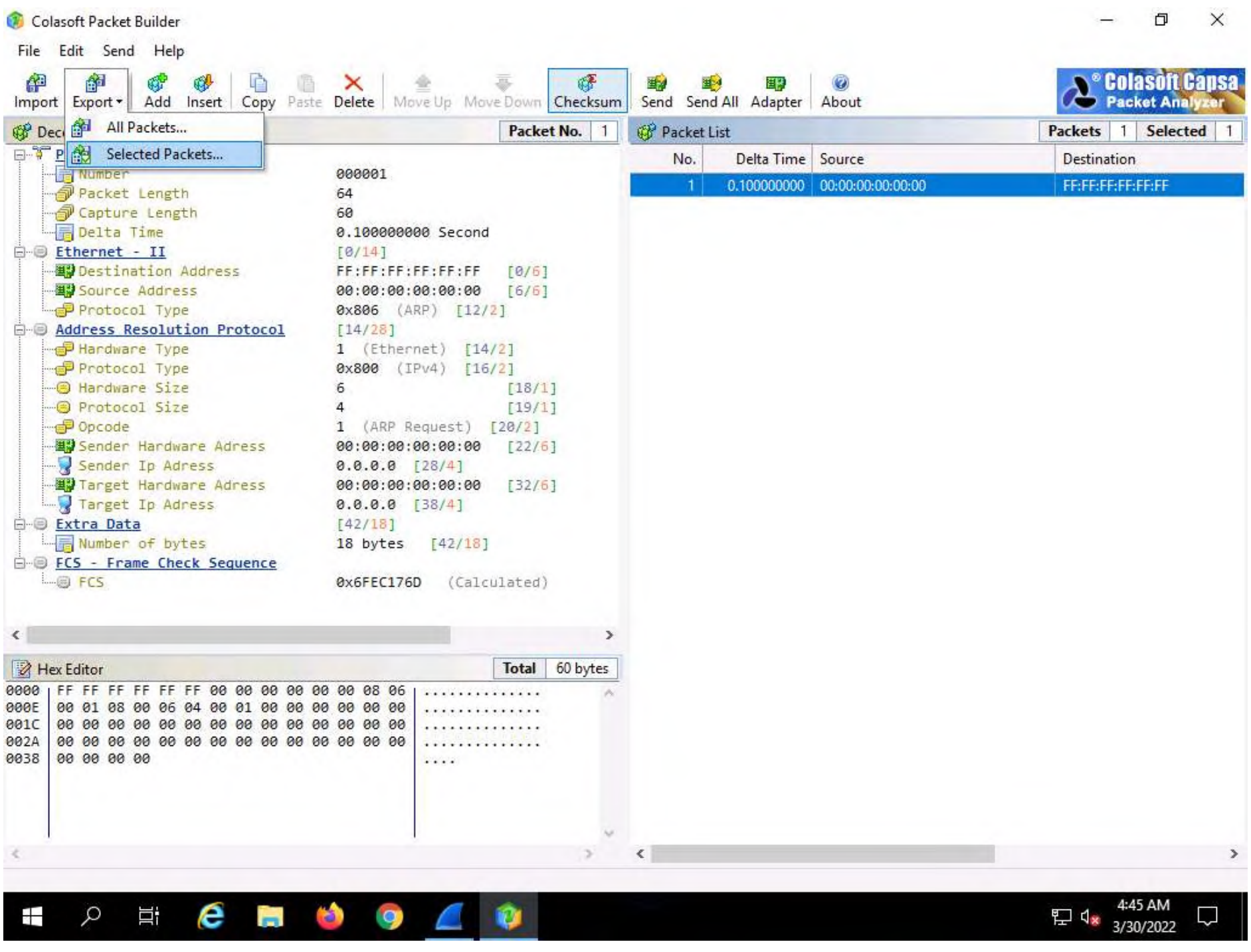
15. Now, when this ARP packet is broadcasted in the network, the active machines receive the packet, and a few start responding with an ARP reply. To evaluate which machine is responding to the ARP packet, you need to observe packets captured by the **Wireshark** tool.

16. In the **Wireshark** window, click on the **Filter** field, type **arp** and press **Enter**. The ARP packets will be displayed, as shown in the screenshot.

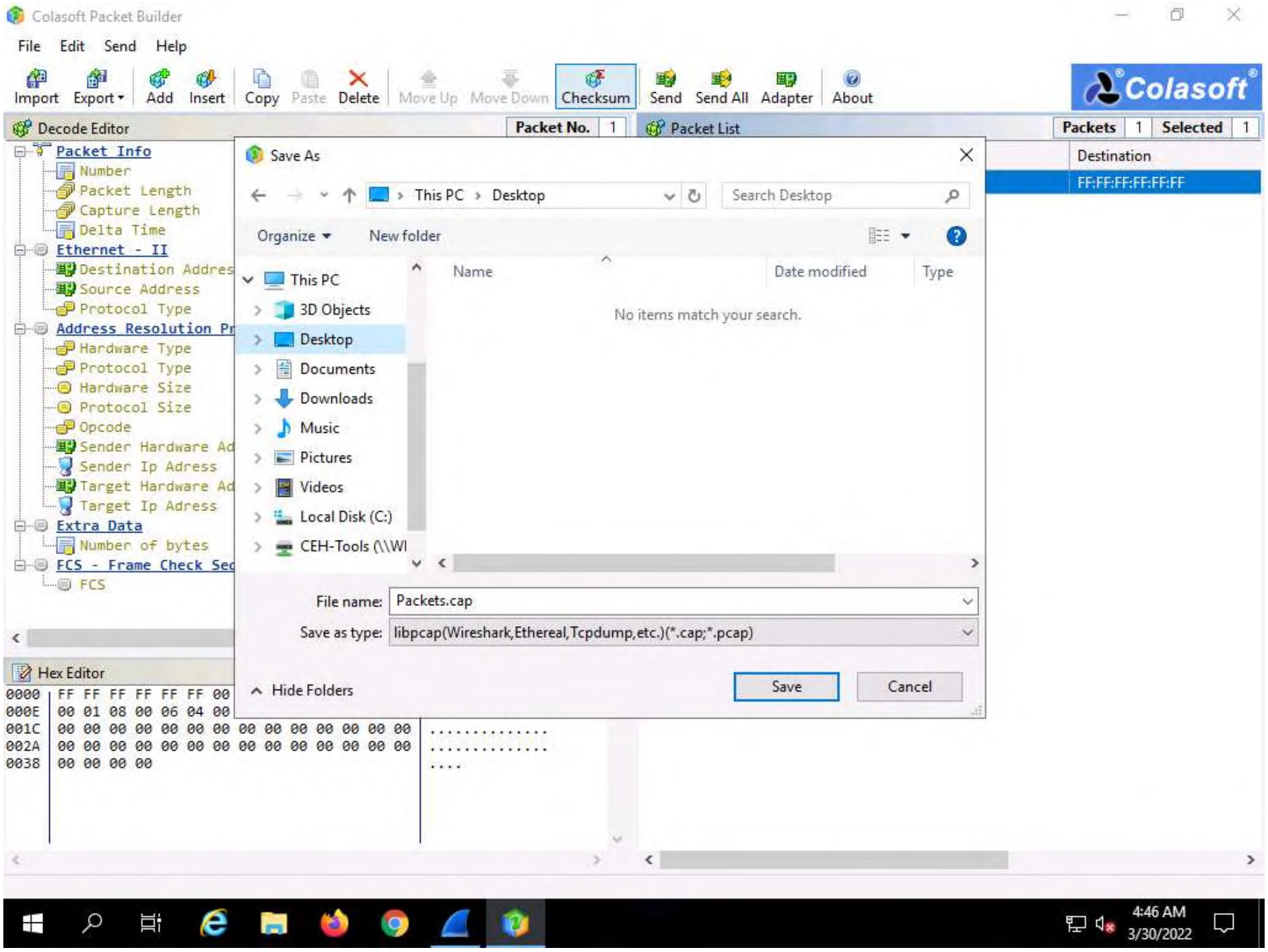
Note: Here, the host machine (**10.10.1.19**) is broadcasting ARP packets, prompting the target machines to reply to the message.



17. Switch back to the **Colasoft Packet Builder** window, to export the packet, click **Export** --> **Selected Packets...**.



18. In the **Save As** window, select a destination folder in the **Save in** field, specify **File name** and **Save as type**, and click **Save**.



19. This saved file can be used for future reference.
20. Attackers can use this packet builder to create fragmented packets to bypass network firewalls and IDS systems. They can also create packets and flood the victim with a very large number of packets, which could result in DoS attacks.
21. This concludes the demonstration of creating a custom TCP packets to scan the target host by bypassing the IDS/firewall.
22. Close all open windows and document all the acquired information.

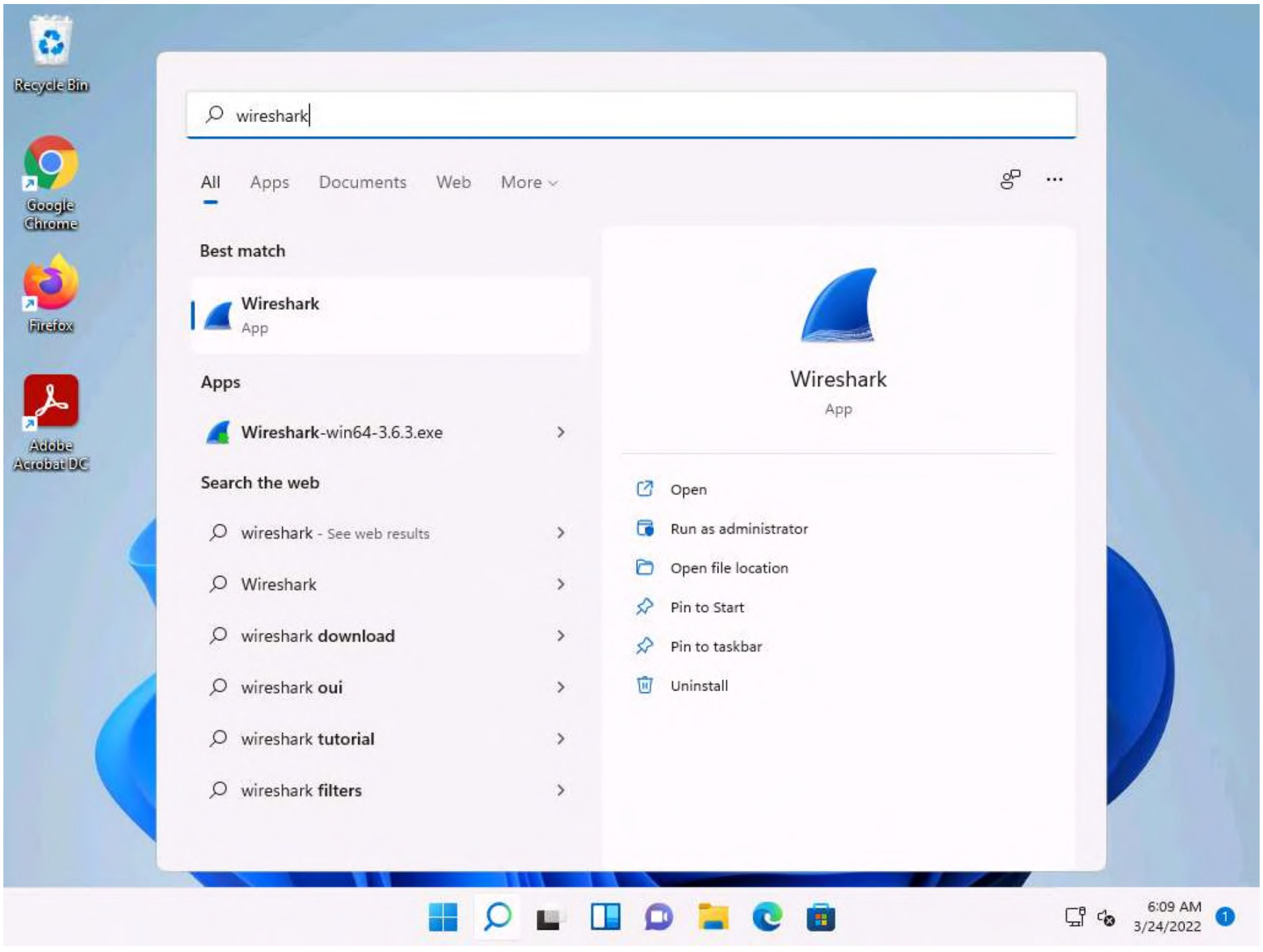
Task 3: Create Custom UDP and TCP Packets using Hping3 to Scan beyond the IDS/Firewall

Hping3 is a scriptable program that uses the TCL language, whereby packets can be received and sent via a binary or string representation describing the packets.

Here, we will use Hping3 to create custom UDP and TCP packets to evade the IDS/firewall in the target network.

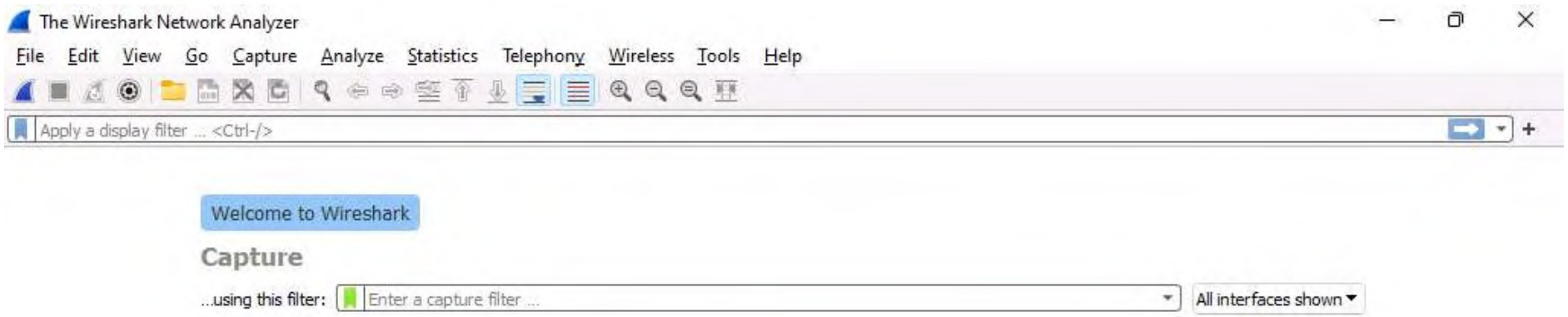
> Note: Before beginning this task, ensure that the **Windows Defender Firewall** in the **Windows 11** machine is enabled.

1. Click **CEHv12 Windows 11** to switch to the **Windows 11** machine.
2. Click **Search** icon ( ) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Open** to launch it.



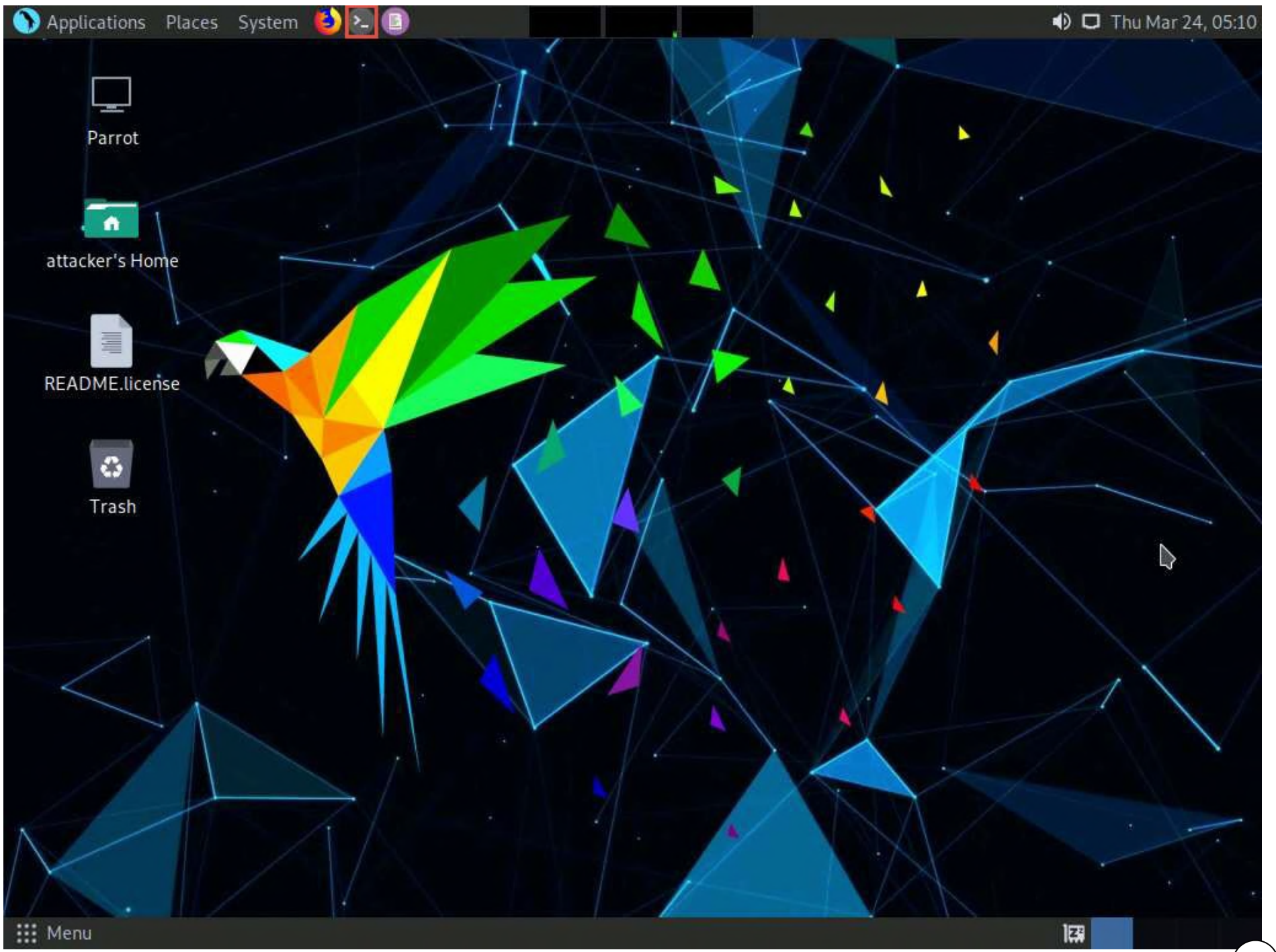
3. The **Wireshark Network Analyzer** window appears, double-click the available ethernet or interface (here, **Ethernet**) to start the packet capture.

Note: If a **Software Update** pop-up appears click on **Remind me later**.



4. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.

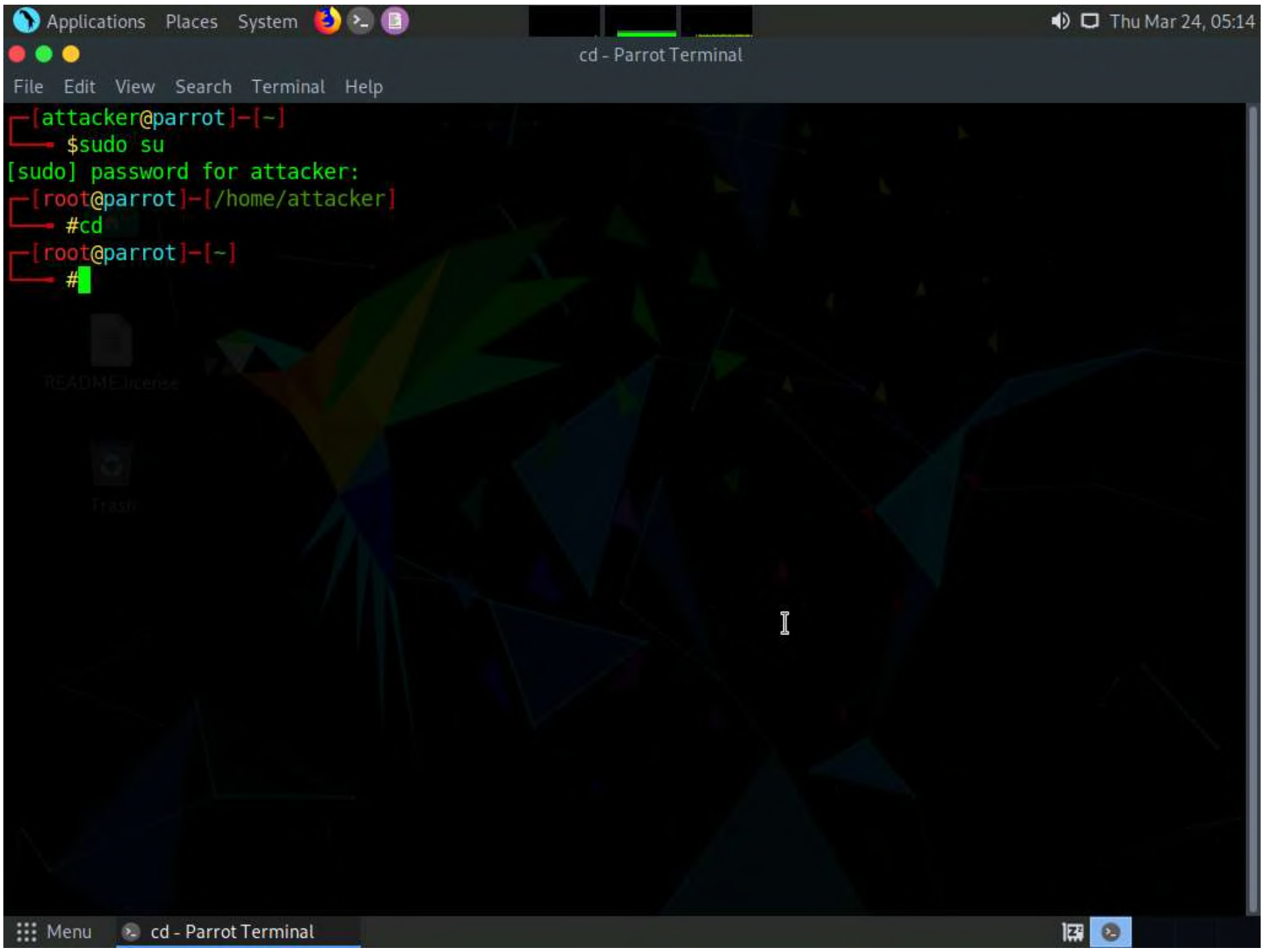
5. Click the **MATE Terminal** icon in the top-left corner of the **Desktop** to open a **Terminal** window.



6. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.
7. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

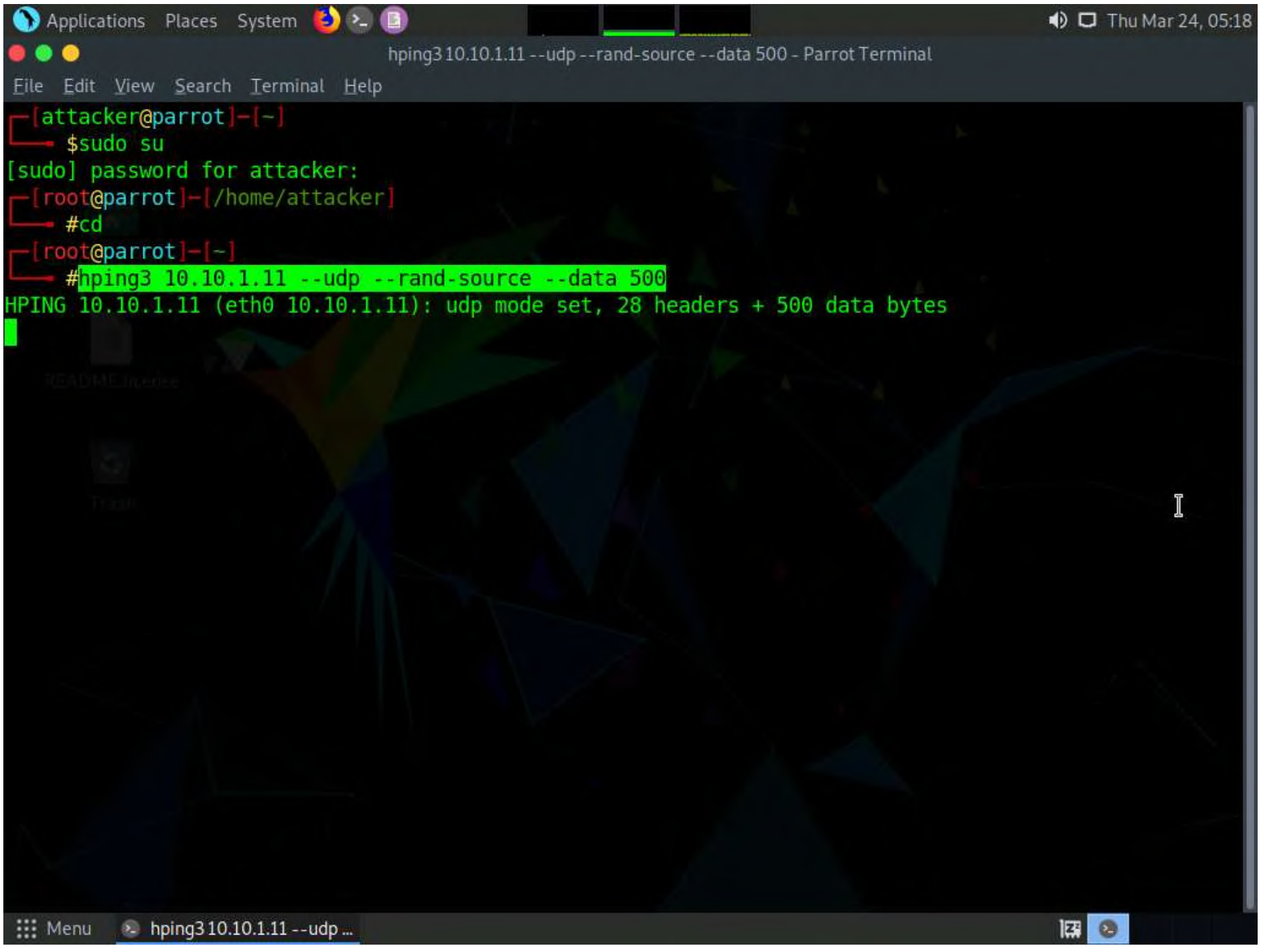
> Note: The password that you type will not be visible.

8. Now, type **cd** and press **Enter** to jump to the root directory.



9. In the **Parrot Terminal** window, type **hping3 [Target IP Address] --udp --rand-source --data 500** (here, the target machine is **Windows 11** [**10.10.1.11**]) and press **Enter**.

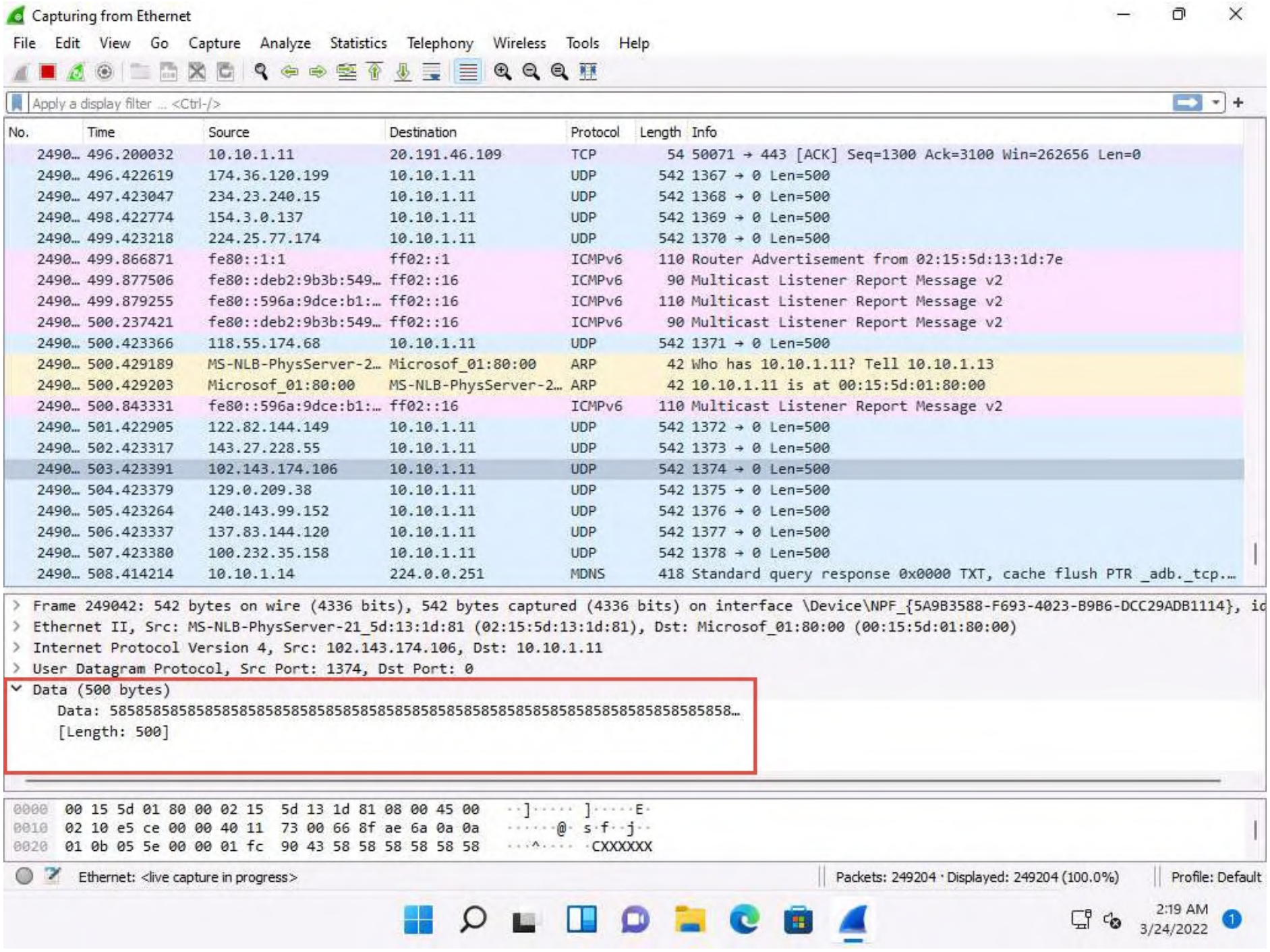
> Note: Here, **--udp** specifies sending the UDP packets to the target host, **--rand-source** enables the random source mode and **--data** specifies the packet body size.
>
> Note: The MAC addresses might differ when you perform this task.



10. Now, click **CEHv12 Windows 11** to switch to the **Windows 11** machine and observe the random UDP packets captured by **Wireshark**.

Note: You can double-click any UDP packet and observe the detail.

11. Expand the **Data** node in the **Packet Details** pane and observe the size of **Data** and its **Length** (the length is the same as the size of the packet body that we specified in Hping3 command, i.e., **500**).



12. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine. In the **Parrot Terminal** window, first press **Control+C** and type **hping3 -S [Target IP Address] -p 80 -c 5** (here, target IP address is **10.10.1.11**), and then press **Enter**.

> Note: Here, **-S** specifies the TCP SYN request on the target machine, **-p** specifies assigning the port to send the traffic, and **-c** is the count of the packets sent to the target machine.

13. In the result, it is indicated that five packets were sent and received through port 80.

```
Applications  Places  System                                   Thu Mar 24, 05:22
hping3 -S 10.10.1.11 -p 80 -c 5 - Parrot Terminal
File  Edit  View  Search  Terminal  Help
┌─[attacker@parrot]─[~]
└──╼ $sudo su
[sudo] password for attacker:
┌─[root@parrot]─[/home/attacker]
└──╼ #cd
┌─[root@parrot]─[~]
└──╼ #hping3 10.10.1.11 --udp --rand-source --data 500
HPING 10.10.1.11 (eth0 10.10.1.11): udp mode set, 28 headers + 500 data bytes
^C
--- 10.10.1.11 hping statistic ---
198 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
┌─[✗]─[root@parrot]─[~]
└──╼ #hping3 -S 10.10.1.11 -p 80 -c 5
HPING 10.10.1.11 (eth0 10.10.1.11): S set, 40 headers + 0 data bytes
len=44 ip=10.10.1.11 ttl=128 DF id=45004 sport=80 flags=SA seq=0 win=65392 rtt=11.9 ms
len=44 ip=10.10.1.11 ttl=128 DF id=45005 sport=80 flags=SA seq=1 win=65392 rtt=3.8 ms
len=44 ip=10.10.1.11 ttl=128 DF id=45006 sport=80 flags=SA seq=2 win=65392 rtt=2.7 ms
len=44 ip=10.10.1.11 ttl=128 DF id=45007 sport=80 flags=SA seq=3 win=65392 rtt=9.7 ms
len=44 ip=10.10.1.11 ttl=128 DF id=45008 sport=80 flags=SA seq=4 win=65392 rtt=1.6 ms

--- 10.10.1.11 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.6/5.9/11.9 ms
┌─[root@parrot]─[~]
└──╼ #
Menu  hping3 -S 10.10.1.11 -p ...
```

14. Now, click **CEHv12 Windows 11** to switch to the target machine (i.e., **Windows 11**) and observe the TCP packets captured via **Wireshark**.

Capturing from Ethernet

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
2501…	690.283405	10.10.1.13	10.10.1.11	TCP	54	1874 → 80 [RST] Seq=1 Win=0 Len=0
2501…	691.282378	10.10.1.13	10.10.1.11	TCP	54	1875 → 80 [SYN] Seq=0 Win=512 Len=0
2501…	691.282487	10.10.1.11	10.10.1.13	TCP	58	80 → 1875 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460
2501…	691.283333	10.10.1.13	10.10.1.11	TCP	54	1875 → 80 [RST] Seq=1 Win=0 Len=0
2501…	692.283556	10.10.1.13	10.10.1.11	TCP	54	1876 → 80 [SYN] Seq=0 Win=512 Len=0
2501…	692.283660	10.10.1.11	10.10.1.13	TCP	58	80 → 1876 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460
2501…	692.285182	10.10.1.13	10.10.1.11	TCP	54	1876 → 80 [RST] Seq=1 Win=0 Len=0
2501…	693.284367	10.10.1.13	10.10.1.11	TCP	54	1877 → 80 [SYN] Seq=0 Win=512 Len=0
2501…	693.284476	10.10.1.11	10.10.1.13	TCP	58	80 → 1877 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460
2501…	693.285239	10.10.1.13	10.10.1.11	TCP	54	1877 → 80 [RST] Seq=1 Win=0 Len=0
2501…	694.284416	10.10.1.13	10.10.1.11	TCP	54	1878 → 80 [SYN] Seq=0 Win=512 Len=0
2501…	694.284523	10.10.1.11	10.10.1.13	TCP	58	80 → 1878 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460
2501…	694.285267	10.10.1.13	10.10.1.11	TCP	54	1878 → 80 [RST] Seq=1 Win=0 Len=0
2501…	695.104921	Microsof_01:80:00	MS-NLB-PhysServer-2…	ARP	42	Who has 10.10.1.13? Tell 10.10.1.11
2501…	695.106724	MS-NLB-PhysServer-2…	Microsof_01:80:00	ARP	42	10.10.1.13 is at 02:15:5d:13:1d:81
2501…	695.502001	MS-NLB-PhysServer-2…	Microsof_01:80:00	ARP	42	Who has 10.10.1.11? Tell 10.10.1.13
2501…	695.502019	Microsof_01:80:00	MS-NLB-PhysServer-2…	ARP	42	10.10.1.11 is at 00:15:5d:01:80:00
2501…	698.002239	fe80::1:1	ff02::1	ICMPv6	110	Router Advertisement from 02:15:5d:13:1d:7e
2501…	698.012020	fe80::596a:9dce:b1:…	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
2501…	698.013163	fe80::deb2:9b3b:549…	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
2501…	698.152593	fe80::1:1	ff02::16	ICMPv6	90	Multicast Listener Report Message v2

```
> Frame 250173: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{5A9B3588-F693-4023-B9B6-DCC29ADB1114}, id 0
> Ethernet II, Src: MS-NLB-PhysServer-21_5d:13:1d:81 (02:15:5d:13:1d:81), Dst: Microsof_01:80:00 (00:15:5d:01:80:00)
> Internet Protocol Version 4, Src: 10.10.1.13, Dst: 10.10.1.11
> Transmission Control Protocol, Src Port: 1874, Dst Port: 80, Seq: 1, Len: 0
```

```
0000  00 15 5d 01 80 00 02 15  5d 13 1d 81 08 00 45 00   ··]····· ]·····E·
0010  00 28 00 00 40 00 40 06  24 a5 0a 0a 01 0d 0a 0a   ·(··@·@· $·······
0020  01 0b 07 52 00 50 3c 2c  73 ba 00 00 00 00 50 04   ···R·P<, s·····P·
```

Ethernet: <live capture in progress> | Packets: 250444 · Displayed: 250444 (100.0%) | Profile: Default

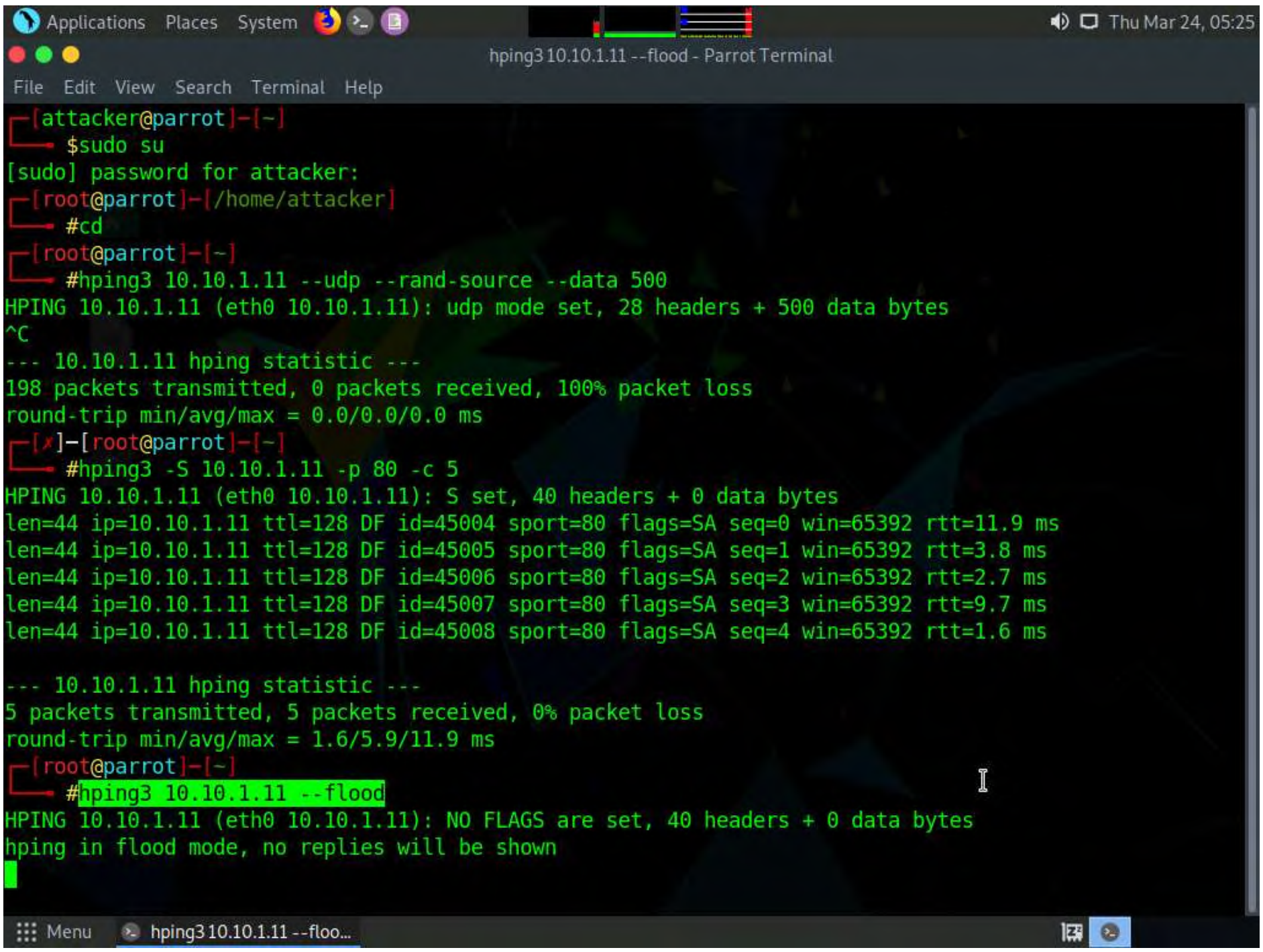
2:24 AM 3/24/2022

15. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine and try to flood the target machine (here, **Windows 11**) with TCP packets.

16. In the **Parrot Terminal** window, type **hping3 [Target IP Address] --flood** (here, target IP address is **10.10.1.11**) and press **Enter**.

> Note: **--flood**: performs the TCP flooding.

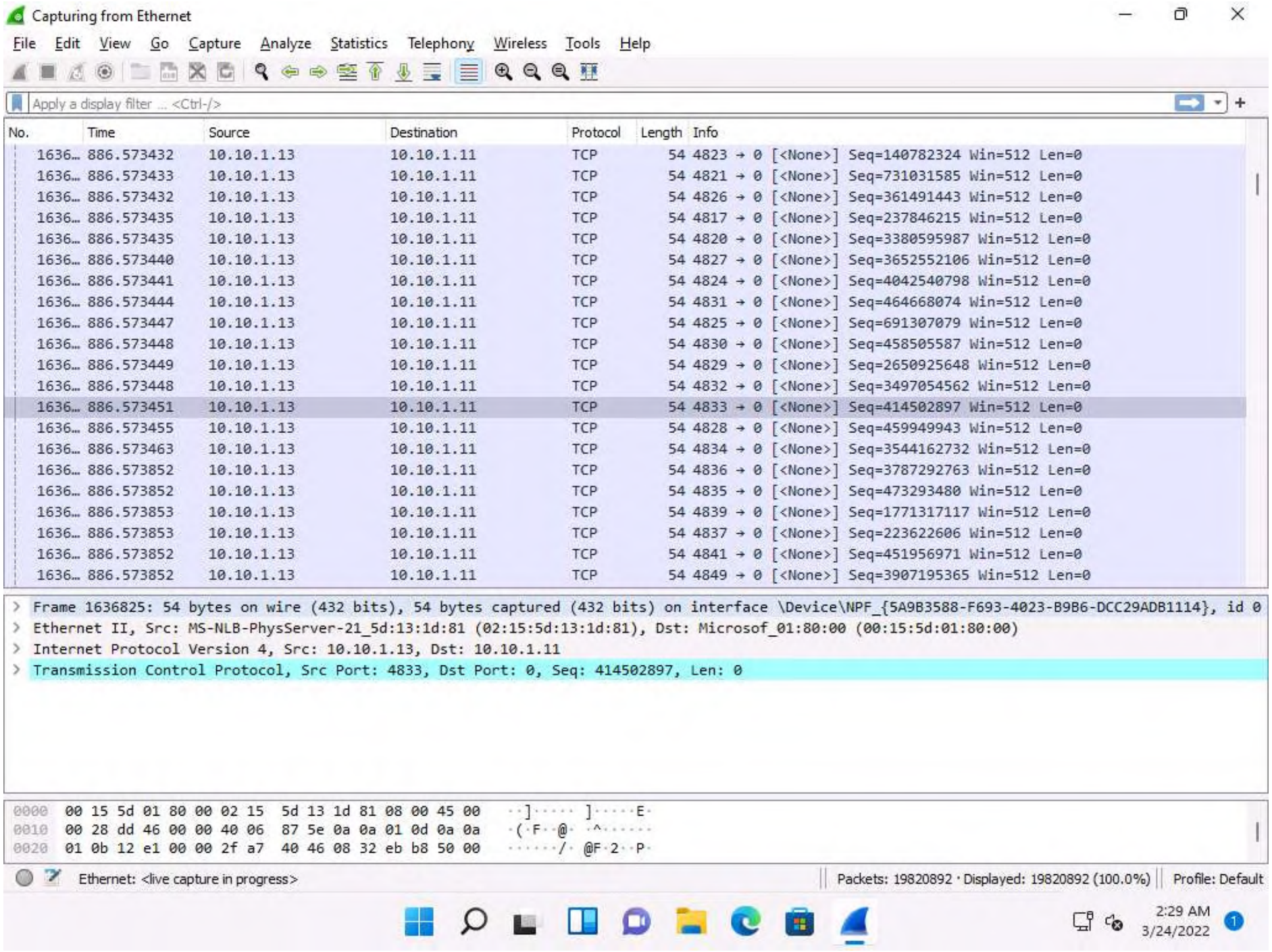
17. Once you flood traffic to the target machine, it will respond in the hping3 terminal.



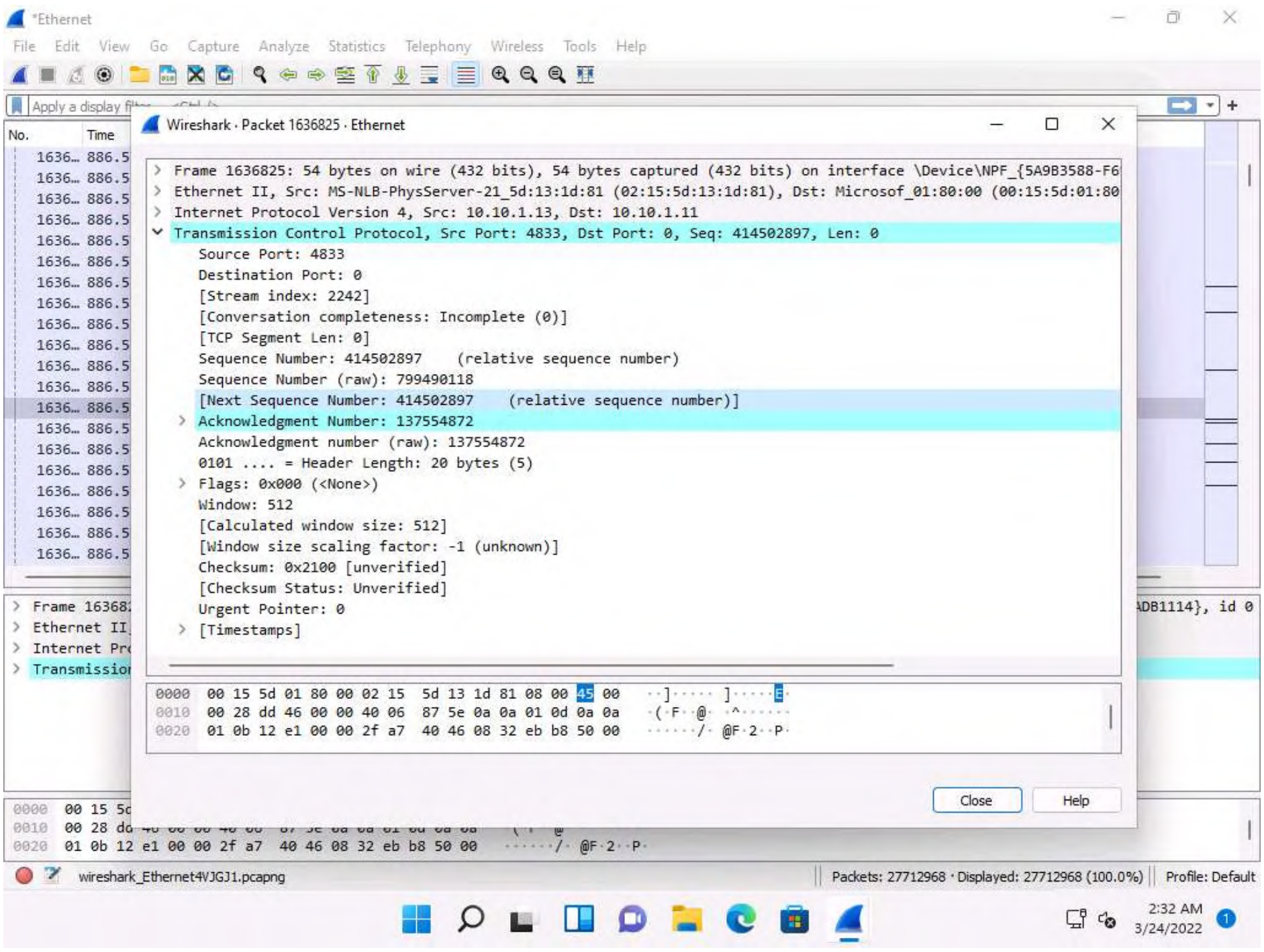
18. Click **CEHv12 Windows 11** to switch to the **Windows 11** (target machine) and stop the packet capture in the **Wireshark** window after a while by click **Stop Capturing Packets** icon in the toolbar.

19. Observe the **Wireshark** window, which displays the TCP packet flooding from the host machine. The attacker employs TCP SYN flooding technique to perform a DoS attack on the target.

> Note: You can double-click the TCP packet stream to observe the TCP packet information.



20. The TCP packet stream displays the complete information of TCP packets such as the source and destination of the captured packet, source port, destination port, etc.



21. Turn off the **Windows Firewall** in the **Windows 11** by navigating to **Control Panel** --> **System and Security** --> **Windows Defender Firewall** --> **Turn Windows Defender Firewall on or off**.
22. This concludes the demonstration of evading the IDS and firewall using various evasion techniques in Hping3.
23. You can also use other packet crafting tools such as **NetScanTools Pro** (https://www.netscantools.com), **Colasoft packet builder** (https://www.colasoft.com), etc. to build custom packets to evade security mechanisms.
24. Close all open windows and document all the acquired information.

Lab 5: Perform Network Scanning using Various Scanning Tools

Lab Scenario

The information obtained in the previous steps might be insufficient to reveal potential vulnerabilities in the target network: there may be more information available that could help in finding loopholes in the target network. As an ethical hacker and pen tester, you should look for as much information as possible about systems in the target network using various network scanning tools when needed. This lab will demonstrate other techniques/commands/methods that can assist you in extracting information about the systems in the target network using various scanning tools.

Lab Objectives

Scan a target network using Metasploit

Overview of Network Scanning Tools

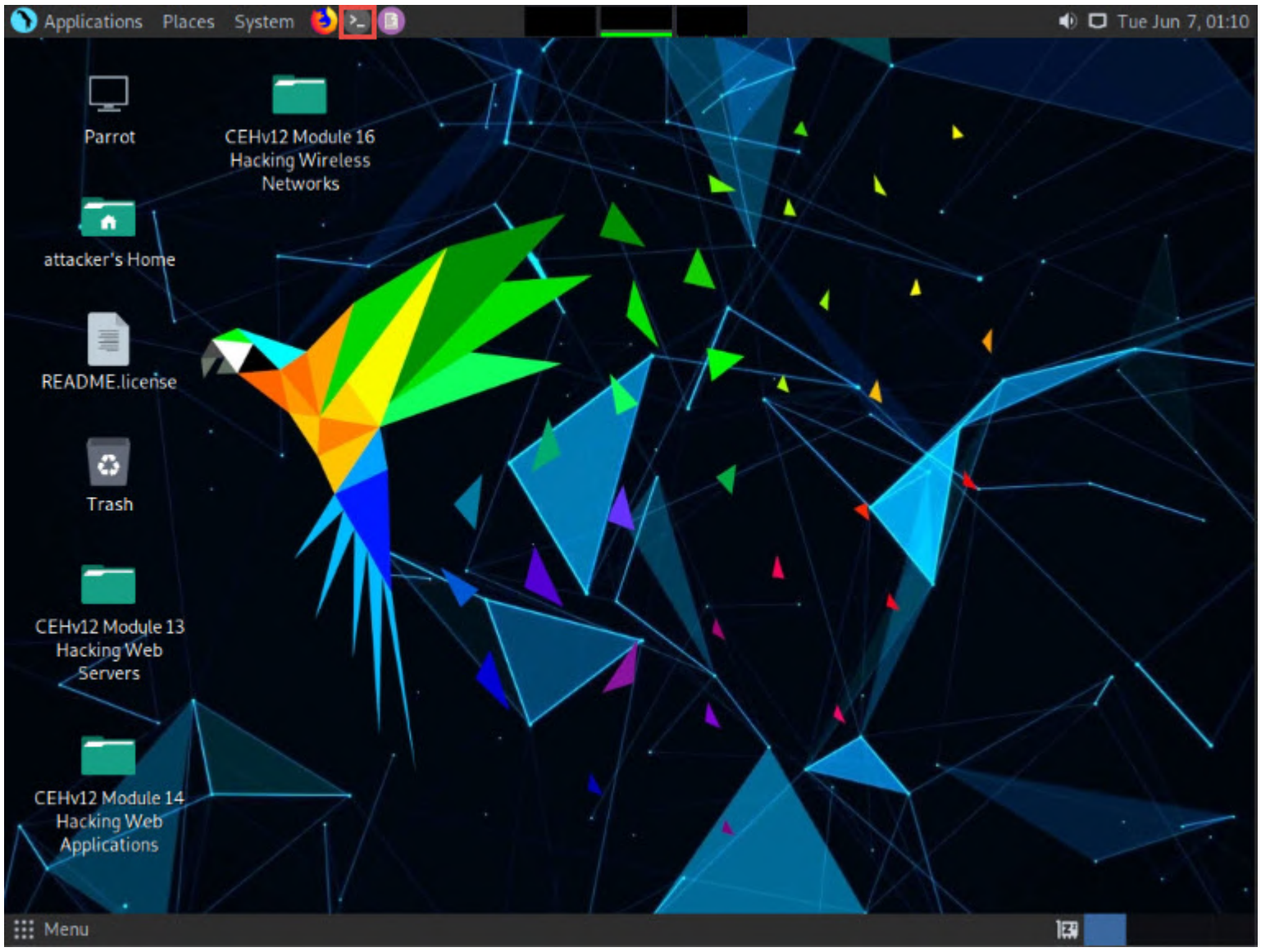
Scanning tools are used to scan and identify live hosts, open ports, running services on a target network, location-info, NetBIOS info, and information about all TCP/IP and UDP open ports. Information obtained from these tools will assist an ethical hacker in creating the profile of the target organization and to scan the network for open ports of the devices connected.

Task 1: Scan a Target Network using Metasploit

Metasploit Framework is a tool that provides information about security vulnerabilities in the target organization's system, and aids in penetration testing and IDS signature development. It facilitates the tasks of attackers, exploit writers, and payload writers. A major advantage of the framework is the modular approach, that is, allowing the combination of any exploit with any payload.

Here, we will use Metasploit to discover active hosts, open ports, services running, and OS details of systems present in the target network.

1. Click **CEHv12 Parrot Security** to switch to the **Parrot Security** machine.
2. Click the **MATE Terminal** icon in the top of the **Desktop** to open a **Terminal** window.

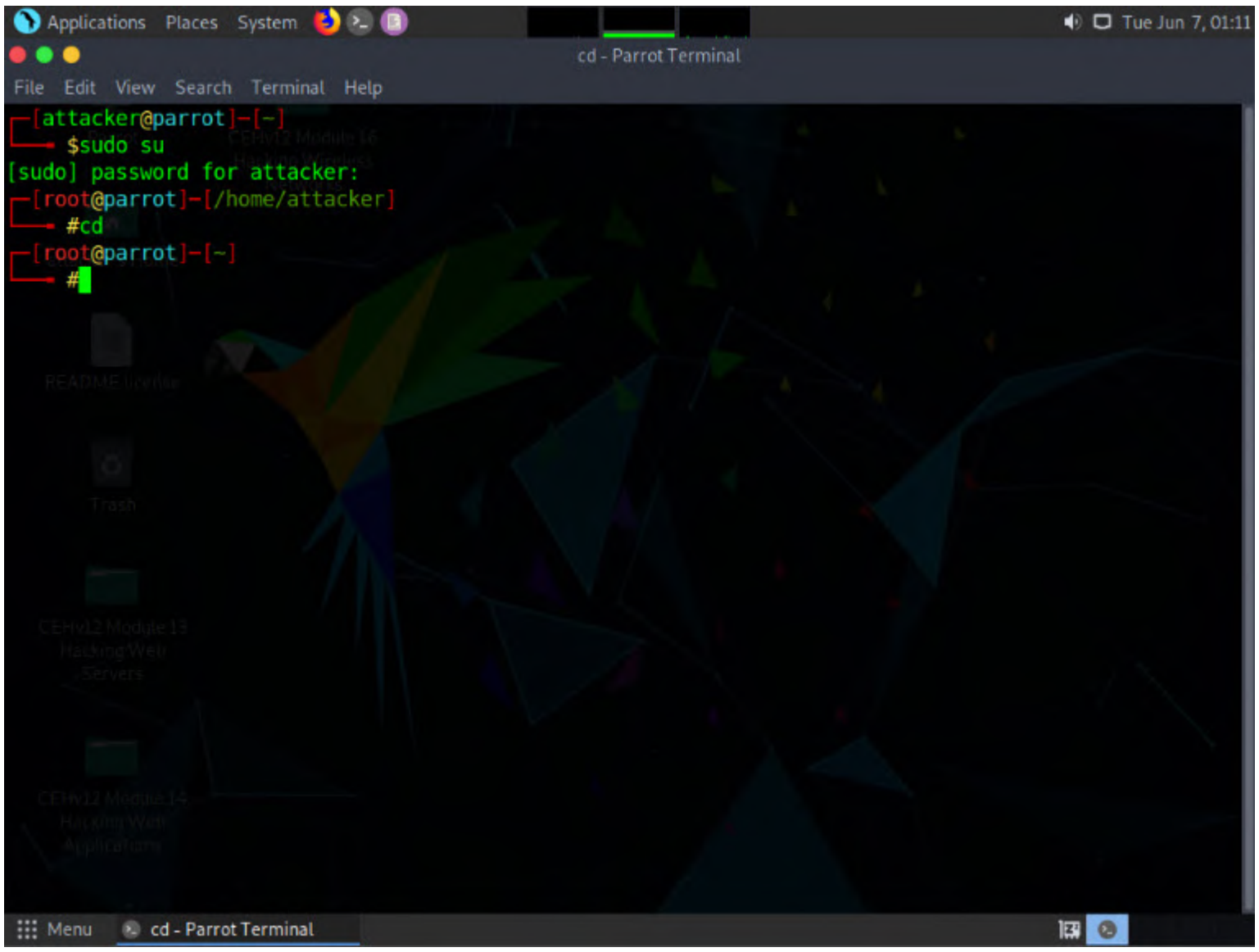


3. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

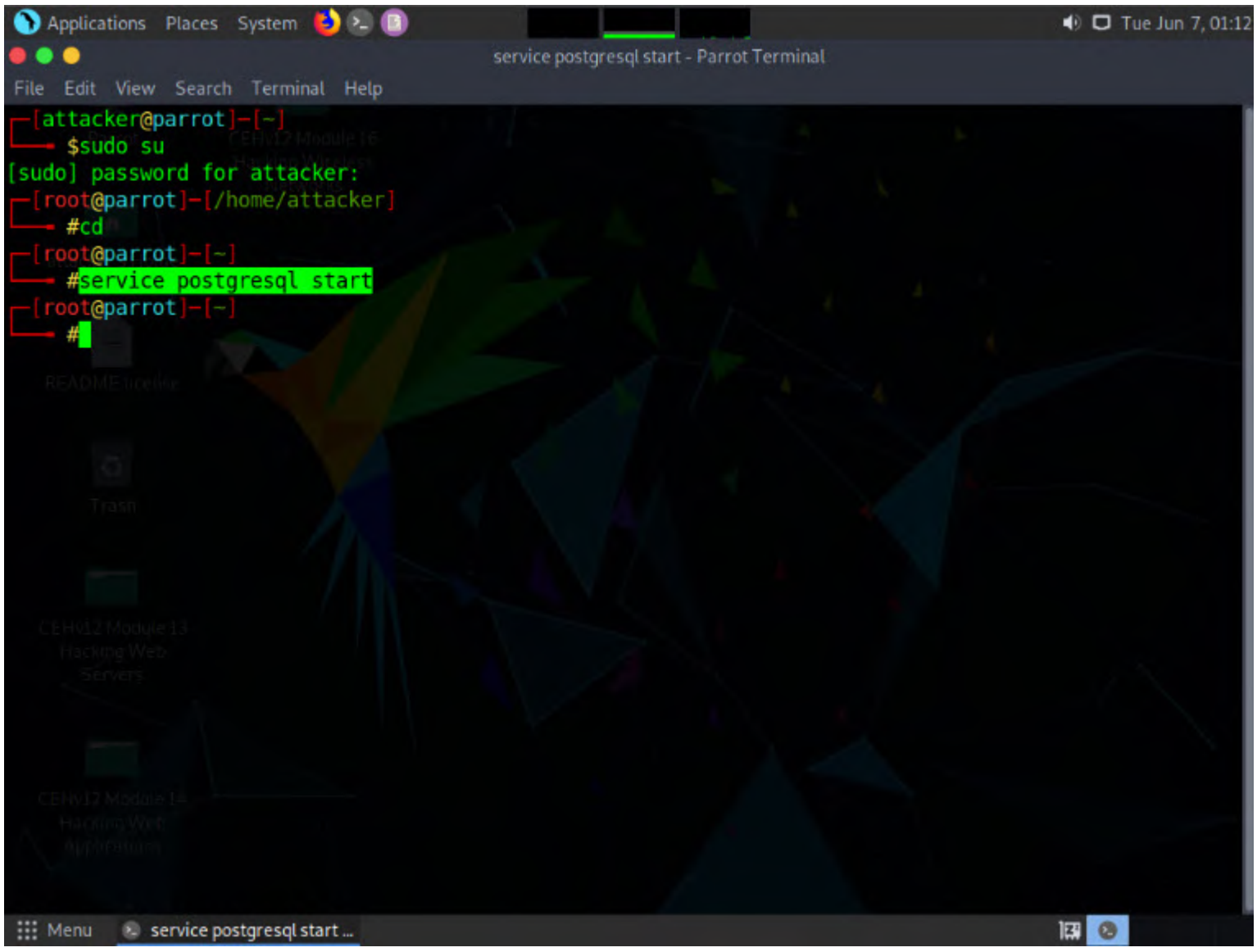
4. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

> Note: The password that you type will not be visible.

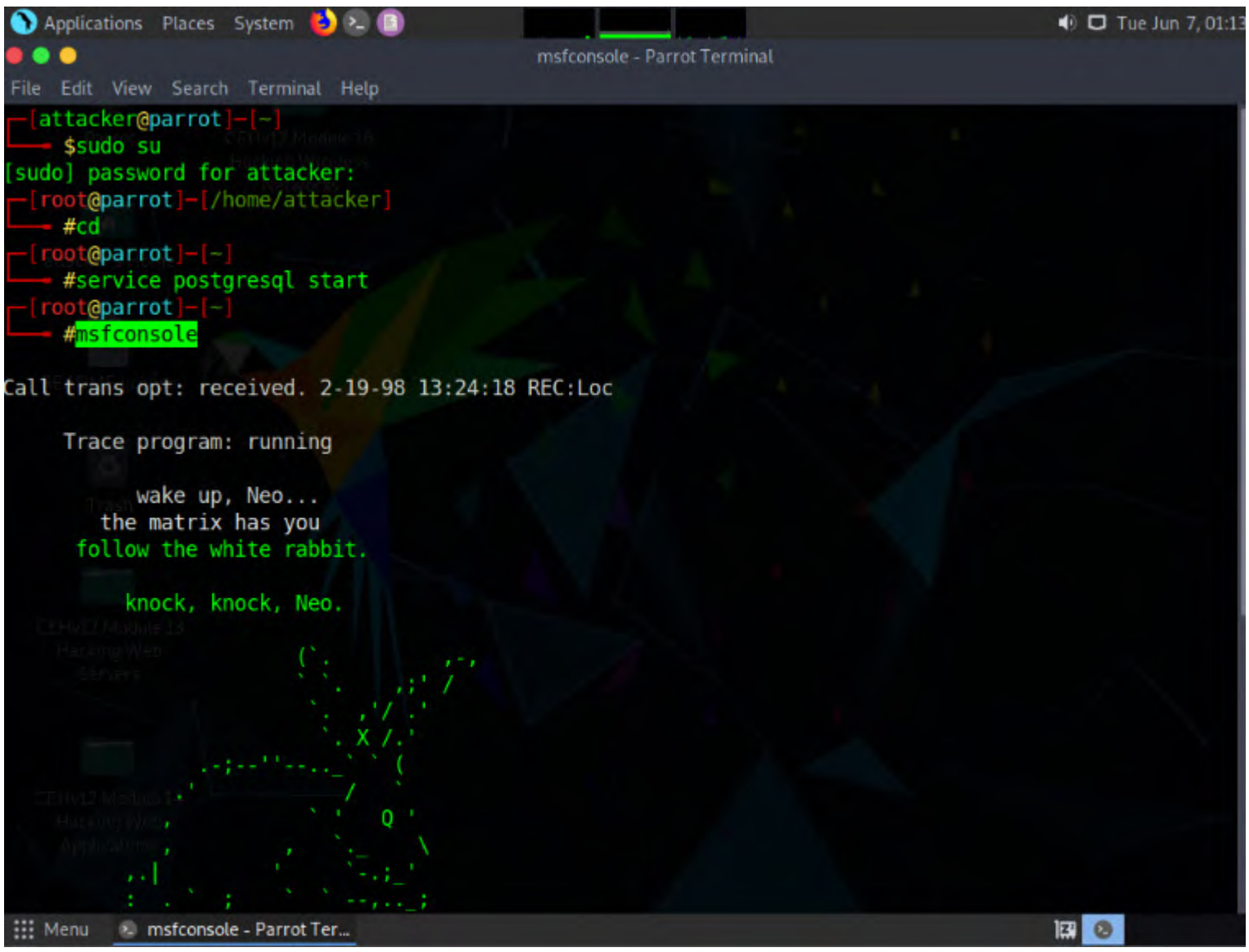
5. Now, type **cd** and press **Enter** to jump to the root directory.



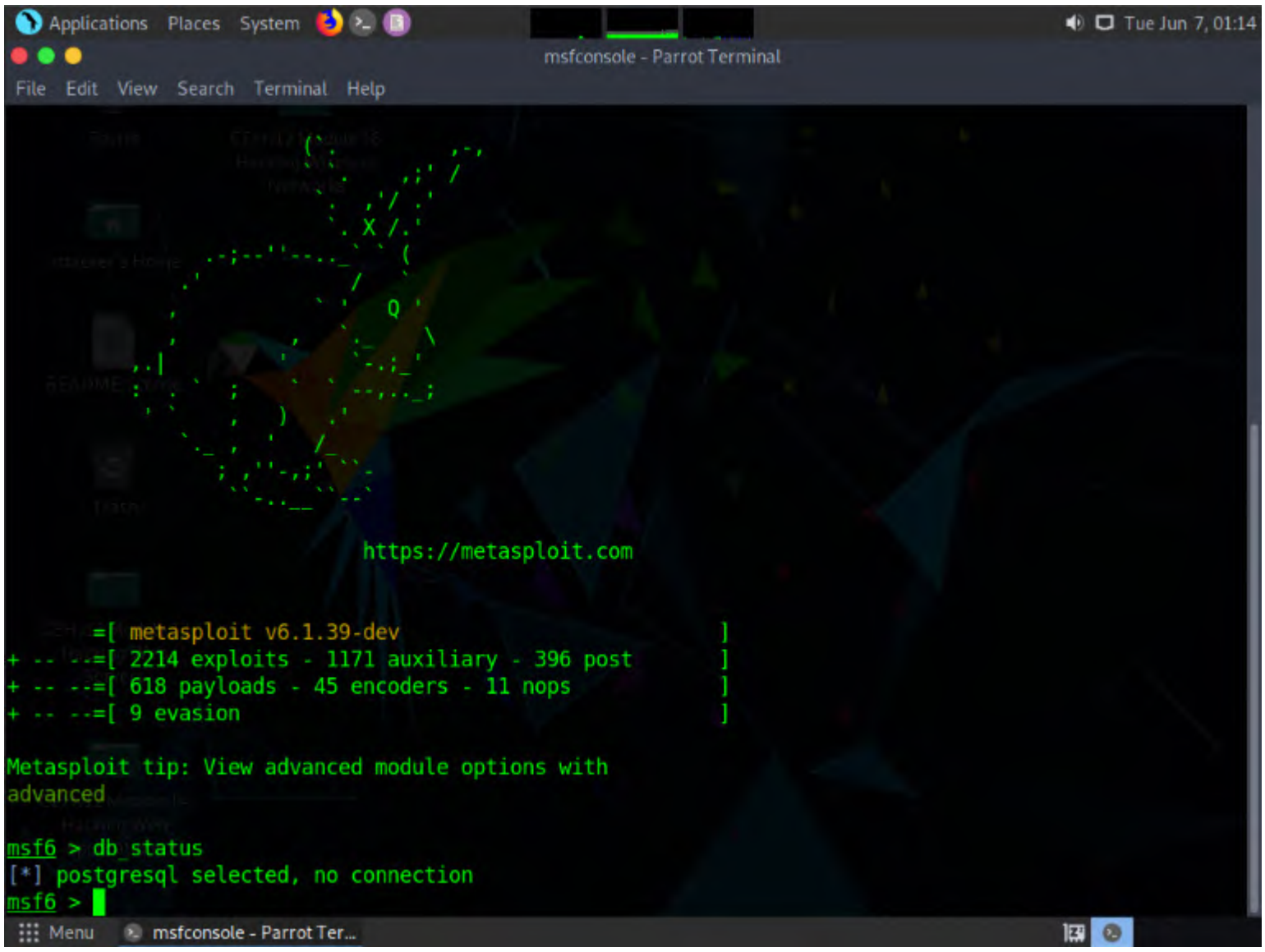
6. In the **Parrot Terminal** window, type **service postgresql start** and hit **Enter**.



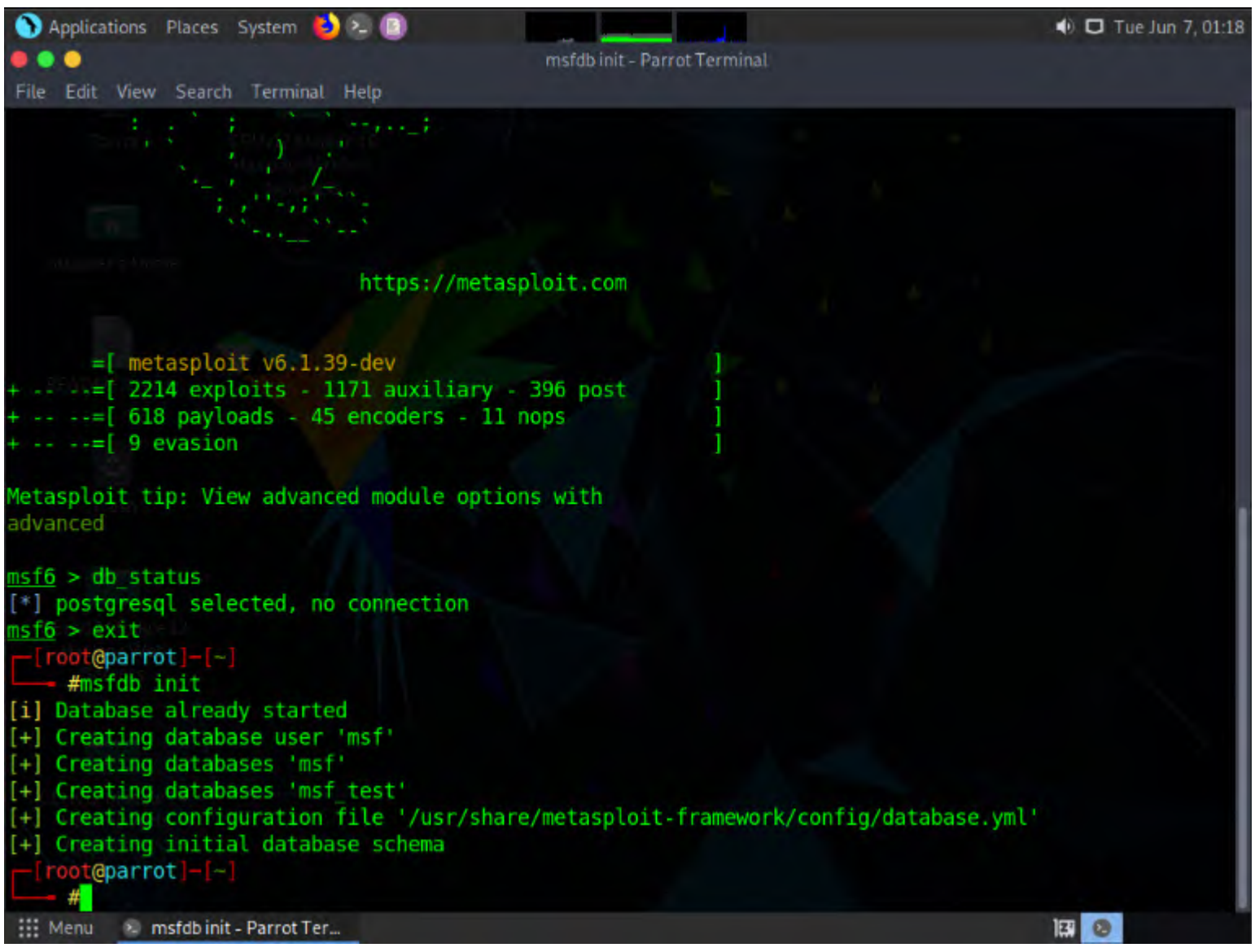
7. Now, type **msfconsole** and hit **Enter** to launch Metasploit.



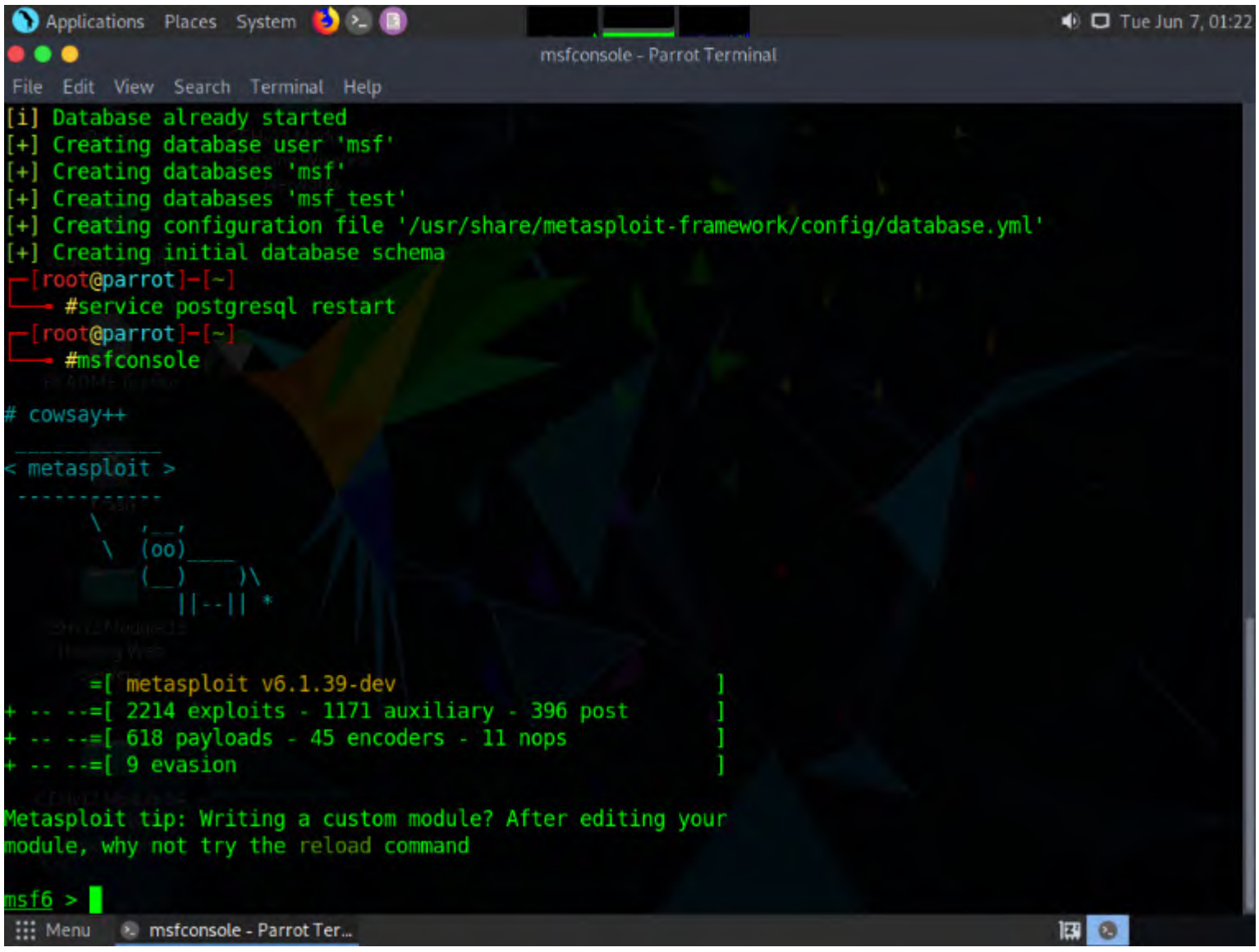
8. An msf command line appears. Type **db_status** and hit **Enter** to check if Metasploit has connected to the database successfully. If you receive the message "**postgresql selected, no connection**," then the database did not connect to msf.



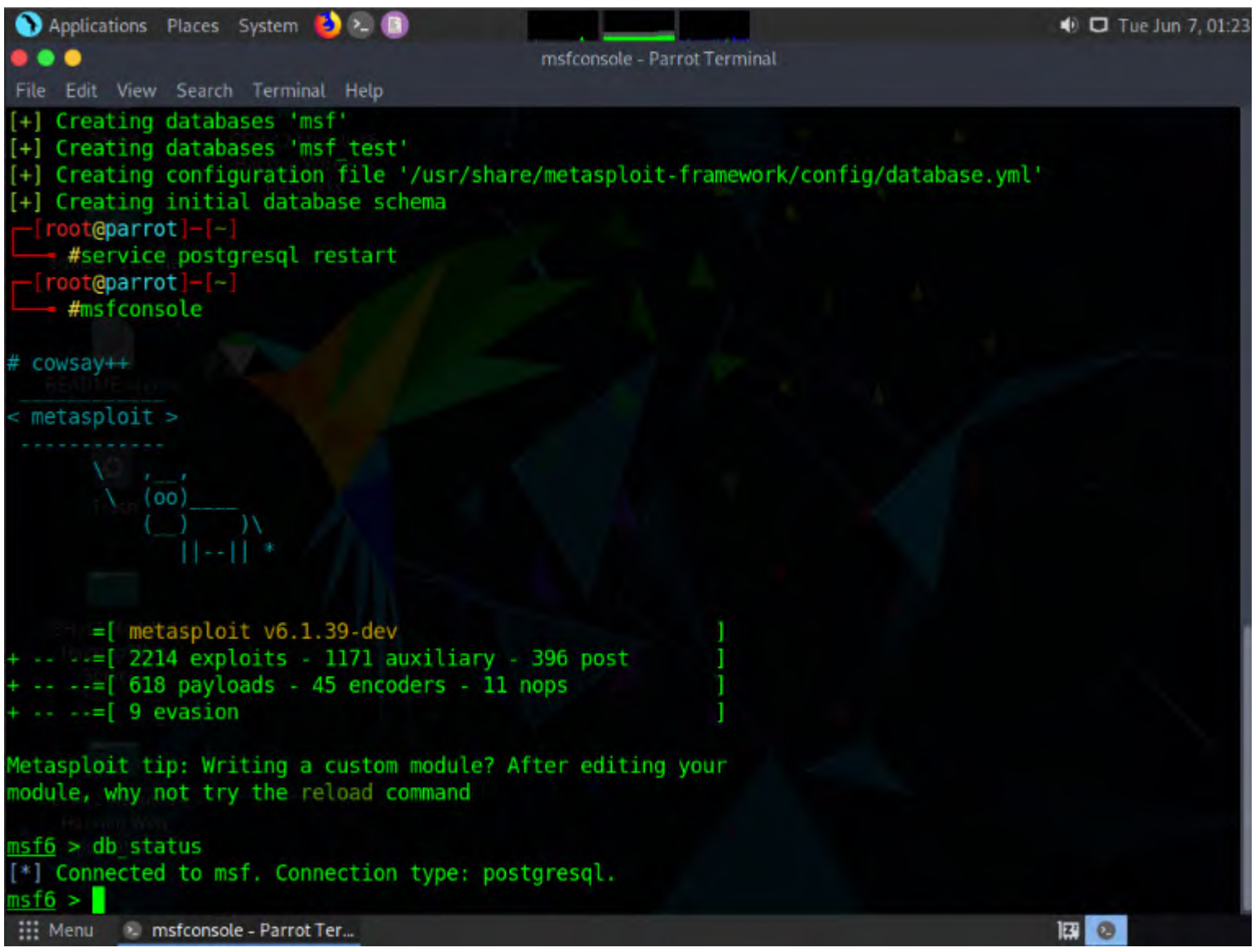
9. Exit the Metasploit framework by typing **exit** and press **Enter**. Then, to initiate the database, type **msfdb init**, and press **Enter**.



10. To restart the postgresql service, type **service postgresql restart** and press **Enter**. Now, start the Metasploit Framework again by typing **msfconsole** and pressing **Enter**.



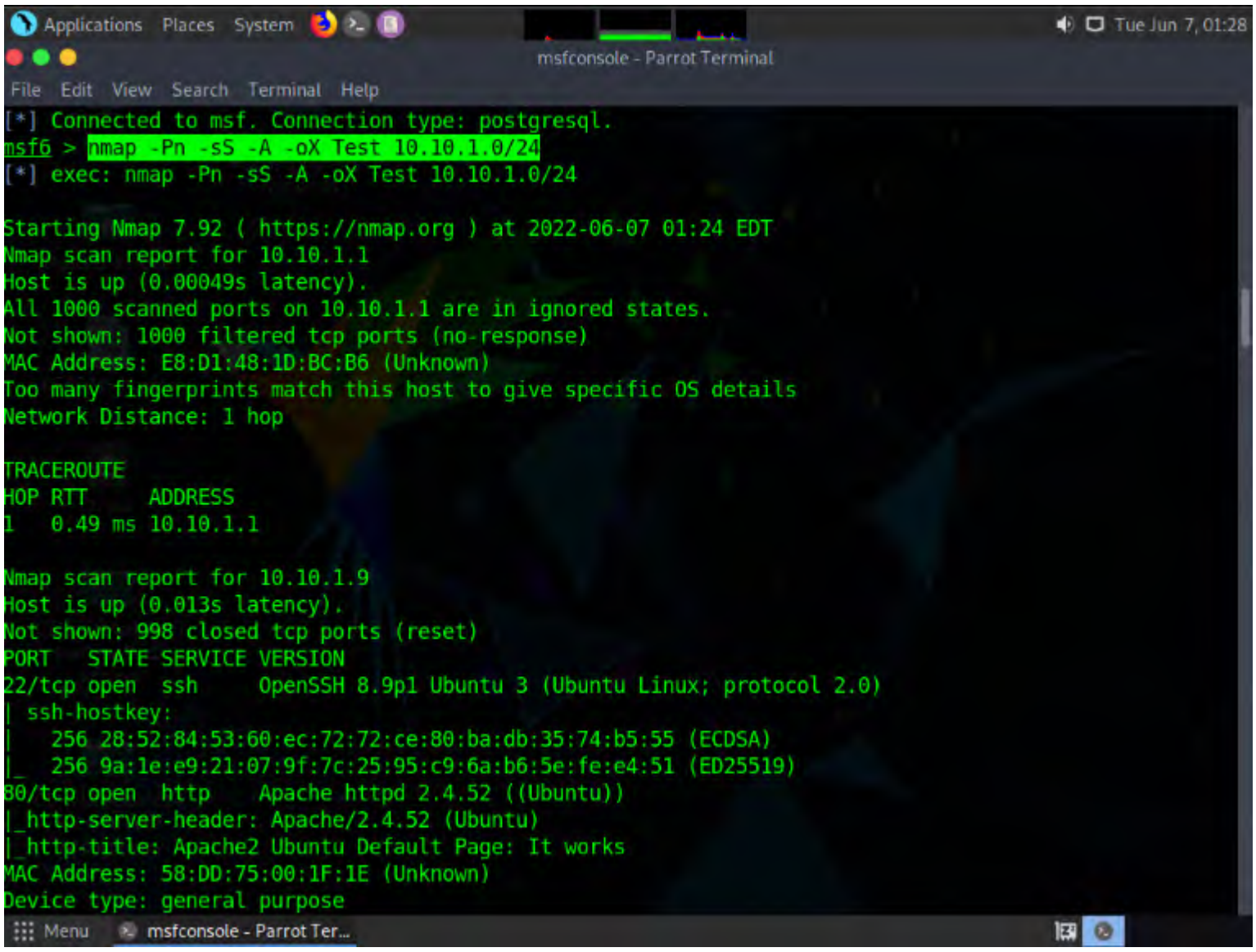
11. Check the database status by typing **db_status** and press **Enter**. This time, the database should successfully connect to msf, as shown in the screenshot.



12. Type **nmap -Pn -sS -A -oX Test 10.10.1.0/24** and hit **Enter** to scan the subnet, as shown in the screenshot.

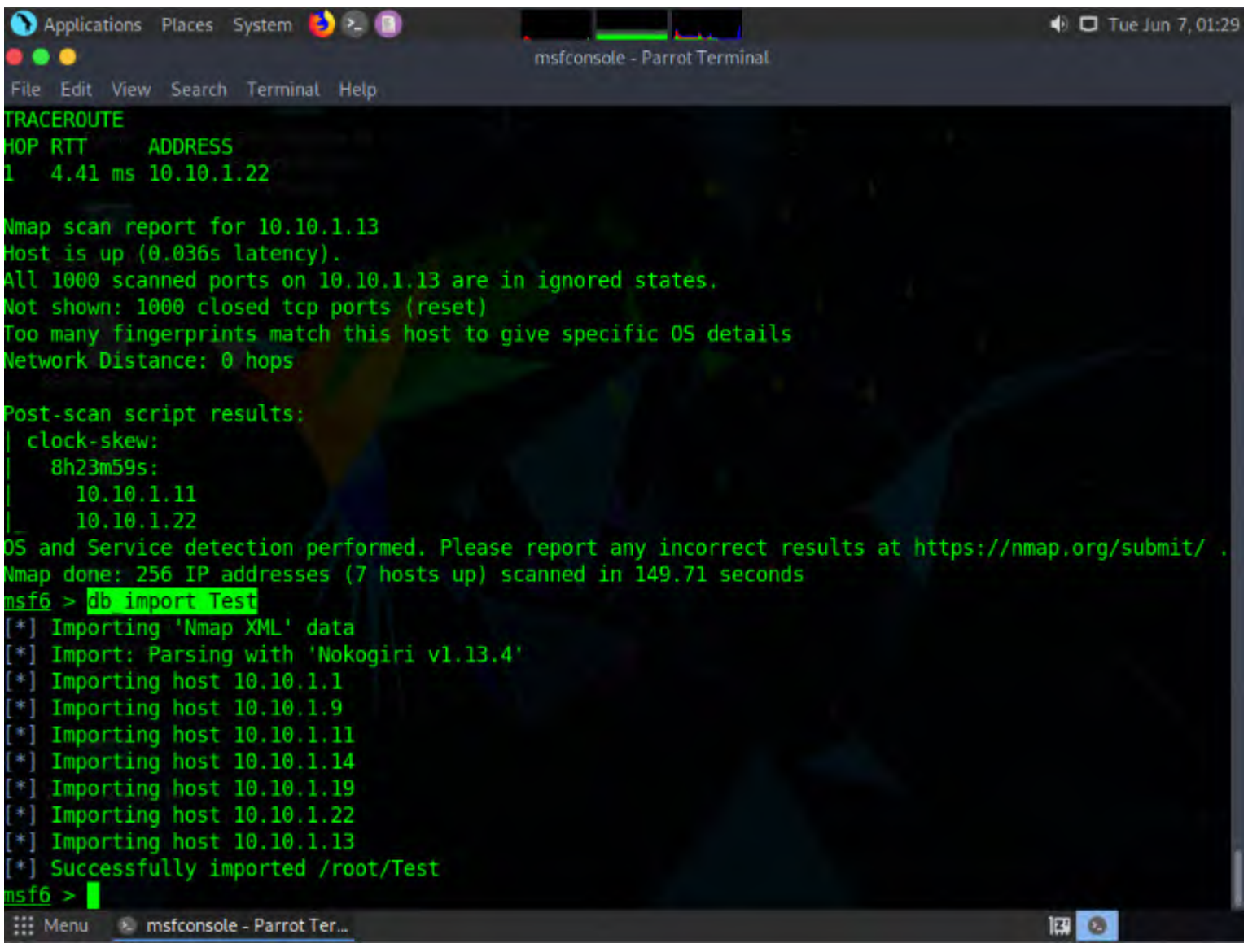
> Note: Here, we are scanning the whole subnet 10.10.1.0/24 for active hosts.

13. Nmap begins scanning the subnet and displays the results. It takes approximately 5 minutes for the scan to complete.



14. After the scan completes, Nmap displays the number of active hosts in the target network (here, **7**).

15. Now, type **db_import Test** and hit **Enter** to import the Nmap results from the database.



16. Type **hosts** and hit **Enter** to view the list of active hosts along with their MAC addresses, OS names, etc. as shown in the screenshot.

```
Applications  Places  System                                   Tue Jun 7, 01:32
                                msfconsole - Parrot Terminal
File  Edit  View  Search  Terminal  Help
[*] Importing host 10.10.1.1
[*] Importing host 10.10.1.9
[*] Importing host 10.10.1.11
[*] Importing host 10.10.1.14
[*] Importing host 10.10.1.19
[*] Importing host 10.10.1.22
[*] Importing host 10.10.1.13
[*] Successfully imported /root/Test
msf6 > hosts

Hosts
=====

address     mac           name          os_name     os_flavor  os_sp  purpose  info  comments
-------     ---           ----          -------     ---------  -----  -------  ----  --------
10.10.1.1   e8:d1:48:1d:                Unknown                       device
            bc:b6
10.10.1.9   58:dd:75:00:                Linux                 4.X     server
            1f:1e
10.10.1.11  b4:b5:78:89:                Windows 10                    client
            75:64
10.10.1.13                              Unknown                       device
10.10.1.14  3e:7d:4f:2c:                Linux                 4.X     server
            4b:d7
10.10.1.19  ac:90:49:48:  www.moviesco  Windows 10                    client
            6f:e6         pe.com
10.10.1.22  1c:5a:12:d9:                Windows 10                    client
            10:bd

msf6 >
Menu   msfconsole - Parrot Ter...
```

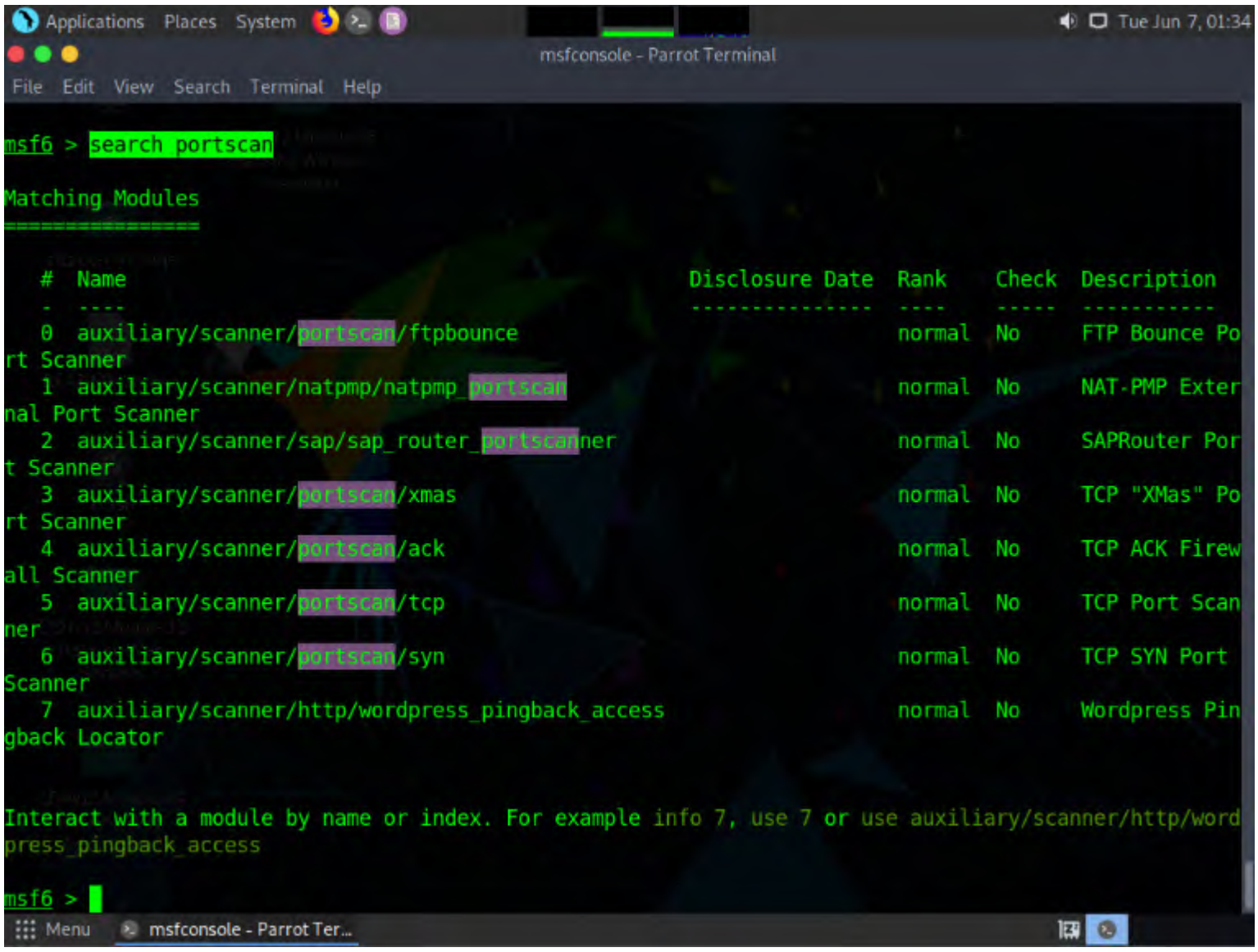
17. Type **services** or **db_services** and hit **Enter** to receive a list of the services running on the active hosts, as shown in the screenshot.

Note: In addition to running Nmap, there are a variety of other port scanners that are available within the Metasploit framework to scan the target systems.

```
Applications  Places  System                                   Tue Jun 7, 01:33
                                msfconsole - Parrot Terminal
File  Edit  View  Search  Terminal  Help
msf6 > services
Services
========

host        port  proto  name             state  info
----        ----  -----  ----             -----  ----
10.10.1.9   22    tcp    ssh              open   OpenSSH 8.9p1 Ubuntu 3 Ubuntu Linux; protocol 2.
                                                 0
10.10.1.9   80    tcp    http             open   Apache httpd 2.4.52 (Ubuntu)
10.10.1.11  80    tcp    http             open   Microsoft IIS httpd 10.0
10.10.1.11  135   tcp    msrpc            open   Microsoft Windows RPC
10.10.1.11  139   tcp    netbios-ssn      open   Microsoft Windows netbios-ssn
10.10.1.11  445   tcp    microsoft-ds     open   Windows 10 Enterprise 22000 microsoft-ds workgro
                                                 up: WORKGROUP
10.10.1.11  3389  tcp    ssl/ms-wbt-server  open
10.10.1.14  5555  tcp    adb              open   Android Debug Bridge device name: android_x86_64
                                                 ; model: Standard PC (i440FX + PIIX, 1996); devi
                                                 ce: x86_64; features: cmd,stat_v2,shell_v2
10.10.1.19  25    tcp    smtp             open   Microsoft ESMTP 10.0.17763.1
10.10.1.19  80    tcp    http             open   Microsoft IIS httpd 10.0
10.10.1.19  135   tcp    msrpc            open   Microsoft Windows RPC
10.10.1.19  139   tcp    netbios-ssn      open   Microsoft Windows netbios-ssn
10.10.1.19  445   tcp    microsoft-ds     open
10.10.1.19  1801  tcp    msmq             open
10.10.1.19  2103  tcp    msrpc            open   Microsoft Windows RPC
10.10.1.19  2105  tcp    msrpc            open   Microsoft Windows RPC
10.10.1.19  2107  tcp    msrpc            open   Microsoft Windows RPC
10.10.1.19  3389  tcp    ms-wbt-server    open   Microsoft Terminal Services
10.10.1.22  53    tcp    domain           open   Simple DNS Plus
10.10.1.22  80    tcp    http             open   Microsoft IIS httpd 10.0
Menu   msfconsole - Parrot Ter...
```

18. Type **search portscan** and hit **Enter**. The Metasploit port scanning modules appear, as shown in the screenshot.



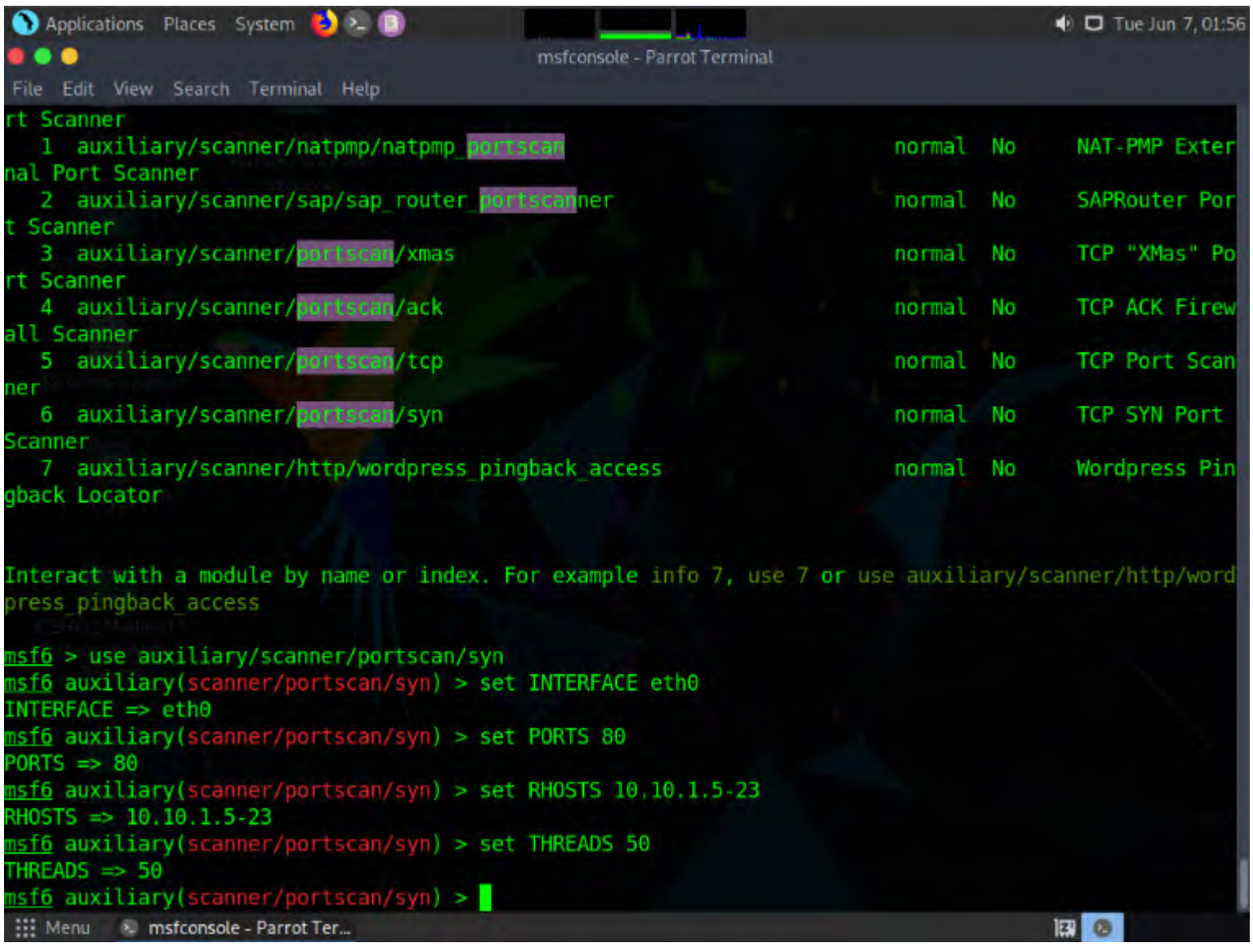
19. Here, we will use the **auxiliary/scanner/portscan/syn** module to perform an SYN scan on the target systems. To do so, type **use auxiliary/scanner/portscan/syn** and press **Enter**.

20. We will use this module to perform an SYN scan against the target IP address range (**10.10.1.5-23**) to look for open port 80 through the eth0 interface.

    To do so, issue the below commands:

    set INTERFACE eth0
    set PORTS 80
    set RHOSTS 10.10.1.5-23
    set THREADS 50

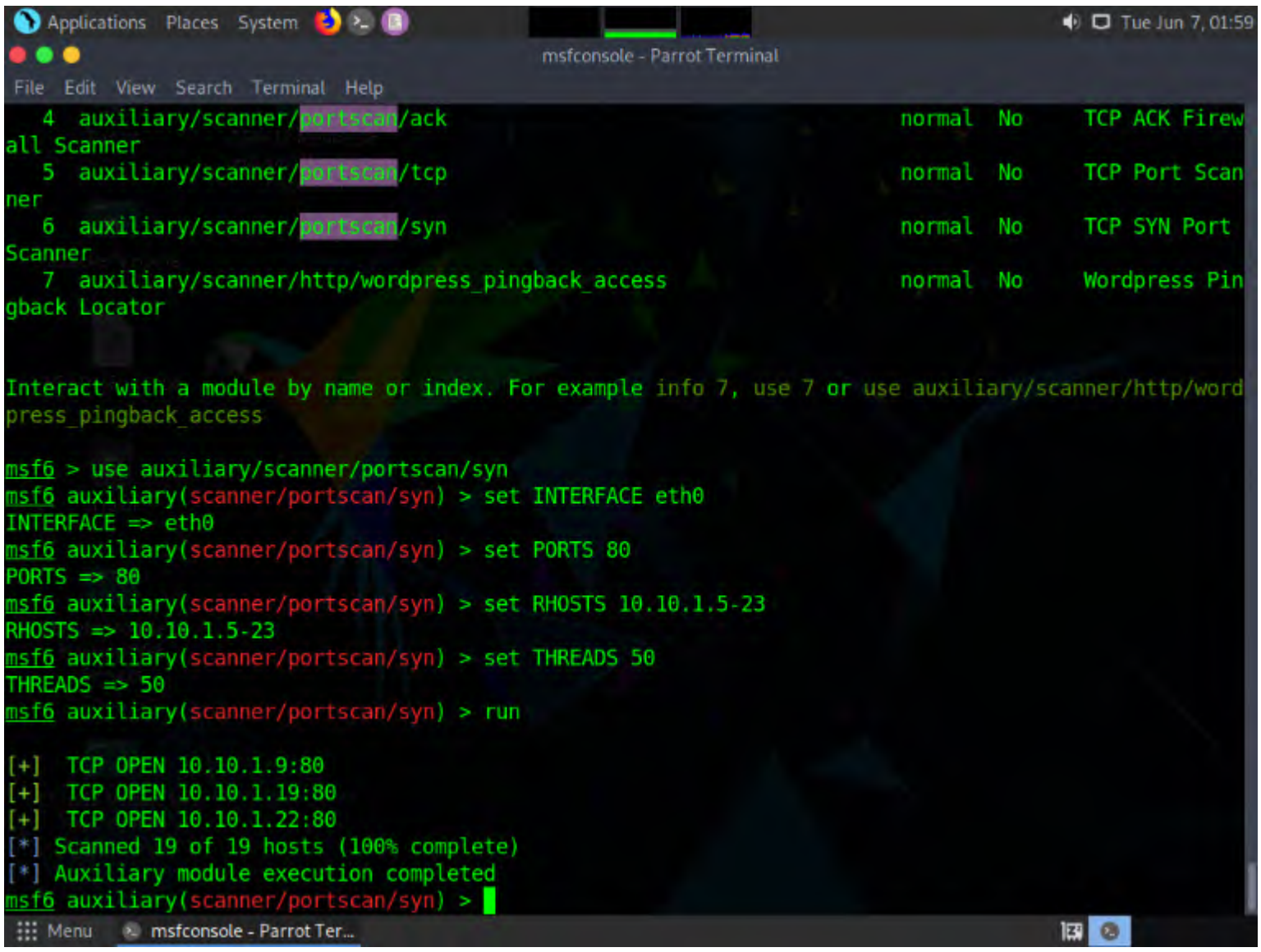
Note: **PORTS**: specifies the ports to scan (e.g., 22-25, 80, 110-900), **RHOSTS**: specifies the target address range or CIDR identifier, and THREADS: specifies the number of concurrent threads (default 1).



21. After specifying the above values, type **run**, and press **Enter** to initiate the scan against the target IP address range.

Note: Similarly, you can also specify a range of ports to be scanned against the target IP address range.

22. The result appears, displaying open port 80 in active hosts, as shown in the screenshot.



23. Now, we will perform a TCP scan for open ports on the target systems.

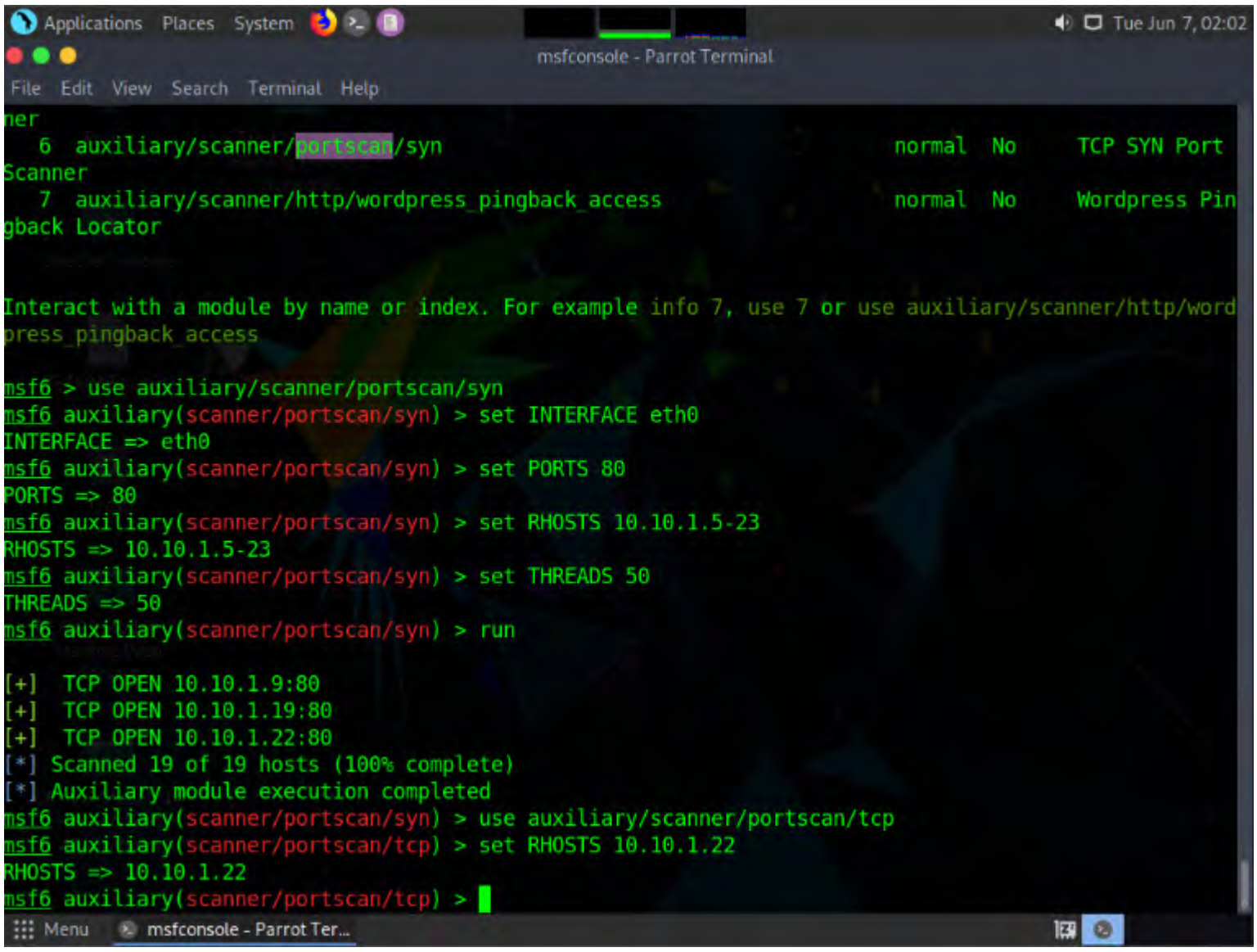
24. To load the **auxiliary/scanner/portscan/tcp** module, type **use auxiliary/scanner/portscan/tcp** and press **Enter**.

25. Type **hosts -R** and press **Enter** to automatically set this option with the discovered hosts present in our database.

    OR

    Type **set RHOSTS [Target IP Address]** and press **Enter**.

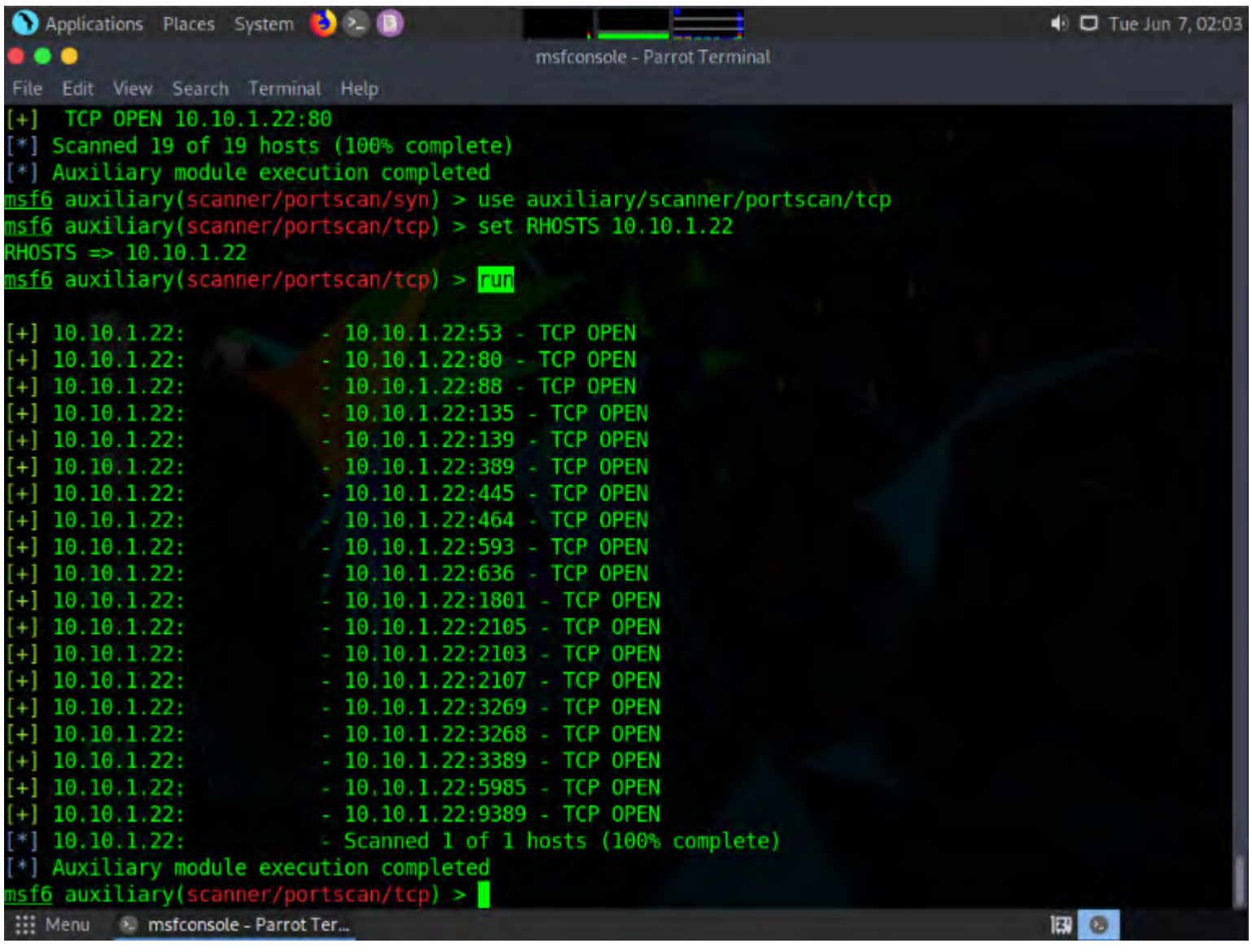
Note: Here, we will perform a TCP scan for open ports on a single IP address (**10.10.1.22**), as scanning multiple IP addresses consumes much time.



26. Type **run** and press **Enter** to discover open TCP ports in the target system.

> Note: It will take approximately 20 minutes for the scan to complete.

27. The results appear, displaying all open TCP ports in the target IP address (10.10.1.22).



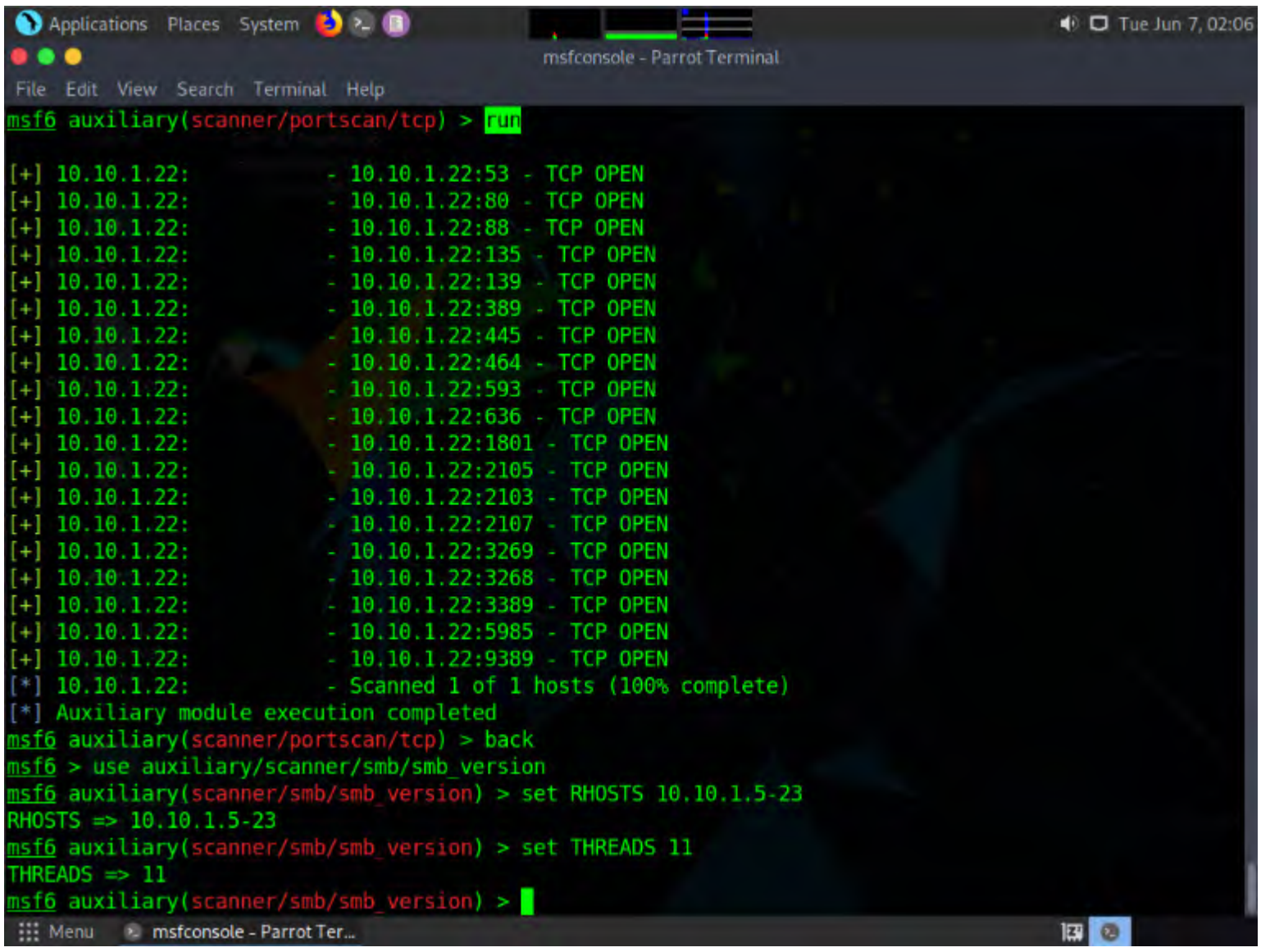
28. Now that we have determined the active hosts on the target network, we can further attempt to determine the OSes running on the target systems. As there are systems in our scan that have port 445 open, we will use the module scanner/smb/version to determine which version of Windows is running on a target and which Samba version is on a Linux host.

29. To do so, first type **back**, and then press **Enter** to revert to the msf command line. Then, type **use auxiliary/scanner/smb/smb_version** and press **Enter**.

30. We will use this module to run a SMB version scan against the target IP address range (**10.10.1.5-23**). To do so, issue the below commands:

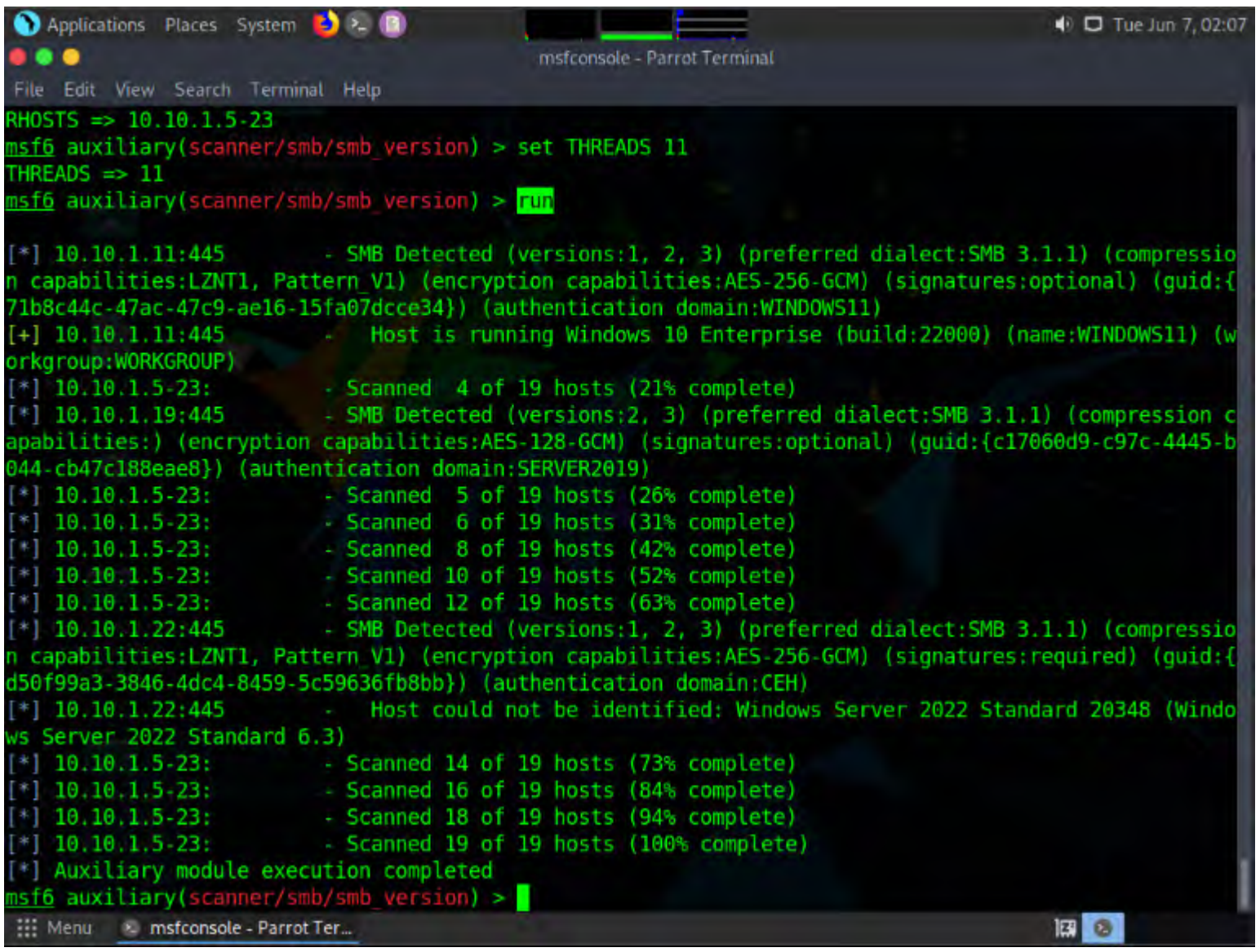
set RHOSTS 10.10.1.5-23

set THREADS 11



31. Type **run** and press **Enter** to discover SMB version in the target systems.

32. The result appears, displaying the OS details of the target hosts.



33. You can further explore various modules of Metasploit such as FTP module to identify the FTP version running in the target host.

34. This information can further be used to perform vulnerability analysis on the open services discovered in the target hosts.

35. This concludes the demonstration of gathering information on open ports, a list of services running on active hosts, and information related to OSes, amongst others.

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