

Module 02 – Footprinting and Reconnaissance

19. Google's crawling functionality begins crawling the target website. Once the crawling is completed, results appear in the lower pane.
20. The results include the domains obtained through scanning along with their severity as low, medium or high is displayed, as shown in the screenshot. Using this information, attackers can further find vulnerabilities in the target domain and exploit them to launch web application attacks.

The screenshot shows the FOCA Open Source 3.4.7.1 application window. The left sidebar displays a project tree for 'www.eccouncil.org' under 'Project'. It includes sections for 'Network', 'Domains' (listing 'comcastbusiness.net', 'eccouncil.org', and 'www.eccouncil.org'), and 'Document Analysis'. The main pane has tabs for 'Attribute' (showing 'Domain - Source' as 'eccouncil.org' with value 'DuckDuckGoWeb > Inferred by aspen.eccouncil.org [eccouncil.org]') and 'Value' (listing 'IP Addresses - Source' for '104.18.21.251' and '104.18.20.251' with the same inferred source). Below these are buttons for 'Technology recognition' (Google, Bing, DuckDuckGo) and 'Crawling' (selected). A 'Log' tab is also present. The bottom section shows a log table with columns 'Time', 'Source', 'Severity', and 'Message'. The log entries show various search attempts and domain discoveries. The status bar at the bottom indicates 'All searchers have finished' and shows system time as 1:10 AM on 3/24/2022.

Time	Source	Severity	Message
12:50:57...	MetadataSearch	error	An error has occurred on DuckDuckGoWeb: The remote server returned an error: (403) Forbidden...
12:51:00...	MetadataSearch	medium	BingWeb search finished successfully!! Total found result count: 3
12:51:00...	MetadataSearch	medium	GoogleWeb search finished successfully!! Total found result count: 90
1:07:49...	Crawling	medium	Domain found: egs.eccouncil.org
1:07:49...	Crawling	medium	Domain found: careers.eccouncil.org
1:07:49...	Crawling	medium	Domain found: codered.eccouncil.org
1:07:49...	Crawling	medium	Domain found: iclass.eccouncil.org
1:07:49...	Crawling	medium	Domain found: ilabs.eccouncil.org
1:07:49...	Crawling	medium	Domain found: store.eccouncil.org
1:07:49...	Crawling	medium	Domain found: cismag.eccouncil.org
1:07:50...	Crawling	medium	Domain found: aware.eccouncil.org
1:07:50...	Crawling	medium	Domain found: foundation.eccouncil.org
1:07:50...	Crawling	medium	Domain found: masterclass.eccouncil.org
1:07:51...	Crawling	medium	Domain found: cyberq.eccouncil.org

- Now, expand the **Document Analysis** node; further expand the **Metadata Summary** node. Here, information regarding users, folders, printers, software, etc. is displayed.

Note: The domain we used does not have information associated with metadata summary.

ID	Type	URL	Download	Download Date	Size	Metadata E.	Malware An...	Mod
0	pdf	https://aspen.eccouncil.org/Docs/Exam-Guides/ECCEExam-RPS-UserGuide...	X		2.98 MB	X	X	
1	html	https://cert.eccouncil.org/announcements.html	X		-	X	X	
2	pdf	https://cert.eccouncil.org/Images/doc/CHFI-New-Blueprint-v3.pdf	X		823.25 KB	X	X	
3	pdf	https://cert.eccouncil.org/Images/doc/CHFI-Exam-Blueprint-v2.1.pdf	X		5.09 MB	X	X	
4	pdf	https://aspen.eccouncil.org/Docs/UserGuides/AccessCourseware-UserGu...	X		433.5 KB	X	X	
5	pdf	https://cert.eccouncil.org/Images/doc/CEH-Handbook_v1.pdf	X		17.07 MB	X	X	
6	pdf	https://cert.eccouncil.org/Images/doc/CEH-Exam-Blueprint-v4.0.pdf	X		158.28 KB	X	X	
7	pdf	https://tyackson.wixsite.com/catholic/individual-presentations	X		-	X	X	
8	pdf	https://ciso.eccouncil.org/wp-content/uploads/2013/09/CCISO-Table-of-Co...	X		933.61 KB	X	X	
9	pdf	https://aspen.eccouncil.org/BecomeAnATC	X		-	X	X	
10	pdf	https://cert.eccouncil.org/Images/doc/CEH-Handbook-v5.pdf	X		6.79 MB	X	X	
11	pdf	https://cert.eccouncil.org/Images/doc/CHFI-Handbook-v5.pdf	X		2.13 MB	X	X	
12	pdf	https://aspen.eccouncil.org/Docs/Applications/ATC application Form v7.0.pdf	X		1.24 MB	X	X	
13	pdf	https://aspen.eccouncil.org/Docs/CISOMAG/CISO-MAG-October2020-Prev ...	X		34.3 MB	X	X	
14	pdf	https://cert.eccouncil.org/Images/doc/CND-Handbook-v5.pdf	X		8.62 MB	X	X	
15	pdf	https://cert.eccouncil.org/Images/doc/CEH-Handbook-v6.pdf	X		5.7 MB	X	X	
16	pdf	https://aspen.eccouncil.org/Docs/UserGuides/Instructions-AccessCoursew...	X		436.24 KB	X	X	
17	pdf	https://cert.eccouncil.org/Images/doc/CND Handbook_v1B.pdf	X		12.36 MB	X	X	
18	pdf	https://cert.eccouncil.org/Images/doc/ECSA Handbook_v1.pdf	X		3.89 MB	X	X	

Time Source Severity Message

12:50:57... MetadataSearch error An error has occurred on DuckDuckGoWeb: The remote server returned an error: (403) Forbidden.
12:51:00... MetadataSearch medium BingWeb search finished successfully!! Total found result count: 3
12:51:00... MetadataSearch medium GoogleWeb search finished successfully!! Total found result count: 90

- This concludes the demonstration of gathering useful information about the target organization using the FOCA tool.

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

24. Turn off the **Windows Server 2019** virtual machine.

Task 5: Footprinting a Target using BillCipher

BillCipher is an information gathering tool for a Website or IP address. Using this tool, you can gather information such as DNS Lookup, Whois lookup, GeoIP Lookup, Subnet Lookup, Port Scanner, Page Links, Zone Transfer, HTTP Header, etc.

Here, we will use the BillCipher tool to footprint a target website URL.

Note: Here, we will consider **www.certifiedhacker.com** as a target website. However, you can select a target domain of your choice.

1. Switch to the **Parrot Security** virtual machine. Click the **MATE Terminal** icon at the top-left corner 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. In the **Parrot Terminal** window, type **cd BillCipher** and press **Enter** to navigate to the BillCipher directory.

```
[attacker@parrot] - [~]
└─$ sudo su
[sudo] password for attacker:
[root@parrot] - [/home/attacker]
└─# cd BillCipher
[root@parrot] - [/home/attacker/BillCipher]
└─#
```

5. Now, type **python3 billcipher.py** and press **Enter** to launch the application.

```
[attacker@parrot] - [~]
└─$ sudo su
[sudo] password for attacker:
[root@parrot] - [/home/attacker]
└─# cd BillCipher
[root@parrot] - [/home/attacker/BillCipher]
└─# python3 billcipher.py
```

6. BillCipher application initializes. In the **Are you want to collect information of website or IP address?** option, type **website** and press **Enter**.

7. In the Enter the website address option, type the target website URL (here, www.certifiedhacker.com) and press Enter.

```
Applications Places System 🌐 🌐 🌐
python3 biltcipher.py - Parrot Terminal
File Edit View Search Terminal Help
#####
#      # # #      #      # # ##### #      # ######
#      # # #      #      # #      # #      # #      #
##### # #      #      # #      # ##### ##### #      #
#      # # #      #      # ##### #      # #      #####
#      # # #      #      # # #      #      # #      #      #
##### # ##### ##### ##### # #      #      # ##### #      # 2.1
Information Gathering tool for a Website or IP address

Are you want to collect information of website or IP address? [website/IP]: website
Enter the website address: www.certifiedhacker.com
```

8. BillCipher displays various available options that you can use to gather information regarding a target website.
 9. In the **What information would you like to collect?** option, type **1** to choose the **DNS Lookup** option and press **Enter**.

- The result appears, displaying the DNS information regarding the target website, as shown in the screenshot.
 - In the **Do you want to continue?** option, type **Yes** and press **Enter** to continue.

```
python3 billcipher.py - Parrot Terminal
File Edit View Search Terminal Help
#      # # #      #      # # #      #      # #      #      #
##### # ##### ##### #### # #      #      # ##### #      # 2.1
Information Gathering tool for a Website or IP address

Are you want to collect information of website or IP address? [website/IP]: website
Enter the website address: www.certifiedhacker.com

1) DNS Lookup          13) Host DNS Finder
2) Whois Lookup        14) Reserve IP Lookup
3) GeoIP Lookup        15) Email Gathering (use Infoga)
4) Subnet Lookup       16) Subdomain listing (use Sublist3r)
5) Port Scanner        17) Find Admin login site (use Breacher)
6) Page Links          18) Check and Bypass CloudFlare (use HatCloud)
7) Zone Transfer        19) Website Copier (use httrack)
8) HTTP Header         20) Host Info Scanner (use WhatWeb)
9) Host Finder          21) About BillCipher
10) IP-Locator         22) Fuck Out Of Here (Exit)
11) Find Shared DNS Servers
12) Get Robots.txt

What information would you like to collect? (1-20): 1
A : 162.241.216.11
MX : 0 mail.certifiedhacker.com.
NS : ns2.bluehost.com.
NS : ns1.bluehost.com.
TXT : "v=spf1 a mx ptr include:bluehost.com ?all"
CNAME : certifiedhacker.com.
SOA : ns1.bluehost.com. dnsadmin.box5331.bluehost.com. 2018011205 86400 7200 3600000 300

Do you want to continue? [Yes/No]: Yes
```

12. Are you want to collect information of website or IP address? option appears, type website and press Enter.
13. In the Enter the website address option, type the target website URL (here, www.certifiedhacker.com) and press Enter.
14. Now, type 3 and press Enter to choose the **GeoIP Lookup** option from the available information gathering options.
15. The result appears, displaying the **GeoIP Lookup** information of the target website, as shown in the screenshot.
16. In the Do you want to continue? option, type Yes and press Enter to continue.

```
python3 billcipher.py - Parrot Terminal
File Edit View Search Terminal Help
CNAME : certifiedhacker.com.
SOA : ns1.bluehost.com. dnsadmin.box5331.bluehost.com. 2018011205 86400 7200 3600000 300
Do you want to continue? [Yes/No]: Yes
Are you want to collect information of website or IP address? [website/IP]: website
Enter the website address: www.certifiedhacker.com

1) DNS Lookup
2) Whois Lookup
3) GeoIP Lookup
4) Subnet Lookup
5) Port Scanner
6) Page Links
7) Zone Transfer
8) HTTP Header
9) Host Finder
10) IP-Locator
11) Find Shared DNS Servers
12) Get Robots.txt
13) Host DNS Finder
14) Reserve IP Lookup
15) Email Gathering (use Infoga)
16) Subdomain listing (use Sublist3r)
17) Find Admin login site (use Breacher)
18) Check and Bypass CloudFlare (use HatCloud)
19) Website Copier (use httrack)
20) Host Info Scanner (use WhatWeb)
21) About BillCipher
22) Fuck Out Of Here (Exit)

What information would you like to collect? (1-20): 3
IP Address: 162.241.216.11
Country: United States
State:
City:
Latitude: 37.751
Longitude: -97.822

Do you want to continue? [Yes/No]: Yes
```

17. Are you want to collect information of website or IP address? option appears, type **website** and press **Enter**.
18. In the **Enter the website address** option, type the target website URL (here, **www.certifiedhacker.com**) and press **Enter**.
19. Now, type **4** and press **Enter** to choose the **Subnet Lookup** option from the available information gathering options.
20. The result appears, displaying the **Subnet Lookup** information of the target website.
21. In the **Do you want to continue?** option, type **Yes** and press **Enter** to continue.

The screenshot shows a terminal window titled "python3 billcipher.py - Parrot Terminal". The window has a dark background with green text. At the top, there's a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The title bar also shows the date and time: "Mon Mar 21, 08:50".
The terminal content is as follows:
Do you want to continue? [Yes/No]: Yes
Are you want to collect information of website or IP address? [website/IP]: website
Enter the website address: www.certifiedhacker.com
A list of options is displayed:
1) DNS Lookup 13) Host DNS Finder
2) Whois Lookup 14) Reserve IP Lookup
3) GeoIP Lookup 15) Email Gathering (use Infoga)
4) Subnet Lookup 16) Subdomain listing (use Sublist3r)
5) Port Scanner 17) Find Admin Login site (use Breacher)
6) Page Links 18) Check and Bypass CloudFlare (use HatCloud)
7) Zone Transfer 19) Website Copier (use httrack)
8) HTTP Header 20) Host Info Scanner (use WhatWeb)
9) Host Finder 21) About BillCipher
10) IP-Locator 22) Fuck Out Of Here (Exit)
11) Find Shared DNS Servers
12) Get Robots.txt
What information would you like to collect? (1-20): 4
Address = 162.241.216.11
Network = 162.241.216.11 / 32
Netmask = 255.255.255.255
Broadcast = not needed on Point-to-Point links
Wildcard Mask = 0.0.0.0
Hosts Bits = 0
Max. Hosts = 1 (2^0 - 0)
Host Range = { 162.241.216.11 - 162.241.216.11 }
Do you want to continue? [Yes/No]: Yes
At the bottom, there's a status bar with "Menu", "[clear - Parrot Terminal]", and "python3 billcipher.py - P...".

22. Are you want to collect information of website or IP address? option appears, type website and press Enter.
23. In the Enter the website address option, type the target website URL (here, www.certifiedhacker.com) and press Enter.
24. Now, type 6 and press Enter to choose the Page Links option from the available information gathering options.
25. The result appears, displaying a list of Visible links and Hidden links of the target website, as shown in the screenshot.
26. In the Do you want to continue? option, type Yes and press Enter to continue.

The screenshot shows a terminal window titled "python3 billcipher.py - Parrot Terminal". The window displays the following text:

```
Are you want to collect information of website or IP address? [website/IP]: website
Enter the website address: www.certifiedhacker.com

1) DNS Lookup          13) Host DNS Finder
2) Whois Lookup        14) Reserve IP Lookup
3) GeoIP Lookup         15) Email Gathering (use Infoga)
4) Subnet Lookup       16) Subdomain listing (use Sublist3r)
5) Port Scanner         17) Find Admin login site (use Breacher)
6) Page Links           18) Check and Bypass CloudFlare (use HatCloud)
7) Zone Transfer        19) Website Copier (use httrack)
8) HTTP Header          20) Host Info Scanner (use WhatWeb)
9) Host Finder          21) About BillCipher
10) IP-Locator          22) Fuck Out Of Here (Exit)
11) Find Shared DNS Servers
12) Get Robots.txt

What information would you like to collect? (1-20): 6
http://certifiedhacker.com/P-folio/index.html
http://certifiedhacker.com/Online Booking/index.htm
http://certifiedhacker.com/corporate-learning-website/01-homepage.html
http://certifiedhacker.com/Real Estates/index.html
http://certifiedhacker.com/Recipes/index.html
http://certifiedhacker.com/Social Media/index.html
http://certifiedhacker.com/Turbo Max/index.htm
http://certifiedhacker.com/Under Construction/index.html
http://certifiedhacker.com/Under the trees/index.html
http://certifiedhacker.com/

Do you want to continue? [Yes/No]: Yes
```

27. Are you want to collect information of website or IP address? option appears, type website and press Enter.
28. In the Enter the website address option, type the target website URL (here, www.certifiedhacker.com) and press Enter.
29. Now, type 8 and press Enter to choose the HTTP Header option from the available information gathering options.
30. The result appears, displaying information regarding the HTTP header of the target website, as shown in the screenshot.
31. In the Do you want to continue? option, type Yes and press Enter to continue.

```
python3 billcipher.py - Parrot Terminal
File Edit View Search Terminal Help

1) DNS Lookup          13) Host DNS Finder
2) Whois Lookup        14) Reserve IP Lookup
3) GeoIP Lookup        15) Email Gathering (use Infoga)
4) Subnet Lookup       16) Subdomain listing (use Sublist3r)
5) Port Scanner        17) Find Admin login site (use Breacher)
6) Page Links          18) Check and Bypass CloudFlare (use HatCloud)
7) Zone Transfer       19) Website Copier (use httrack)
8) HTTP Header         20) Host Info Scanner (use WhatWeb)
9) Host Finder         21) About BillCipher
10) IP-Locator         22) Fuck Out Of Here (Exit)
11) Find Shared DNS Servers
12) Get Robots.txt

What information would you like to collect? (1-20): 8
HTTP/1.1 200 OK
Date: Mon, 21 Mar 2022 12:52:46 GMT
Server: nginx/1.19.10
Content-Type: text/html
Content-Length: 3228
Last-Modified: Thu, 10 Feb 2011 11:01:38 GMT
Vary: Accept-Encoding
Content-Encoding: gzip
Host-header: c2hhcmVklmJsdWob3N0LmNvbQ==
X-Server-Cache: true
X-Proxy-Cache: HIT
Accept-Ranges: bytes

Do you want to continue? [Yes/No]: Yes
```

32. Are you want to collect information of website or IP address? option appears, type website and press Enter.
33. In the Enter the website address option, type the target website URL (here, www.certifiedhacker.com) and press Enter.
34. Now, type 9 and press Enter to choose Host Finder option from the available information gathering option.
35. The result appears, displaying information regarding the IP address of the target website, as shown in the screenshot.3
36. In the Do you want to continue? option, type Yes and press Enter to continue.

The screenshot shows a terminal window titled "python3 billcipher.py - Parrot Terminal". The window has a dark theme with green text. At the top, there's a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The title bar shows the window name and the date and time: "Mon Mar 21, 08:53".
The terminal content starts with some red-colored HTTP header information:
Content-Encoding: gzip
host-header: c2hhcmVklmJsdWob3N0LmNvb0==
X-Server-Cache: true
X-Proxy-Cache: HIT
Accept-Ranges: bytes
Below this, a question is displayed: "Do you want to continue? [Yes/No]: Yes".
The next line asks: "Are you want to collect information of website or IP address? [website/IP]: website".
The user enters: "Enter the website address: www.certifiedhacker.com".
A list of options follows, numbered 1 through 22:

- 1) DNS Lookup
- 2) Whois Lookup
- 3) GeoIP Lookup
- 4) Subnet Lookup
- 5) Port Scanner
- 6) Page Links
- 7) Zone Transfer
- 8) HTTP Header
- 9) Host Finder
- 10) IP-Locator
- 11) Find Shared DNS Servers
- 12) Get Robots.txt
- 13) Host DNS Finder
- 14) Reserve IP Lookup
- 15) Email Gathering (use Infoga)
- 16) Subdomain listing (use Sublist3r)
- 17) Find Admin login site (use Breacher)
- 18) Check and Bypass CloudFlare (use HatCloud)
- 19) Website Copier (use httrack)
- 20) Host Info Scanner (use WhatWeb)
- 21) About BillCipher
- 22) Fuck Out Of Here (Exit)

The user then types: "What information would you like to collect? (1-20): 9".
The terminal then displays the IP address: "www.certifiedhacker.com, 162.241.216.11".
Finally, the question "Do you want to continue? [Yes/No]: Yes" is shown again at the bottom of the terminal window.

37. Are you want to collect information of website or IP address? option appears, type website and press Enter.
38. In the Enter the website address option, type the target website URL (here, www.certifiedhacker.com) and press Enter.
39. Now, type 19 and press Enter to choose the Website Copier (use httrack) option from the available information gathering options.
40. The tool starts mirroring the target website; this will take approximately 5 minutes.
41. After completion of the mirroring process, the mirrored website gets saved in the folder websource, as shown in the screenshot.
42. In the Do you want to continue? option, type No and press Enter to exit BillCiper.

The screenshot shows a terminal window titled "python3 billcipher.py - Parrot Terminal". The window has a dark theme with green text. It displays the following interaction:

```
Do you want to continue? [Yes/No]: Yes
Are you want to collect information of website or IP address? [website/IP]: website
Enter the website address: www.certifiedhacker.com
1) DNS Lookup
2) Whois Lookup
3) GeoIP Lookup
4) Subnet Lookup
5) Port Scanner
6) Page Links
7) Zone Transfer
8) HTTP Header
9) Host Finder
10) IP-Locator
11) Find Shared DNS Servers
12) Get Robots.txt
13) Host DNS Finder
14) Reserve IP Lookup
15) Email Gathering (use Infoga)
16) Subdomain listing (use Sublist3r)
17) Find Admin Login site (use Breacher)
18) Check and Bypass CloudFlare (use HatCloud)
19) Website Copier (use httrack)
20) Host Info Scanner (use WhatWeb)
21) About BillCipher
22) Fuck Out Of Here (Exit)

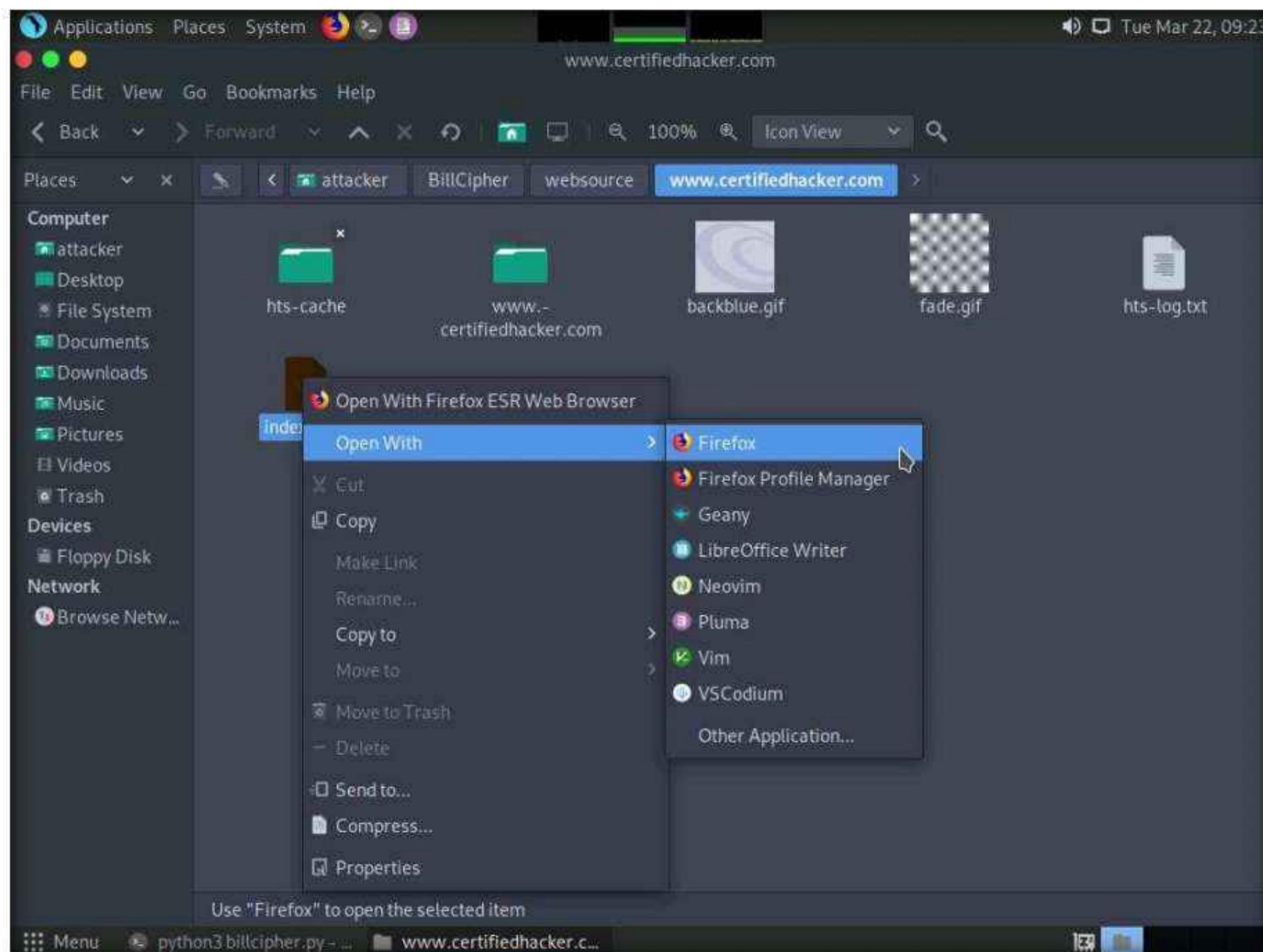
What information would you like to collect? (1-20): 19
WARNING! You are running this program as root!
It might be a good idea to run as a different user
Mirror launched on Tue, 22 Mar 2022 09:14:47 by HTTrack Website Copier/3.49-2+libhttplib.so.2 [XR&CO' 2014]
mirroring www.certifiedhacker.com with the wizard help...
Done.41: www.certifiedhacker.com/images/content/skin-changer/skin-changer-overlay.png (0 bytes) - OK
Thanks for using HTTrack!
The website source code was saved in folder 'websource'

Do you want to continue? [Yes/No]: No
```

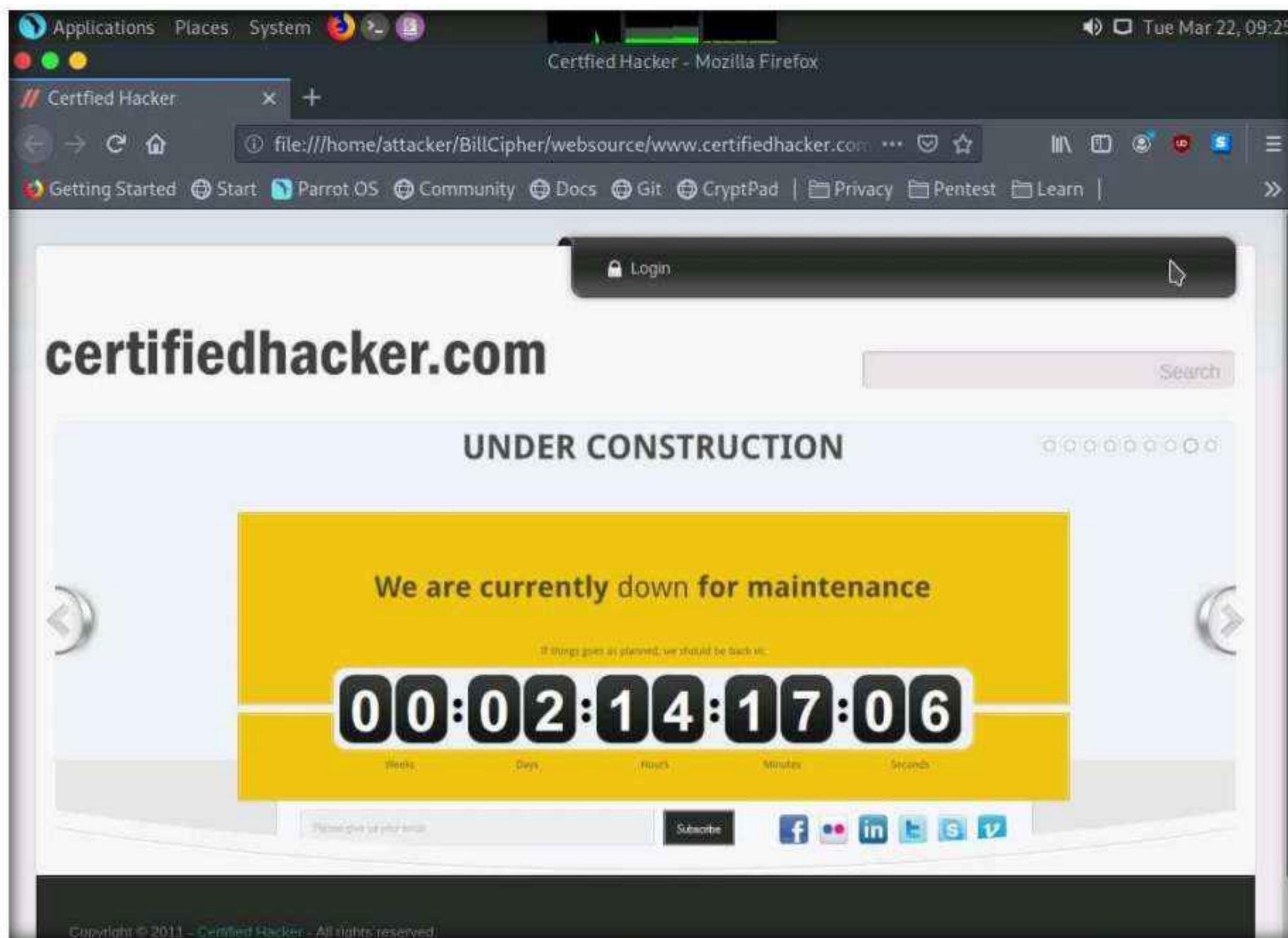
43. Now, click **Places** from the top section of the **Desktop** and click **Home Folder** from the context menu.



44. The **attacker** window appears, navigate to **BillCipher** --> **websource** --> **www.certifiedhacker.com** --> **www.certifiedhacker.com**. Right-click the **index.html** file and navigate to **Open With** --> **Firefox** to open the mirrored website.



45. The mirror target website (www.certifiedhacker.com) appears in the **Mozilla Firefox** browser, as shown in the screenshot.



46. Similarly, you can use other information gathering options to gather information about the target.
47. This concludes the demonstration of footprinting the target website URL using BillCipher.
48. Close all open windows and document all the acquired information.
49. Turn off the **Parrot Security** virtual machine.

Task 6: Footprinting a Target using OSINT Framework

OSINT Framework is an open-source intelligence gathering framework that helps security professionals for performing automated footprinting and reconnaissance, OSINT research, and intelligence gathering. It is focused on gathering information from free tools or resources. This framework includes a simple web interface that lists various OSINT tools arranged by category and is shown as an OSINT tree structure on the web interface.

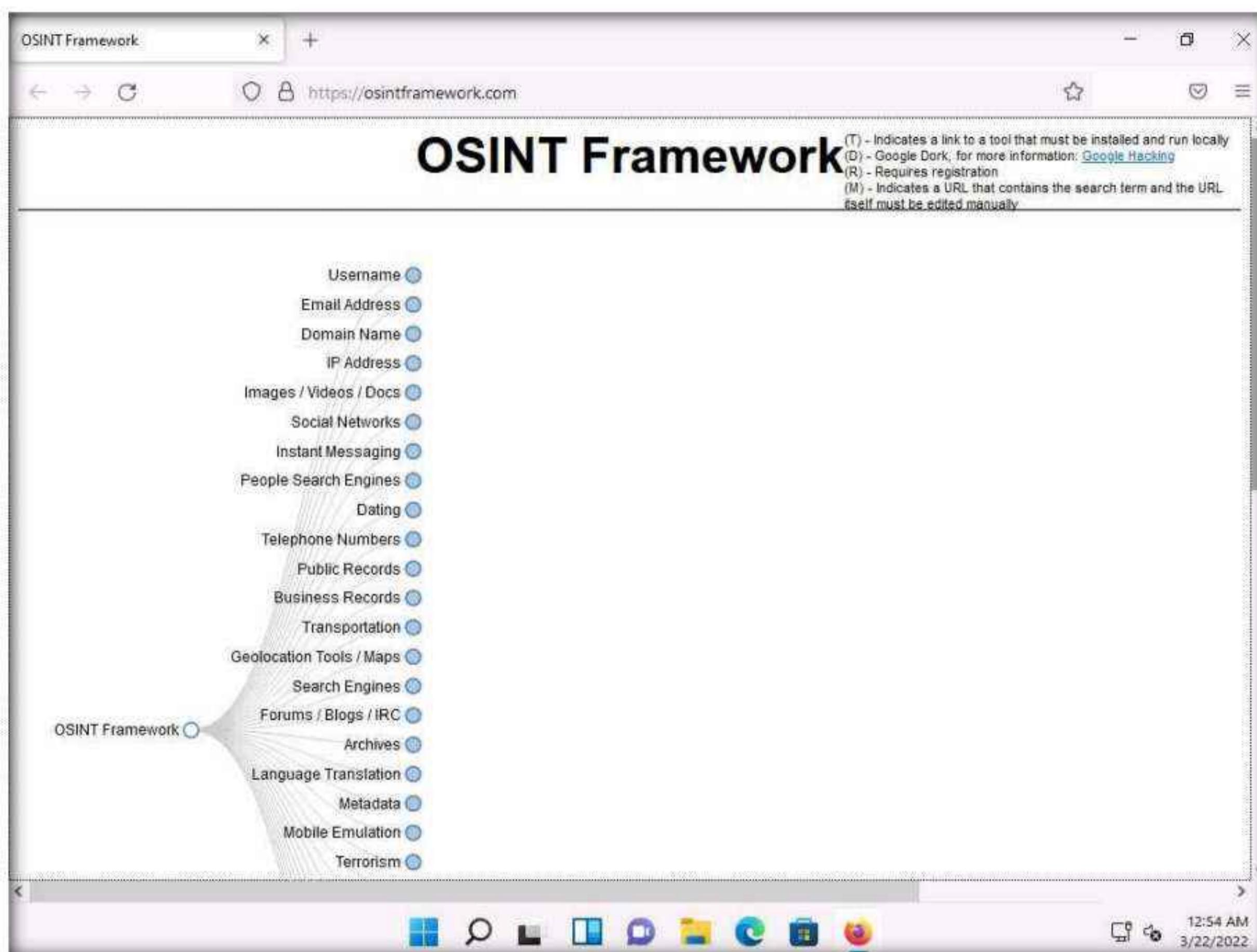
The OSINT Framework includes the following indicators with the available tools:

- (T) - Indicates a link to a tool that must be installed and run locally
- (D) - Google Dork

- (R) - Requires registration
- (M) - Indicates a URL that contains the search term and the URL itself must be edited manually

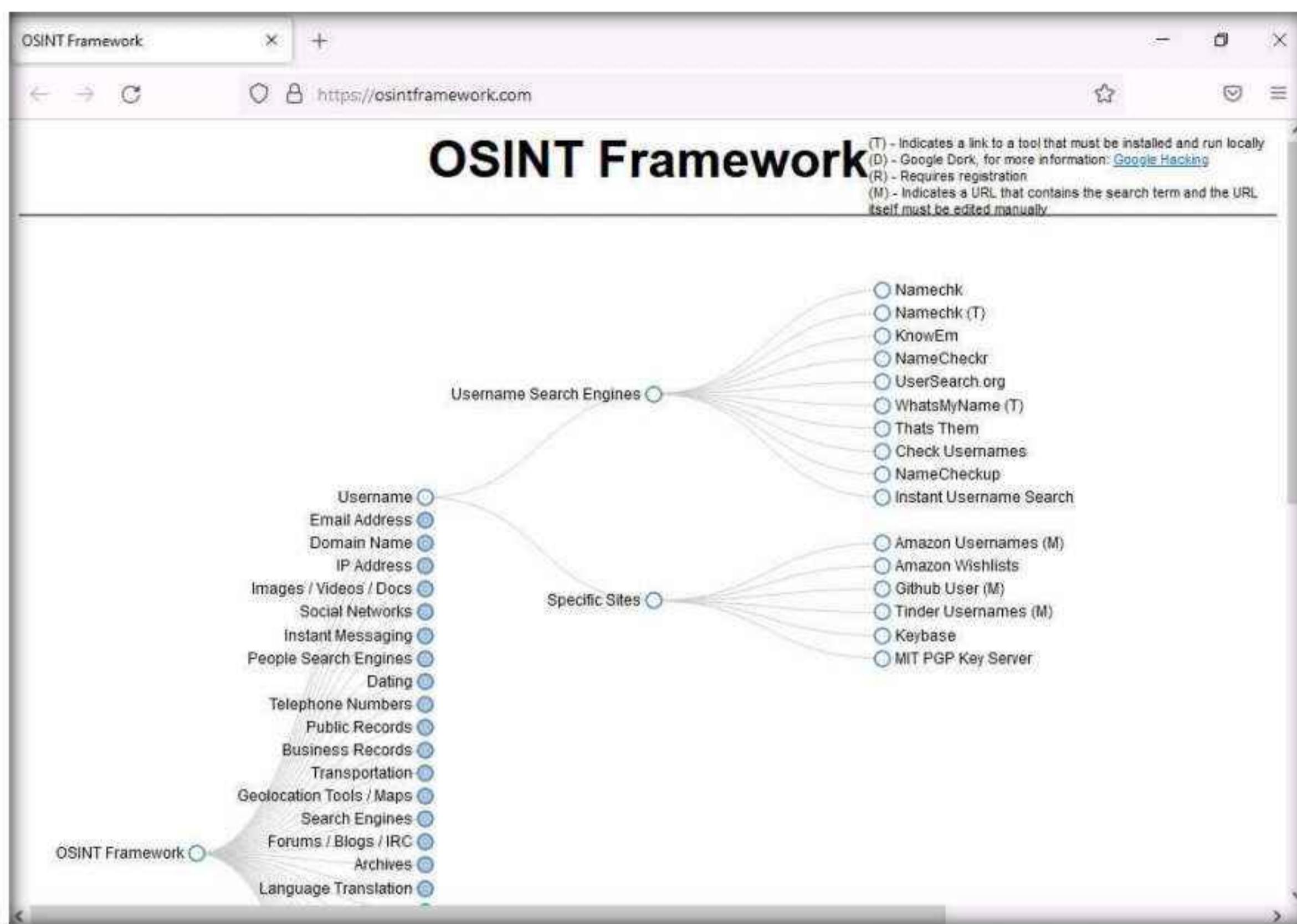
Here, we will use the OSINT Framework to explore footprinting categories and associated tools.

1. Switch to the **Windows 11** virtual machine.
2. Open any web browser (here, **Mozilla Firefox**). In the address bar of the browser place your mouse cursor, type <https://osintframework.com/> and press **Enter**.
3. **OSINT Framework** website appears; you can observe the OSINT tree on the left side of screen, as shown in the screenshot.

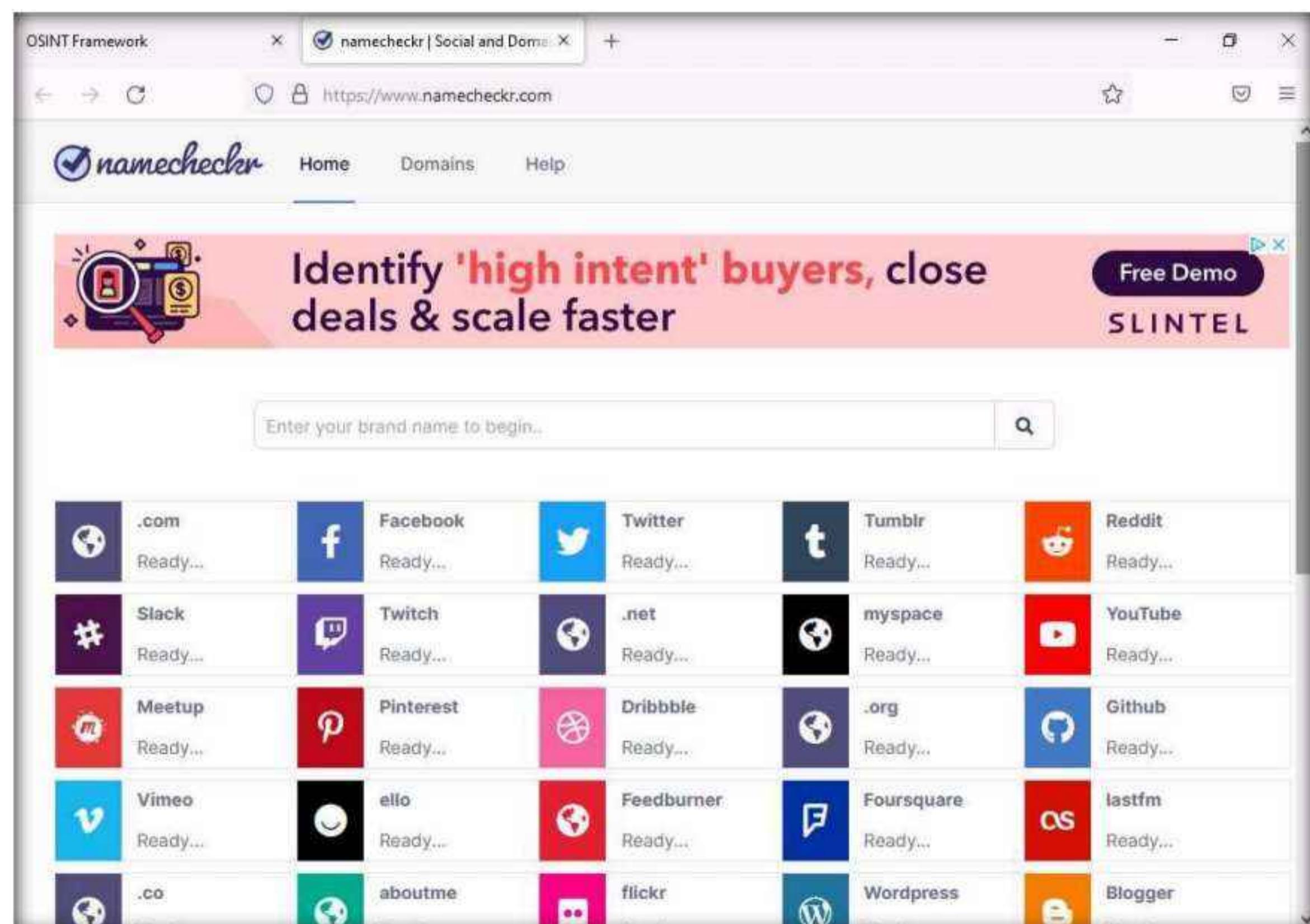


4. Clicking on any of the categories such as **Username**, **Email Address**, or **Domain Name** will make many useful resources appear on the screen in the form of a sub-tree.
5. Click the **Username** category and click to expand the **Username Search Engines** and **Specific Sites** sub-categories.
6. You can observe a list of OSINT tools filtered by sub-categories (**Username Search Engines** and **Specific Sites** sub-categories).

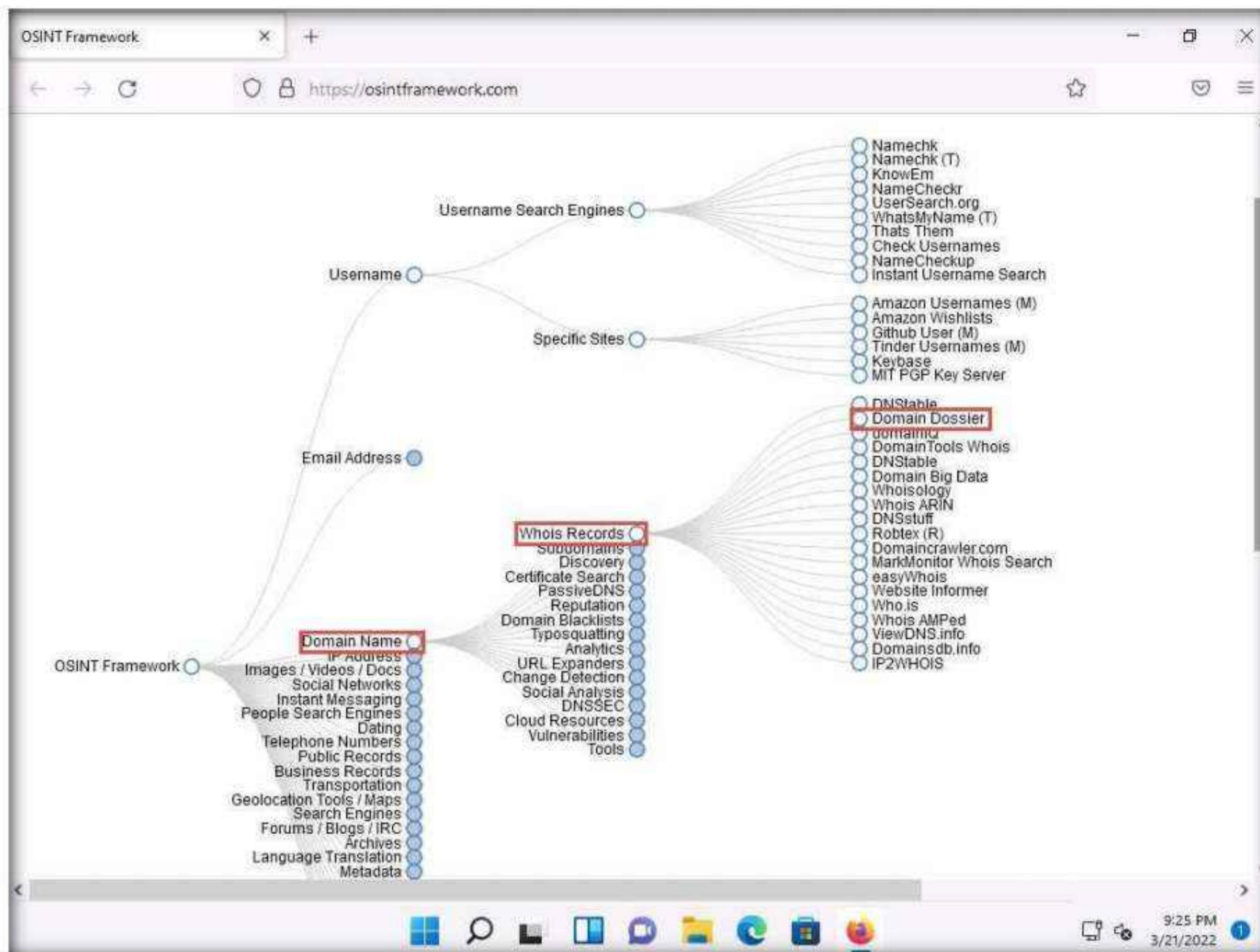
Module 02 – Footprinting and Reconnaissance



7. From the list of available tools under the **Username Search Engines** category, click on the **NameCheckr** tool to navigate to the **NameCheckr** website.
8. The **NameCheckr** website appears, as shown in the screenshot.



9. Close the current tab to navigate back to the OSINT Framework webpage.
10. Similarly, you can explore other tools from the list of mentioned tools under the **Username Search Engines** and **Specific Sites** sub-categories.
11. Now, click the **Domain Name** category, and its sub-categories appear. Click to expand the **Whois Records** sub-category.
12. A list of tools under the **Whois Records** sub-category appears; click the **Domain Dossier** tool.



13. The Domain Dossier website appears, as shown in the screenshot.

Note: The Domain Dossier tool generates reports from public records about domain names and IP addresses to help solve problems, investigate cybercrime, or just to better understand how things are set up.

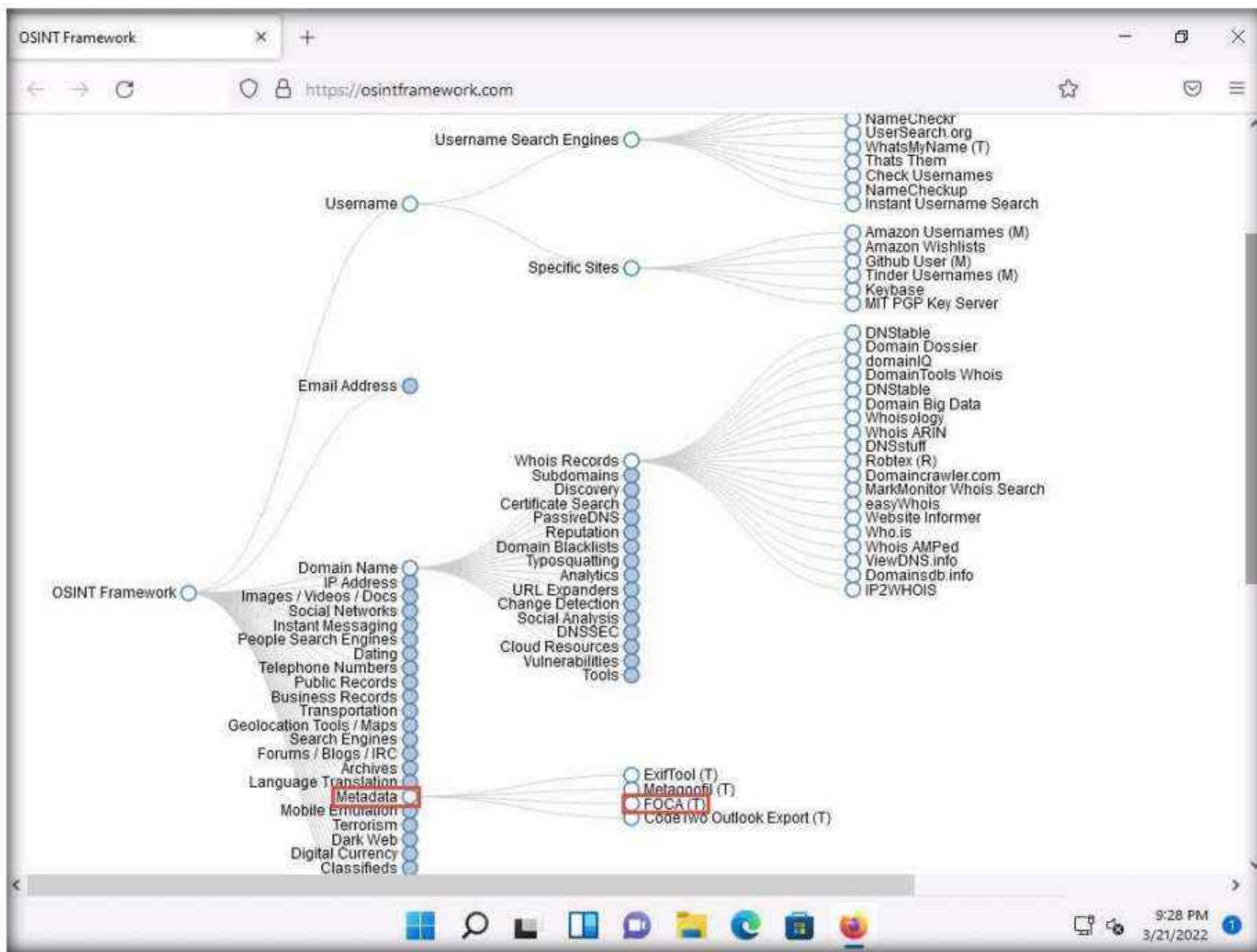
The Domain Dossier tool generates **reports from public records** about domain names and IP addresses to help solve problems, investigate cybercrime, or just better understand how things are set up. These reports may show you:

- Owner's contact information
- Registrar and registry information
- The company that is hosting a Web site
- Where an IP address is geographically located
- What type of server is at the address
- The upstream networks of a site
- and much more

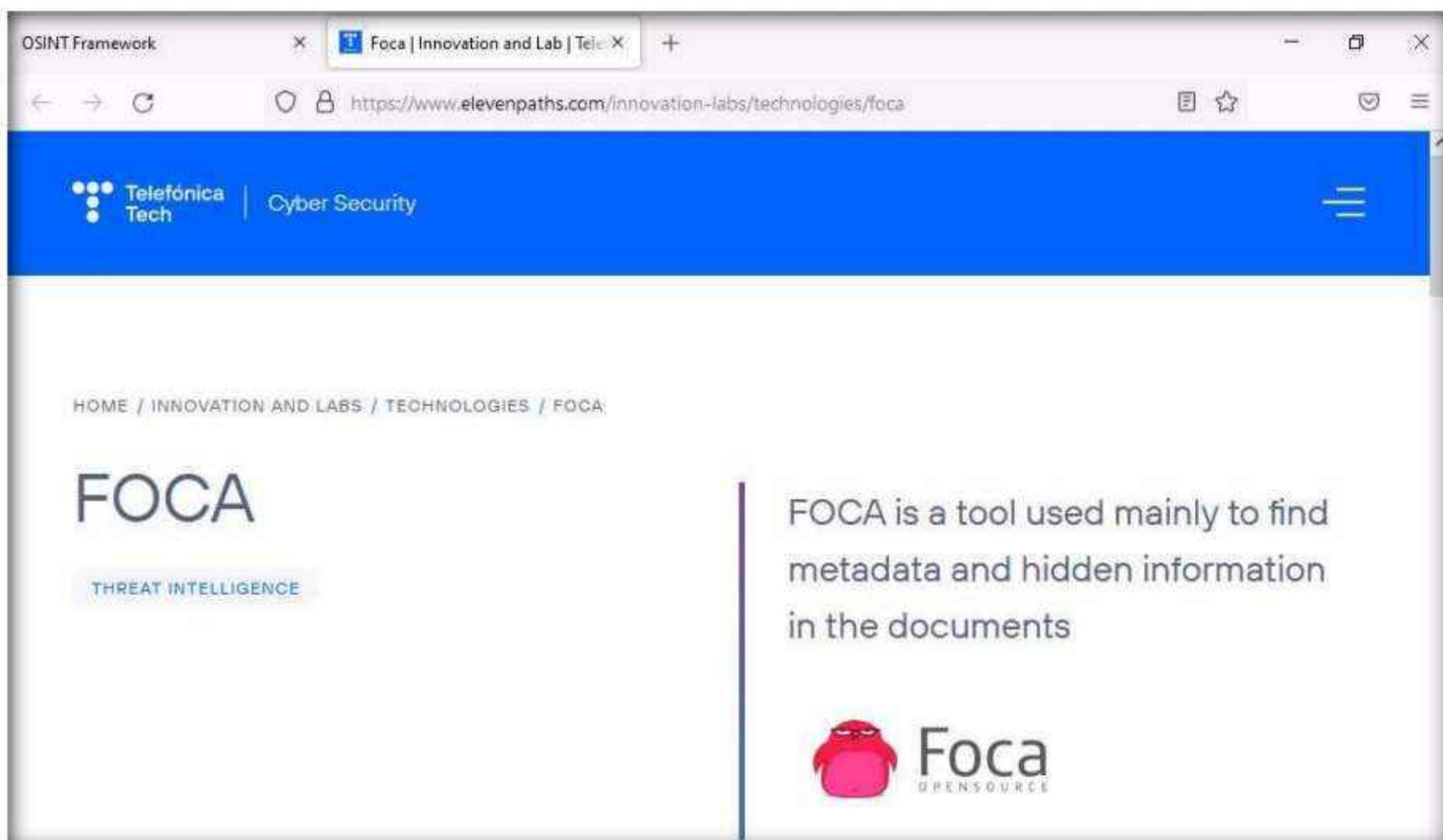
Domain Dossier normally gets records from their original sources *at the time you request them*, but it does keep copies in memory for up to 24 hours. Thus, if someone has already requested a particular Dossier, the records shown *could be up to a day old*.

14. Close the current tab to navigate back to the OSINT Framework webpage.

15. Now, click the **Metadata** category and click the **FOCA** tool from a list of available tools.



16. The **FOCA** website appears, displaying information about the tool along with its download link, as shown in the screenshot.



17. Similarly, you can explore other available categories such as **Email Address, IP Address, Social Networks, Instant Messaging**, etc. and the tools associated with each category. Using these tools, you can perform footprinting on the target organization.
18. This concludes the demonstration of performing footprinting using the OSINT Framework.
19. You can also use footprinting tools such as **Recon-Dog** (<https://www.github.com>), **Grecon** (<https://github.com>), **Th3Inspector** (<https://github.com>), **Raccoon** (<https://github.com>), **Orb** (<https://github.com>), etc. to gather additional information related to the target company.
20. Close all open windows and document all the acquired information.
21. Turn off the **Windows 11** virtual machine.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> CyberQ

CEH Lab Manual

Scanning Networks

Module 03

Scanning Networks

Network scanning refers to a set of procedures performed to identify the hosts, ports, and services running in a network.

Lab 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.

Lab 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

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Android virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 130 Minutes

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:

Lab No.	Lab Exercise Name	Core*	Self-study**	CyberQ ***
1	Perform Host Discovery	√	√	√
	1.1 Perform Host Discovery using Nmap	√		√
	1.2 Perform Host Discovery using Angry IP Scanner		√	√
2	Perform Port and Service Discovery	√	√	√
	2.1 Perform Port and Service Discovery using MegaPing		√	√
	2.2 Perform Port and Service Discovery using NetScanTools Pro		√	√

Module 03 – Scanning Networks

	2.3 Perform Port Scanning using sx Tool		✓	✓
	2.4 Explore Various Network Scanning Techniques using Nmap	✓		✓
	2.5 Explore Various Network Scanning Techniques using Hping3		✓	✓
3	Perform OS Discovery	✓	✓	✓
	3.1 Identify the Target System's OS with Time-to-Live (TTL) and TCP Window Sizes using Wireshark		✓	✓
	3.2 Perform OS Discovery using Nmap Script Engine (NSE)	✓		✓
	3.3 Perform OS Discovery using Unicornscan		✓	✓
4	Scan beyond IDS and Firewall	✓	✓	✓
	4.1 Scan beyond IDS/Firewall using various Evasion Techniques	✓		✓
	4.2 Create Custom Packets using Colasoft Packet Builder to Scan beyond IDS/Firewall		✓	✓
	4.3 Create Custom UDP and TCP Packets using Hping3 to Scan beyond IDS/Firewall	✓		✓
	4.4 Browse Anonymously using Proxy Switcher		✓	
	4.5 Browse Anonymously using CyberGhost VPN		✓	
5	Perform Network Scanning using Various Scanning Tools	✓		✓
	5.1 Scan a Target Network using Metasploit	✓		✓

Remark

EC-Council has prepared a considered amount of lab exercises for student to practice during the 5-day class and at their free time to enhance their knowledge and skill.

***Core** - Lab exercise(s) marked under Core are recommended by EC-Council to be practised during the 5-day class.

****Self-study** - Lab exercise(s) marked under self-study is for students to practise at their free time. Steps to access the additional lab exercises can be found in the first page of CEHv12 volume 1 book.

*****CyberQ** - Lab exercise(s) marked under CyberQ are available in our CyberQ solution. CyberQ is a cloud-based virtual lab environment preconfigured with vulnerabilities, exploits, tools and scripts, and can be accessed from anywhere with an Internet connection. If you are interested to learn more about our CyberQ solution, please contact your training center or visit <https://www.cyberq.io/>.

Lab Analysis

Analyze and document the results related to this lab exercise. Give an opinion on your target's security posture.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Lab

1

Perform Host Discovery

Host discovery is the process of identifying active hosts in the target network.

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

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Android virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 10 Minutes

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

Lab Tasks

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.

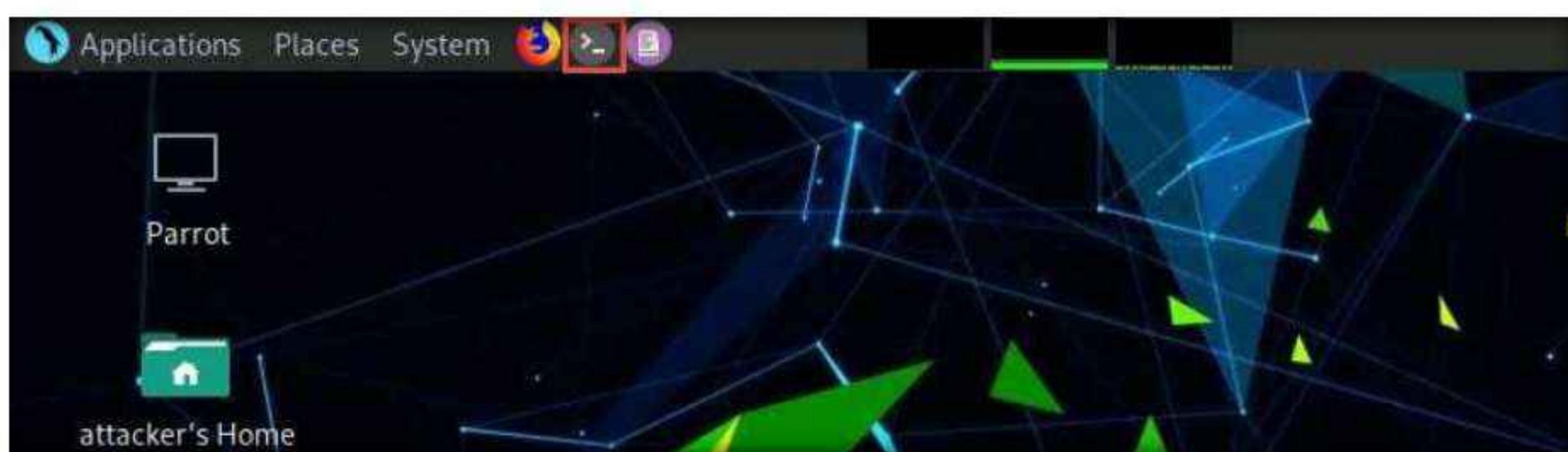
Here, we will use Nmap to discover a list of live hosts in the target network. We can use Nmap to scan the active hosts in the target network using various host discovery techniques such as ARP ping scan, UDP ping scan, ICMP ECHO ping scan, ICMP ECHO ping sweep, etc.

1. Turn on the **Windows 11**, **Windows Server 2022**, **Windows Server 2019**, **Parrot Security**, **Ubuntu**, and **Android** virtual machines.
2. In the login page of **Parrot Security** machine, 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.

The screenshot shows a Parrot OS desktop environment. A terminal window titled "Parrot Terminal" is open. The terminal window has a dark background with white text. It displays the following command sequence:

```
[attacker@parrot] ~
$ sudo su
[sudo] password for attacker:
[root@parrot] ~
# nmap -sn -PR 10.10.1.22
```

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.

The screenshot shows a Parrot OS desktop environment. A terminal window titled "nmap -sn -PR 10.10.1.22 - Parrot Terminal" is open. The terminal window has a dark background with white text. It displays the command entered and the resulting Nmap scan report:

```
[attacker@parrot] ~
$ sudo su
[sudo] password for attacker:
[root@parrot] ~
# nmap -sn -PR 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:11 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00052s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
[root@parrot] ~
#
```

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.

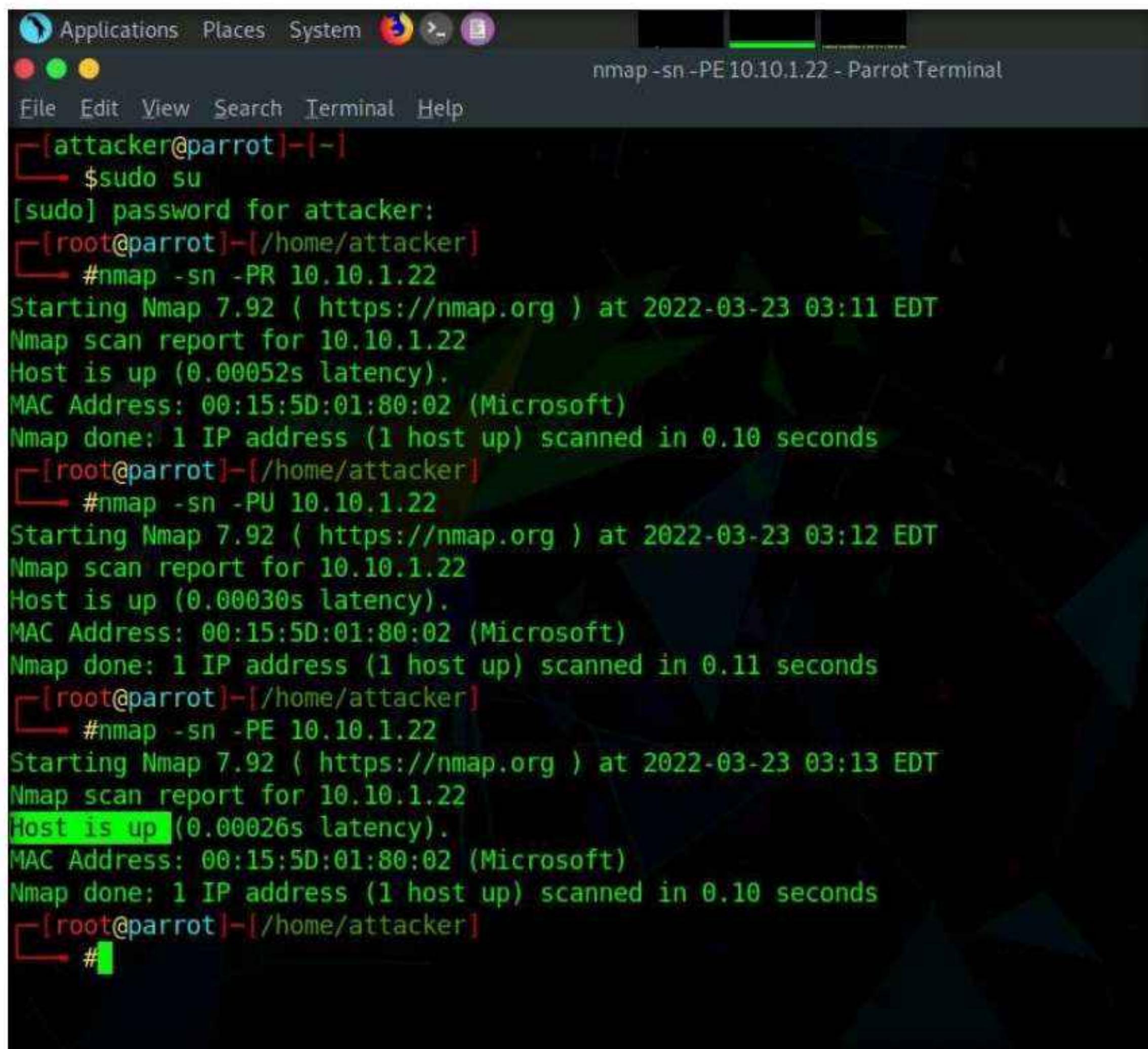
Note: The UDP ping scan sends UDP packets to the target host; a UDP response means that the host is active. If the target host is offline or unreachable, various error messages such as “host/network unreachable” or “TTL exceeded” could be returned.

```
[attacker@parrot](-)
└─$ sudo su
[sudo] password for attacker:
[root@parrot](-[/home/attacker]
└─# nmap -sn -PR 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:11 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00052s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
[root@parrot](-[/home/attacker]
└─# nmap -sn -PU 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:12 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00030s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
[root@parrot](-[/home/attacker]
└─#
```

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.



The screenshot shows a terminal window titled "nmap -sn -PE 10.10.1.22 - Parrot Terminal". The terminal output is as follows:

```
[attacker@parrot] -[~]
└─$ sudo su
[sudo] password for attacker:
[root@parrot] -[~/home/attacker]
└─# nmap -sn -PR 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:11 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00052s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
[root@parrot] -[~/home/attacker]
└─# nmap -sn -PU 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:12 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00030s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
[root@parrot] -[~/home/attacker]
└─# nmap -sn -PE 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:13 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00026s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
[root@parrot] -[~/home/attacker]
└─#
```

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.

```
nmap -sn -PE 10.10.1.10-23 - Parrot Terminal
[...]
# nmap -sn -PE 10.10.1.10-23
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:55 EDT
Nmap scan report for 10.10.1.11
Host is up (0.0011s latency).
MAC Address: 00:15:5D:01:80:00 (Microsoft)
Nmap scan report for 10.10.1.14
Host is up (0.00096s latency).
MAC Address: 02:15:5D:19:04:A7 (Unknown)
Nmap scan report for www.moviescope.com (10.10.1.19)
Host is up (0.00094s latency).
MAC Address: 02:15:5D:19:04:A4 (Unknown)
Nmap scan report for 10.10.1.22
Host is up (0.00021s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap scan report for 10.10.1.13
Host is up.
Nmap done: 14 IP addresses (5 hosts up) scanned in 1.33 seconds
[...]
#
```

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.

```
nmap -sn -PP 10.10.1.22 - Parrot Terminal
[...]
# nmap -sn -PP 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-23 03:58 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00070s latency).
MAC Address: 00:15:5D:01:80:02 (Microsoft)
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
[...]
#
```

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.

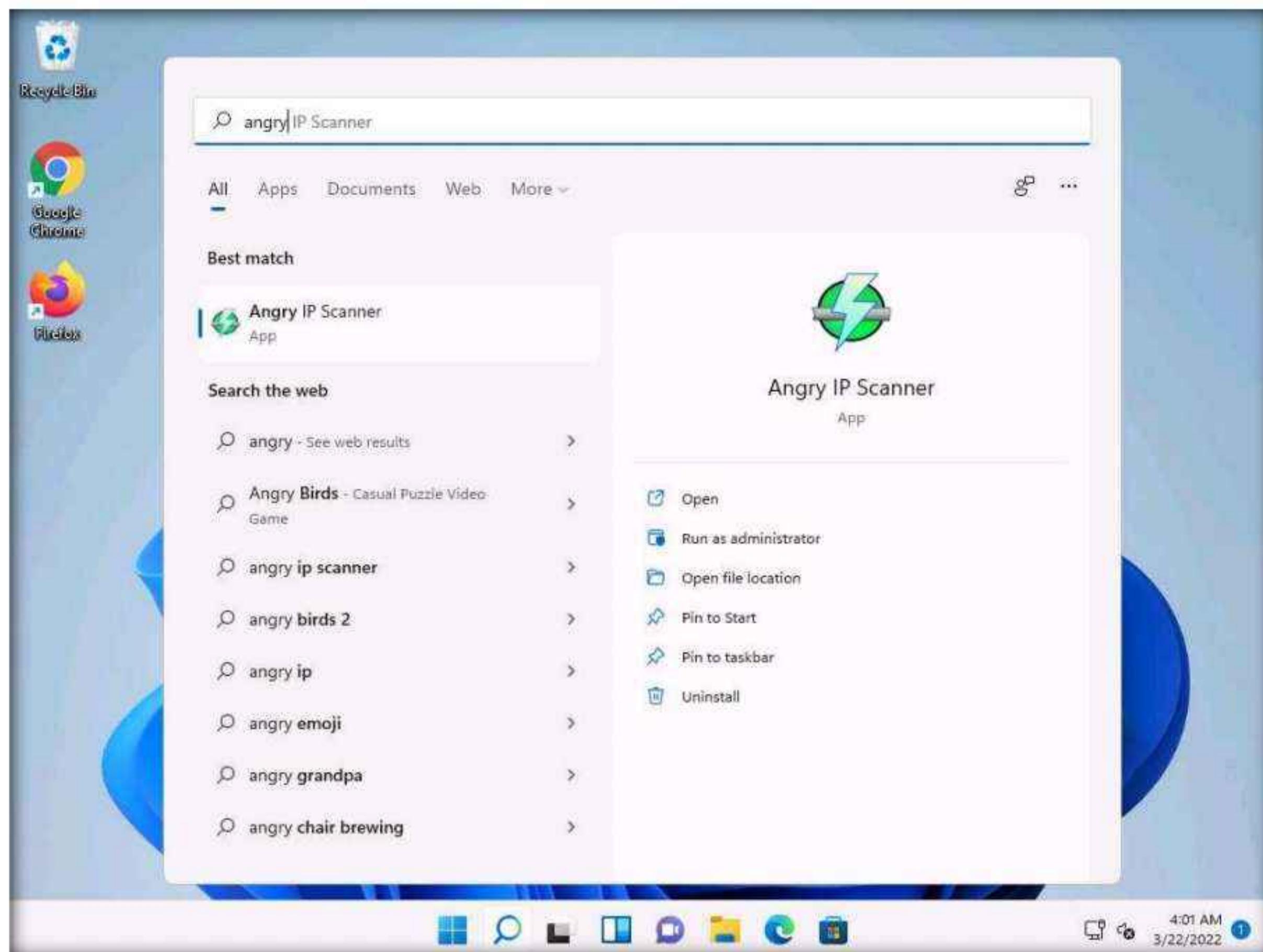
Note: Ensure that all the virtual machines are running.

1. Switch to the **Windows 11** virtual machine. By default, **Admin** user profile is selected, type **Pa\$\$w0rd** in the **Password** field and press **Enter** to login.

Note: If **Welcome to Windows** wizard appears, click **Continue** and in **Sign in with Microsoft** wizard, click **Cancel**.

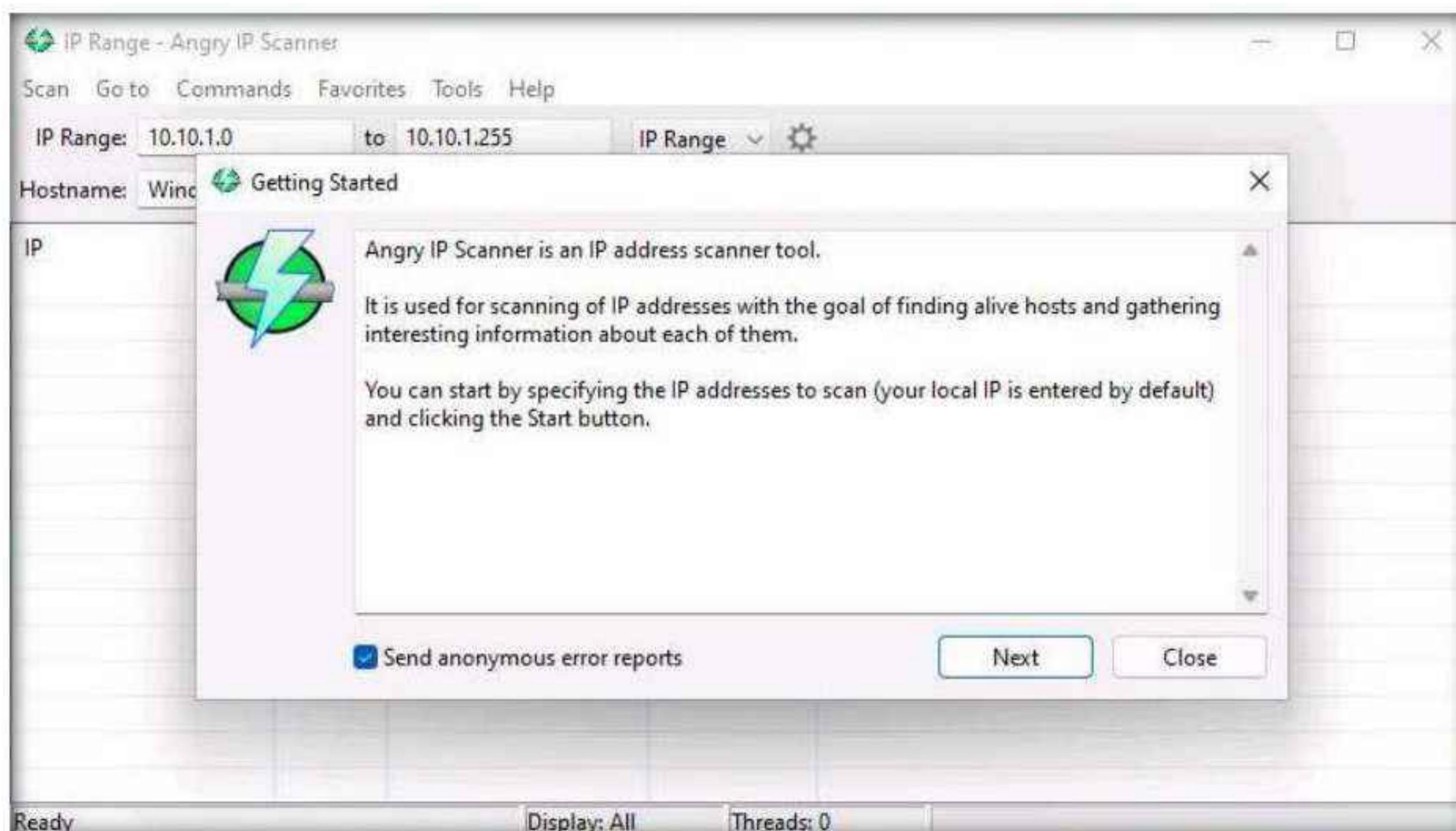
Note: Networks screen appears, click **Yes** to allow your PC to be discoverable by other PCs and devices on the network.

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.



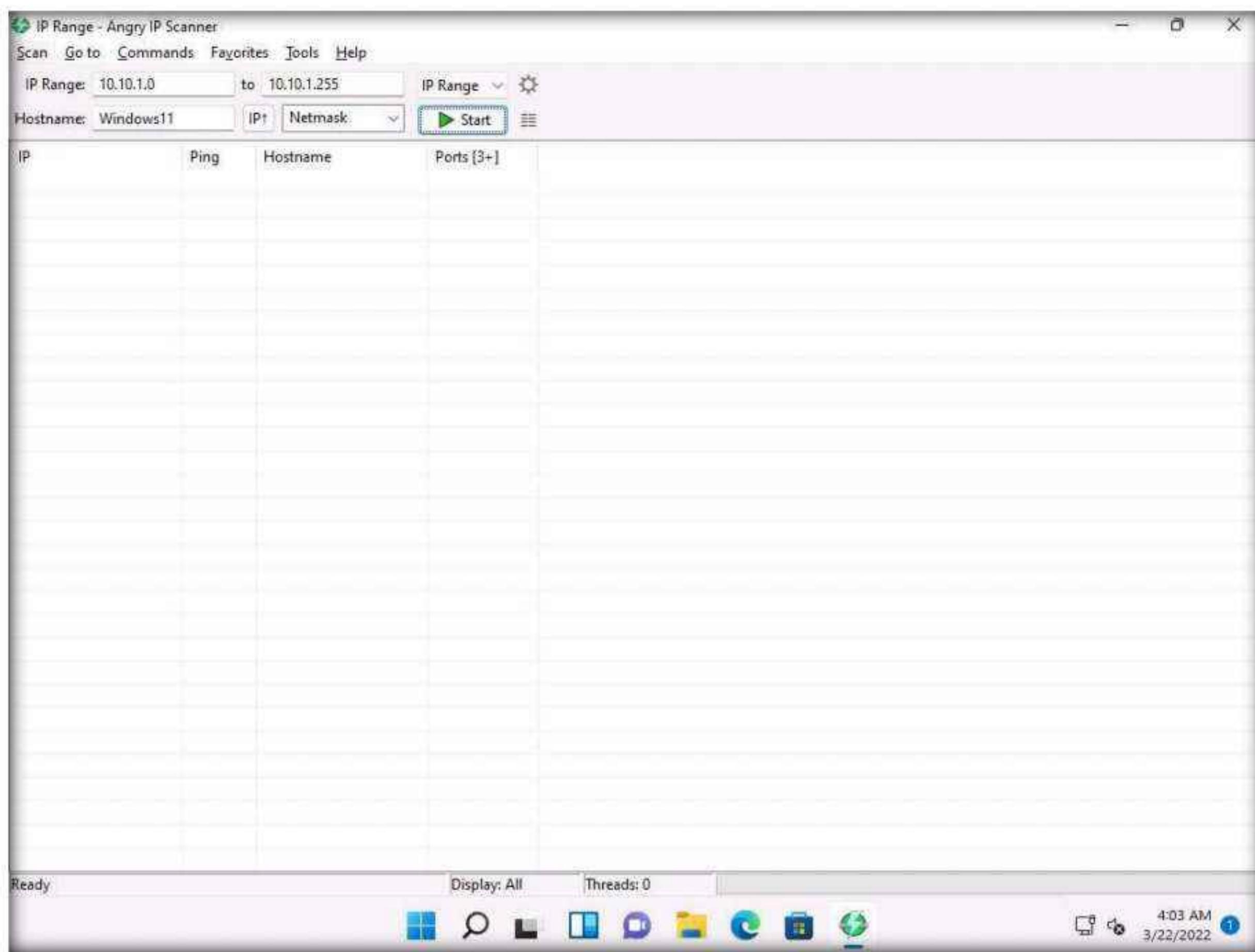
3. Angry IP Scanner starts, and a Getting Started window pops up. Click **Next**, follow the wizard, and click **Close**.

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

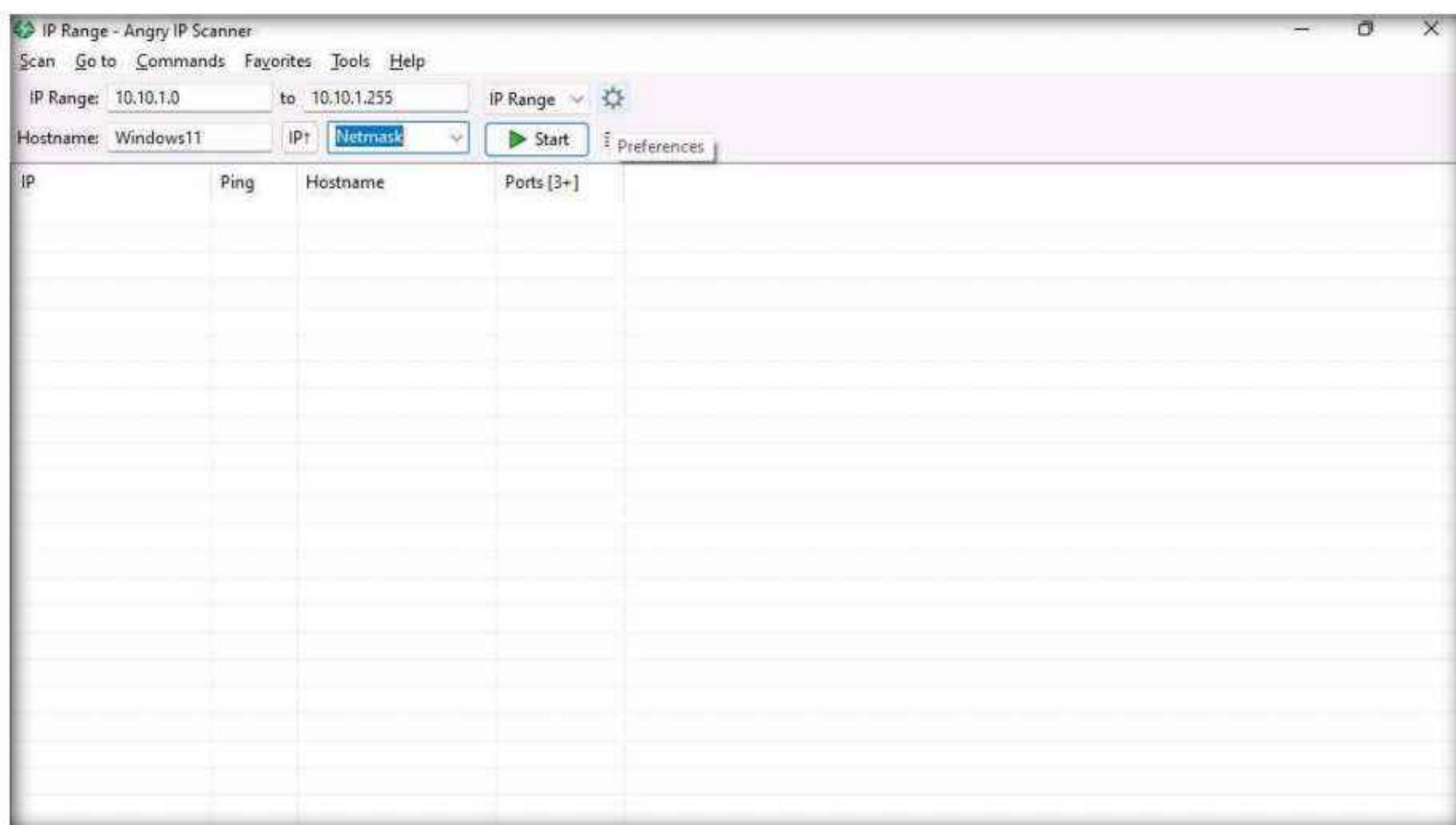


Module 03 – Scanning Networks

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

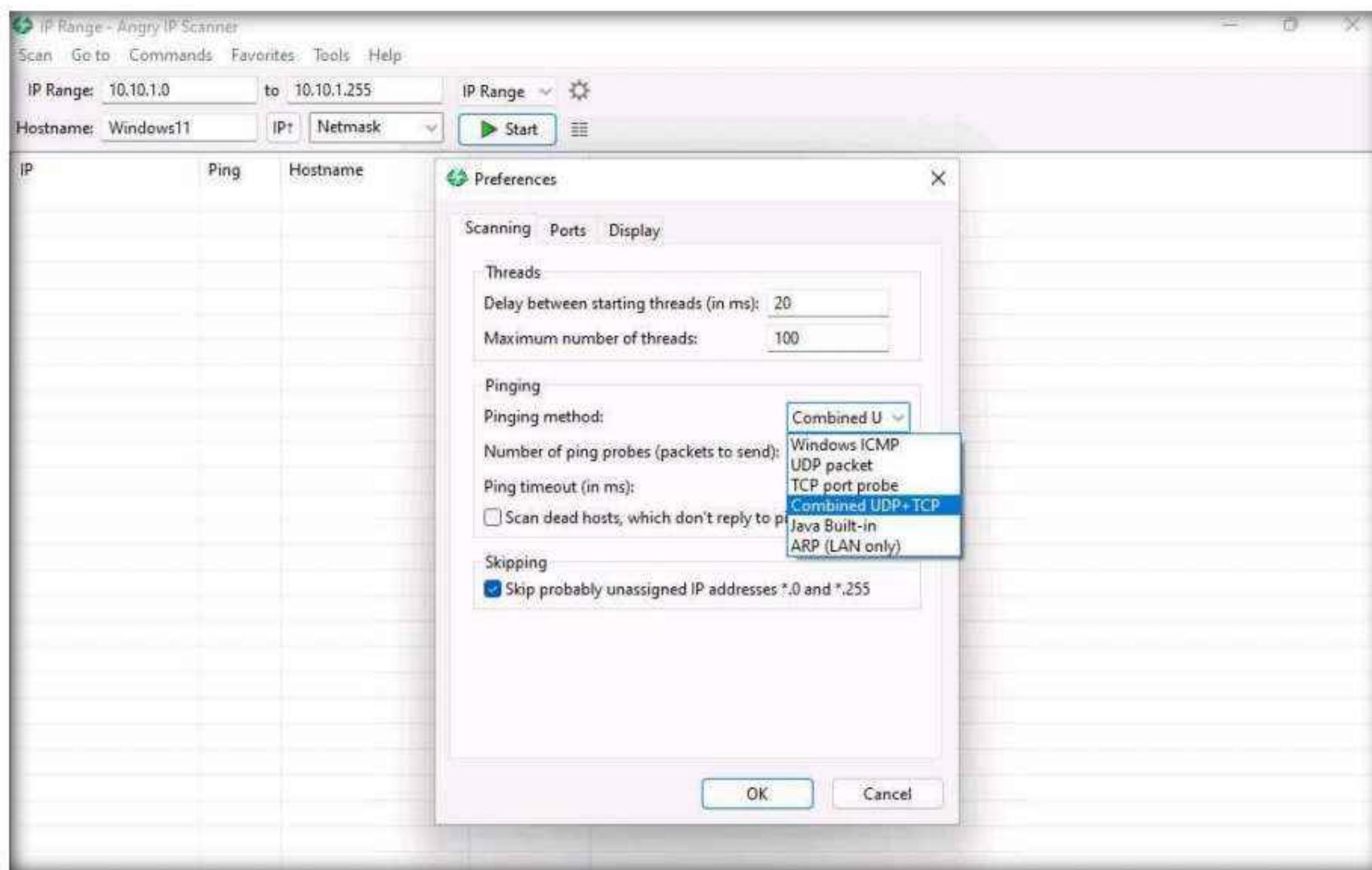


5. In the IP Range fields, type the IP range as **10.10.1.0** to **10.10.1.255** and click the Preferences icon beside the IP Range menu, as shown in the screenshot.

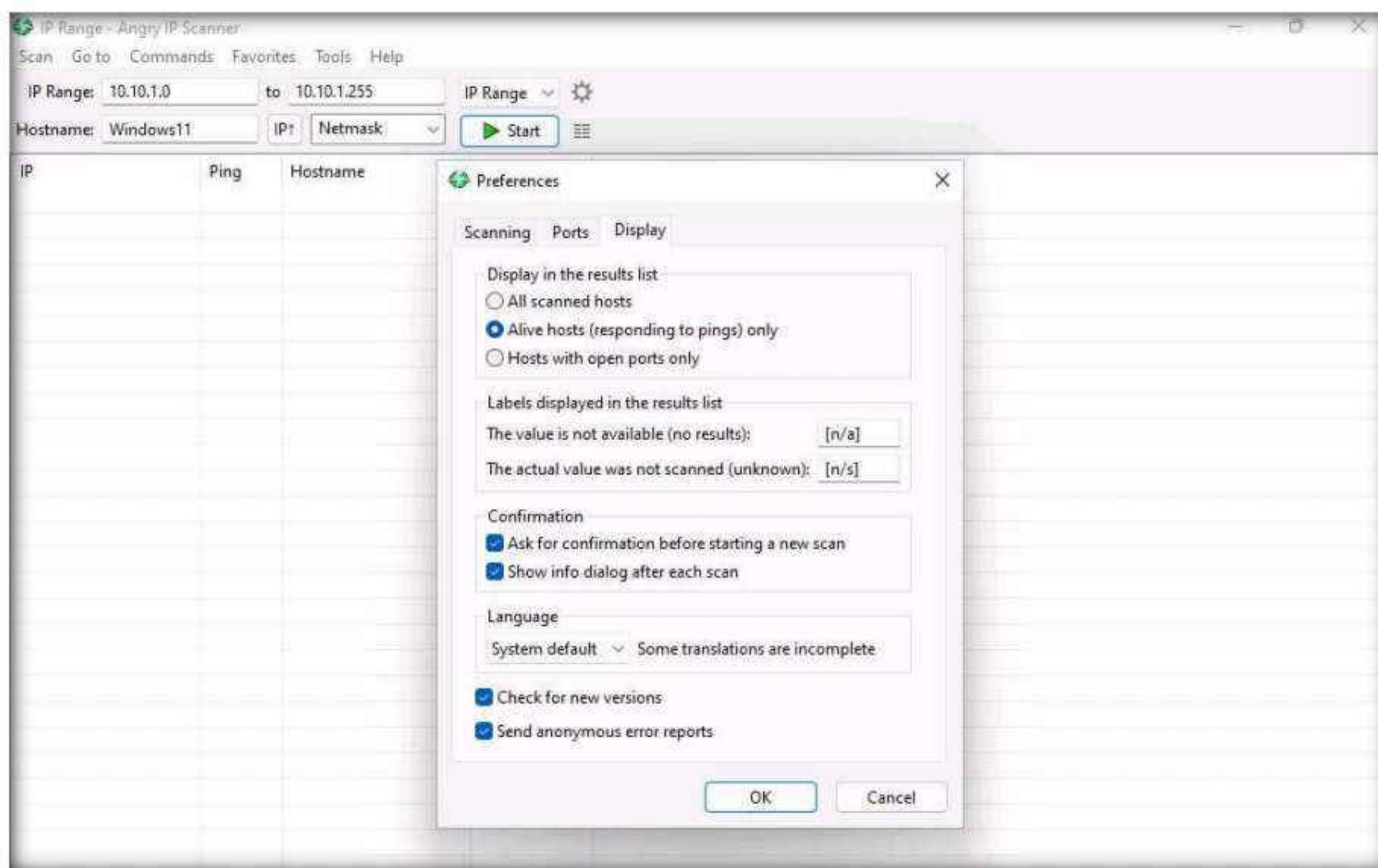


Module 03 – Scanning Networks

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.

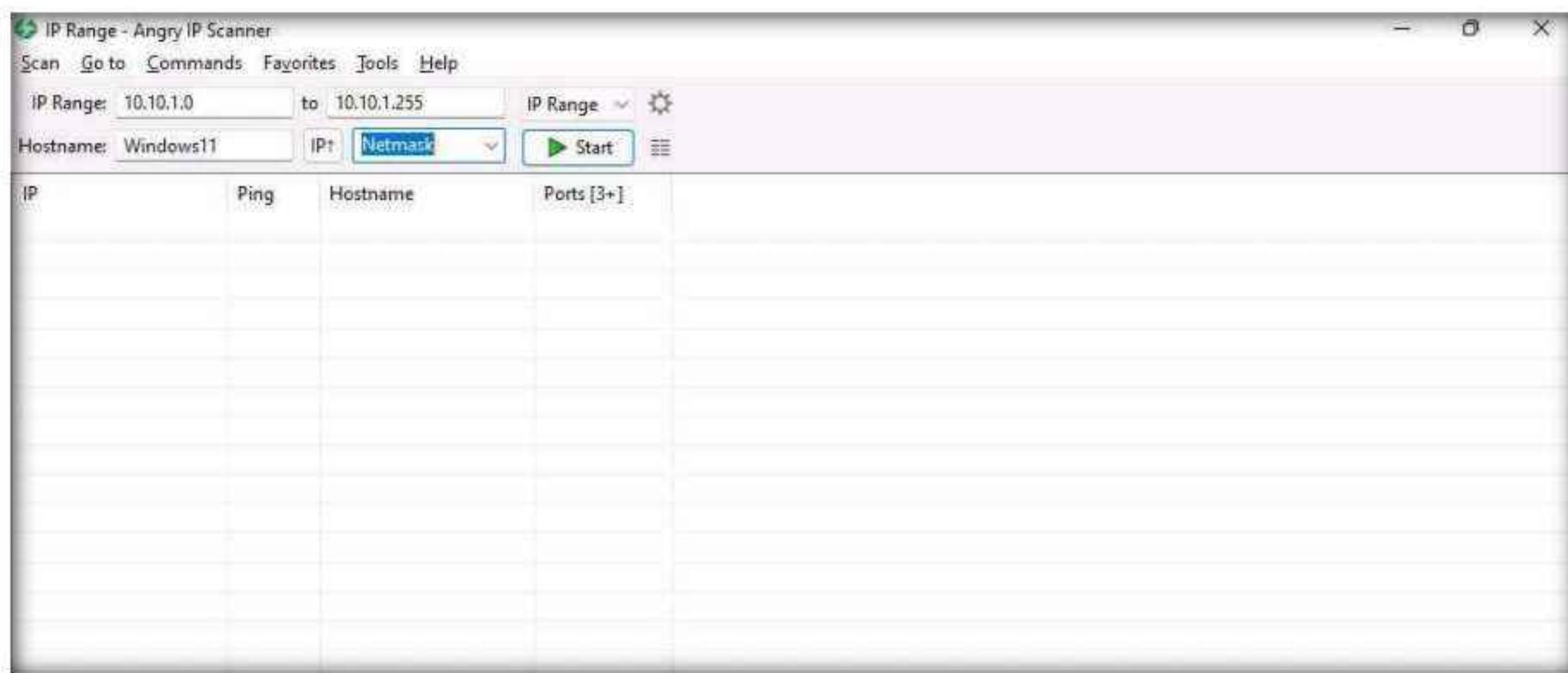


7. Now, switch to the **Display** tab. Under the **Display in the results list** section, select the **Alive hosts (responding to pings) only** radio button and click **OK**.

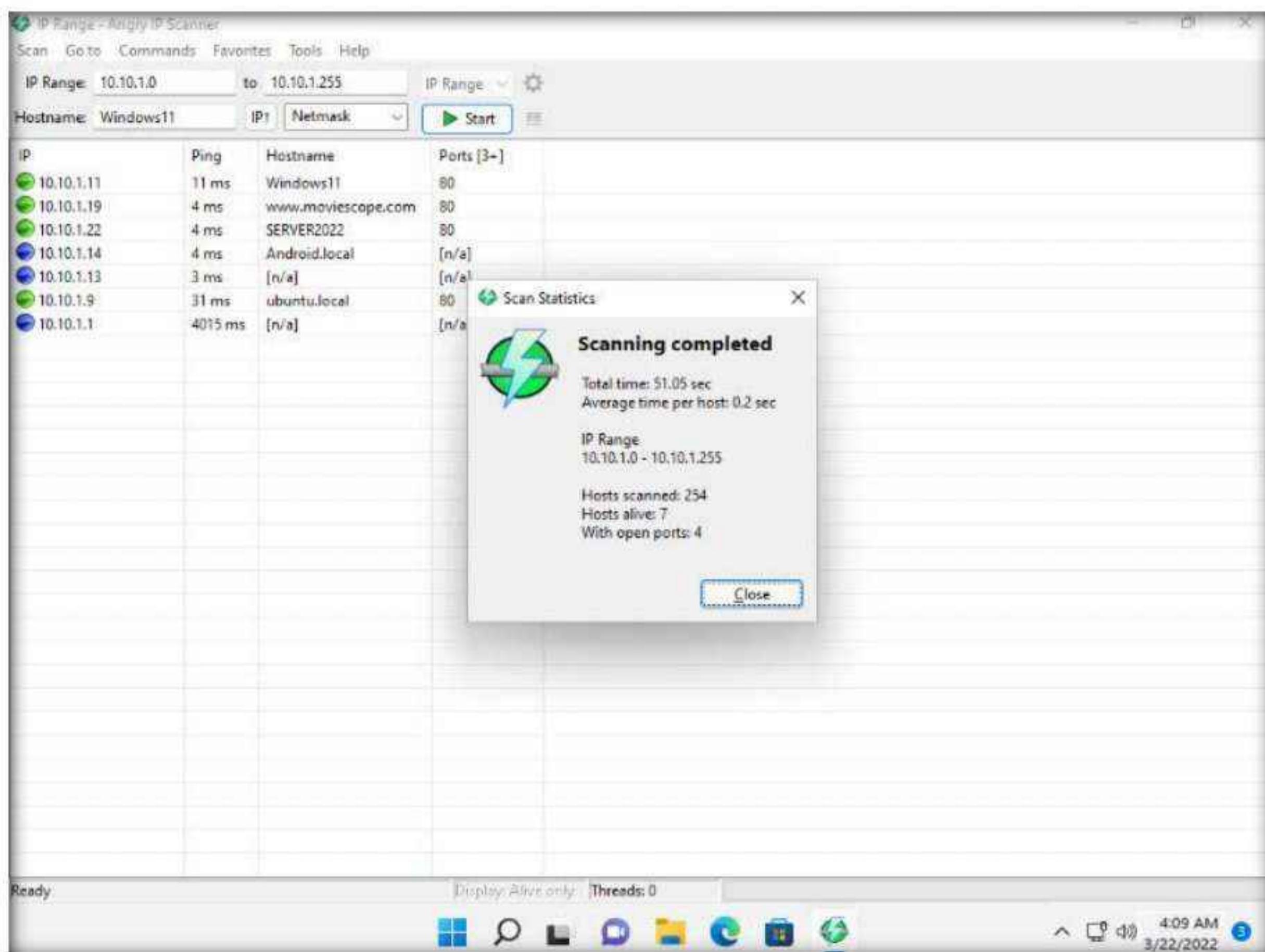


Module 03 – Scanning Networks

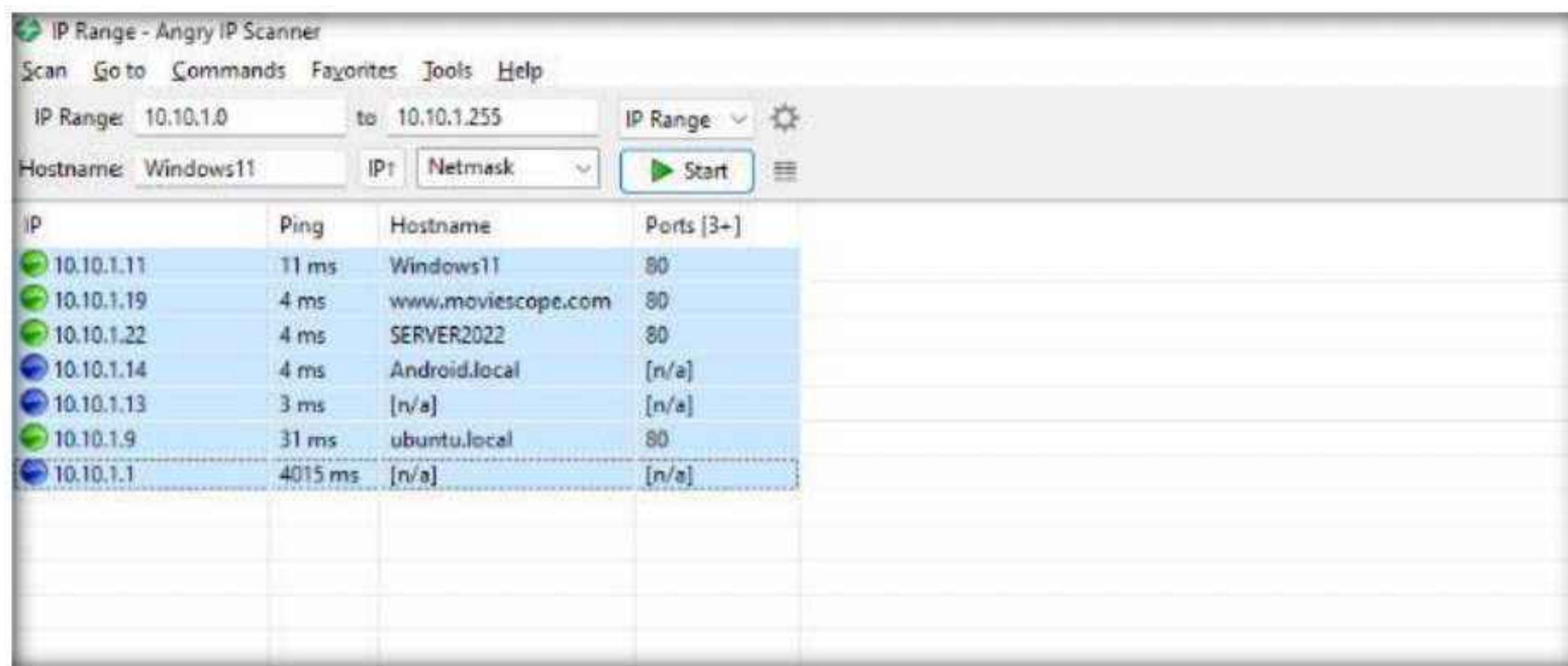
8. In the **IP Range - Angry IP Scanner** window, click the **Start** button to start scanning the IP range that you entered.



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.



- | IP | Ping | Hostname | Ports [3+] |
|------------|---------|--------------------|------------|
| 10.10.1.11 | 11 ms | Windows11 | 80 |
| 10.10.1.19 | 4 ms | www.moviescope.com | 80 |
| 10.10.1.22 | 4 ms | SERVER2022 | 80 |
| 10.10.1.14 | 4 ms | Android.local | [n/a] |
| 10.10.1.13 | 3 ms | [n/a] | [n/a] |
| 10.10.1.9 | 31 ms | ubuntu.local | 80 |
| 10.10.1.1 | 4015 ms | [n/a] | [n/a] |
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.
15. Turn off all the virtual machines (**Windows 11, Windows Server 2022, Windows Server 2019, Parrot Security, Ubuntu, and Android**).

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> CyberQ

Lab**2**

Perform Port and Service Discovery

Port and service discovery is the process of identifying open ports and services running on the target IP addresses/active hosts.

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

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Android virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 45 Minutes

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

Lab Tasks

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. Before beginning this task, turn on the **Windows 11, Windows Server 2022, Windows Server 2019, Ubuntu, Parrot Security, and Android** virtual machines.
2. Switch to the **Windows 11** virtual machine. By default, **Admin** user profile is selected, type **Pa\$\$w0rd** in the **Password** field and press **Enter** to login.

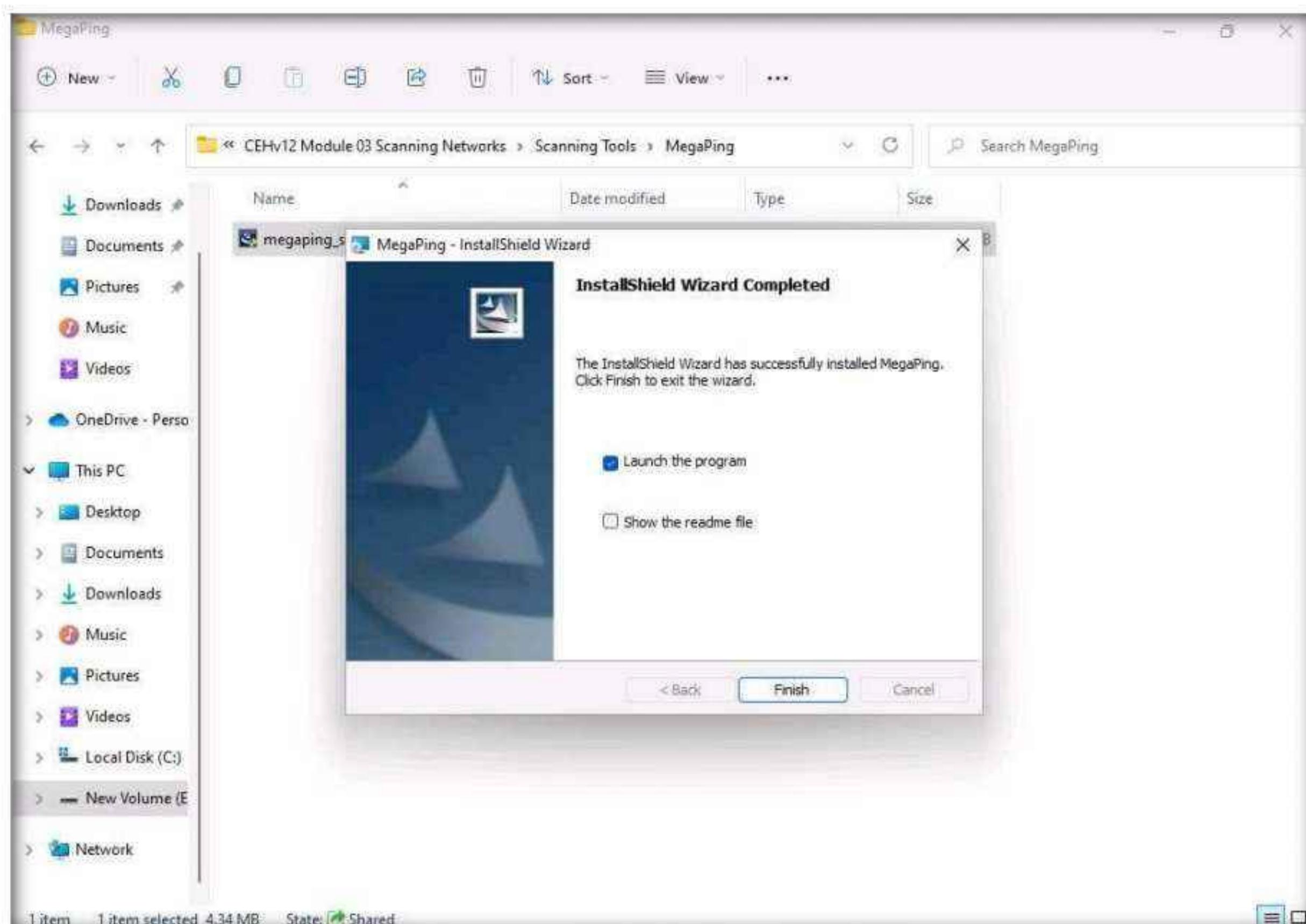
Note: If **Welcome to Windows** wizard appears, click **Continue** and in **Sign in with Microsoft** wizard, click **Cancel**.

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

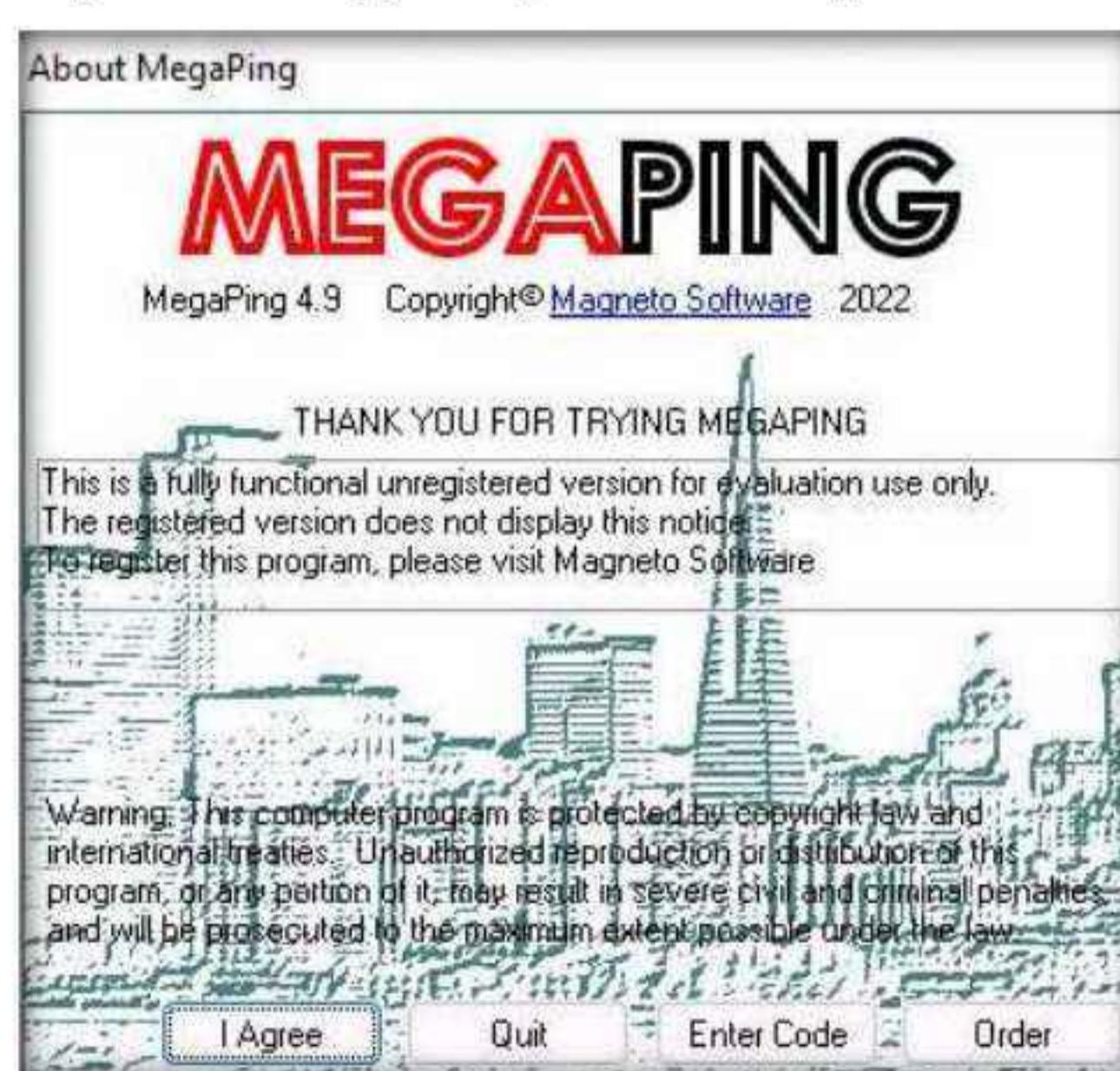
3. Navigate to **E:\CEH-Tools\CEHv12 Module 03 Scanning Networks\Scanning Tools\MegaPing** and double-click **megaping_setup.exe**.

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

4. The **MegaPing - InstallShield Wizard** window appears; click **Next** and follow the wizard-driven installation steps to install **MegaPing**.
5. After the completion of the installation, click on the **Launch the program** checkbox and click **Finish**.

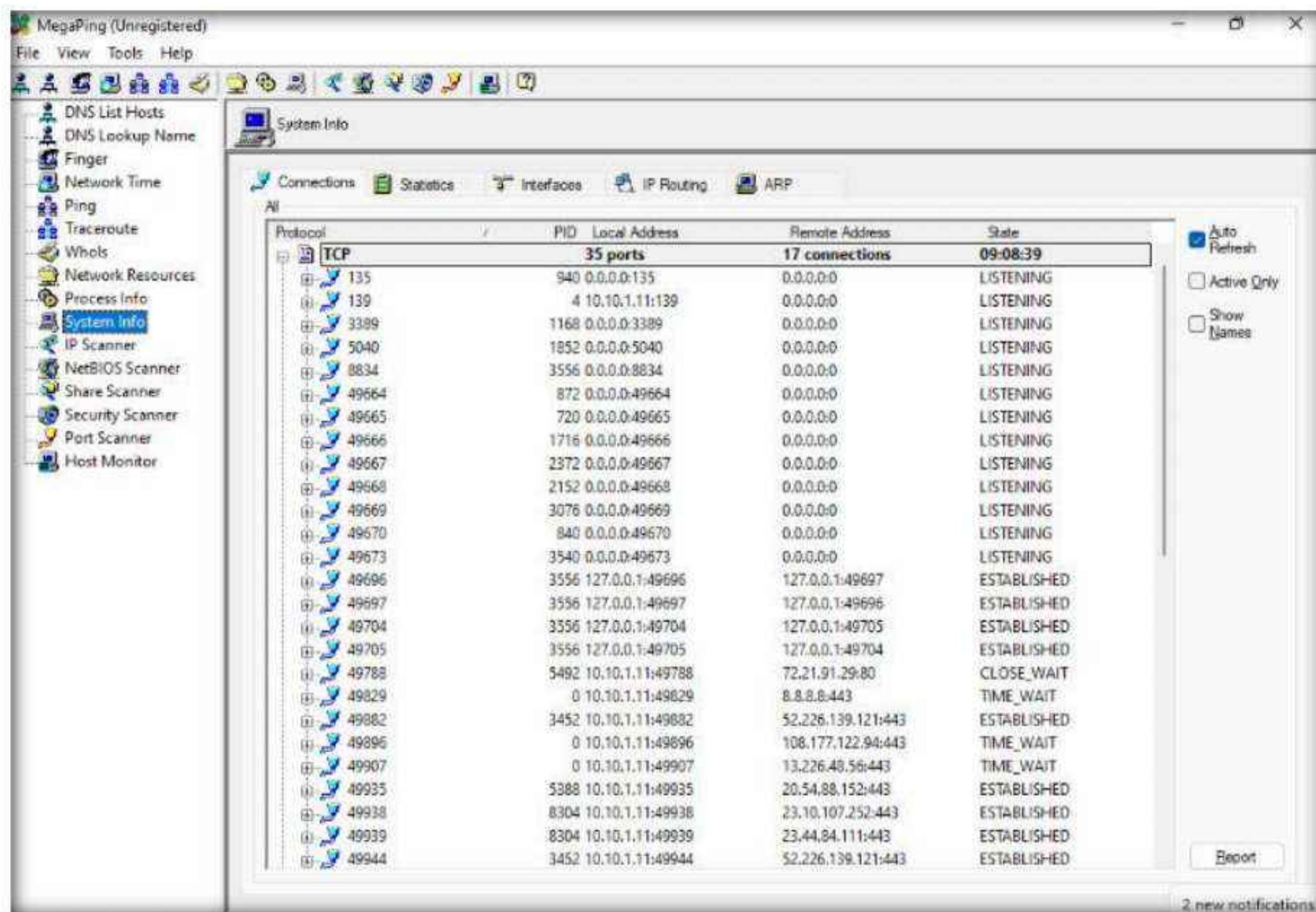


6. The **About MegaPing** window appears; click the **I Agree** button.

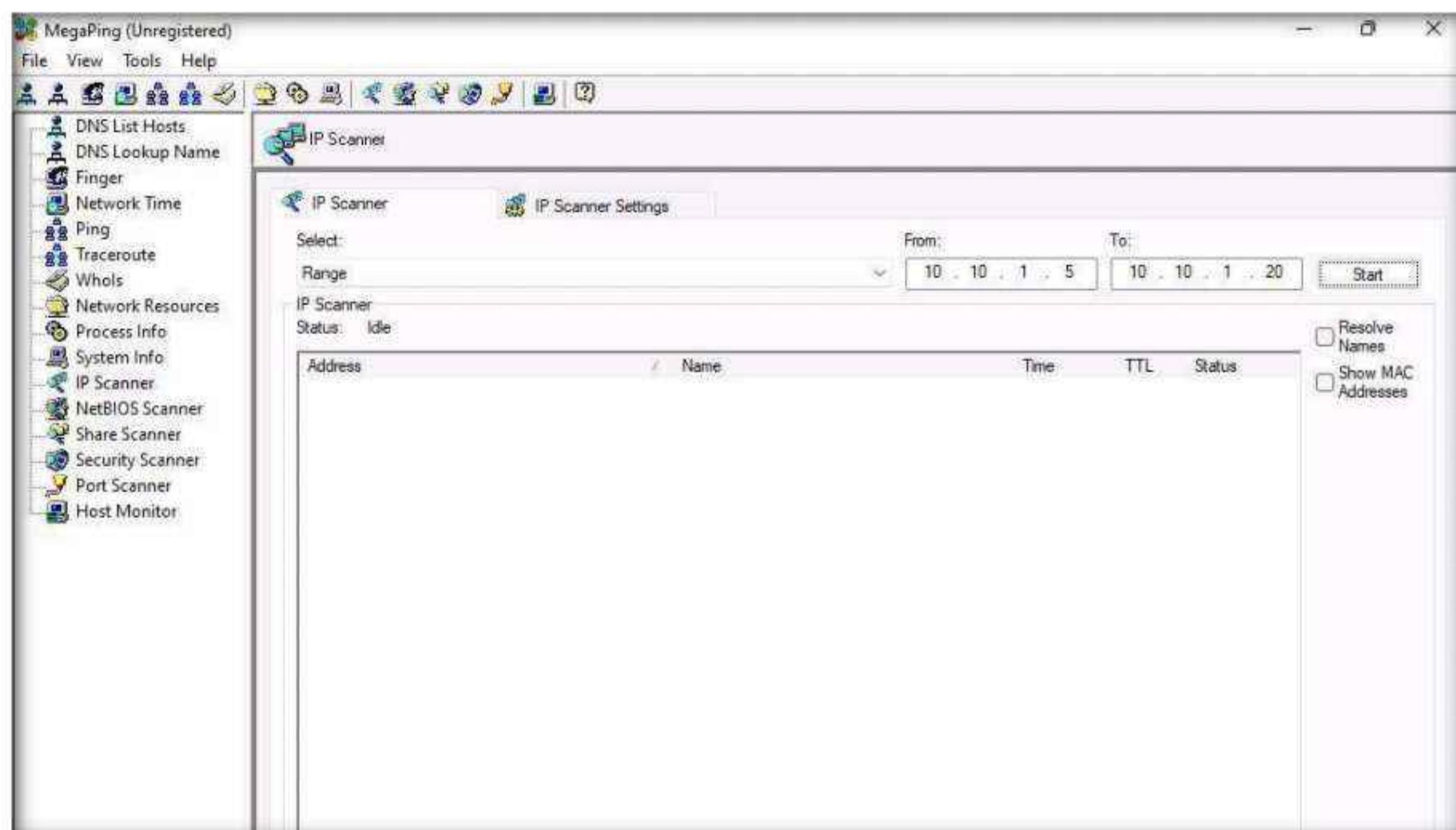


Module 03 – Scanning Networks

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

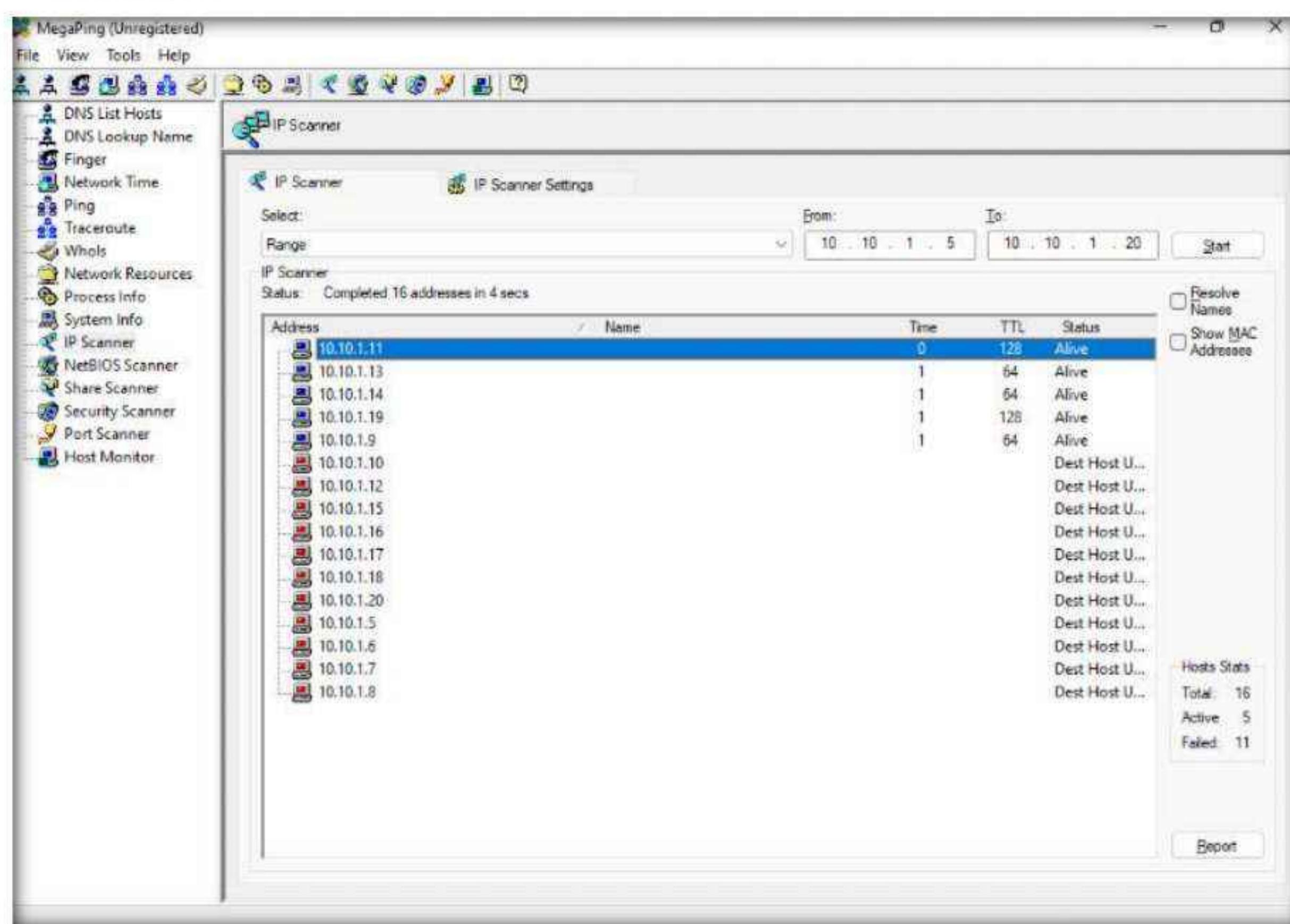


8. Select the **IP Scanner** option from the left pane. In the **IP Scanner** tab in the right-hand pane, enter the IP range in the **From** and **To** fields; in this lab, the IP range is **10.10.1.5** to **10.10.1.20**; then, click **Start**.

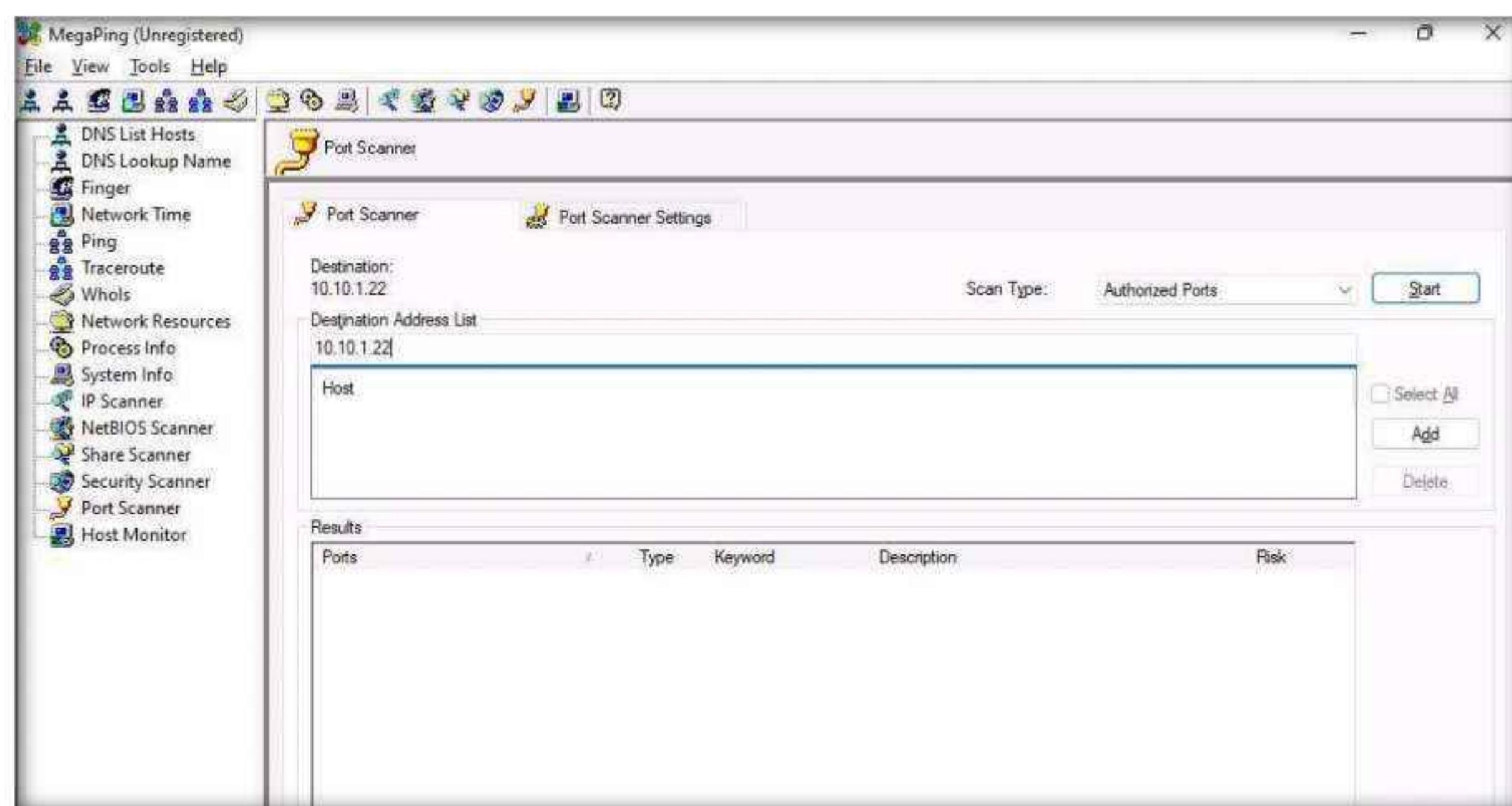


Module 03 – Scanning Networks

9. 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.

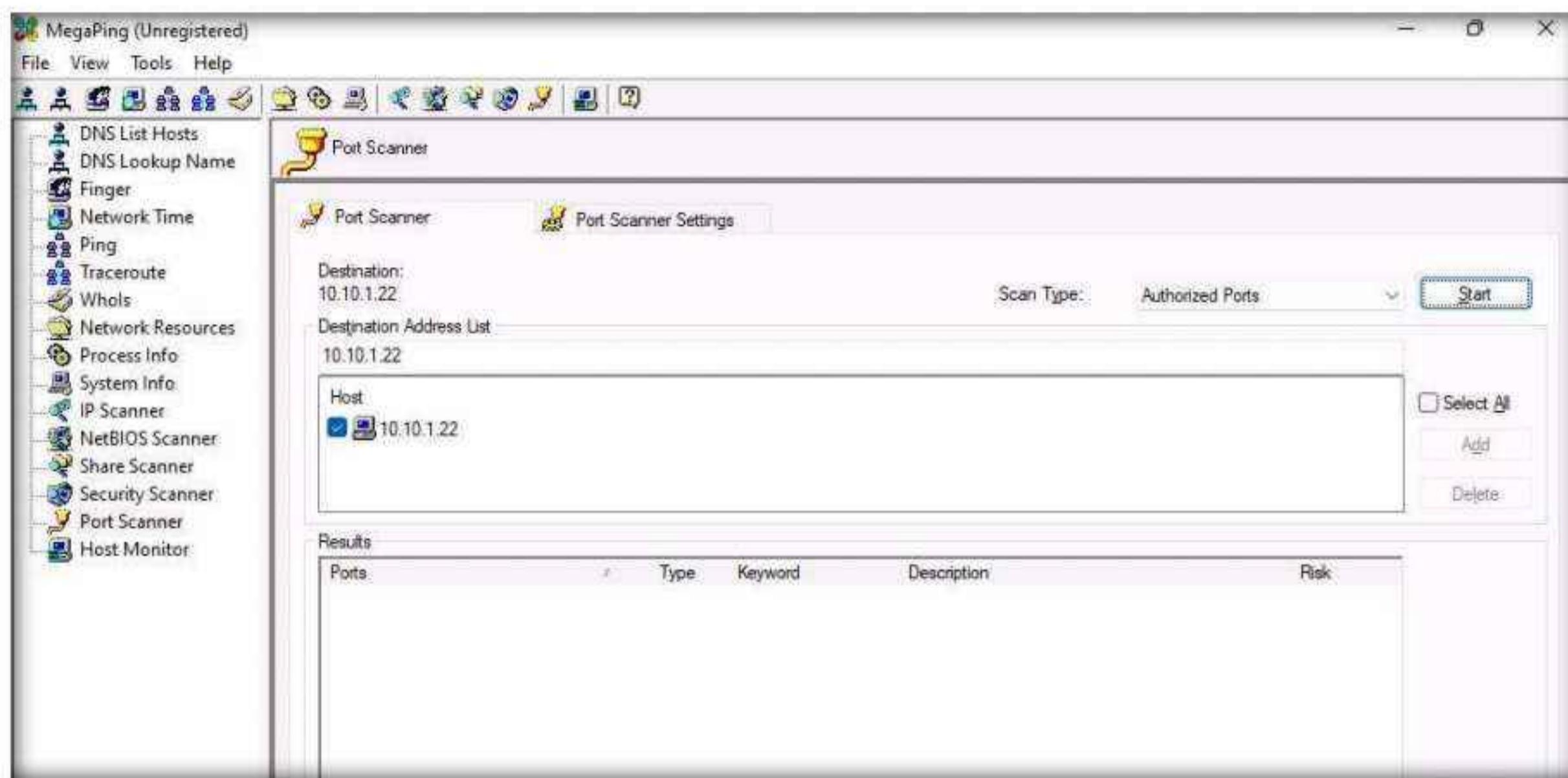


10. Select the **Port Scanner** option from the left-hand pane. In the **Port Scanner** tab in the right-hand pane, enter the IP address of the **Windows Server 2022 (10.10.1.22)** machine into the **Destination Address List** field and click **Add**.

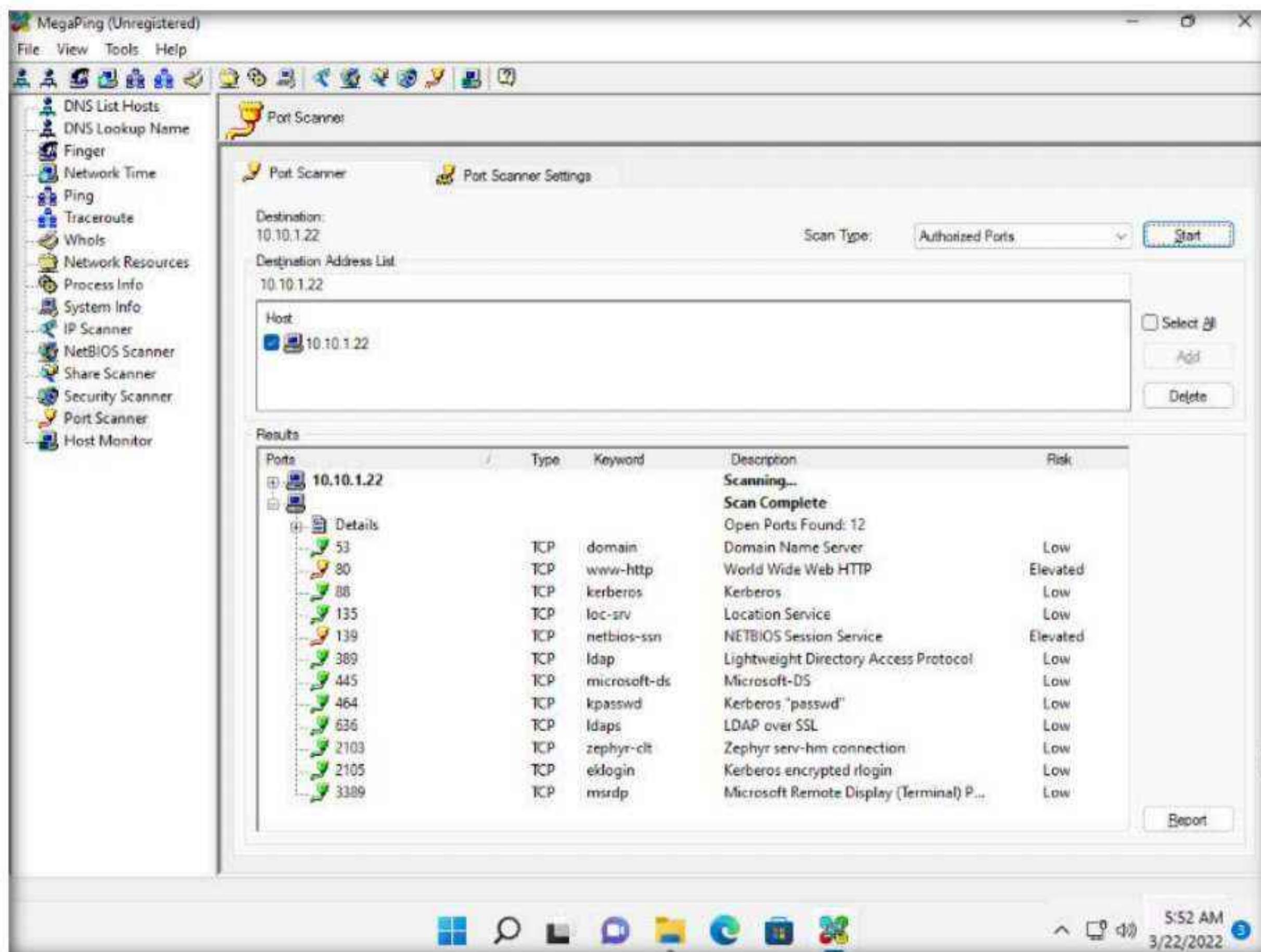


Module 03 – Scanning Networks

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



12. 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.



13. Similarly, you can perform port and service scanning on other target machines.

14. This concludes the demonstration of discovering open ports and services running on the target IP address using MegaPing.
15. 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.

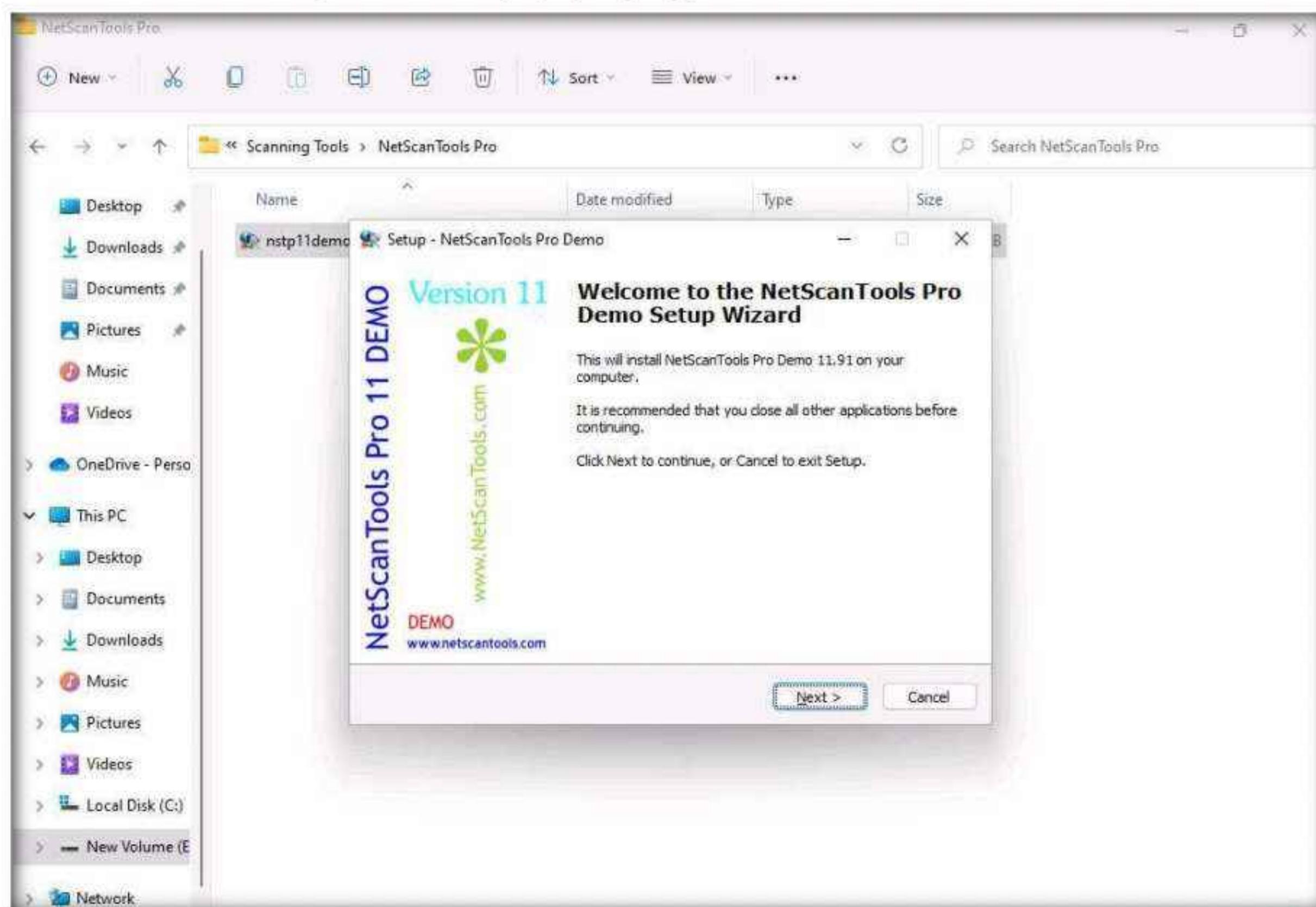
Here, we will use the NetScanTools Pro tool to discover open ports and services running on the target range of IP addresses.

1. Ensure that the virtual machines (**Windows Server 2022**, **Windows Server 2019**, **Ubuntu**, **Parrot Security**, and **Android**) are running.
2. 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**.

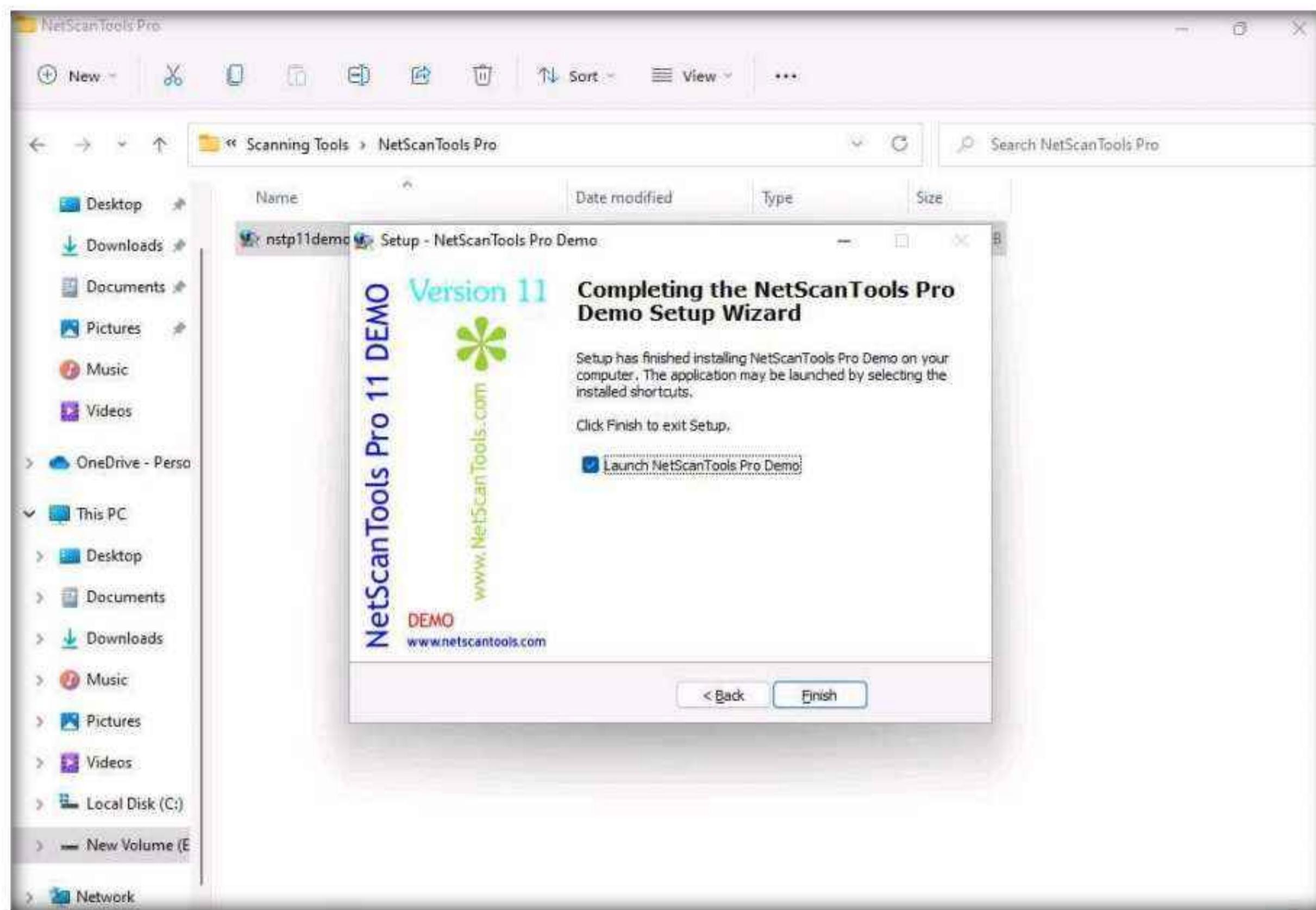
3. The **Setup - NetScanTools Pro Demo** window appears, click **Next** and follow the wizard-driven installation steps to install **NetScanTools Pro**.

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

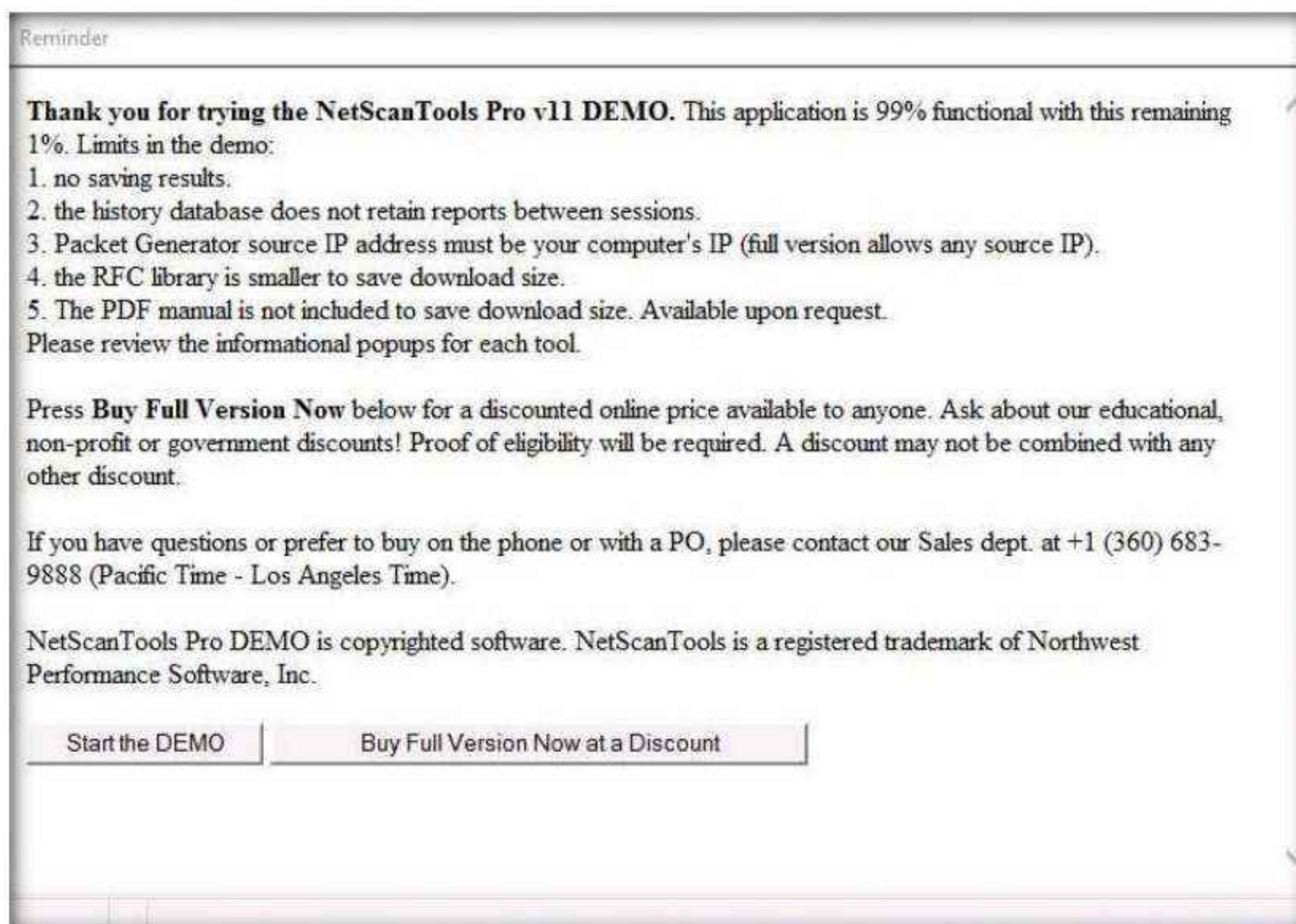


Module 03 – Scanning Networks

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

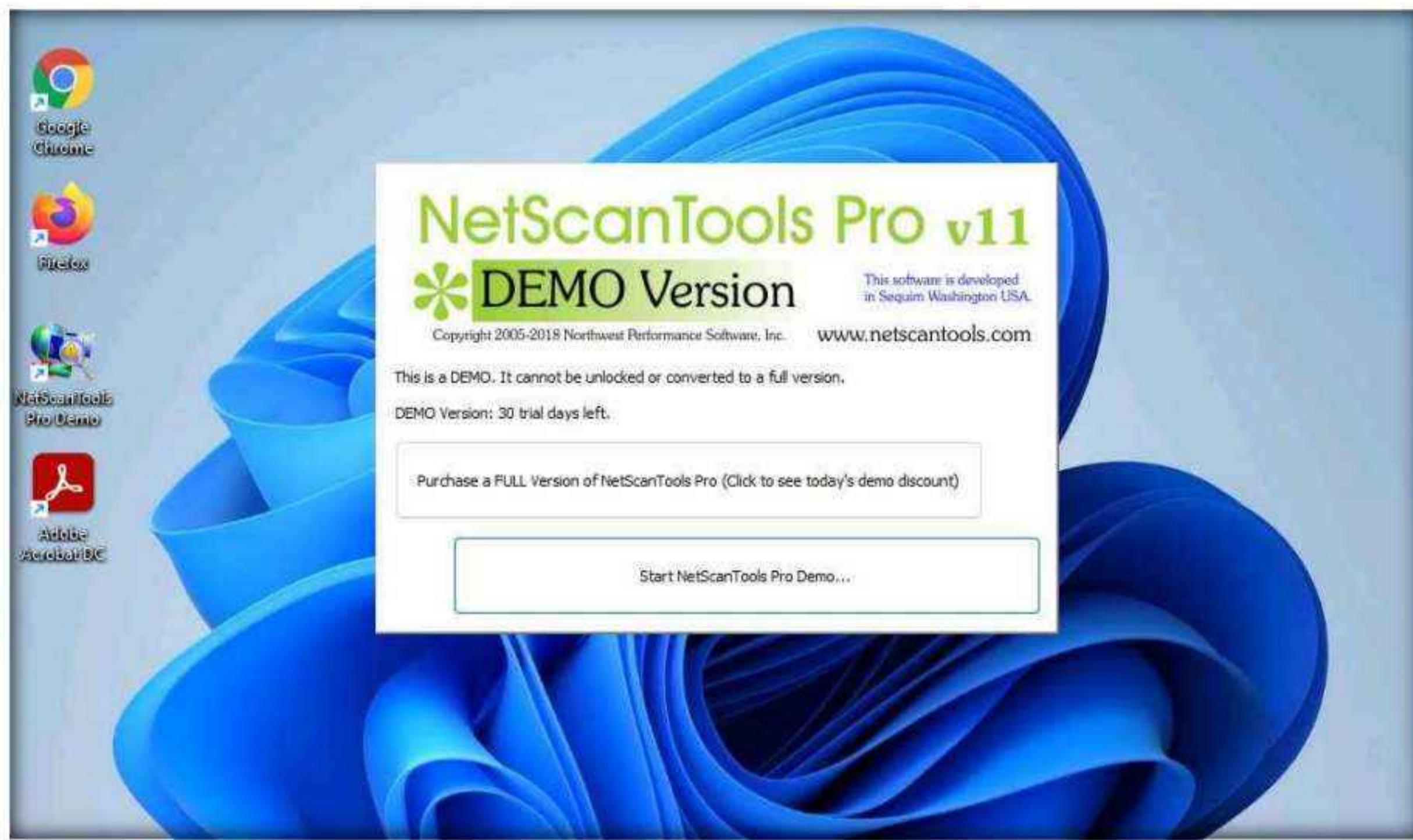


5. The **Reminder** window appears; if you are using a demo version of NetScanTools Pro, click the **Start the DEMO** button.



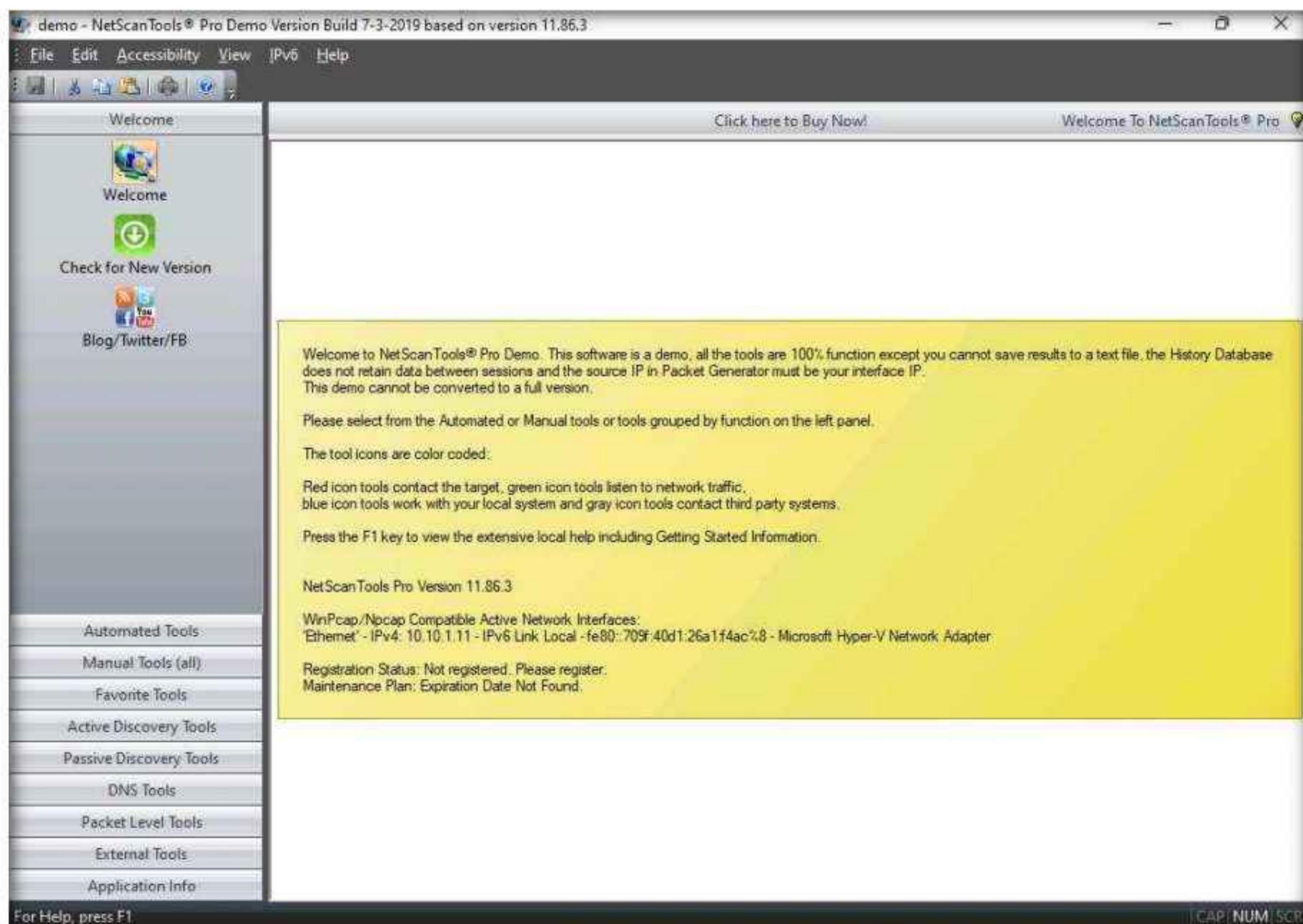
Module 03 – Scanning Networks

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



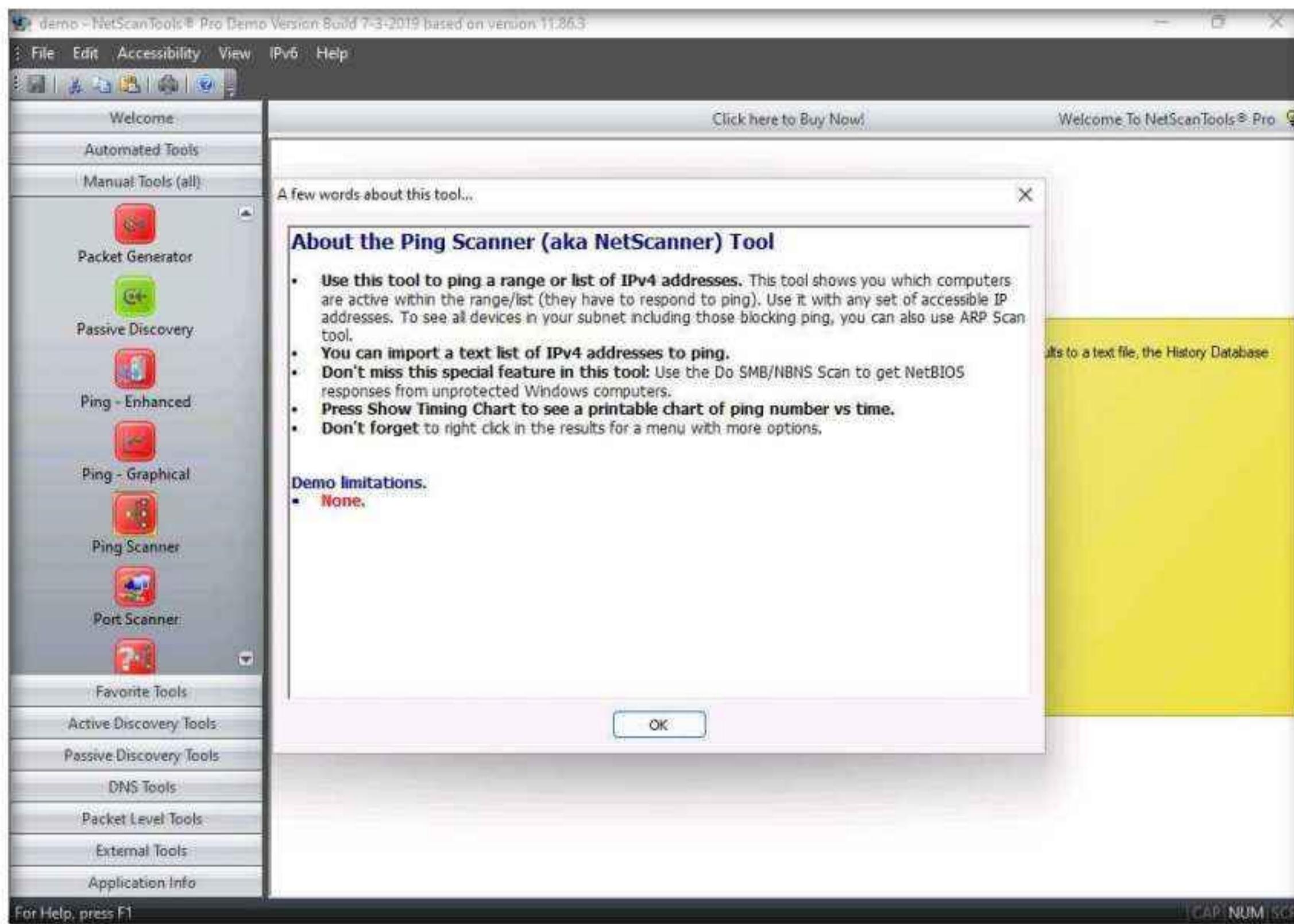
7. The **NetScanTools Pro** main window appears, as shown in the screenshot.

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



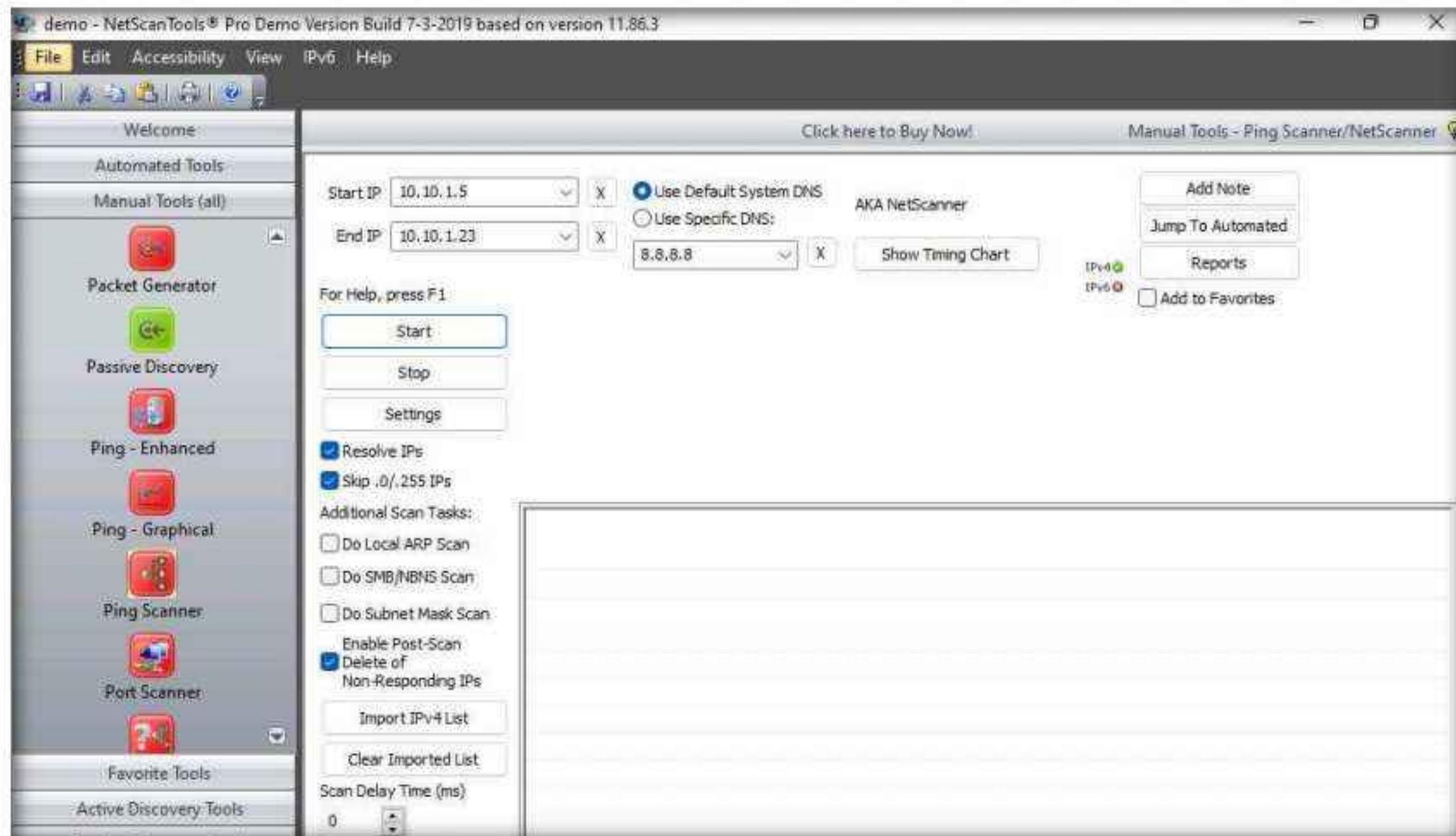
Module 03 – Scanning Networks

8. In the left-hand pane, under the **Manual Tools (all)** section, scroll down and click the **Ping Scanner** option, as shown in the screenshot.
9. A dialog box opens explaining the **Ping Scanner** tool; click **OK**.

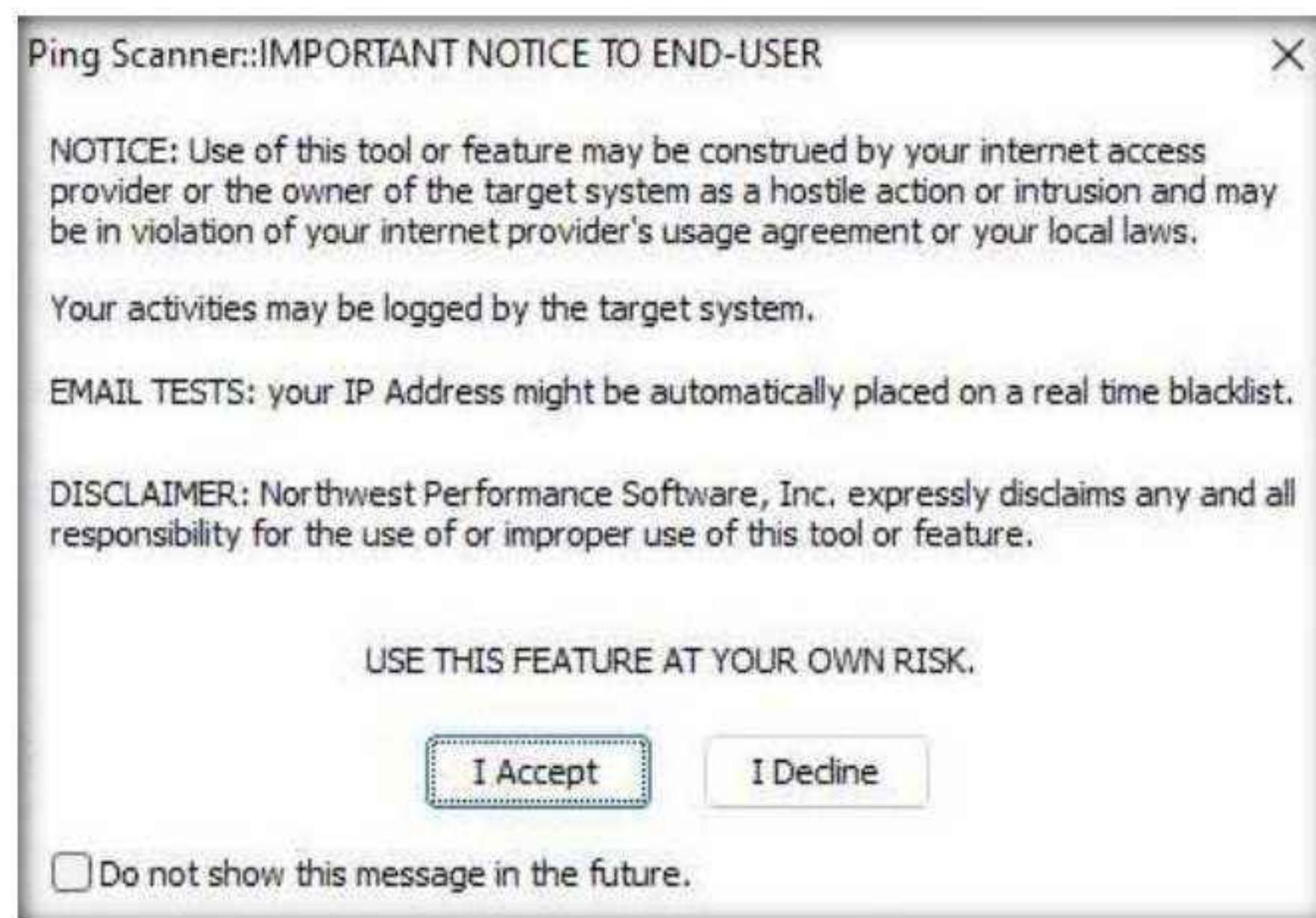


10. Ensure that **Use Default System DNS** is selected. Enter the range of IP addresses into the **Start IP** and **End IP** fields (here, **10.10.1.5 - 10.10.1.23**); then, click **Start**.

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



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



12. 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**.

A screenshot of a Google Chrome browser window displaying a scan report. The title bar says "NetScanTools® Pro Report". The main content area shows the following information:

NetScanTools® Pro v11
Reports Created with DEMO v11.11
Buy from: www.netscantools.com

Report created with NetScanTools Pro v11 DEMO.
[Purchase NetScanTools Pro at www.netscantools.com](http://www.netscantools.com).

Statistics for Ping Scanner

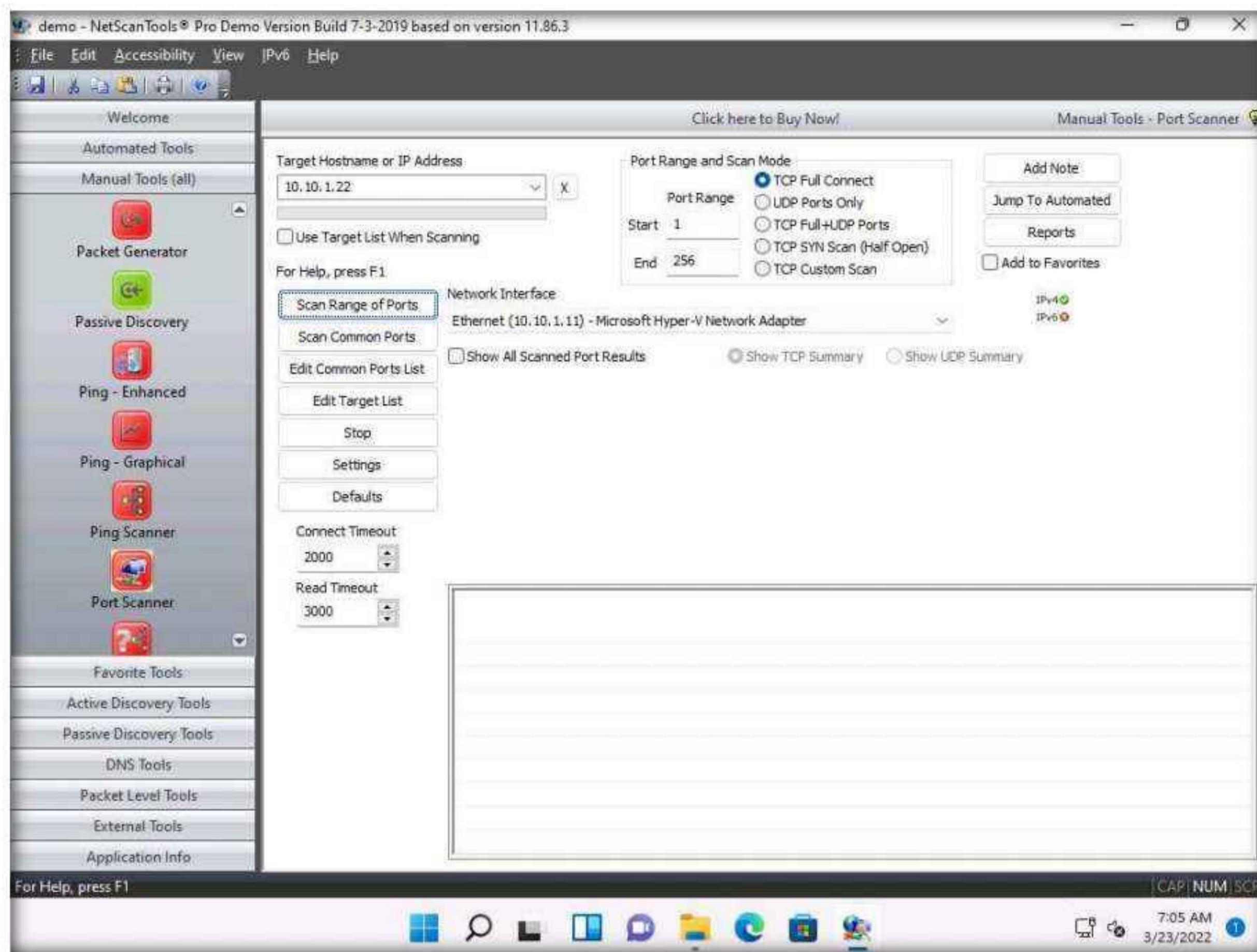
Report Timestamp	Wednesday, March 23, 2022 07:02:40
Scan Start Timestamp	Wednesday, March 23, 2022 07:02:33
Total Scan Time	5.641 seconds
Start IP address	10.10.1.5
End IP address	10.10.1.23
Number of target IP addresses	19
Number of IP addresses responding to pings	6
Number of intermediate routers responding to pings	0
Number of successful NetBIOS queries	0
Number of MAC addresses obtained by ARP or NetBIOS queries	0
Number of successful Subnet Mask queries	0

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

14. 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**.

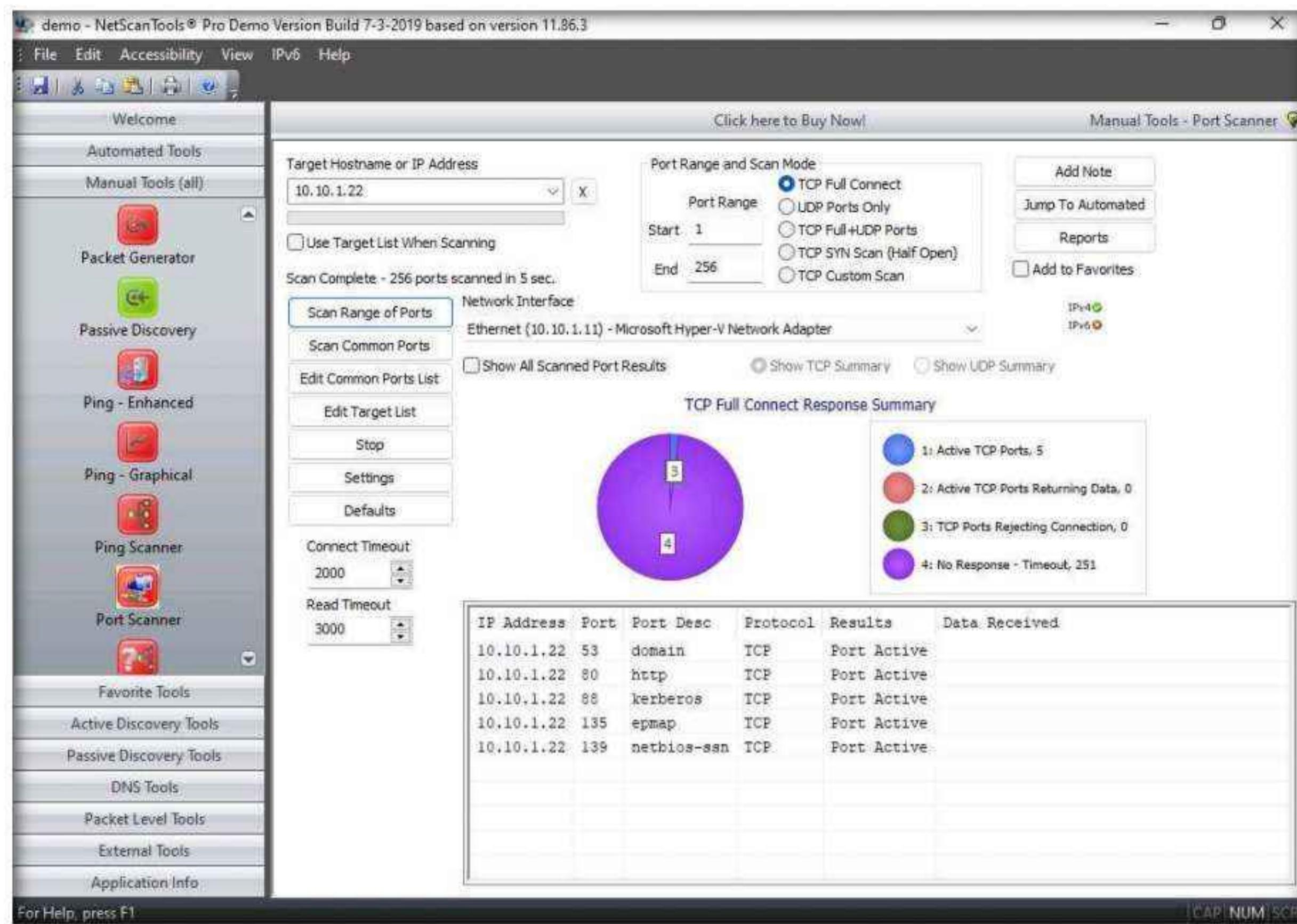
15. In the **Target Hostname or IP Address** field, enter the IP address of the target (here, **10.10.1.22**). Ensure that **TCP Full Connect** radio button is selected, and then click the **Scan Range of Ports** button.



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



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



18. 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.
19. This concludes the demonstration of discovering open ports and services running on the target IP address using NetScanTools Pro.
20. 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. Ensure that the virtual machines (**Windows 11, Windows Server 2022, Windows Server 2019, Ubuntu, and Android**) are running.
2. Switch to the **Parrot Security** virtual machine.

- 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.

- Click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.
- A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.

- 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.

- 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.

```
[root@parrot]~[~/home/attacker]
└─#sx arp 10.10.1.0/24
10.10.1.9      8e:99:65:2c:99:18
10.10.1.19     32:84:ba:4d:18:47
10.10.1.22     ac:f2:ac:87:8d:60
10.10.1.11     36:92:e8:06:ea:27
10.10.1.1      f8:3b:43:f3:b9:24
10.10.1.14     7a:db:e9:41:c4:ad
[root@parrot]~[~/home/attacker]
└─#
```

- 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.

```
[root@parrot]~[~/home/attacker]
└─#sx arp 10.10.1.0/24 --json | tee arp.cache
{"ip":"10.10.1.14","mac":"7a:db:e9:41:c4:ad","vendor":""}
{"ip":"10.10.1.11","mac":"36:92:e8:06:ea:27","vendor":""}
{"ip":"10.10.1.9","mac":"8e:99:65:2c:99:18","vendor":""}
{"ip":"10.10.1.1","mac":"f8:3b:43:f3:b9:24","vendor":""}
{"ip":"10.10.1.22","mac":"ac:f2:ac:87:8d:60","vendor":""}
{"ip":"10.10.1.19","mac":"32:84:ba:4d:18:47","vendor":""}
[root@parrot]~[~/home/attacker]
└─#
```

9. 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**).

```
[root@parrot]~[~/home/attacker]
└─#sx arp 10.10.1.0/24 --json | tee arp.cache
{"ip":"10.10.1.14","mac":"7a:db:e9:41:c4:ad","vendor":""}
{"ip":"10.10.1.11","mac":"36:92:e8:06:ea:27","vendor":""}
{"ip":"10.10.1.9","mac":"8e:99:65:2c:99:18","vendor":""}
{"ip":"10.10.1.1","mac":"f8:3b:43:f3:b9:24","vendor":""}
{"ip":"10.10.1.22","mac":"ac:f2:ac:87:8d:60","vendor":""}
{"ip":"10.10.1.19","mac":"32:84:ba:4d:18:47","vendor":""}
[root@parrot]~[~/home/attacker]
└─#cat arp.cache | sx tcp -p 1-65535 10.10.1.11
10.10.1.11      135
10.10.1.11      49664
10.10.1.11      49673
10.10.1.11      21
10.10.1.11      5040
10.10.1.11      7680
10.10.1.11      49666
10.10.1.11      49668
10.10.1.11      80
10.10.1.11      445
10.10.1.11      49667
10.10.1.11      8834
10.10.1.11      49665
10.10.1.11      3389
10.10.1.11      49670
10.10.1.11      139
10.10.1.11      49669
[root@parrot]~[~/home/attacker]
└─#
```

10. 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.

```

Applications Places System sx help - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[~/home/attacker]
#sx help
Fast, modern, easy-to-use network scanner

Usage:
sx [command]

Available Commands:
arp      Perform ARP scan
docker   Perform Docker scan
elastic  Perform Elasticsearch scan
help     Help about any command
icmp    Perform ICMP scan
socks   Perform SOCKS5 scan
tcp     Perform TCP scan
udp     Perform UDP scan

Flags:
-h, --help  help for sx

Use "sx [command] --help" for more information about a command.

```

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

12. 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.

13. 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.

```

Applications Places System sx udp --json -p 53 10.10.1.11 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[~/home/attacker]
#cat arp.cache | sx udp --json -p 53 10.10.1.11
{"scan":"udp","ip":"10.10.1.11","ttl":128,"icmp":{"type":3,"code":3}}
[root@parrot]~[~/home/attacker]
#
```

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

```
[root@parrot]~[~/home/attacker]
└─# cat arp.cache | sx udp --json -p 500 10.10.1.11
{"scan":"udp","ip":"10.10.1.11","ttl":128,"icmp":{"type":3,"code":3}}
[root@parrot]~[~/home/attacker]
└─# cat arp.cache | sx udp --json -p 500 10.10.1.11
[root@parrot]~[~/home/attacker]
└─#
```

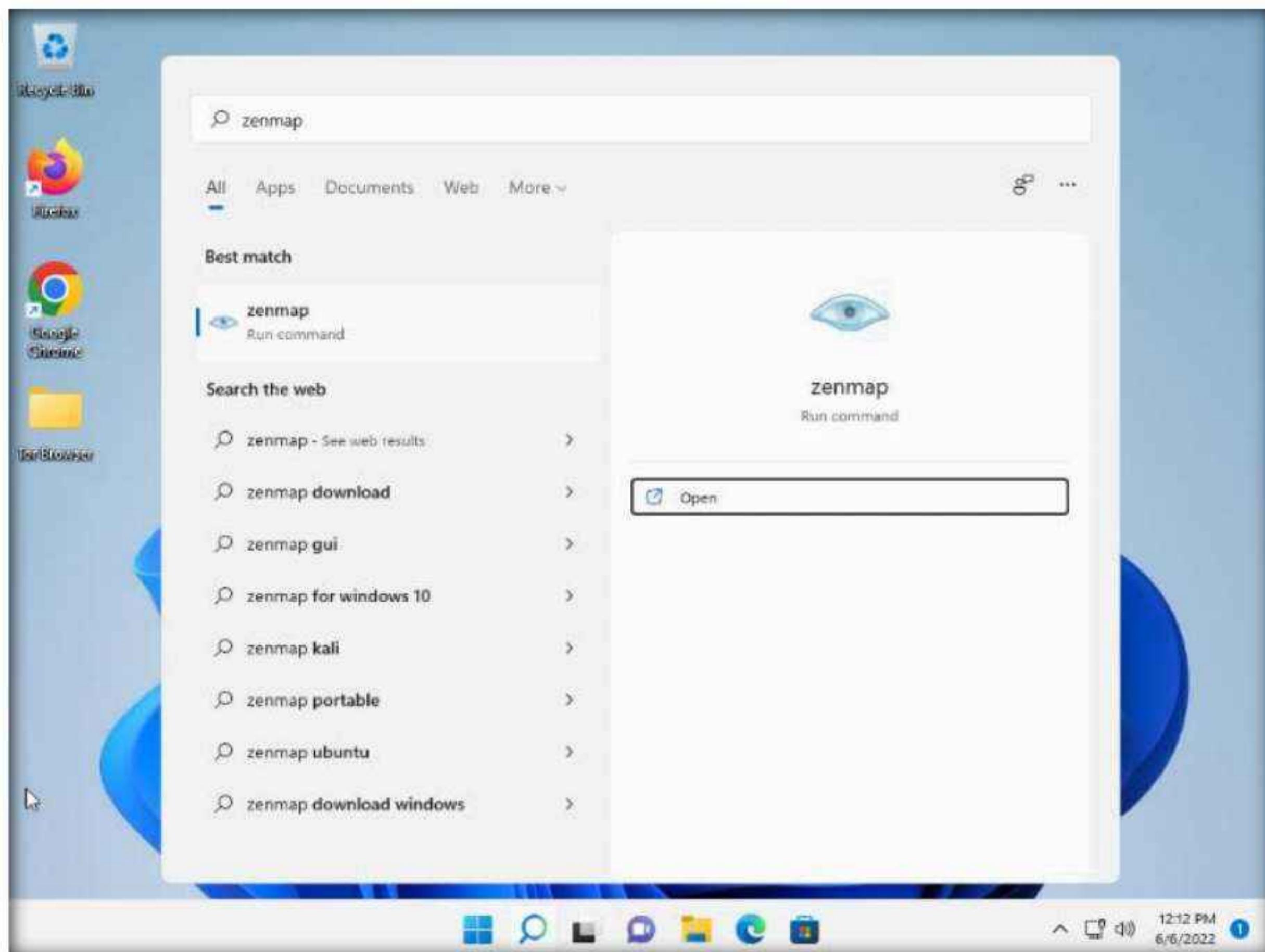
15. You can observe that sx does not return any code in the above command, which states that the target port is open.
16. This concludes the demonstration of port scanning using sx Tool.
17. Close all open windows and document all acquired information.
18. Turn off the virtual machines (**Windows Server 2019**, **Android**, and **Parrot Security**).

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.

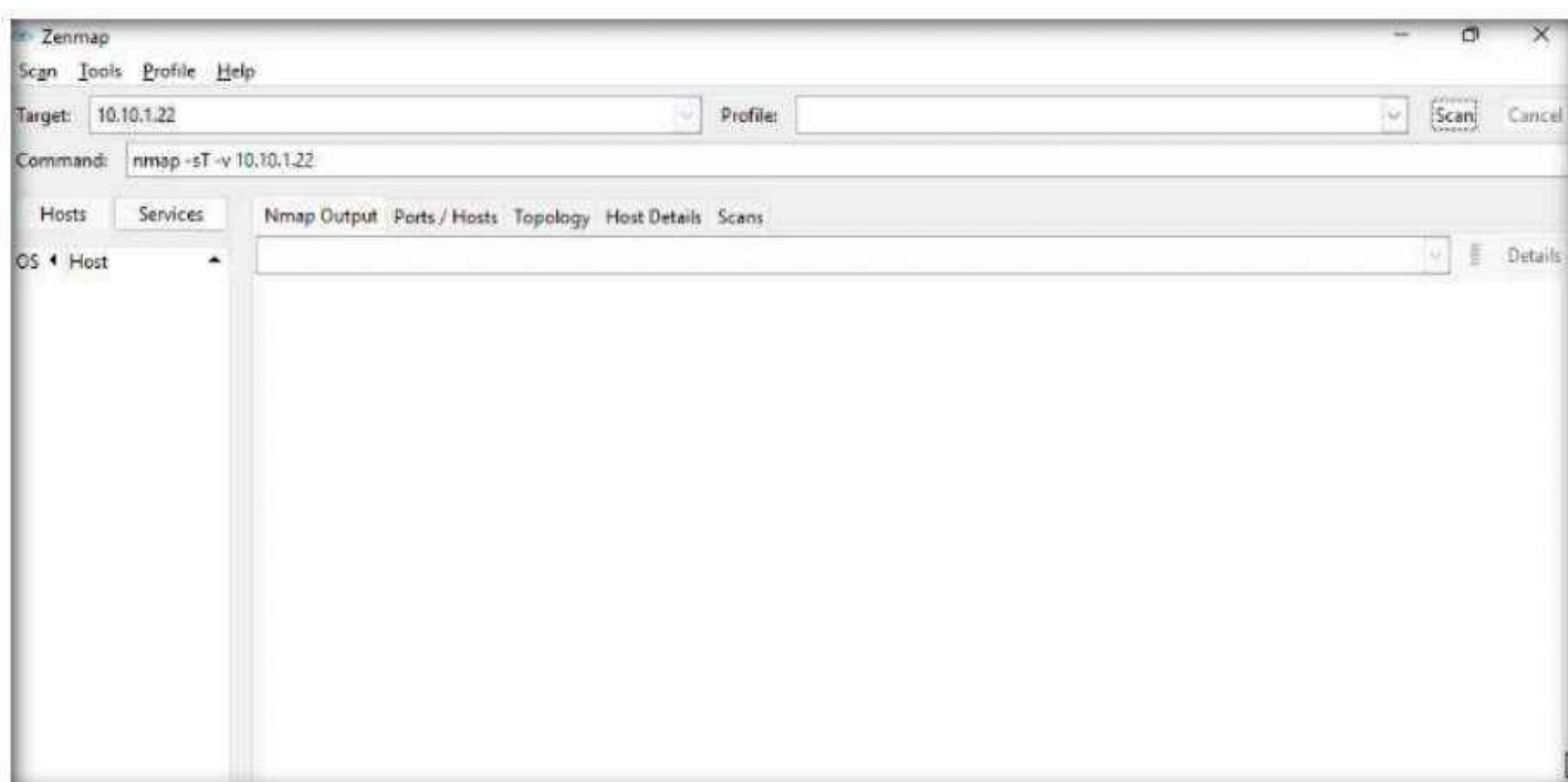
1. Ensure that the **Windows 11**, **Ubuntu** and **Windows Server 2022** virtual machines are running.
2. Switch to the **Windows 11** virtual machine. In the **Windows 11** machine, click **Search** icon (🔍) on the **Desktop**. Type **zenmap** in the search field, the **Zenmap** appears in the results, click **Open** to launch it.



3. 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.



Module 03 – Scanning Networks

- 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.

Zenmap

Scan Tools Profile Help

Target: 10.10.1.22

Command: nmap -sT -v 10.10.1.22

Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans

OS Host 10.10.1.22

nmap -sT -v 10.10.1.22

Discovered open port 3268/tcp on 10.10.1.22

Discovered open port 389/tcp on 10.10.1.22

Connect Scan Timing: About 44.30% done; ETC: 12:15 (0:00:39 remaining)

Discovered open port 636/tcp on 10.10.1.22

Discovered open port 593/tcp on 10.10.1.22

Discovered open port 3289/tcp on 10.10.1.22

Discovered open port 2105/tcp on 10.10.1.22

Discovered open port 2107/tcp on 10.10.1.22

Discovered open port 1801/tcp on 10.10.1.22

Discovered open port 88/tcp on 10.10.1.22

Completed Connect Scan at 12:15, 66.92s elapsed (1000 total ports)

Nmap scan report for 10.10.1.22

Host is up (0.00090s latency).

Not shown: 983 filtered tcp ports (no-response)

PORT	STATE	SERVICE
53/tcp	open	domain
80/tcp	open	http
88/tcp	open	kerberos-sec
135/tcp	open	msrpc
139/tcp	open	netbios-ssn
389/tcp	open	ldap
445/tcp	open	microsoft-ds
464/tcp	open	kpasswd5
593/tcp	open	http-rpc-epmap
636/tcp	open	ldaps
1801/tcp	open	msmq
2103/tcp	open	zephyr-clt
2105/tcp	open	eklogin
2107/tcp	open	msmq-mgmt
3268/tcp	open	globalcatLDAP
3269/tcp	open	globalcatLDAPssl
3389/tcp	open	ms-wbt-server

MAC Address: 70:89:C3:9A:88:48 (Unknown)

Read data files from: C:\Program Files (x86)\Nmap

Nmap done: 1 IP address (1 host up) scanned in 67.21 seconds

Raw packets sent: 1 (288) | Rcvd: 1 (288)

Filter Hosts

- 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.

Zenmap

Scan Tools Profile Help

Target: 10.10.1.22

Command: nmap -sT -v 10.10.1.22

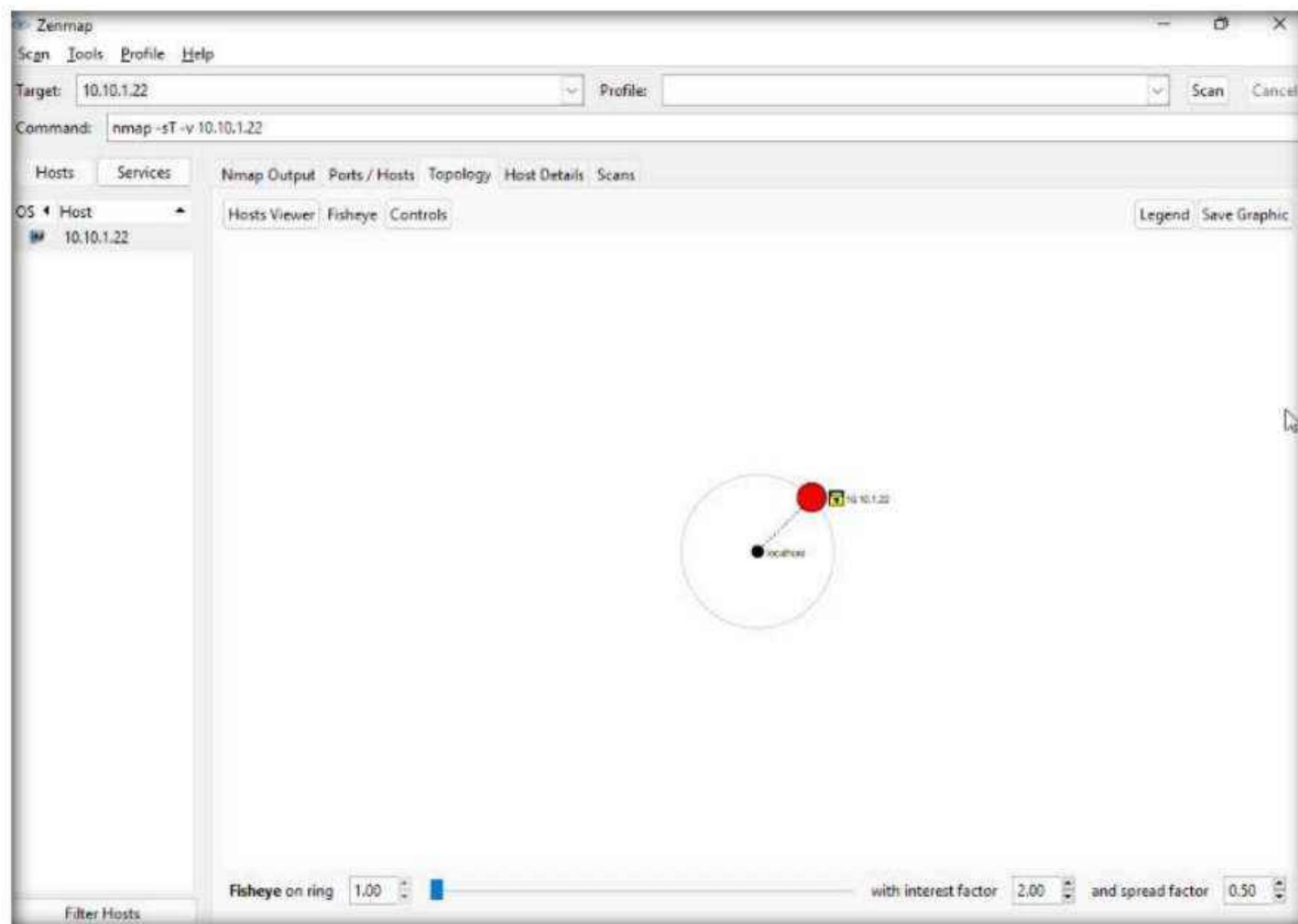
Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans

OS Host 10.10.1.22

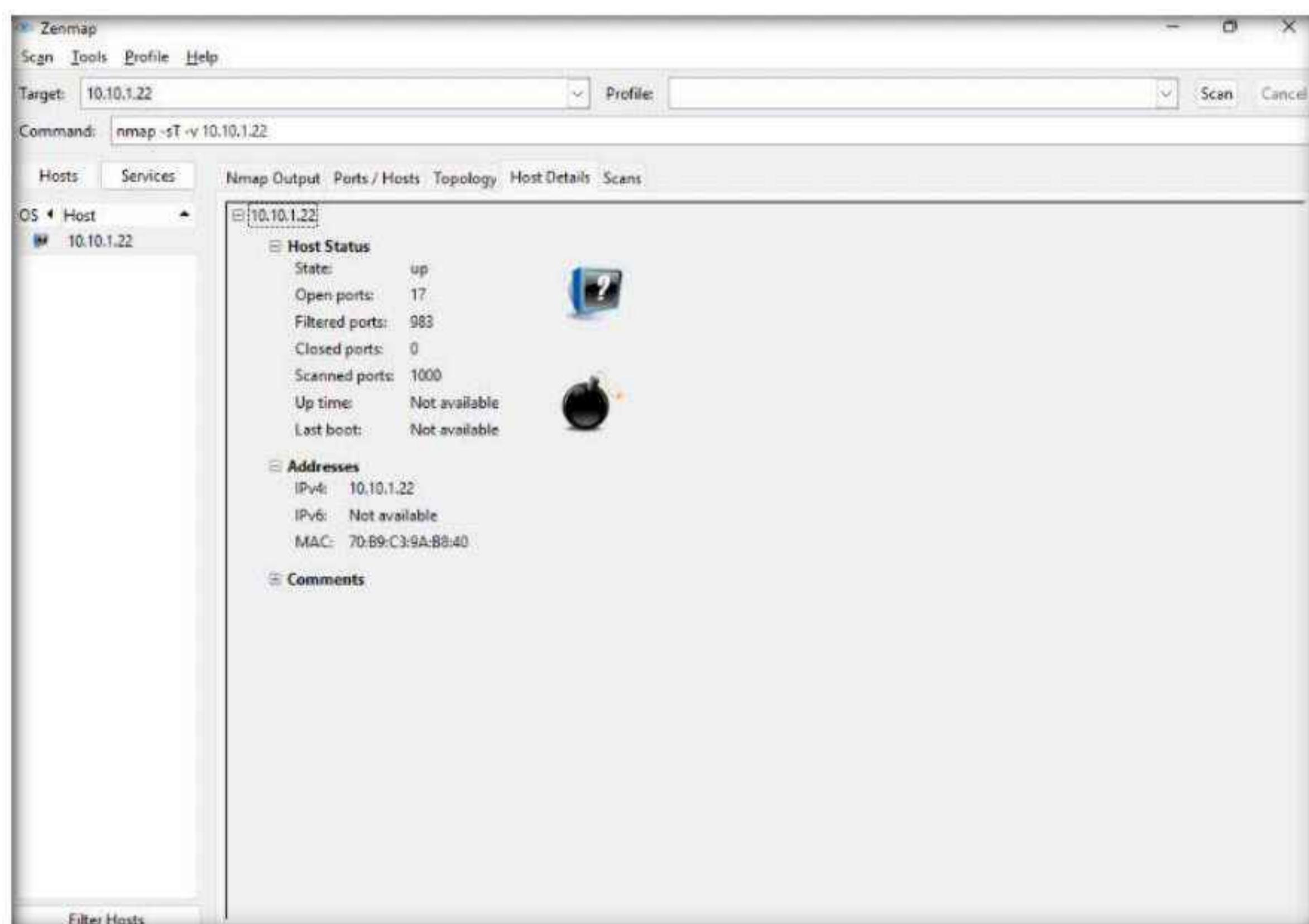
Port	Protocol	State	Service	Version
53	tcp	open	domain	
80	tcp	open	http	
88	tcp	open	kerberos-sec	
135	tcp	open	msrpc	
139	tcp	open	netbios-ssn	
389	tcp	open	ldap	
445	tcp	open	microsoft-ds	
464	tcp	open	kpasswd5	
593	tcp	open	http-rpc-epmap	
636	tcp	open	ldaps	
1801	tcp	open	msmq	
2103	tcp	open	zephyr-clt	
2105	tcp	open	eklogin	
2107	tcp	open	msmq-mgmt	
3268	tcp	open	globalcatLDAP	
3269	tcp	open	globalcatLDAPssl	
3389	tcp	open	ms-wbt-server	

Module 03 – Scanning Networks

6. 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.

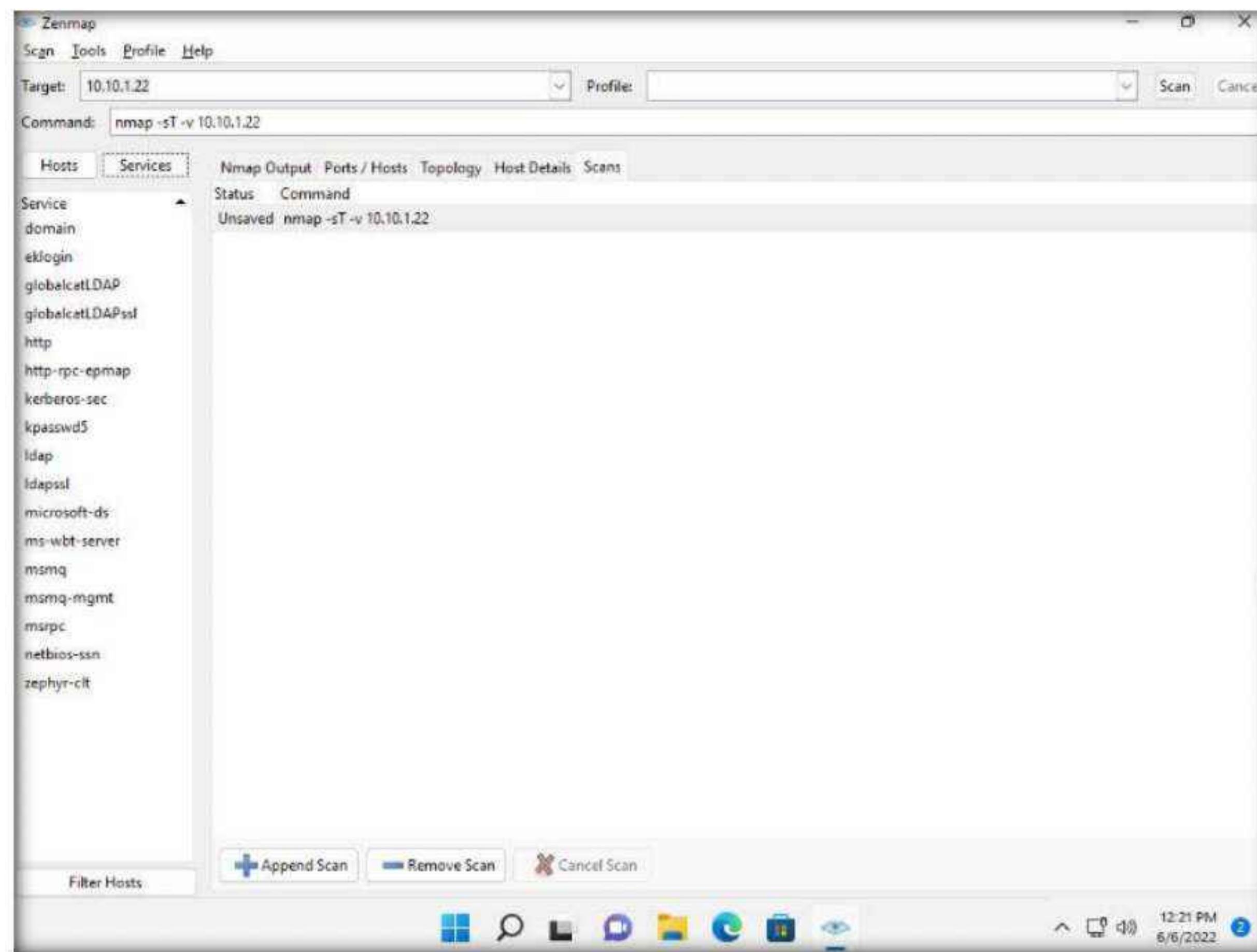


7. In the same way, click the **Host Details** tab to view the details of the TCP connect scan.



Module 03 – Scanning Networks

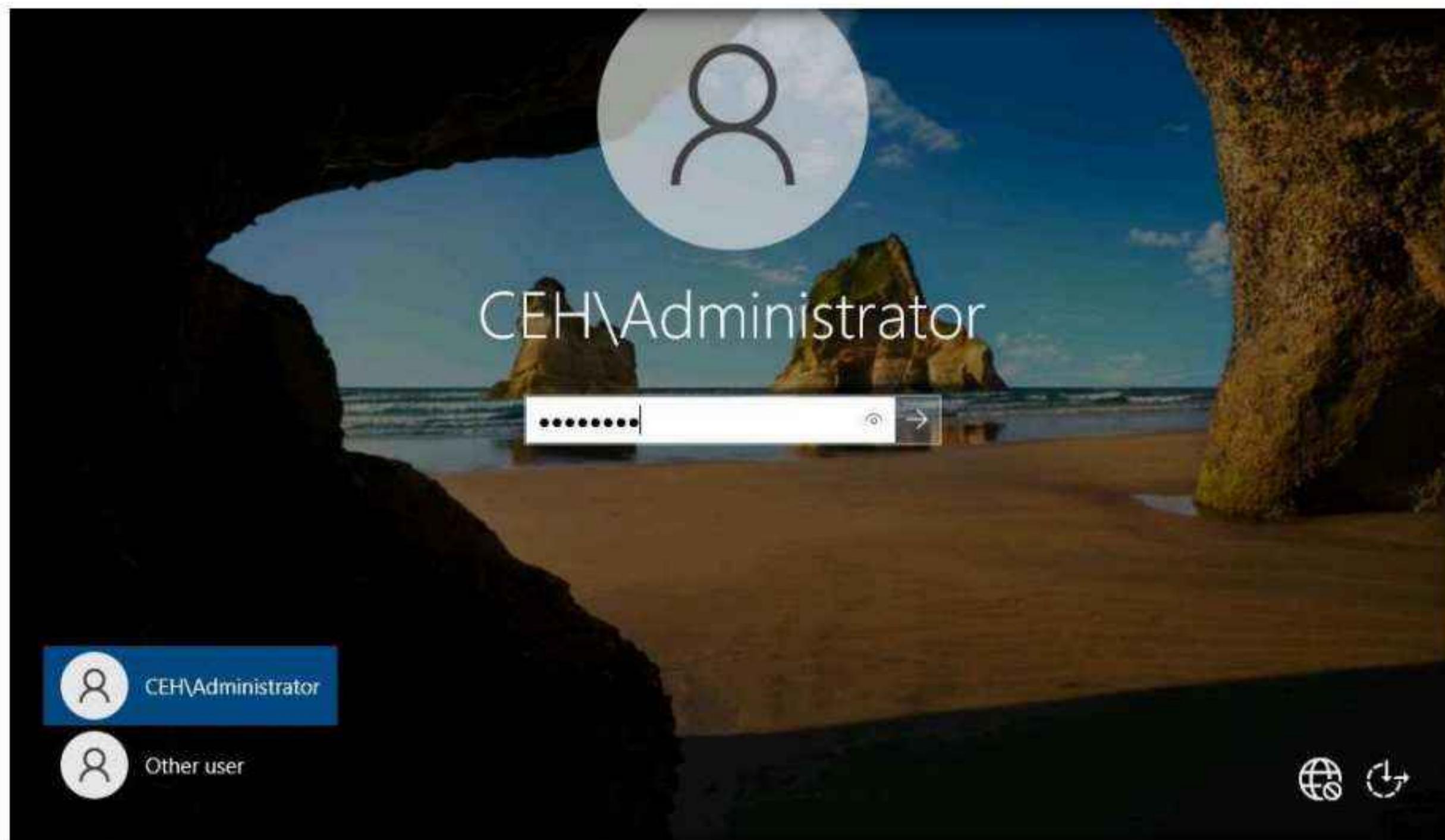
8. Click the **Scans** tab to view the command used to perform TCP connect/full open scan.
9. Click the **Services** tab located in the left pane of the window. This tab displays a list of services.



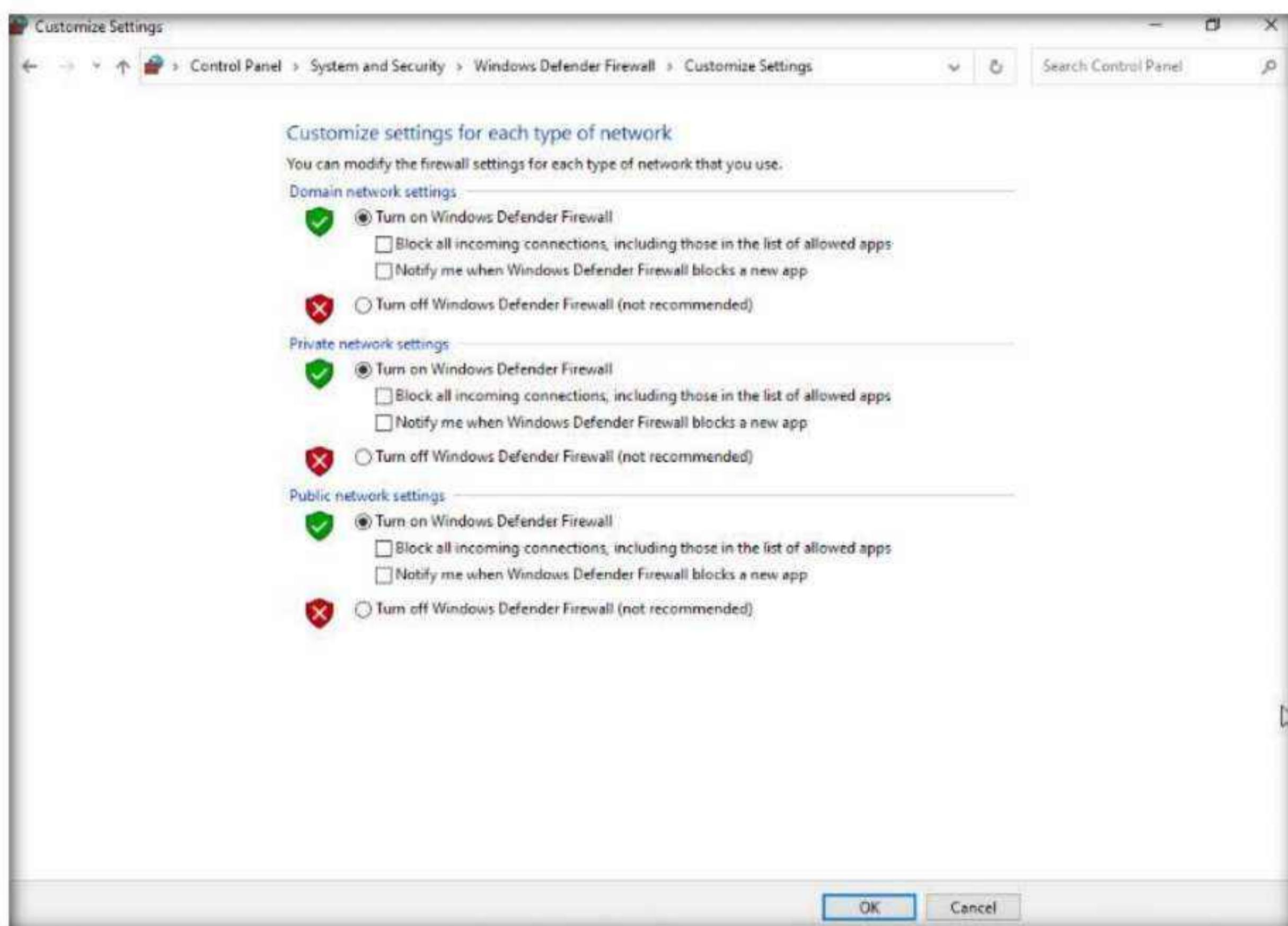
Note: You can use any of these services and their open ports to enter into the target network/host and establish a connection.

10. 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.
11. Switch to the **Windows Server 2022** virtual machine.

12. 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.



13. 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.



14. Now, switch to the **Windows 11** virtual 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).

15. 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.

```

Zenmap
Scan Tools Profile Help
Target: 10.10.1.22 Profile:
Command: nmap -sS -v 10.10.1.22

Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans
Service
domain
eklogin
globalcatLDAP
globalcatLDAPssl
http
http+rpc+epmap
kerberos-sec
kpasswd5
ldap
ldapssl
microsoft-ds
ms-wbt-server
msmq
msmq-mgmt
msrpc
netbios-ssn
zephyr-clt

nmap -sS -v 10.10.1.22
Discovered open port 593/tcp on 10.10.1.22
Discovered open port 2105/tcp on 10.10.1.22
Discovered open port 3268/tcp on 10.10.1.22
Discovered open port 1801/tcp on 10.10.1.22
Discovered open port 2103/tcp on 10.10.1.22
Discovered open port 88/tcp on 10.10.1.22
Discovered open port 464/tcp on 10.10.1.22
Discovered open port 2107/tcp on 10.10.1.22
Discovered open port 389/tcp on 10.10.1.22
Discovered open port 636/tcp on 10.10.1.22
Completed SYN Stealth Scan at 12:35, 4.75s elapsed (1000 total ports)
Nmap scan report for 10.10.1.22
Host is up (0.0013s latency).
Not shown: 983 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http+rpc+epmap
636/tcp   open  ldapssl
1801/tcp  open  msmq
2103/tcp  open  zephyr-clt
2105/tcp  open  eklogin
2107/tcp  open  msmq-mgmt
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
MAC Address: 70:B9:C3:9A:B8:40 (Unknown)

Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 4.93 seconds
Raw packets sent: 1984 (87.280KB) | Rcvd: 18 (776B)

```

16. 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.

17. 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).

18. 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.

The screenshot shows the Zenmap interface. The 'Targets' field contains '10.10.1.22'. The 'Command' field shows 'nmap -sX -v 10.10.1.22'. The 'Hosts' tab is selected. The main pane displays the Nmap output for host 10.10.1.22. The output shows the scan starting at 12:36 Pacific Daylight Time, performing an ARP ping scan, parallel DNS resolution, and an XMAS scan. It reports 1000 scanned ports, all in ignored states, with 1000 open|filtered ports (no-response). The MAC address is listed as 70:89:C3:9A:B8:40 (Unknown). Scan statistics show 1 IP address scanned in 23.10 seconds with 2001 raw packets sent and 2 received.

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-06 12:36 Pacific Daylight Time
Initiating ARP Ping Scan at 12:36
Scanning 10.10.1.22 [1 port]
Completed ARP Ping Scan at 12:36, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:36
Completed Parallel DNS resolution of 1 host. at 12:36, 0.02s elapsed
Initiating XMAS Scan at 12:36
Scanning 10.10.1.22 [1000 ports]
Completed XMAS Scan at 12:37, 22.91s elapsed (1000 total ports)
Nmap scan report for 10.10.1.22
Host is up (0.0010s latency).
All 1000 scanned ports on 10.10.1.22 are in ignored states.
Not shown: 1000 open|filtered tcp ports (no-response)
MAC Address: 70:89:C3:9A:B8:40 (Unknown)

Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 23.10 seconds
Raw packets sent: 2001 (80.028KB) | Rcvd: 2 (80B)
```

19. 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).

20. 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.

The screenshot shows the Zenmap interface. The 'Targets' field contains '10.10.1.22'. The 'Command' field shows 'nmap -sM -v 10.10.1.22'. The 'Services' tab is selected. The main pane displays the Nmap output for host 10.10.1.22:

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-06 12:39 Pacific Daylight Time
Initiating ARP Ping Scan at 12:39
Scanning 10.10.1.22 [1 port]
Completed ARP Ping Scan at 12:39, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:39
Completed Parallel DNS resolution of 1 host. at 12:39, 0.02s elapsed
Initiating Maimon Scan at 12:39
Scanning 10.10.1.22 [1000 ports]
Completed Maimon Scan at 12:39, 22.91s elapsed (1000 total ports)
Nmap scan report for 10.10.1.22
Host is up (0.0010s latency).
All 1000 scanned ports on 10.10.1.22 are in ignored states.
Not shown: 1000 open|filtered tcp ports (no-response)
MAC Address: 70:B9:C3:9A:B8:40 (Unknown)

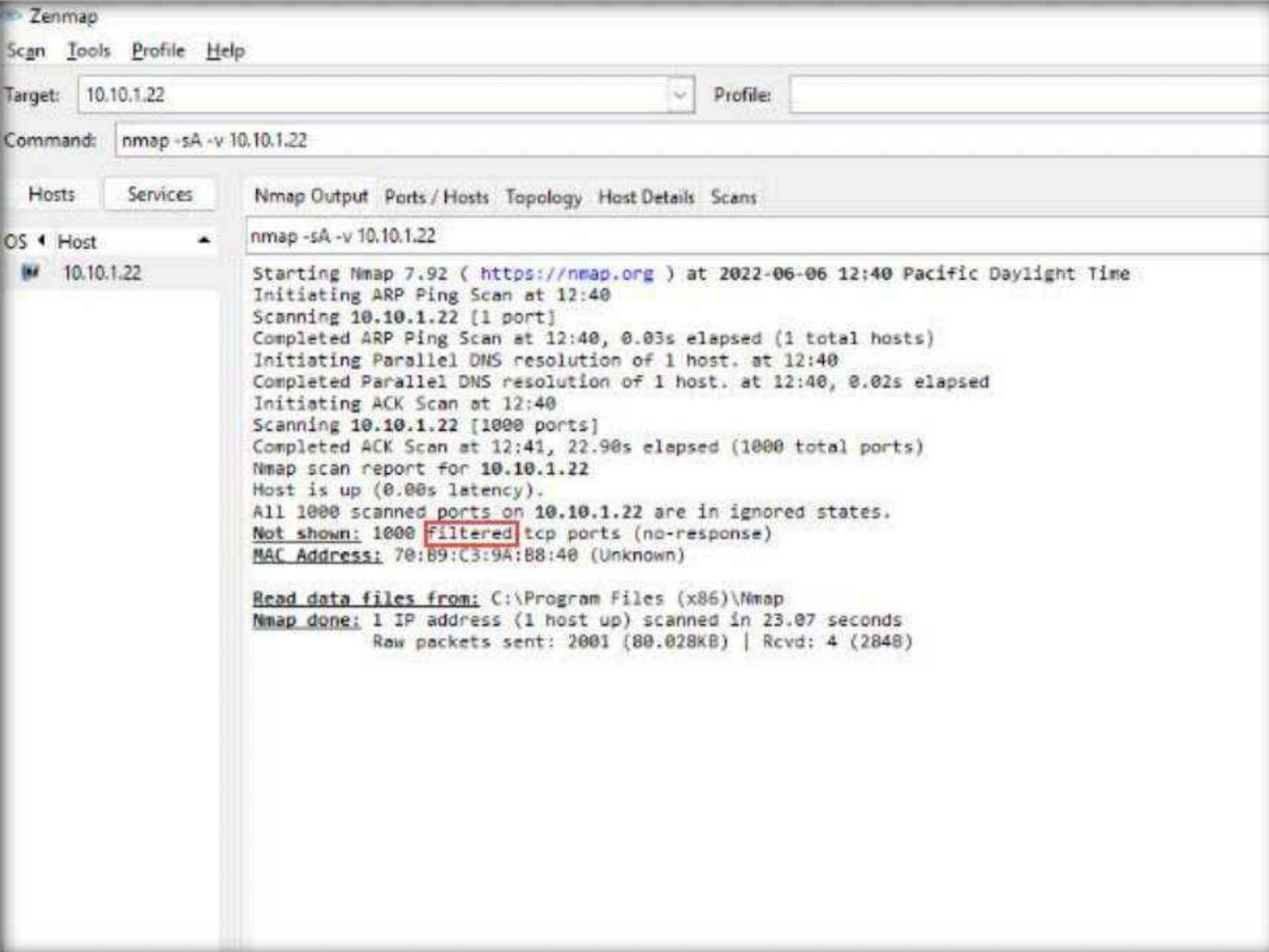
Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 23.10 seconds
Raw packets sent: 2001 (80.028KB) | Rcvd: 1 (288)
```

21. 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).

22. 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.



Zenmap

Scan Tools Profile Help

Target: 10.10.1.22 Profile:

Command: nmap -sA -v 10.10.1.22

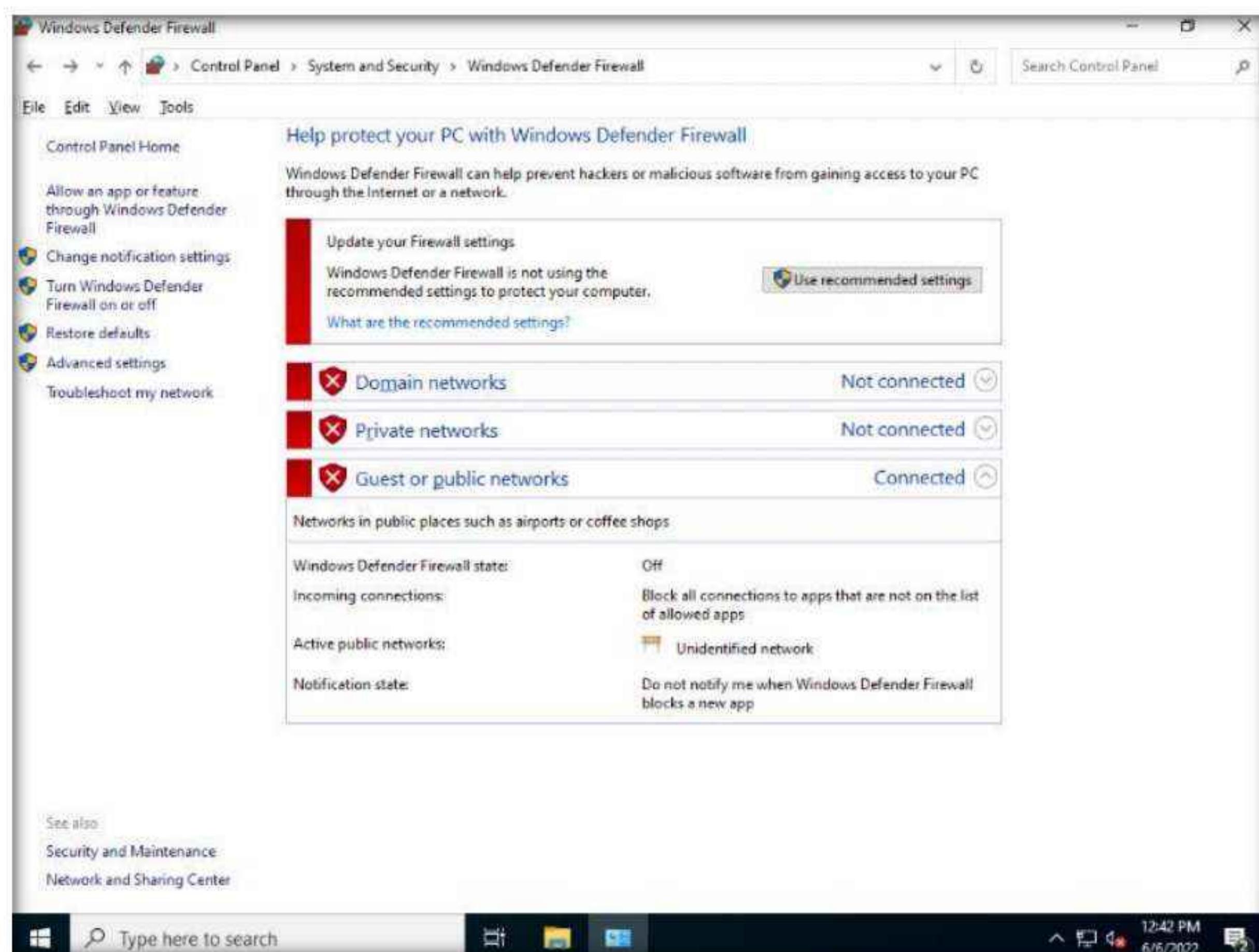
Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans

OS Host ▾ 10.10.1.22

```
nmap -sA -v 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-06 12:40 Pacific Daylight Time
Initiating ARP Ping Scan at 12:40
Scanning 10.10.1.22 [1 port]
Completed ARP Ping Scan at 12:40, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:40
Completed Parallel DNS resolution of 1 host. at 12:40, 0.02s elapsed
Initiating ACK Scan at 12:40
Scanning 10.10.1.22 [1000 ports]
Completed ACK Scan at 12:41, 22.90s elapsed (1000 total ports)
Nmap scan report for 10.10.1.22
Host is up (0.00s latency).
All 1000 scanned ports on 10.10.1.22 are in ignored states.
Not shown: 1000 Filtered tcp ports (no-response)
MAC Address: 70:89:C3:9A:B8:40 (Unknown)

Read data files from: C:\Program Files (x86)\Nmap
Nmap done: 1 IP address (1 host up) scanned in 23.07 seconds
Raw packets sent: 2001 (80.028KB) | Rcvd: 4 (2848)
```

23. Switch to the **Windows Server 2022** virtual machine.
24. If you are logged out of the **Windows Server 2022** virtual 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.
25. Turn off the **Windows Defender Firewall** from **Control Panel**.



26. Now, switch back to the **Windows 11** virtual 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).

27. 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.

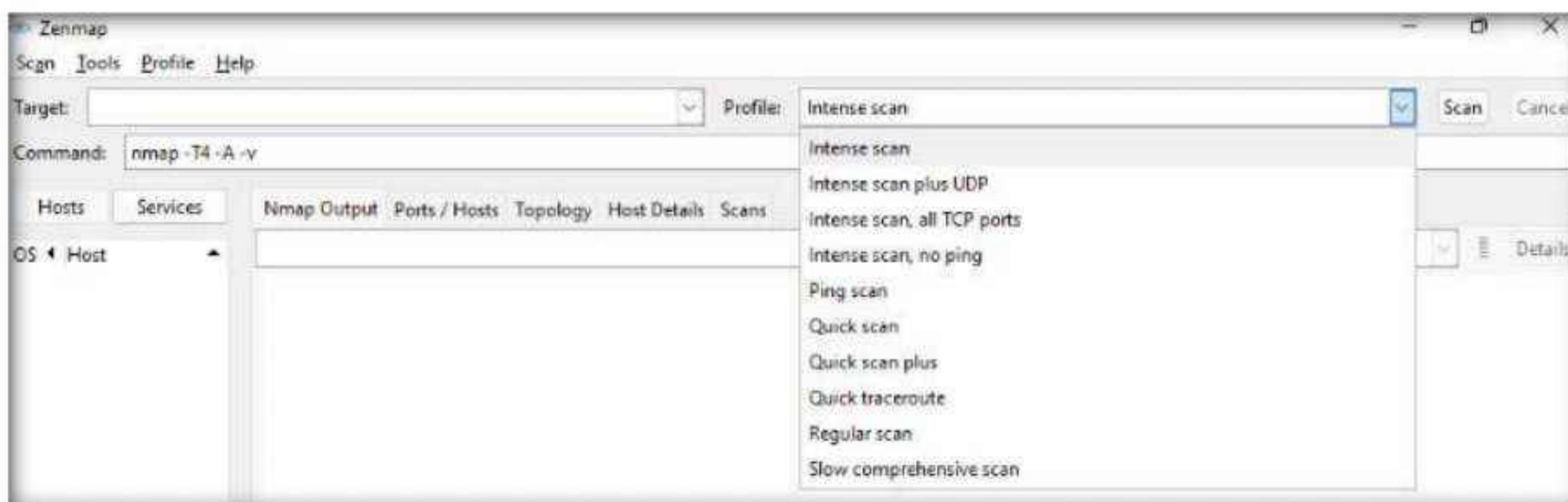
Note: The UDP scan uses UDP protocol instead of the TCP. There is no three-way handshake for the UDP scan. It sends UDP packets to the target host; no response means that the port is open. If the port is closed, an ICMP port unreachable message is received.

Module 03 – Scanning Networks

The screenshot shows the Zenmap interface with the following details:

- Target:** 10.10.1.22
- Command:** nmap -sU -v 10.10.1.22
- OS:** Host
- Host:** 10.10.1.22
- Scan Output:**
 - UDP Scan Timing: About 98.67% done; ETC: 13:03 (0:01:50 remaining)
 - Discovered open port 389/udp on 10.10.1.22
 - Completed UDP Scan at 13:04, 1197.32s elapsed (1000 total ports)
 - Nmap scan report for 10.10.1.22
 - Host is up (0.00071s latency).
 - Not shown: 976 closed udp ports (port-unreach)
 - PORT STATE SERVICE
 - 53/udp open domain
 - 88/udp open|filtered kerberos-sec
 - 123/udp open ntp
 - 137/udp open netbios-ns
 - 138/udp open|filtered netbios-dgm
 - 161/udp open snmp
 - 389/udp open ldap
 - 464/udp open|filtered kpasswd5
 - 500/udp open|filtered isakmp
 - 3389/udp open|filtered ms-wbt-server
 - 4500/udp open|filtered nat-t-ike
 - 5353/udp open|filtered zeroconf
 - 5355/udp open|filtered llmnr
 - 56141/udp open|filtered unknown
 - 57172/udp open|filtered unknown
 - 57489/udp open|filtered unknown
 - 57410/udp open|filtered unknown
 - 57813/udp open|filtered unknown
 - 57843/udp open|filtered unknown
 - 57958/udp open|filtered unknown
 - 57977/udp open|filtered unknown
 - 58002/udp open|filtered unknown
 - 58075/udp open|filtered unknown
 - 58178/udp open|filtered unknown
 - MAC Address: 70:B9:C3:9A:B8:40 (Unknown)
- Statistics:**
 - Read data files from: C:\Program Files (x86)\Nmap
 - Nmap done: 1 IP address (1 host up) scanned in 1197.52 seconds
 - Raw packets sent: 1290 (65.131KB) | Rcvd: 1007 (73.910KB)

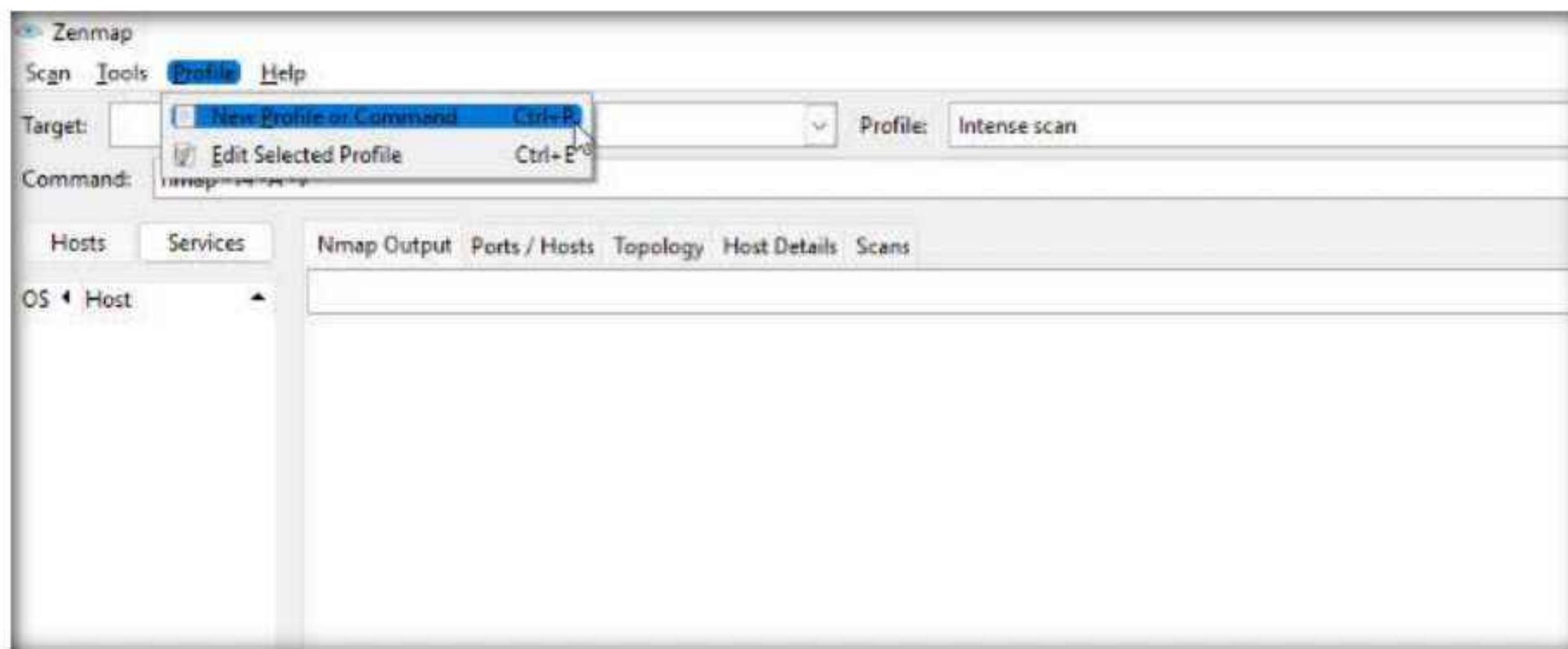
28. Close the **Zenmap** window.
29. You can create your scan profile, or you can also choose the default scan profiles available in Nmap to scan a network.
30. Click **Search icon** (🔍) on the **Desktop**. Type **zenmap** in the search field, the **Nmap - Zenmap GUI** appears in the results, click **Open** to launch it.
31. 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.



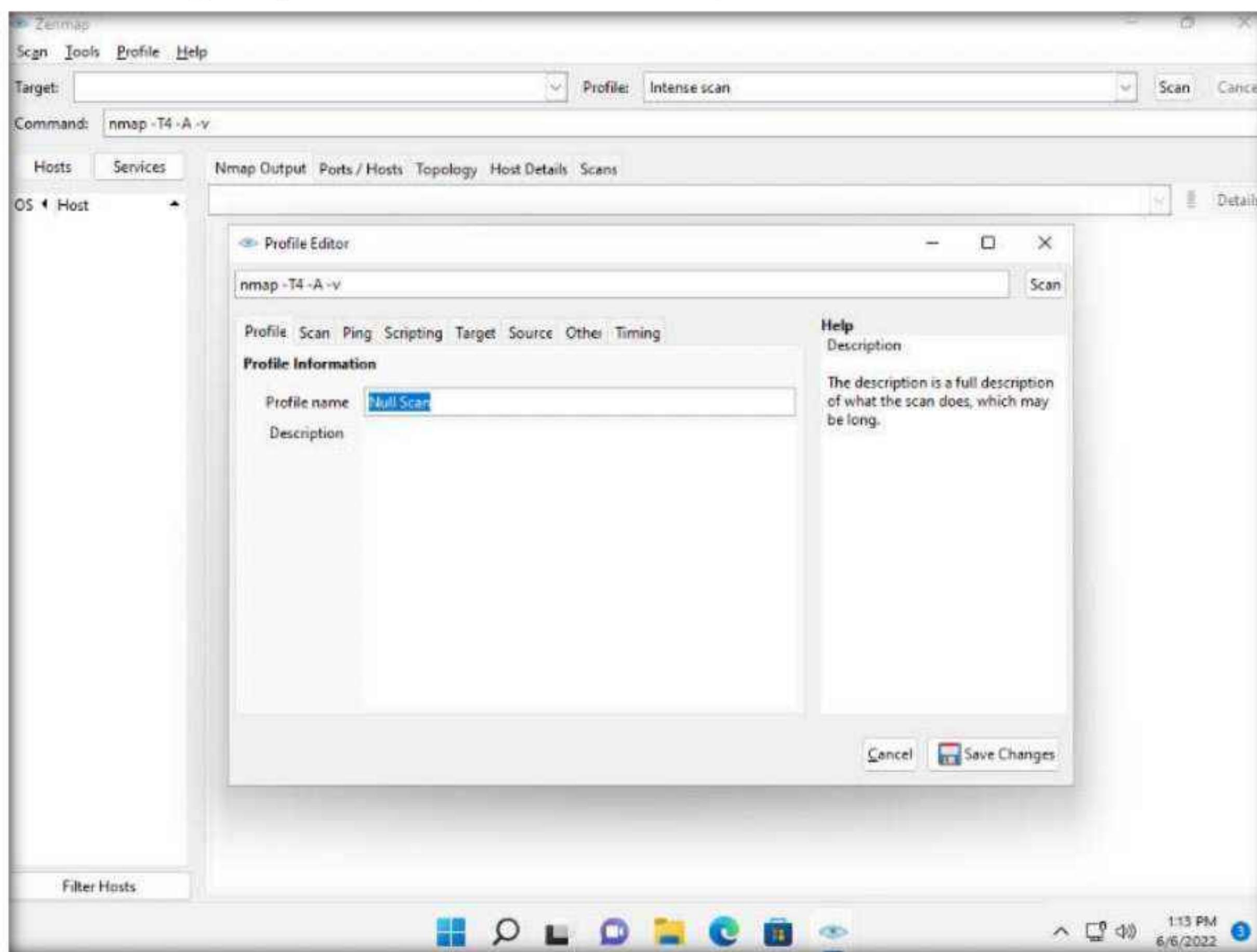
Module 03 – Scanning Networks

32. To create a scan profile; click **Profile** → **New Profile or Command**.

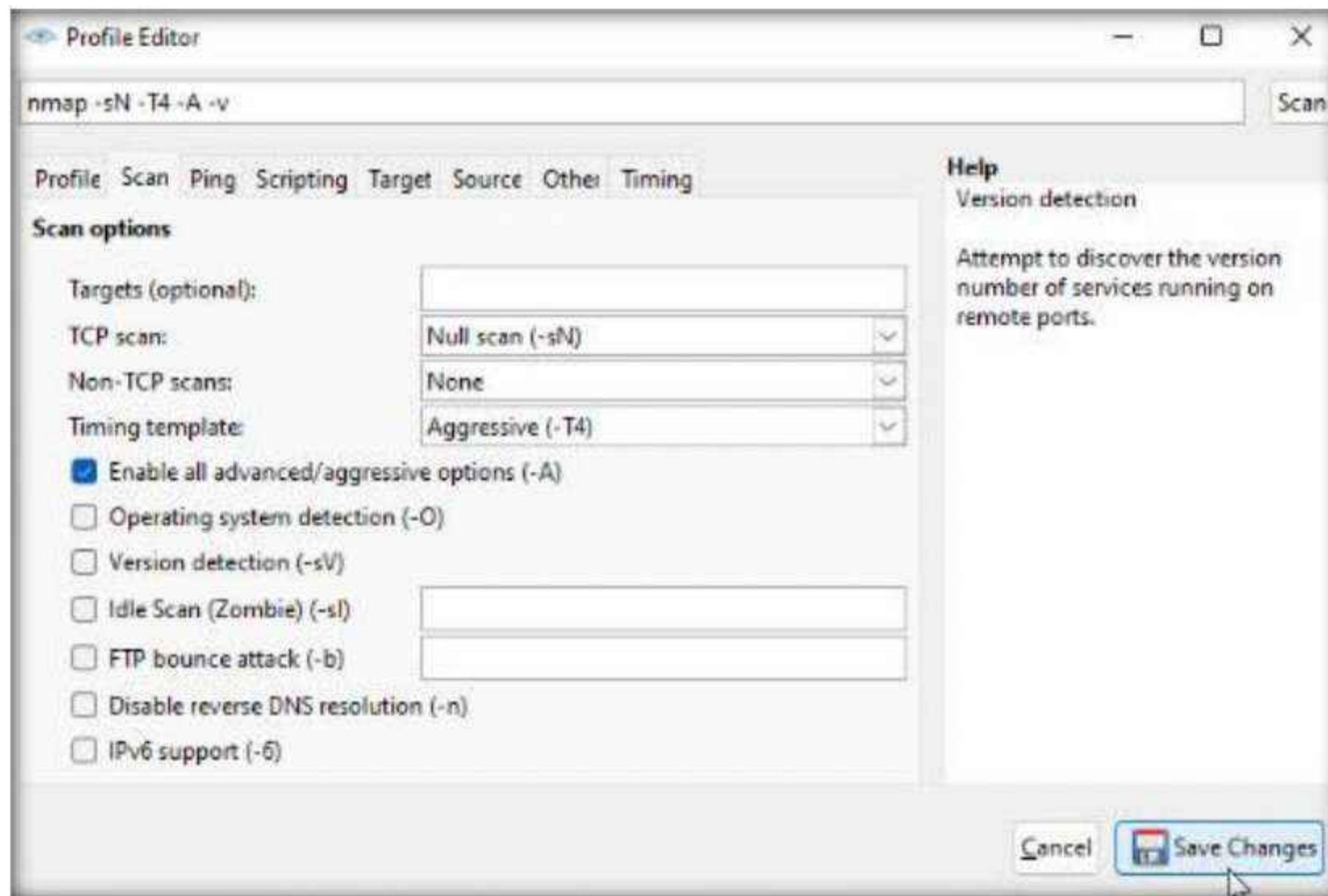
Note: If a User Account Control pop-up appears, click **Yes**.



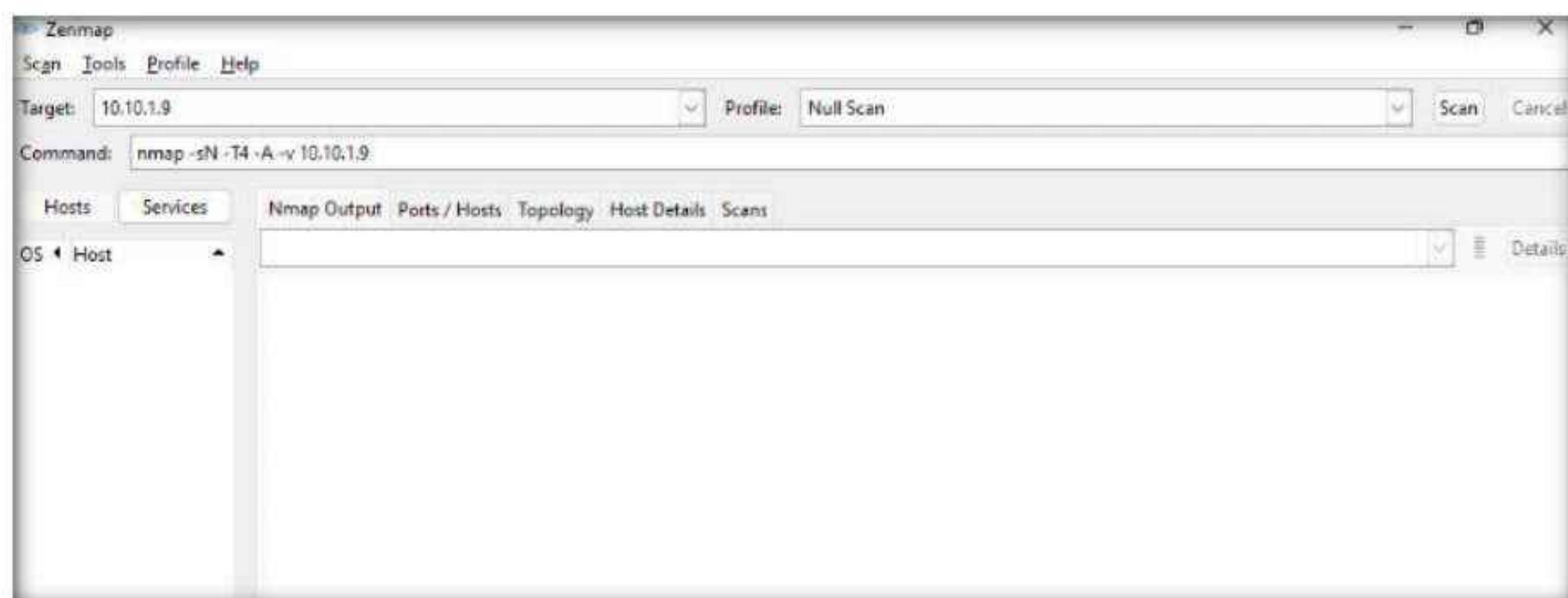
33. The **Profile Editor** window appears. In the **Profile** tab, under the **Profile Information** section, input a profile name (here, **Null Scan**) into the **Profile name** field.



34. Now, click the **Scan** tab and select the scan option (here, **Null scan (-sN)**) from the **TCP scan** drop-down list.
 35. 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.
36. This will create a new profile, and will thus be added to the profile list.



37. In this sub-task, we will be targeting the **Ubuntu** machine (**10.10.1.9**).
38. 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**.



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

```

Zenmap
Scan Tools Profile Help
Target: 10.10.1.9 Profile: Null Scan
Command: nmap -sN -T4 -A -v 10.10.1.9

Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans
OS Host 10.10.1.9
nmap -sN -T4 -A -v 10.10.1.9
Not shown: 998 closed tcp ports (reset)
PORT STATE SERVICE VERSION
22/tcp open ssh OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 28:52:84:53:60:ec:72:72:ce:80:ba:db:35:74:b5:55 (ECDSA)
|   256 9a:1e:e9:21:07:9f:7c:25:95:c9:6a:b6:5e:fe:e4:51 (ED25519)
80/tcp open http Apache HTTPD 2.4.52 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-methods:
|   _ Supported Methods: OPTIONS HEAD GET POST
|_http-server-header: Apache/2.4.52 (Ubuntu)
MAC Address: 38:14:F4:D2:1C:D3 (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Uptime guess: 23.095 days (since Sat May 14 11:06:22 2022)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=260 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT ADDRESS
1 0.65 ms 10.10.1.9

NSE: Script Post-scanning.
Initiating NSE at 13:23
Completed NSE at 13:23, 0.00s elapsed
Initiating NSE at 13:23
Completed NSE at 13:23, 0.00s elapsed
Initiating NSE at 13:23
Completed NSE at 13:23, 0.00s elapsed
Read data files from: C:\Program Files (x86)\Nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.98 seconds
Raw packets sent: 1025 (41.886KB) | Rcvd: 1013 (41.198KB)

```

40. 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]

41. 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.

42. 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.

```

nmap -sV 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-06 13:25 Pacific Daylight Time
Nmap scan report for 10.10.1.22
Host is up (0.00048s latency).
Not shown: 983 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
53/tcp    open  domain      Simple DNS Plus
80/tcp    open  http        Microsoft IIS httpd 10.0
88/tcp    open  Kerberos-sec Microsoft Windows Kerberos (server time: 2022-06-06 20:25:12Z)
135/tcp   open  msrpc       Microsoft Windows RPC
139/tcp   open  netbios-ssn Microsoft Windows netbios-ssn
389/tcp   open  ldap        Microsoft Windows Active Directory LDAP (Domain: CEH.com., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: CEH)
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
1801/tcp  open  msmq?
2103/tcp  open  msrpc       Microsoft Windows RPC
2105/tcp  open  msrpc       Microsoft Windows RPC
2107/tcp  open  msrpc       Microsoft Windows RPC
3268/tcp  open  ldap        Microsoft Windows Active Directory LDAP (Domain: CEH.com., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
MAC Address: 70:B9:C3:9A:8B:40 (Unknown)
Service Info: Host: SERVER2022; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 54.39 seconds

```

43. 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.

44. 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.

Module 03 – Scanning Networks

The screenshot shows the Zenmap interface after performing an OS scan on the target 10.10.1.22. The left pane lists hosts, and the right pane displays the Nmap output. The output details the operating system as Windows 10 Enterprise 22H2 (Windows 10 Enterprise 6.3), with a CPE string of cpe:/o:microsoft:windows_10::-. It also provides host script results, including SMB security mode (SMB2) and message signing information. Post-scan script results show clock skew for both the scanner and the target host.

```
nmap -A 10.10.1.22
[...]
OS: Windows 10 Enterprise 22H2 (Windows 10 Enterprise 6.3)
OS CPE: cpe:/o:microsoft:windows_10::-
Computer name: Windows11
NetBIOS computer name: WINDOWS11\x00
Workgroup: WORKGROUP\x00
System time: 2022-06-06T13:31:07-07:00
clock-skew: mean: 1h24m00s, deviation: 3h07m51s, median: 0s
[...]
```

45. 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.

The screenshot shows the Zenmap interface with the Host Details tab selected for the host 10.10.1.22. The left pane shows other hosts, and the right pane displays detailed information about the selected host. The Host Status section shows the host is up with 17 open ports. The Addresses section lists IPv4 (10.10.1.22), IPv6 (Not available), and MAC (70:89:C3:9A:B8:40). The Operating System section identifies it as Microsoft Windows 10 1703 with 97% accuracy. The Ports used section lists open and closed ports. The OS Classes section shows it belongs to the general purpose class for Microsoft Windows 10 with 97% accuracy. The TCP Sequence and IP ID section are also visible at the bottom.

46. 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.
47. Close all open windows and document all the acquired information.
48. Turn off the **Windows 11** and **Ubuntu** virtual machines.

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.

Note: Ensure that the **Windows Server 2022** virtual machine is running.

1. Turn on the **Parrot Security** virtual 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. Now, type **cd** and press **Enter** to jump to the root directory.

The screenshot shows a terminal window titled "cd - Parrot Terminal". The terminal session starts with the user "[attacker@parrot]". The user runs the command "\$sudo su". A password prompt "(sudo) password for attacker:" appears. The user enters "toor" and presses Enter. The terminal then shows the user is now root, with the prompt "[root@parrot]". The user then runs the command "#cd". Finally, the user is back at the root prompt "[root@parrot]".

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 screenshot.

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.

The screenshot shows a terminal window titled "hping3-A 10.10.1.22 -p 80 -c 5 - Parrot Terminal". The terminal session starts with the user becoming root via "sudo su". The user then runs "hping3 -A 10.10.1.22 -p 80 -c 5". The output shows 5 packets transmitted and 5 packets received with 0% packet loss, indicating a successful ACK scan where the port was found to be open.

```
[attacker@parrot] ~
→ $sudo su
[sudo] password for attacker:
[root@parrot] ~
→ #cd
[root@parrot] ~
→ #hping3 -A 10.10.1.22 -p 80 -c 5
HPING 10.10.1.22 (eth0 10.10.1.22): A set, 40 headers + 0 data bytes
len=40 ip=10.10.1.22 ttl=128 DF id=0 sport=80 flags=R seq=0 win=0 rtt=7.7 ms
len=40 ip=10.10.1.22 ttl=128 DF id=1 sport=80 flags=R seq=1 win=0 rtt=7.5 ms
len=40 ip=10.10.1.22 ttl=128 DF id=2 sport=80 flags=R seq=2 win=0 rtt=7.4 ms
len=40 ip=10.10.1.22 ttl=128 DF id=3 sport=80 flags=R seq=3 win=0 rtt=11.1 ms
len=40 ip=10.10.1.22 ttl=128 DF id=4 sport=80 flags=R seq=4 win=0 rtt=10.9 ms

... 10.10.1.22 hping statistic ...
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 7.4/8.9/11.1 ms
[root@parrot] ~
→ #
```

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.

```
[root@parrot]~[-]
→ #hping3 -8 0-100 -S 10.10.1.22 -V
using eth0, addr: 10.10.1.13, MTU: 1500
Scanning 10.10.1.22 (10.10.1.22), port 0-100
101 ports to scan, use -V to see all the replies
+-----+
|port| serv name | flags |ttl| id | win | len |
+-----+
 0   : ..R.A... 128 1280 0 40
 1  tcpmux   : ..R.A... 128 1536 0 40
 2  nntp    : ..R.A... 128 1792 0 40
 3   : ..R.A... 128 2048 0 40
 4  echo     : ..R.A... 128 2304 0 40
 5   : ..R.A... 128 2560 0 40
 6  zip      : ..R.A... 128 2816 0 40
 7  echo     : ..R.A... 128 3072 0 40
 8   : ..R.A... 128 3328 0 40
 9  discard   : ..R.A... 128 3584 0 40
10   : ..R.A... 128 3840 0 40
11  systat   : ..R.A... 128 4096 0 40
12   : ..R.A... 128 4352 0 40
13  daytime   : ..R.A... 128 4608 0 40
14   : ..R.A... 128 4864 0 40
15  netstat   : ..R.A... 128 5120 0 40
16   : ..R.A... 128 5376 0 40
17  qotd     : ..R.A... 128 5632 0 40
18   : ..R.A... 128 5888 0 40
19  chargen   : ..R.A... 128 6144 0 40
20  ftp-data  : ..R.A... 128 6400 0 40
21  ftp      : ..R.A... 128 6656 0 40
+-----+
[[ Menu ] hping3 -8 0-100 -S 10.10.1.22 -V]
```

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.

```
[root@parrot]~[-]
→ #hping3 -F -P -U 10.10.1.22 -p 80 -c 5
HPING 10.10.1.22 (eth0 10.10.1.22): FPU set, 40 headers + 0 data bytes
len=40 ip=10.10.1.22 ttl=128 DF id=61155 sport=80 flags=RA seq=0 win=0 rtt=3.8 ms
len=40 ip=10.10.1.22 ttl=128 DF id=61156 sport=80 flags=RA seq=1 win=0 rtt=3.6 ms
len=40 ip=10.10.1.22 ttl=128 DF id=61157 sport=80 flags=RA seq=2 win=0 rtt=3.4 ms
len=40 ip=10.10.1.22 ttl=128 DF id=61158 sport=80 flags=RA seq=3 win=0 rtt=3.2 ms
len=40 ip=10.10.1.22 ttl=128 DF id=61159 sport=80 flags=RA seq=4 win=0 rtt=3.1 ms

--- 10.10.1.22 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 3.1/3.4/3.8 ms
```

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.

```
[root@parrot] ~
[root@parrot] ~# hping3 --scan 0-100 -S 10.10.1.22
Scanning 10.10.1.22 (10.10.1.22), port 0-100
101 ports to scan, use -V to see all the replies
+---+-----+-----+-----+
|port| serv name | flags |ttl| id | win | len |
+---+-----+-----+-----+
 53 domain   : .S..A... 128 7663 65392 44
 80 http     : .S..A... 128 14575 65392 44
 88 kerberos : .S..A... 128 16623 65392 44
All replies received. Done.
Not responding ports:
```

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.

```
[root@parrot] ~
[root@parrot] ~# hping3 -1 10.10.1.22 -p 80 -c 5
HPING 10.10.1.22 (eth0 10.10.1.22): icmp mode set, 28 headers + 0 data bytes
len=28 ip=10.10.1.22 ttl=128 id=61440 icmp_seq=0 rtt=7.8 ms
len=28 ip=10.10.1.22 ttl=128 id=61441 icmp_seq=1 rtt=7.7 ms
len=28 ip=10.10.1.22 ttl=128 id=61442 icmp_seq=2 rtt=7.5 ms
len=28 ip=10.10.1.22 ttl=128 id=61443 icmp_seq=3 rtt=3.2 ms
len=28 ip=10.10.1.22 ttl=128 id=61444 icmp_seq=4 rtt=7.1 ms

--- 10.10.1.22 hping statistic ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 3.2/6.6/7.8 ms
[root@parrot] ~
[root@parrot] ~#
```

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.
20. Turn off the **Parrot Security** and **Windows Server 2022** virtual machines.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> CyberQ

Lab**3**

Perform OS Discovery

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

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

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 15 Minutes

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

Lab Tasks

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.

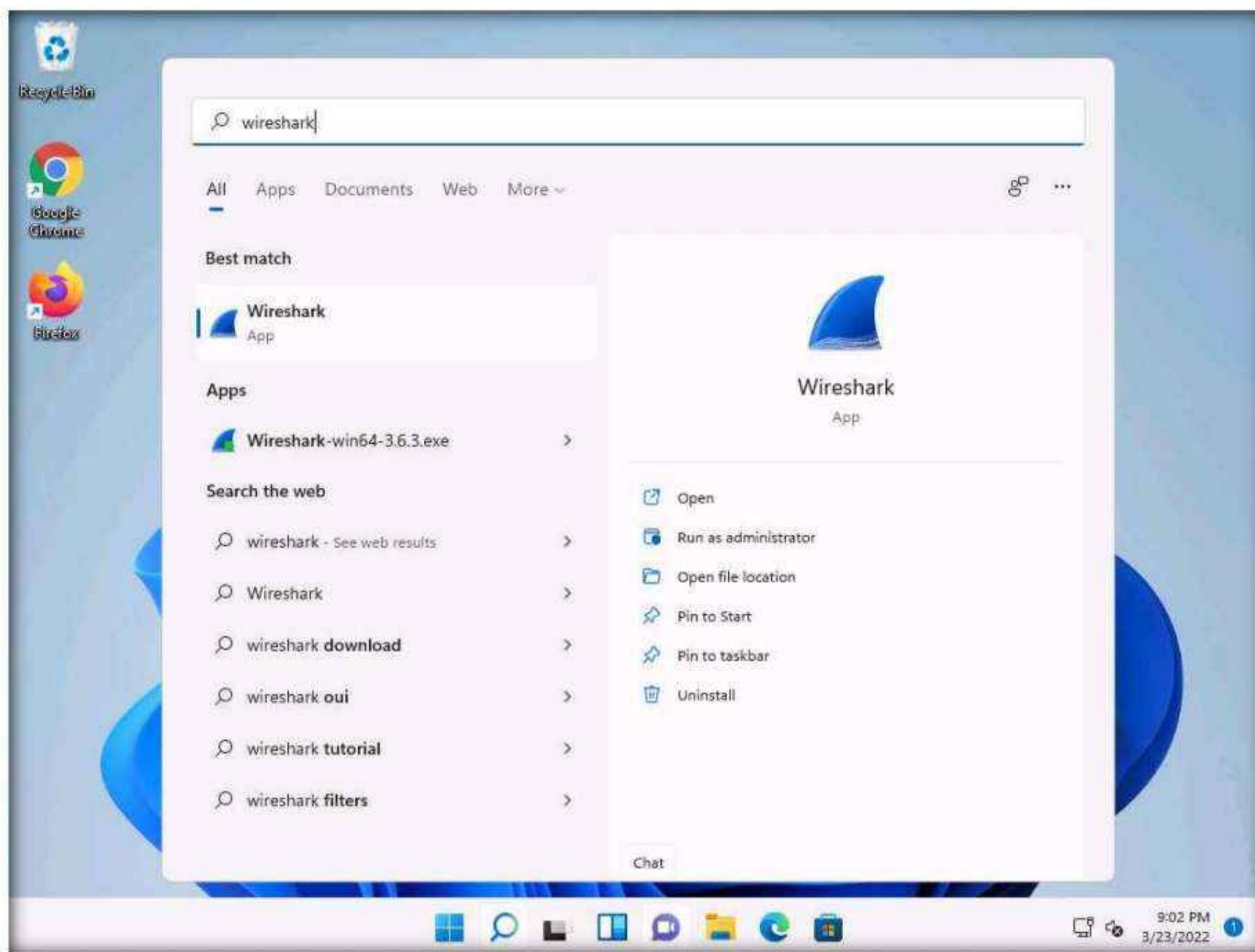
Here, we will use the Wireshark tool to perform OS discovery on the target host(s).

1. Turn on the **Windows 11**, **Windows Server 2022** and **Ubuntu** virtual machines.
2. Switch to the **Windows 11** virtual machine. By default, **Admin** user profile is selected, type **Pa\$\$w0rd** in the **Password** field and press **Enter** to login.

Note: If **Welcome to Windows** wizard appears, click **Continue** and in **Sign in with Microsoft** wizard, click **Cancel**.

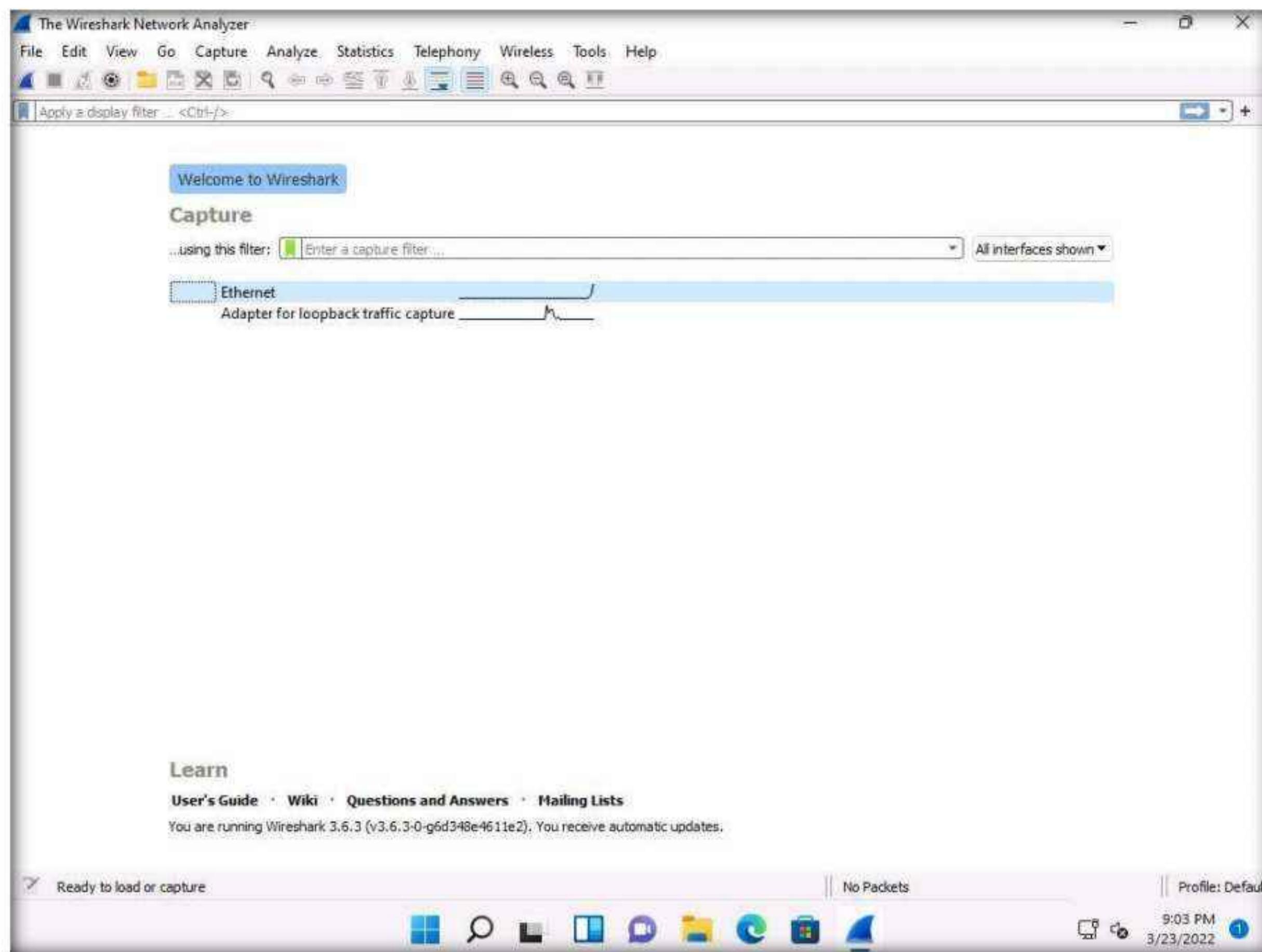
Note: Networks screen appears, click **Yes** to allow your PC to be discoverable by other PCs and devices on the network.

3. Click **Search** icon (🔍) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Open** 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, as shown in the screenshot.

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



5. 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.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.22000.469]
(c) Microsoft Corporation. All rights reserved.

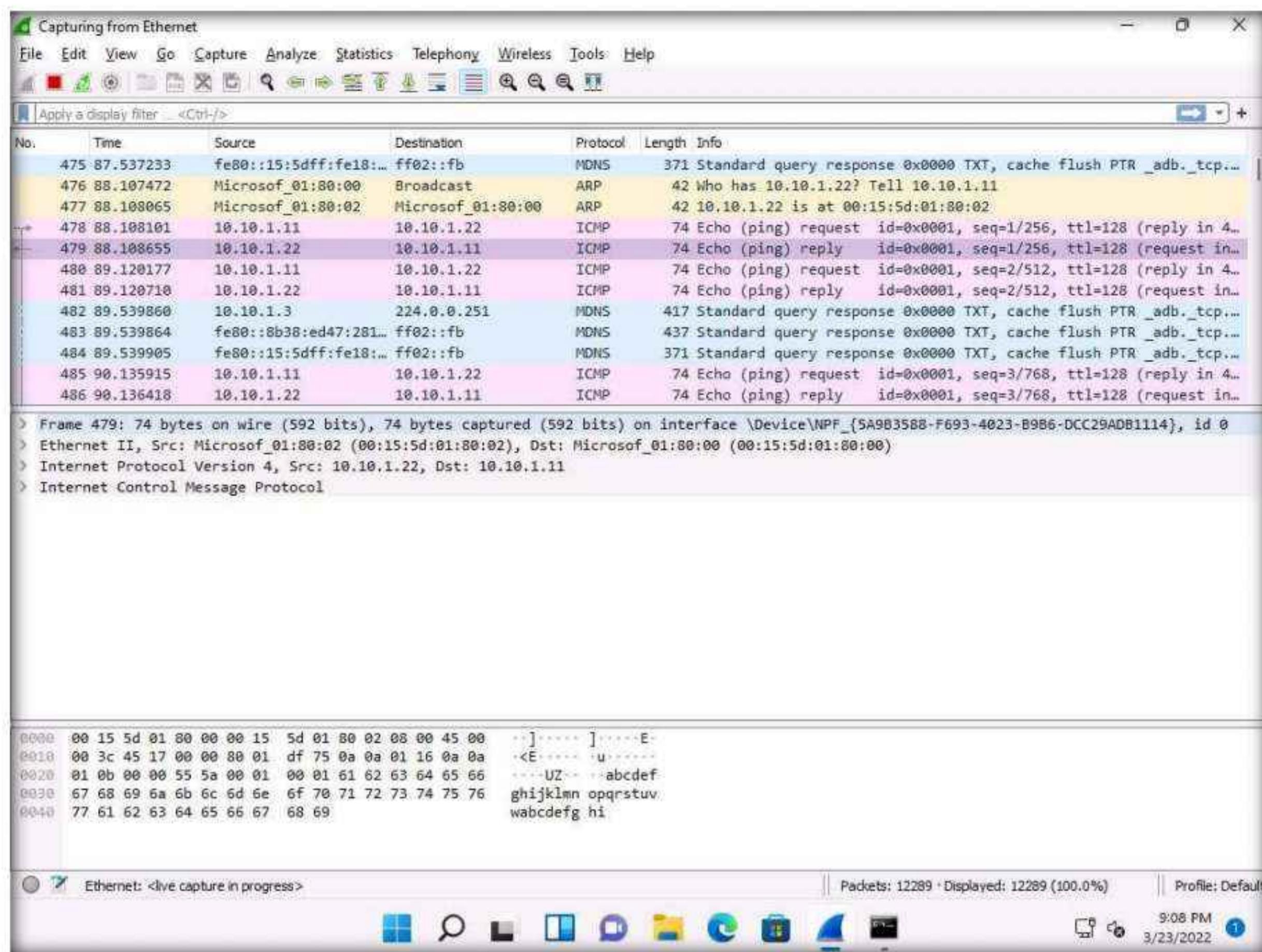
C:\Users\Admin>ping 10.10.1.22

Pinging 10.10.1.22 with 32 bytes of data:
Reply from 10.10.1.22: bytes=32 time=1ms TTL=128
Reply from 10.10.1.22: bytes=32 time<1ms TTL=128
Reply from 10.10.1.22: bytes=32 time<1ms TTL=128
Reply from 10.10.1.22: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.1.22:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

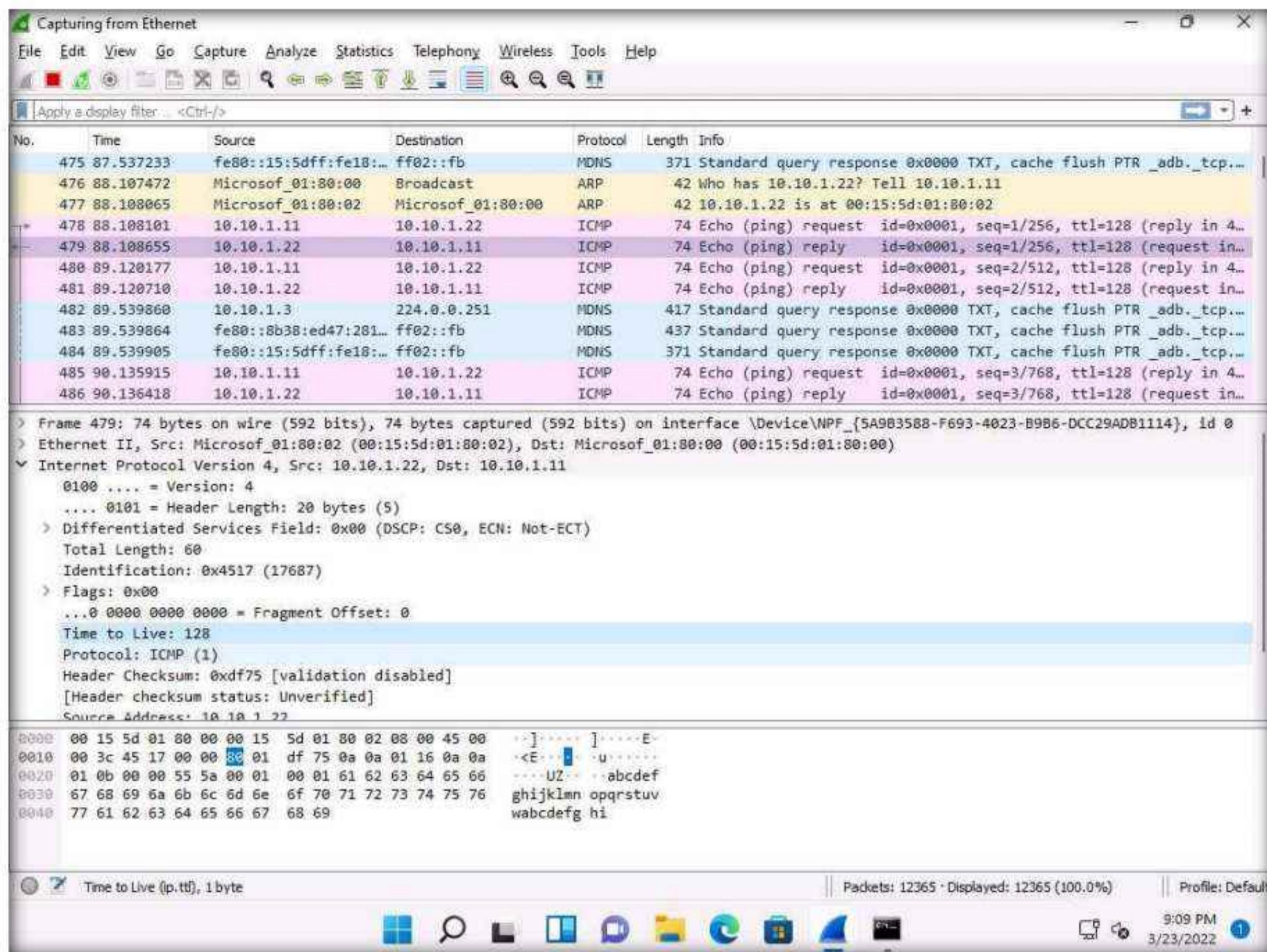
C:\Users\Admin>
```

6. Observe the packets captured by Wireshark.

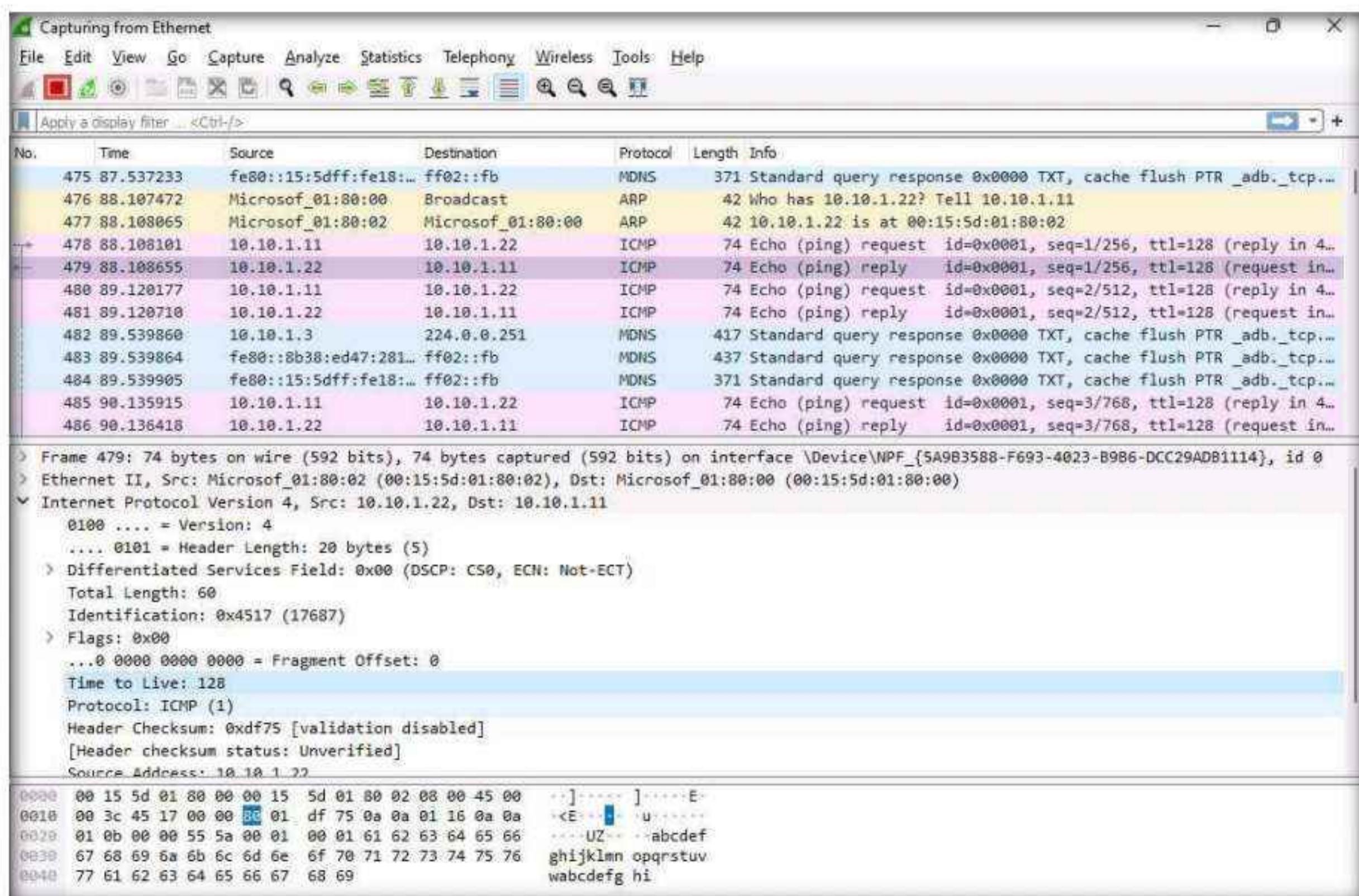


7. Choose any packet of the ICMP reply from the **Windows Server 2022 (10.10.1.22)** to **Windows 11 (10.10.1.11)** machines and expand the **Internet Protocol Version 4** node in the **Packet Details** pane.
8. The TTL value is recorded as **128**, which means that the ICMP reply possibly came from a Windows-based machine.

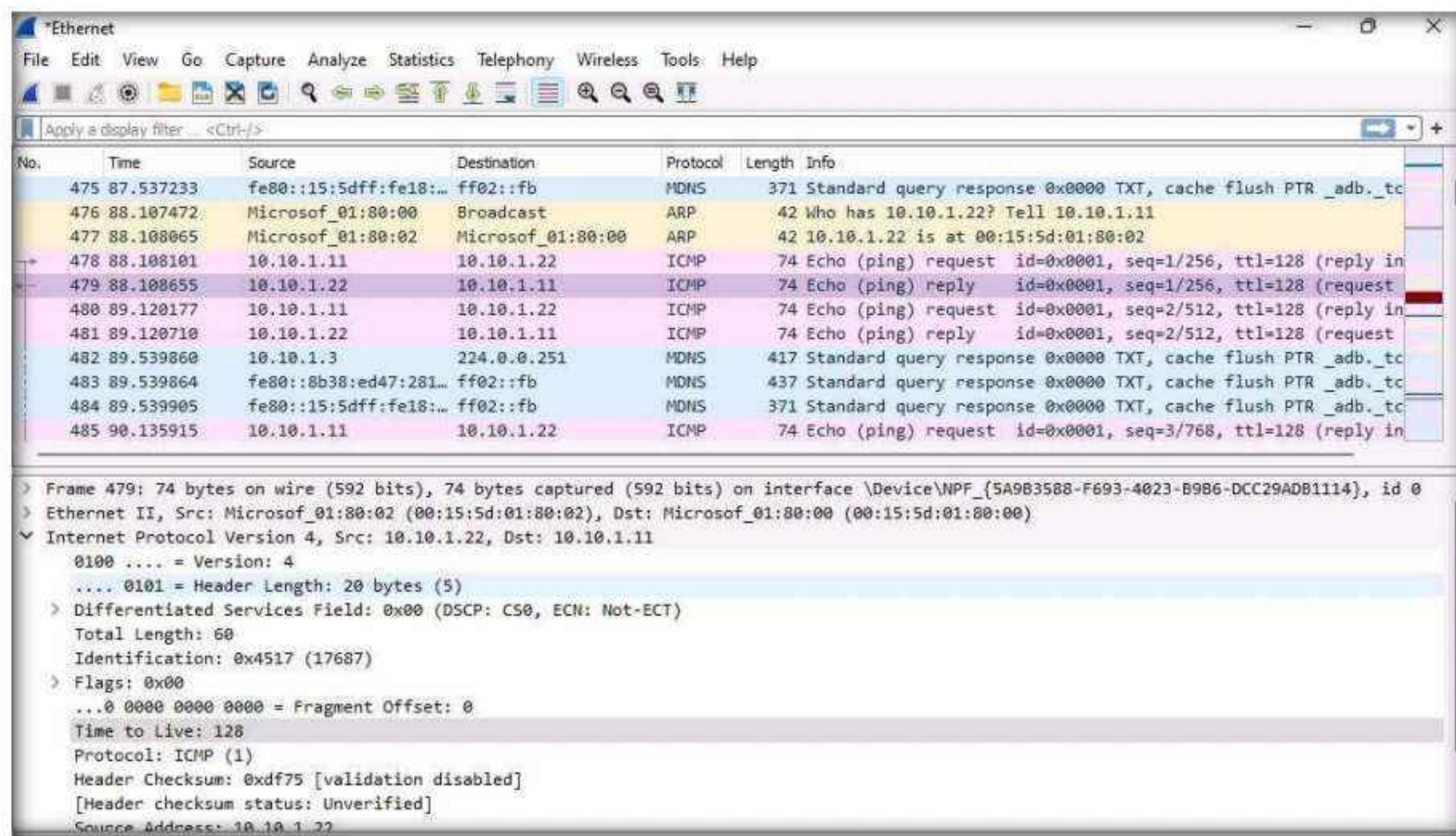
Module 03 – Scanning Networks



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



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



11. Wireshark will start capturing the new packets.

12. 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.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.22000.469]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>ping 10.10.1.22

Pinging 10.10.1.22 with 32 bytes of data:
Reply from 10.10.1.22: bytes=32 time=1ms TTL=128
Reply from 10.10.1.22: bytes=32 time<1ms TTL=128
Reply from 10.10.1.22: bytes=32 time<1ms TTL=128
Reply from 10.10.1.22: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.1.22:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

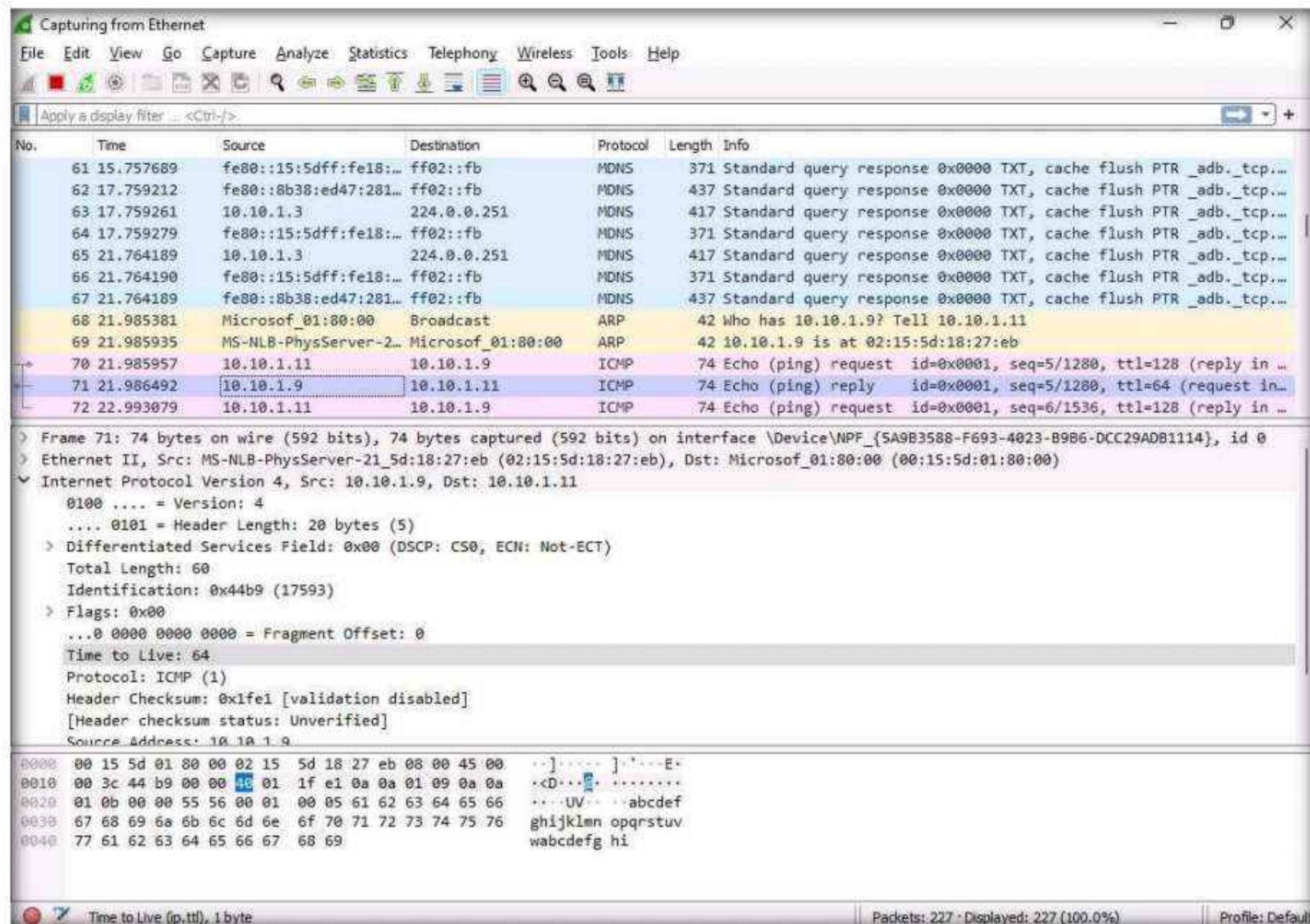
C:\Users\Admin>ping 10.10.1.9

Pinging 10.10.1.9 with 32 bytes of data:
Reply from 10.10.1.9: bytes=32 time=1ms TTL=64
Reply from 10.10.1.9: bytes=32 time<1ms TTL=64
Reply from 10.10.1.9: bytes=32 time<1ms TTL=64
Reply from 10.10.1.9: bytes=32 time<1ms TTL=64

Ping statistics for 10.10.1.9:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\Admin>
```

13. Observe the packets captured by Wireshark.
14. 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.
15. The TTL value is recorded as **64**, which means the ICMP reply possibly came from a Linux-based machine.



16. Stop the capture in the **Wireshark** window by clicking on the Stop button.
17. This concludes the demonstration of identifying the OS of the target system using **Wireshark**.
18. Close all open windows and document all the acquired information.
19. Turn off the **Windows 11** and **Ubuntu** virtual machines.

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.

Note: Ensure that the **Windows Server 2022** virtual machine is running.

1. Turn on the **Parrot Security** virtual 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.

The screenshot shows a terminal window titled "nmap -A 10.10.1.22 - Parrot Terminal". The output displays service information for the host SERVER2022, which is running Windows. It includes details like NetBIOS name, NetBIOS user, NetBIOS MAC address, and various SMB security modes. The "Host script results" section provides comprehensive information about the operating system, including its version (Windows Server 2022 Standard 20348), computer name (Server2022), NetBIOS computer name (SERVER2022\x00), domain name (CEH.com), forest name (CEH.com), FQDN (Server2022.CEH.com), and system time (2022-06-07T03:38:44-07:00). It also shows clock skew statistics and SMB security mode details. The "TRACEROUTE" section shows a single hop to the target IP (10.10.1.22) with a RTT of 2.27 ms. The final message encourages reporting any incorrect results to the nmap.org submission page.

```
nmap -A 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
Service Info: Host: SERVER2022; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2022-06-07T10:38:44
|   start_date: N/A
| nbstat: NetBIOS name: SERVER2022, NetBIOS user: <unknown>, NetBIOS MAC: 1c:5a:12:d9:10:bd (unknown)
| smb2-security-mode:
|   3.1.1:
|     Message signing enabled and required
| smb-os-discovery:
|   OS: Windows Server 2022 Standard 20348 (Windows Server 2022 Standard 6.3)
|   Computer name: Server2022
|   NetBIOS computer name: SERVER2022\x00
|   Domain name: CEH.com
|   Forest name: CEH.com
|   FQDN: Server2022.CEH.com
|   System time: 2022-06-07T03:38:44-07:00
|   clock-skew: mean: 8h23m59s, deviation: 3h07m49s, median: 6h59m59s
| smb-security-mode:
|   account used: guest
|   authentication_level: user
|   challenge_response: supported
|   message_signing: required
|
TRACEROUTE
HOP RTT      ADDRESS
1  2.27 ms  10.10.1.22

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

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.

The screenshot shows a terminal window titled "nmap -O 10.10.1.22 - Parrot Terminal". The terminal output is as follows:

```
|root@parrot|~/home/attacker|
# nmap -O 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-06 23:46 EDT
Nmap scan report for 10.10.1.22
Host is up (0.0018s latency).
Not shown: 983 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
1801/tcp  open  msmq
2103/tcp  open  zephyr-clt
2105/tcp  open  eklogin
2107/tcp  open  msmq-mgmt
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
MAC Address: 1C:5A:12:D9:10:BD (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
```

TCP/IP fingerprint:

```
OS:SCAN(V=7.92%E=4%D=6/6%OT=53%CT=1%CU=37325%PV=Y%DS=1%DC=D%G=Y%M=1C5A12%TM
OS:=629ECA27%P=x86_64-pc-linux-gnu)SEQ(SP=106%GCD=1%ISR=10C%TI=I%CI=I%II=I%
OS:SS=S%TS=A)OPS(01=M5B4NW8ST11%02=M5B4NW8ST11%03=M5B4NW8NNT11%04=M5B4NW8ST
```

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.

Module 03 – Scanning Networks

The terminal window displays the output of an Nmap scan using the script `smb-os-discovery.nse` against the target IP address `10.10.1.22`. The scan identifies several open ports and their corresponding services, including domain, http, kerberos-sec, msrpc, netbios-ssn, ldap, microsoft-ds, kpasswd5, http-rpc-epmap, ldapssl, msmq, zephyr-clt, eklogin, msmq-mgmt, globalcatLDAP, globalcatLDAPssl, and ms-wbt-server. The MAC address of the host is listed as `1C:5A:12:D9:10:BD (Unknown)`. The `smb-os-discovery` script provides detailed OS information, identifying the operating system as `Windows Server 2022 Standard 20348 (Windows Server 2022 Standard 6.3)` and the computer name as `Server2022`.

```
nmap --script smb-os-discovery.nse 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
-[root@parrot]-[~/home/attacker]
-- #nmap --script smb-os-discovery.nse 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-06 23:49 EDT
Nmap scan report for 10.10.1.22
Host is up (0.14s latency).
Not shown: 983 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
1801/tcp  open  msmq
2103/tcp  open  zephyr-clt
2105/tcp  open  eklogin
2107/tcp  open  msmq-mgmt
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
MAC Address: 1C:5A:12:D9:10:BD (Unknown)

Host script results:
smb-os-discovery:
OS: Windows Server 2022 Standard 20348 (Windows Server 2022 Standard 6.3)
Computer name: Server2022

nmap --script smb-os-discovery.nse 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
1801/tcp  open  msmq
2103/tcp  open  zephyr-clt
2105/tcp  open  eklogin
2107/tcp  open  msmq-mgmt
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
MAC Address: 1C:5A:12:D9:10:BD (Unknown)

Host script results:
smb-os-discovery:
OS: Windows Server 2022 Standard 20348 (Windows Server 2022 Standard 6.3)
Computer name: Server2022
NetBIOS computer name: SERVER2022\x00
Domain name: CEH.com
Forest name: CEH.com
FQDN: Server2022.CEH.com
System time: 2022-06-07T03:49:25-07:00

Nmap done: 1 IP address (1 host up) scanned in 2.20 seconds
```

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.

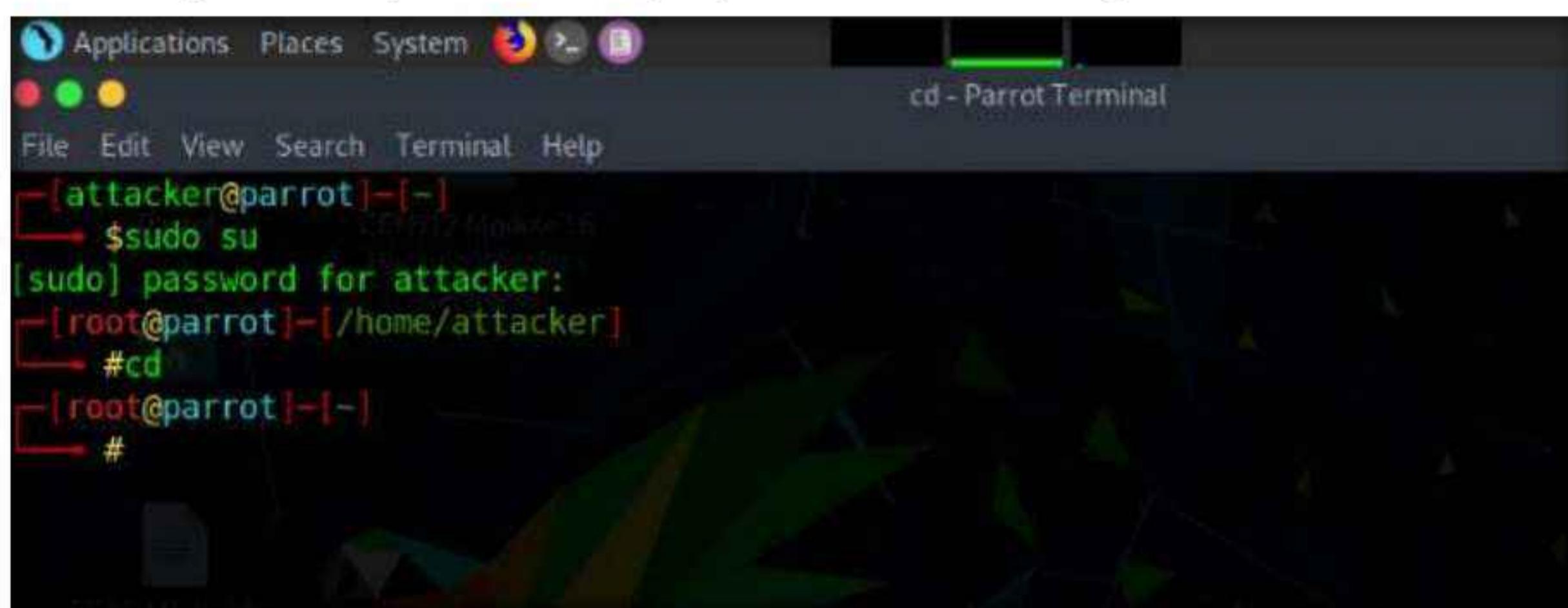
Task 3: Perform OS Discovery using Unicornscan

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.

Note: Ensure that the **Windows Server 2022** virtual machine is running.

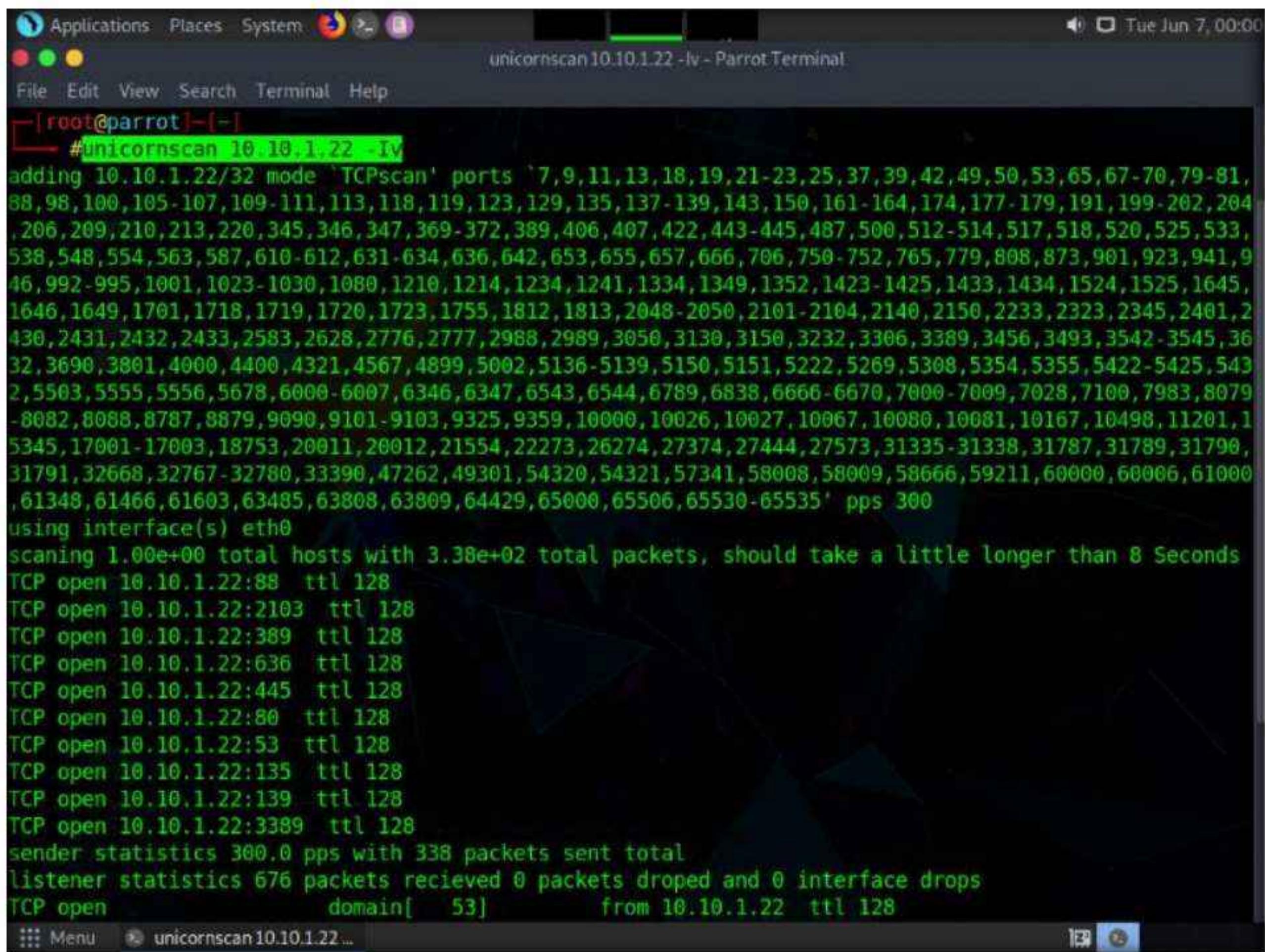
1. Turn on the **Ubuntu** virtual machine.
2. Switch to the **Parrot Security** virtual machine and click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.
3. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run 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.



The screenshot shows a terminal window titled "cd - Parrot Terminal". The window has a dark background with green text. The terminal session starts with the prompt "[attacker@parrot]~" followed by "\$sudo su". The user is prompted for a password with "[sudo] password for attacker:". After entering the password "toor", the prompt changes to "[root@parrot]~" followed by "#cd". Finally, the prompt changes to "[root@parrot]~" followed by "#".

6. In the terminal window, type **unicornscan [Target IP Address] -lv** (here, the target machine is **Windows Server 2022 [10.10.1.22]**) and press **Enter**.
Note: In this command, **-l** specifies an immediate mode and **v** specifies a verbose mode.
7. 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)**.



The screenshot shows a terminal window titled "unicornscan 10.10.1.22 -lv - Parrot Terminal". The terminal is running as root on a Parrot OS system. The user has run the command "#unicornscan 10.10.1.22 -lv". The output of the scan is displayed in green text. It shows that the scan mode is 'TCPscan' and it's scanning ports 7 through 65535. The scan is using interface eth0 and will take about 8 seconds. It lists several open TCP ports with their TTL values: 10.10.1.22:88 (TTL 128), 10.10.1.22:2103 (TTL 128), 10.10.1.22:389 (TTL 128), 10.10.1.22:636 (TTL 128), 10.10.1.22:445 (TTL 128), 10.10.1.22:80 (TTL 128), 10.10.1.22:53 (TTL 128), 10.10.1.22:135 (TTL 128), 10.10.1.22:139 (TTL 128), and 10.10.1.22:3389 (TTL 128). The output also includes sender and listener statistics.

```
#unicornscan 10.10.1.22 -lv
adding 10.10.1.22/32 mode 'TCPscan' ports 7,9,11,13,18,19,21-23,25,37,39,42,49,50,53,65,67-70,79-81,88,98,100,105-107,109-111,113,118,119,123,129,135,137-139,143,150,161-164,174,177-179,191,199-202,204,206,209,210,213,220,345,346,347,369-372,389,406,407,422,443-445,487,500,512-514,517,518,520,525,533,538,548,554,563,587,610-612,631-634,636,642,653,655,657,666,706,750-752,765,779,808,873,901,923,941,946,992-995,1001,1023-1030,1080,1210,1214,1234,1241,1334,1349,1352,1423-1425,1433,1434,1524,1525,1645,1646,1649,1701,1718,1719,1720,1723,1755,1812,1813,2048-2050,2101-2104,2140,2150,2233,2323,2345,2401,2430,2431,2432,2433,2583,2628,2776,2777,2988,2989,3050,3130,3150,3232,3306,3389,3456,3493,3542-3545,3632,3690,3801,4000,4400,4321,4567,4899,5002,5136-5139,5150,5151,5222,5269,5308,5354,5355,5422-5425,5432,5503,5555,5556,5678,6000-6007,6346,6347,6543,6544,6789,6838,6666-6670,7000-7009,7028,7100,7983,8079-8082,8088,8787,8879,9090,9101-9103,9325,9359,10000,10026,10027,10067,10080,10081,10167,10498,11201,115345,17001-17003,18753,20011,20012,21554,22273,26274,27374,27444,27573,31335-31338,31787,31789,31790,31791,32668,32767-32780,33390,47262,49301,54320,54321,57341,58008,58009,58666,59211,60000,60006,61000,61348,61466,61603,63485,63808,63809,64429,65000,65506,65530-65535* pps 300
using interface(s) eth0
scanning 1.00e+00 total hosts with 3.38e+02 total packets, should take a little longer than 8 Seconds
TCP open 10.10.1.22:88 ttl 128
TCP open 10.10.1.22:2103 ttl 128
TCP open 10.10.1.22:389 ttl 128
TCP open 10.10.1.22:636 ttl 128
TCP open 10.10.1.22:445 ttl 128
TCP open 10.10.1.22:80 ttl 128
TCP open 10.10.1.22:53 ttl 128
TCP open 10.10.1.22:135 ttl 128
TCP open 10.10.1.22:139 ttl 128
TCP open 10.10.1.22:3389 ttl 128
sender statistics 300.0 pps with 338 packets sent total
listener statistics 676 packets received 0 packets dropped and 0 interface drops
TCP open domain[ 53] from 10.10.1.22 ttl 128
::: Menu unicornscan10.10.1.22...
```

8. In the **Parrot Terminal** window, type **unicornscan [Target IP Address] -lv** (here, the target machine is **Ubuntu [10.10.1.9]**) and press **Enter**.
9. 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.

```

Applications Places System unicornscan10.10.1.9 -l - Parrot Terminal
File Edit View Search Terminal Help
TCP open      ldap[ 389]      from 10.10.1.22 ttl 128
TCP open      microsoft-ds[ 445]      from 10.10.1.22 ttl 128
TCP open      ldaps[ 636]      from 10.10.1.22 ttl 128
TCP open      zephyr-clt[ 2103]      from 10.10.1.22 ttl 128
TCP open      ms-wbt-server[ 3389]      from 10.10.1.22 ttl 128
[root@parrot]~[~]
#unicornscan 10.10.1.9 -l
adding 10.10.1.9/32 mode 'TCPscan' ports '7,9,11,13,18,19,21-23,25,37,39,42,49,50,53,65,67-70,79-81,8
8,98,100,105-107,109-111,113,118,119,123,129,135,137-139,143,150,161-164,174,177-179,191,199-202,204,
206,209,210,213,220,345,346,347,369-372,389,406,407,422,443-445,487,500,512-514,517,518,520,525,533,5
38,548,554,563,587,610-612,631-634,636,642,653,655,657,666,706,750-752,765,779,808,873,901,923,941,94
6,992-995,1001,1023-1030,1080,1210,1214,1234,1241,1334,1349,1352,1423-1425,1433,1434,1524,1525,1645,1
646,1649,1701,1718,1719,1720,1723,1755,1812,1813,2048-2050,2101-2104,2140,2150,2233,2323,2345,2401,24
30,2431,2432,2433,2583,2628,2776,2777,2988,2989,3050,3130,3150,3232,3306,3389,3456,3493,3542-3545,363
2,3690,3801,4000,4400,4321,4567,4899,5002,5136-5139,5150,5151,5222,5269,5308,5354,5355,5422-5425,5432
,5503,5555,5556,5678,6000-6007,6346,6347,6543,6544,6789,6838,6666-6670,7000-7009,7028,7100,7983,8079-
8082,8088,8787,8879,9090,9101-9103,9325,9359,10000,10026,10027,10067,10080,10081,10167,10498,11201,15
345,17001-17003,18753,20011,20012,21554,22273,26274,27374,27444,27573,31335-31338,31787,31789,31790,3
1791,32668,32767-32780,33390,47262,49301,54320,54321,57341,58008,58009,58666,59211,60000,60006,61000,
61348,61466,61603,63485,63808,63809,64429,65000,65506,65530-65535' pps 300
using interface(s) eth0
scanning 1.00e+00 total hosts with 3.38e+02 total packets, should take a little longer than 8 Seconds
TCP open 10.10.1.9:22 ttl 64
TCP open 10.10.1.9:80 ttl 64
sender statistics 0.5 pps with 338 packets sent total
listener statistics 676 packets received 0 packets dropped and 0 interface drops
TCP open      ssh[ 22]      from 10.10.1.9 ttl 64
TCP open      http[ 80]      from 10.10.1.9 ttl 64
[root@parrot]~[~]
#
```

10. This concludes the demonstration of discovering the OS of the target machine using Unicornscan.
11. Close all open windows and document all the acquired information.
12. Turn off the **Parrot Security**, **Windows Server 2022** and **Ubuntu** virtual machines.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required

Yes

No

Platform Supported

Classroom

CyberQ

Lab**4**

Scan beyond IDS and Firewall

Scanning beyond IDS and firewall is a process of sending intended packets to the target system in order to exploit IDS/firewall limitations.

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
- Browse anonymously using Proxy Switcher
- Browse anonymously using CyberGhost VPN

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 40 Minutes

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

Lab Tasks

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.

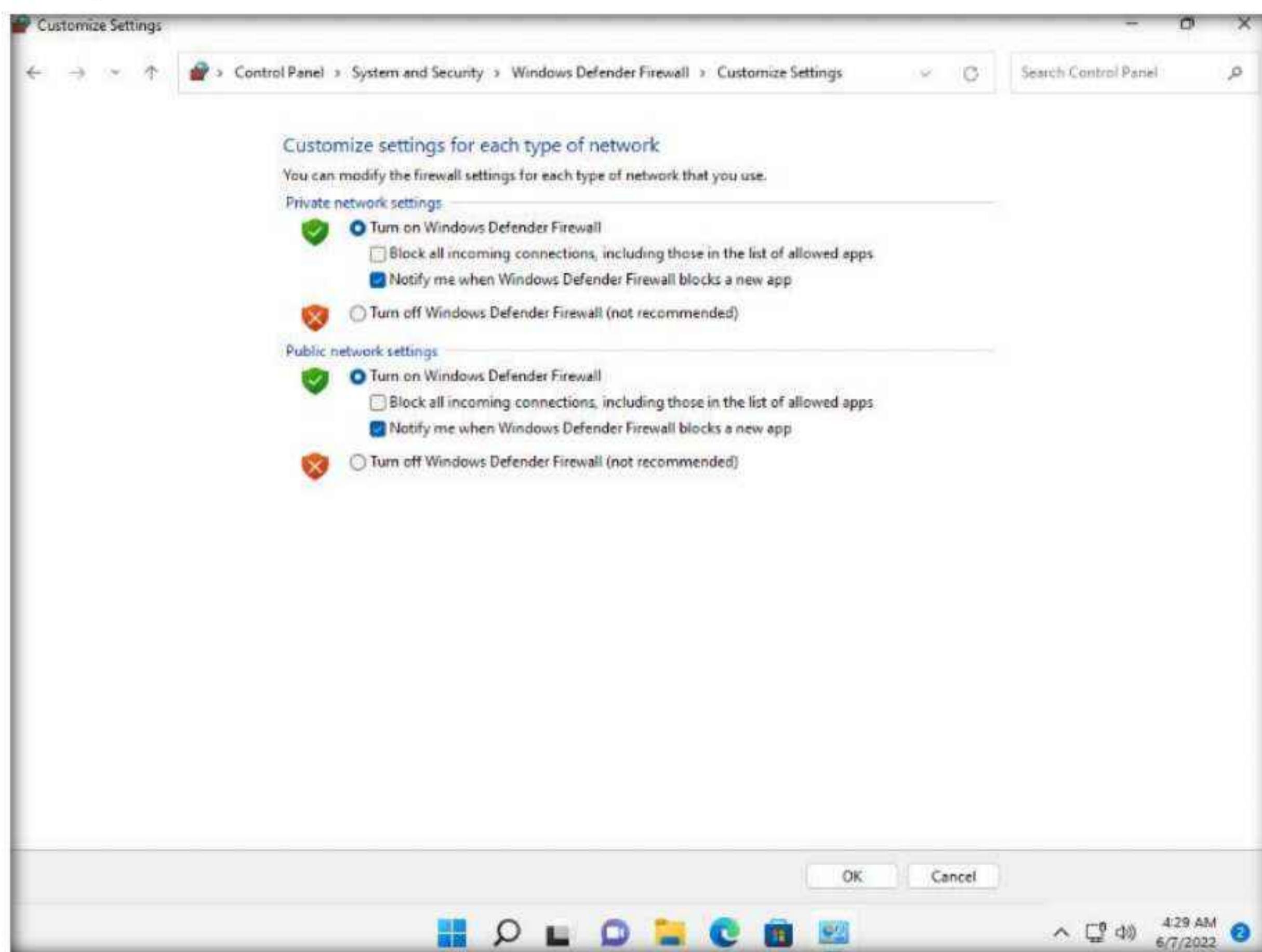
1. Turn on the **Windows 11** and **Parrot Security** virtual machines.

2. Switch to the **Windows 11** virtual machine. By default, **Admin** user profile is selected, type **Pa\$\$w0rd** in the **Password** field and press **Enter** to login.

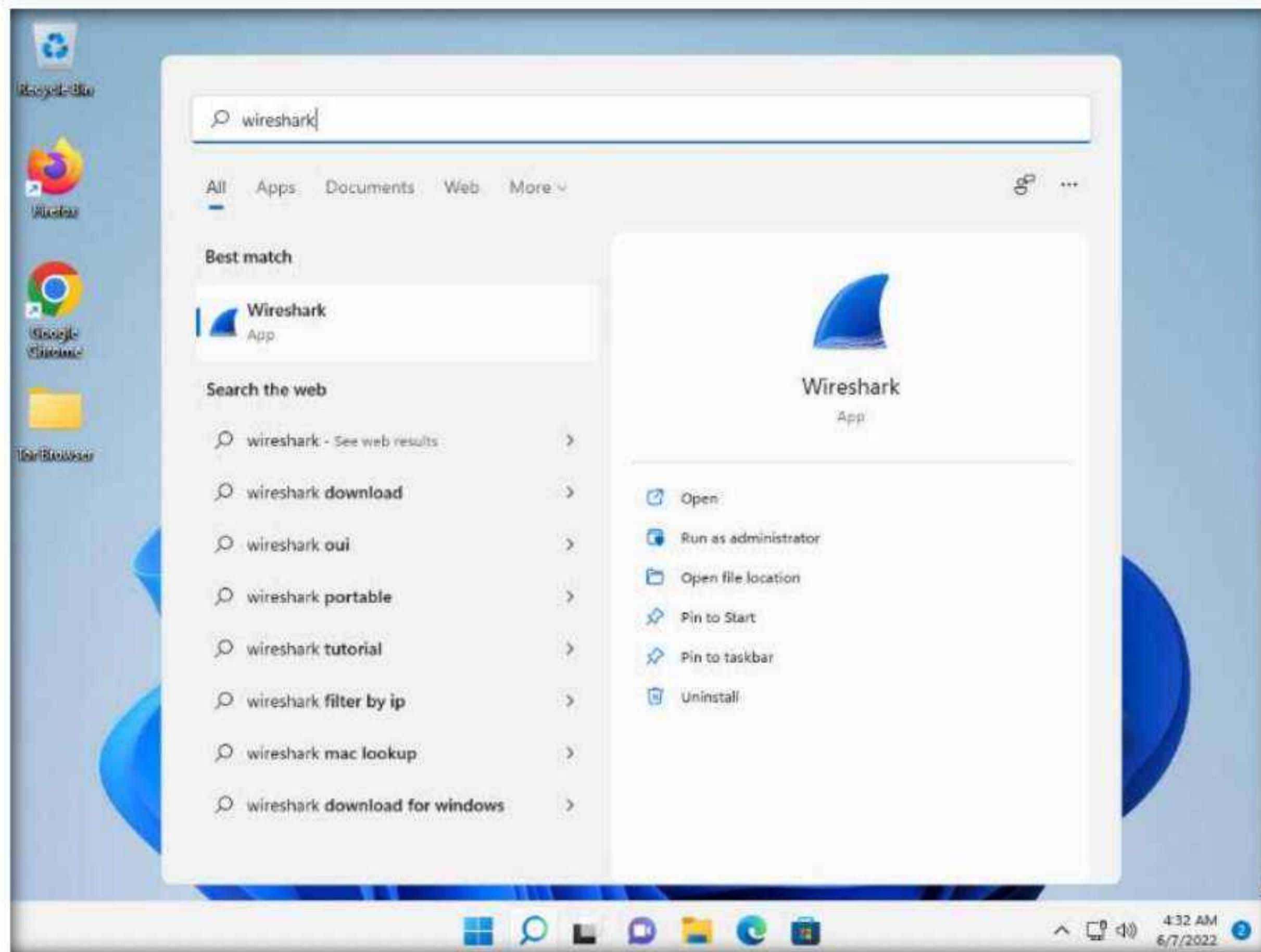
Note: If **Welcome to Windows** wizard appears, click **Continue** and in **Sign in with Microsoft** wizard, click **Cancel**.

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

3. 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.

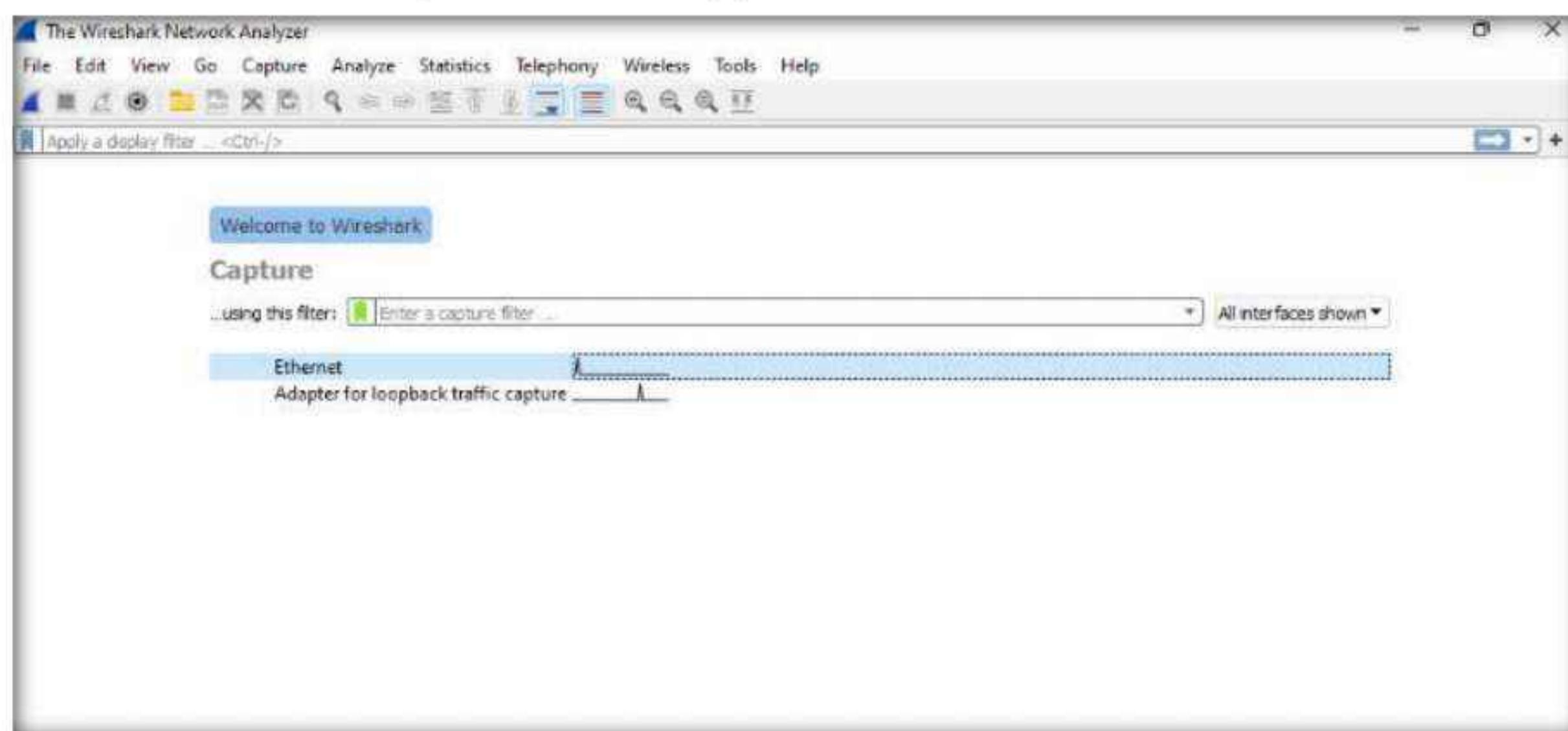


4. Minimize the Control Panel window, click **Search icon** (🔍) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Open** to launch it.



5. 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**.



6. Switch to the **Parrot Security** virtual machine.
7. 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.

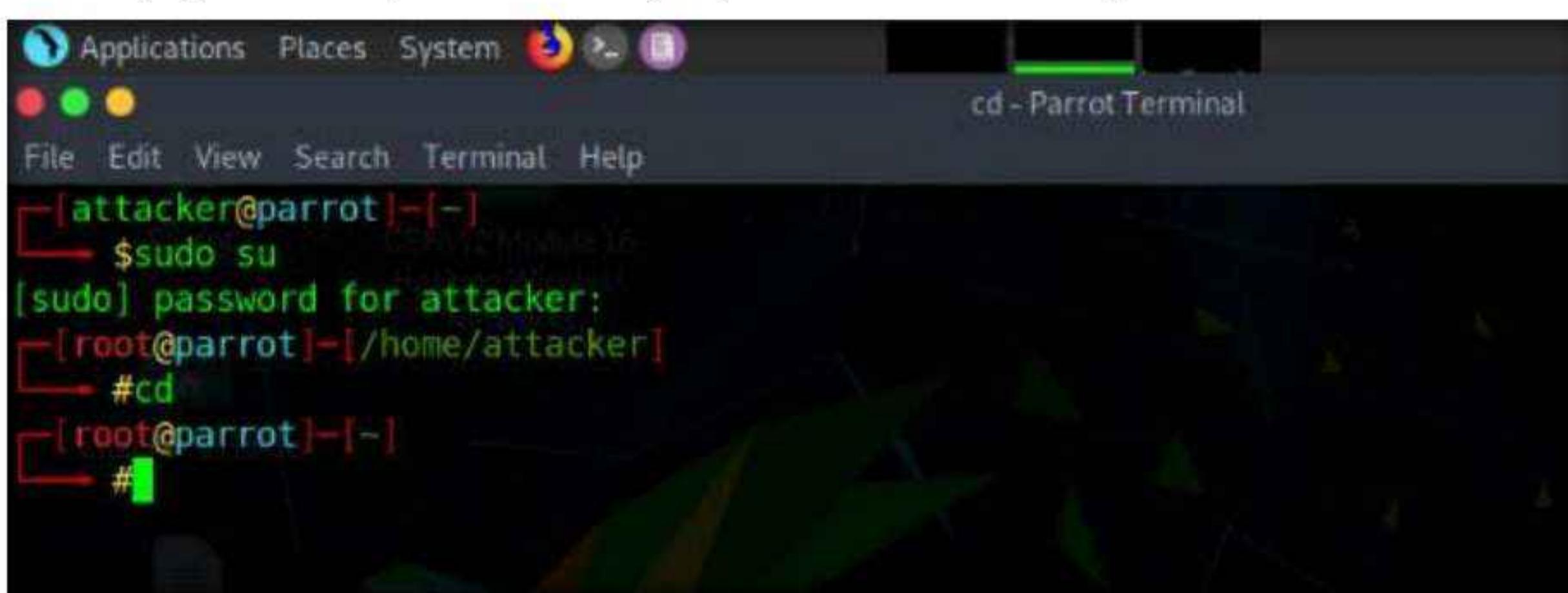
Note: If a **Parrot Updater** pop-up appears at the top-right corner of **Desktop**, ignore and close it.

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

8. Click the **MATE Terminal** icon in the top-left corner of the **Desktop** to open a **Terminal** window.
9. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.
10. 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.

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



The screenshot shows a terminal window titled "cd - Parrot Terminal". The window has a dark background with white text. At the top, there's a menu bar with "Applications", "Places", "System", and other icons. Below the menu, the terminal prompt is "[attacker@parrot]~[-]". The user types "\$sudo su" and presses Enter. A password prompt "[sudo] password for attacker:" appears. The user types "toor" and presses Enter. The terminal then shows "[root@parrot]~[-]/home/attacker" and "#cd". Finally, the user types "#" and presses Enter, resulting in the prompt "[root@parrot]~[-]" followed by another "#".

12. 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.

13. 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.

```

nmap -f 10.10.1.11 - Parrot Terminal

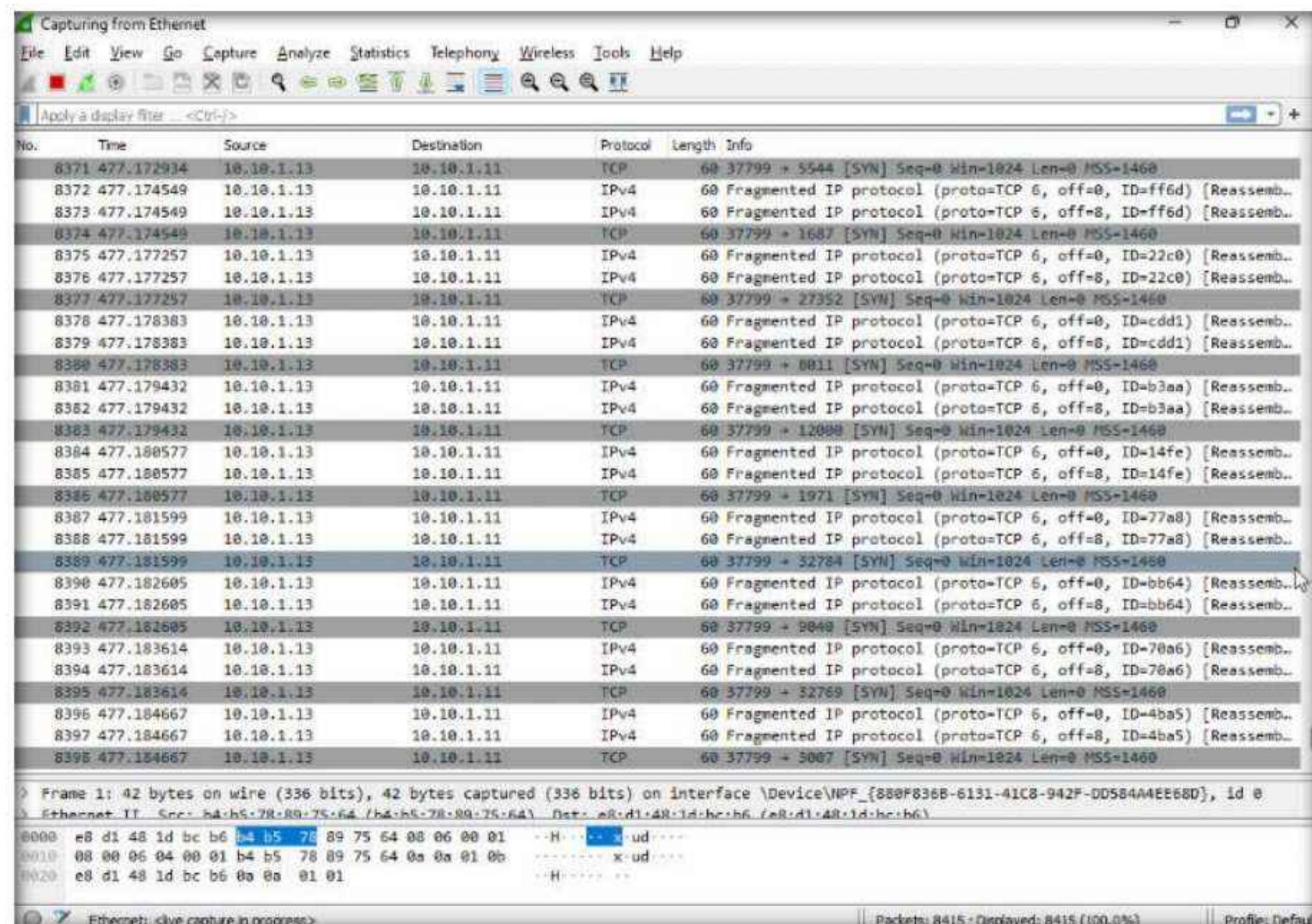
File Edit View Search Terminal Help
[attacker@parrot]~[-]
└─$ sudo su
[sudo] password for attacker:
[root@parrot]~[~/home/attacker]
└─# cd
[root@parrot]~[-]
└─# nmap -f 10.10.1.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-07 00:42 EDT
Nmap scan report for 10.10.1.11
Host is up (0.049s latency).

Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
MAC Address: B4:B5:78:89:75:64 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 9.61 seconds
[root@parrot]~[-]
└─#

```

14. Switch to the **Windows 11** virtual machine (target machine). You can observe the fragmented packets captured by the Wireshark, as shown in the screenshot.



15. Switch to the **Parrot Security** virtual machine.

16. 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.

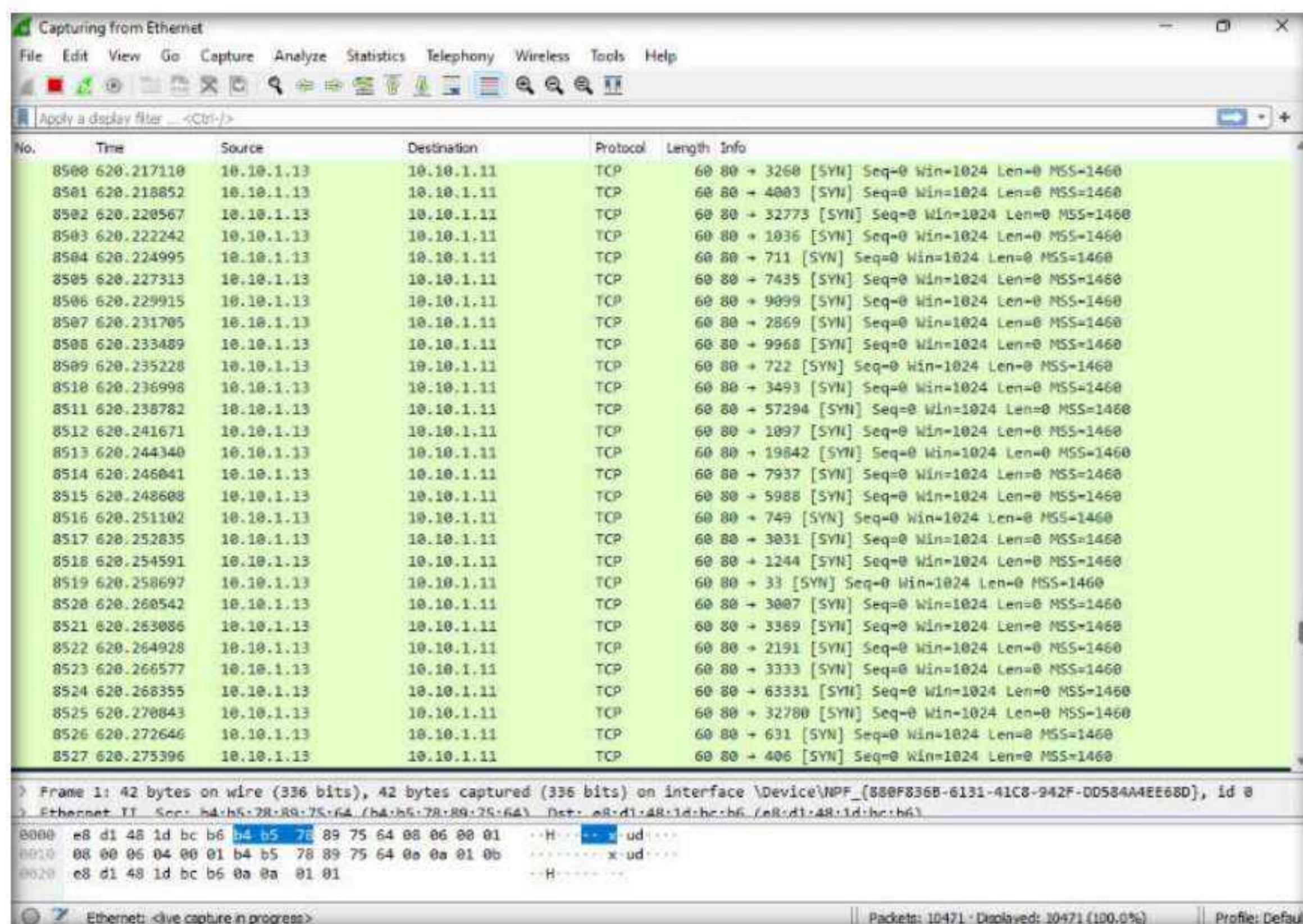
Note: Source port manipulation refers to manipulating actual port numbers with common port numbers to evade IDS/firewall: this is useful when the firewall is configured to allow packets from well-known ports like HTTP, DNS, FTP, etc.

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

```
[root@parrot]#[-]
[root@parrot]# nmap -g 80 10.10.1.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-07 00:45 EDT
Nmap scan report for 10.10.1.11
Host is up (0.039s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
MAC Address: B4:B5:78:89:75:64 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 7.11 seconds
```

18. Switch to the **Windows 11** virtual 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.



19. Switch to the **Parrot Security** virtual machine.
20. Now, type **nmap -mtu 8 [Target IP Address]** (here, target IP address is **10.10.1.11**) and press **Enter**.

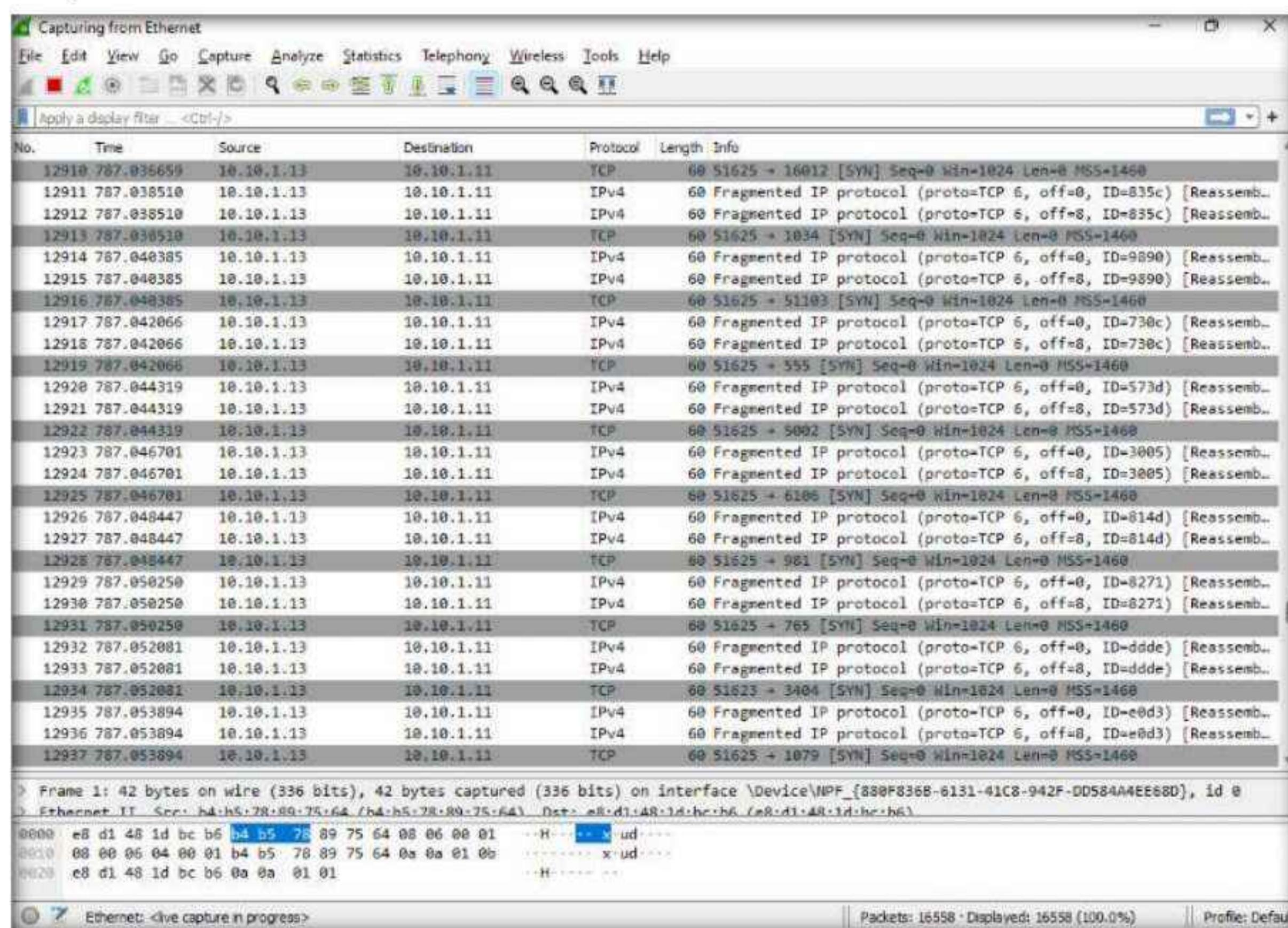
Note: In this command, **-mtu**: specifies the number of Maximum Transmission Unit (MTU) (here, **8** bytes of packets).

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.

```
root@parrot:~# nmap -mtu 8 10.10.1.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-07 00:48 EDT
Nmap scan report for 10.10.1.11
Host is up (0.042s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
MAC Address: B4:B5:78:89:75:64 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 9.71 seconds
```

21. Switch to the virtual **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.



22. Switch to the **Parrot Security** virtual machine.
23. 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 random and non-reserved IP addresses (here, **10**).

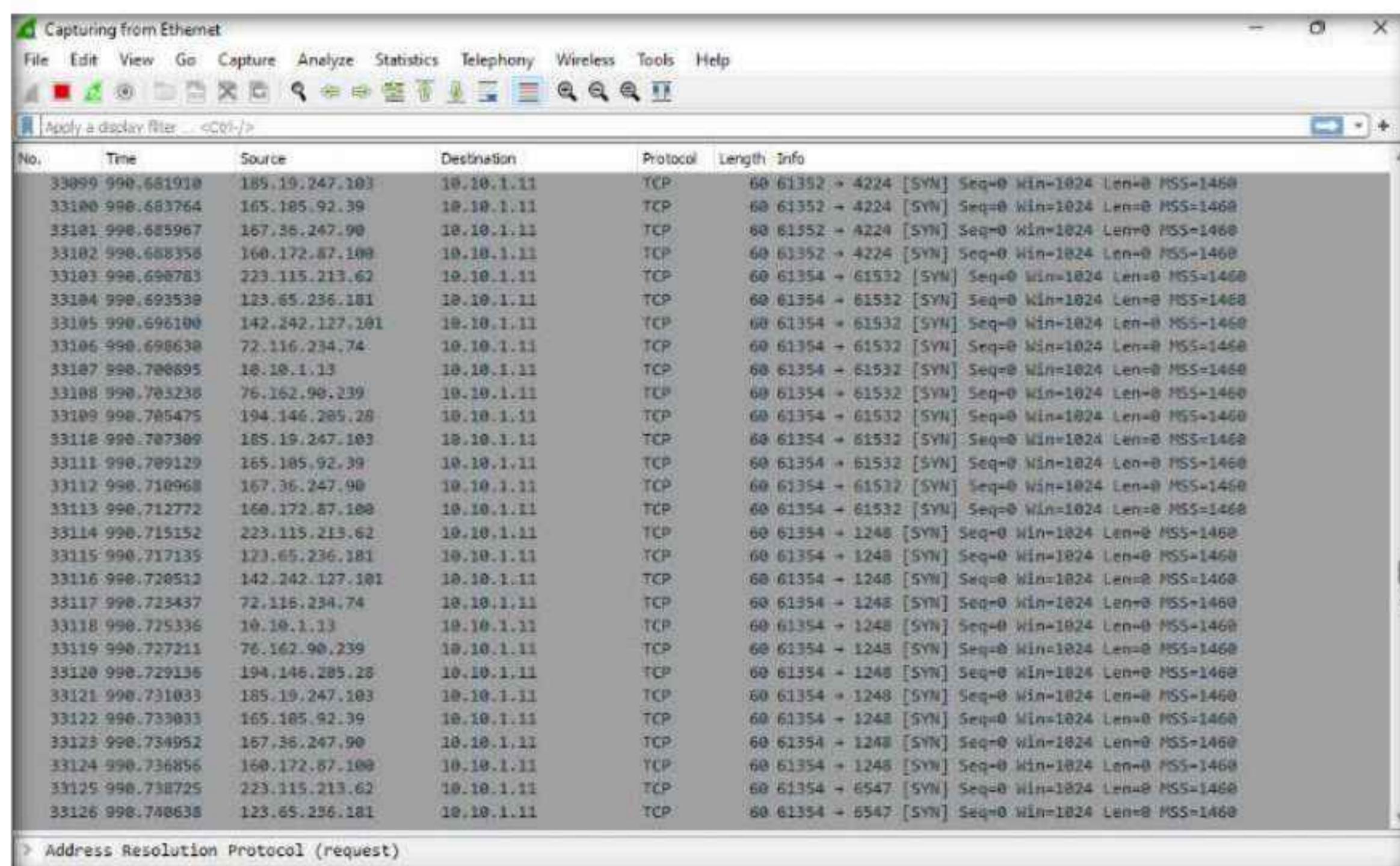
Note: The IP address decoy technique refers to generating or manually specifying IP addresses of the decoys to evade IDS/firewall. This technique makes it difficult for the IDS/firewall to determine which IP address was actually scanning the network and which IP addresses were decoys.

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.

```
[root@parrot] ~
# nmap -D RND:10 10.10.1.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-07 00:51 EDT
Nmap scan report for 10.10.1.11
Host is up (0.38s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
MAC Address: B4:85:78:89:75:64 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 68.76 seconds
```

24. Now, switch to the **Windows 11** virtual 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.



25. Switch to the **Parrot Security** virtual machine.
26. 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.

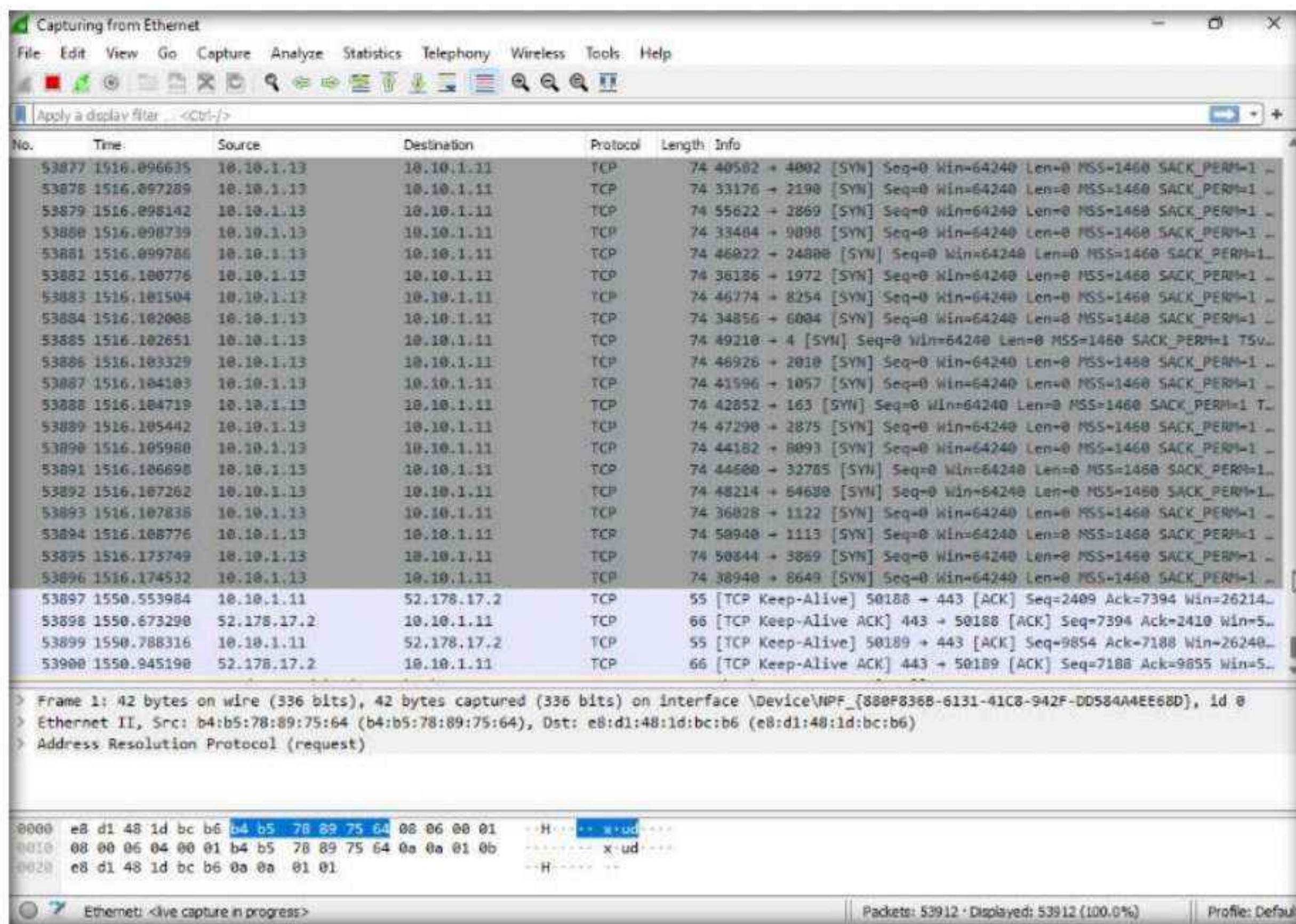
```
nmap -sT -Pn --spoof-mac 0 10.10.1.11 - Parrot Terminal

Host is up (0.38s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
MAC Address: B4:85:78:89:75:64 (Unknown)

Nmap done; 1 IP address (1 host up) scanned in 68.76 seconds
-[root@parrot]-[~]
└─# nmap -sT -Pn --spoof-mac 0 10.10.1.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-07 01:00 EDT
Spoofing MAC address AD:22:E0:B0:C8:53 (No registered vendor)
You have specified some options that require raw socket access.
These options will not be honored for TCP Connect scan.
Nmap scan report for 10.10.1.11
Host is up (0.0034s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server

Nmap done; 1 IP address (1 host up) scanned in 4.60 seconds
-[root@parrot]-[~]
└─#
```

27. Switch to the **Windows 11** virtual machine (target machine). In the Wireshark window, scroll-down and you can observe the captured TCP, as shown in the screenshot.



28. This concludes the demonstration of evading IDS and firewall using various evasion techniques in Nmap.

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

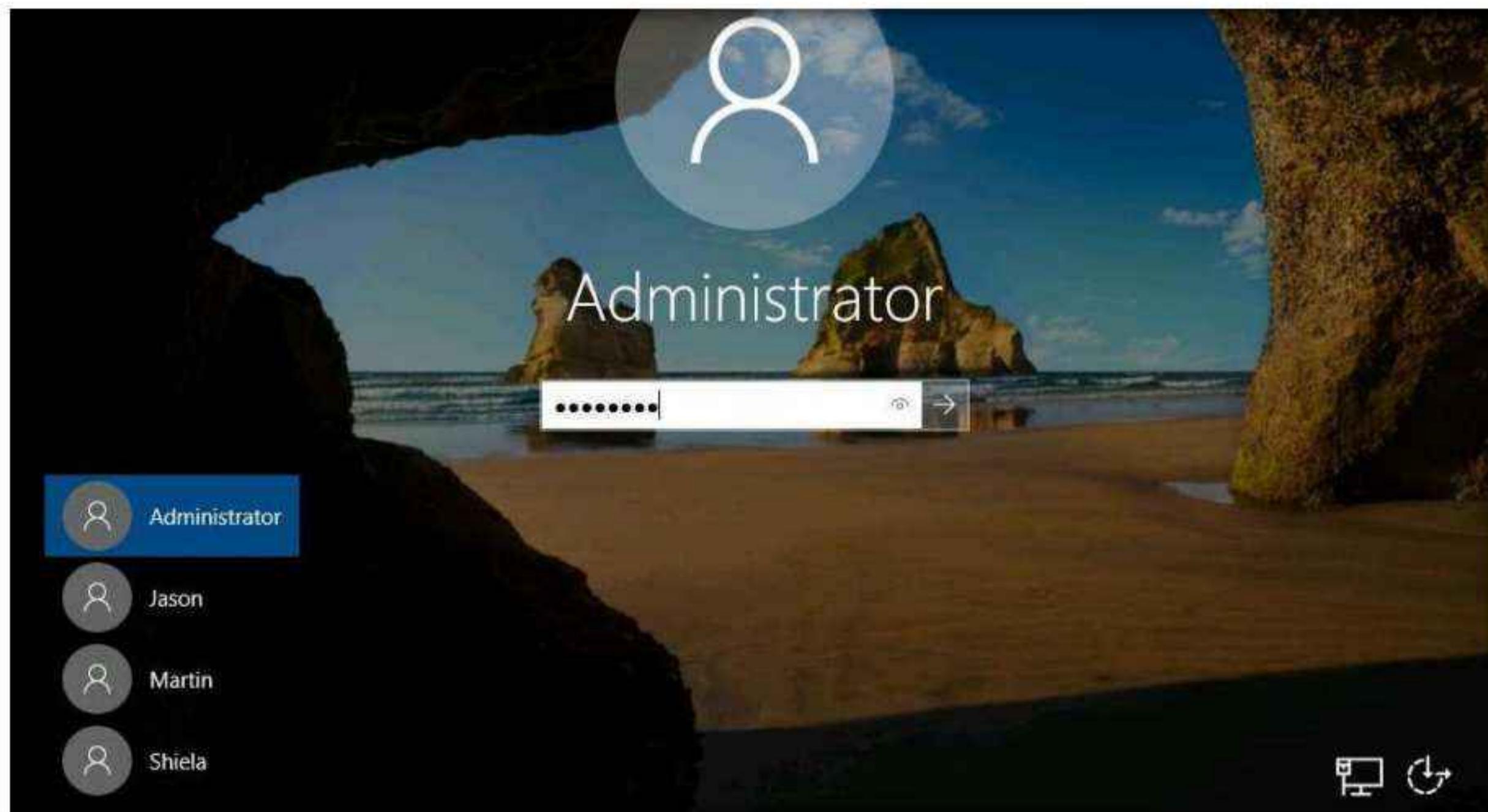
Task 2: Create Custom Packets using Colasoft Packet Builder to Scan beyond the IDS/Firewall

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.

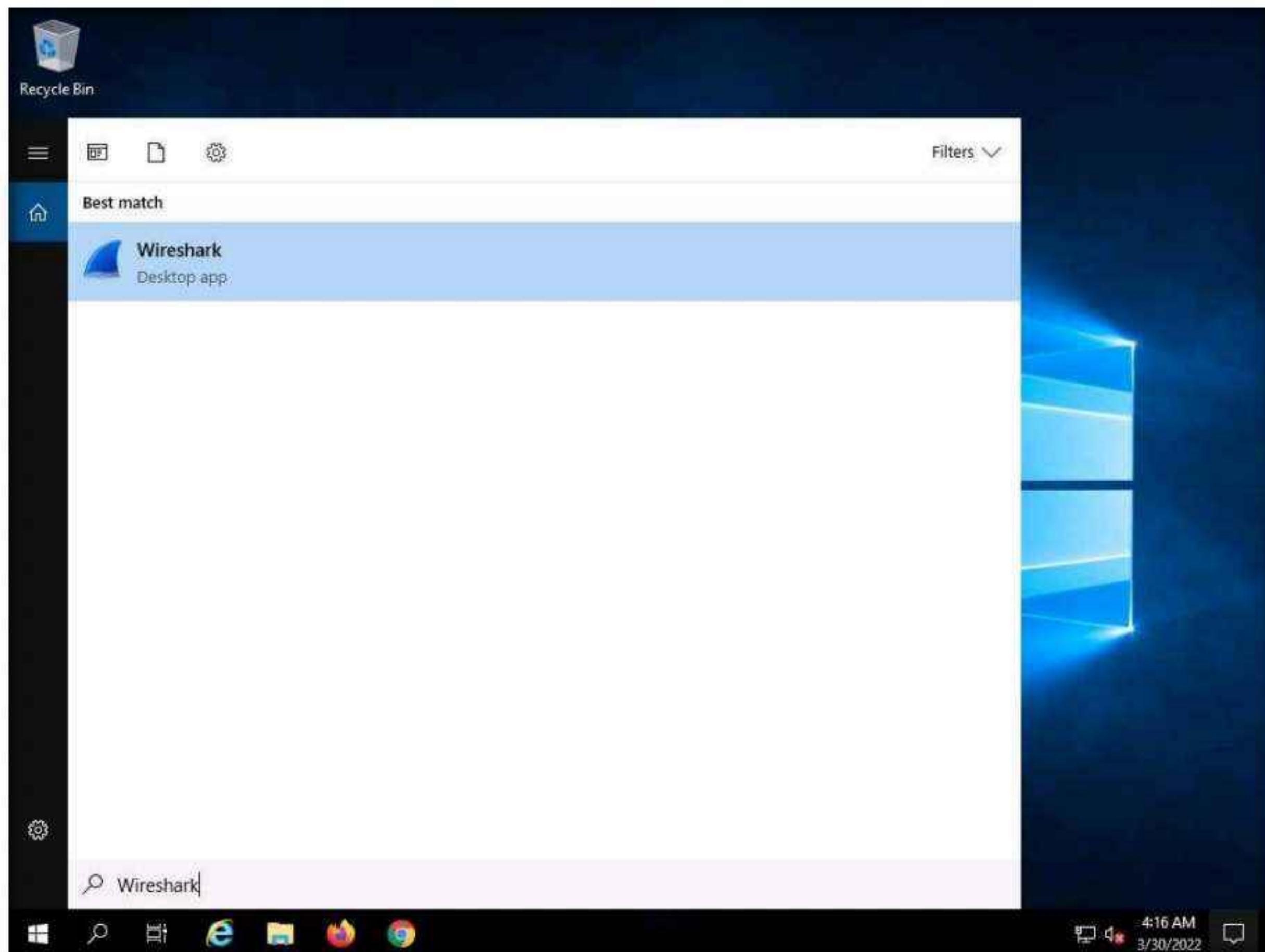
Here, we will use the Colasoft Packet Builder tool to create custom TCP packets to scan the target host by bypassing the IDS/firewall.

1. Turn on the **Windows Server 2019** virtual machine.
2. In the **Windows Server 2019** virtual machine, 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.

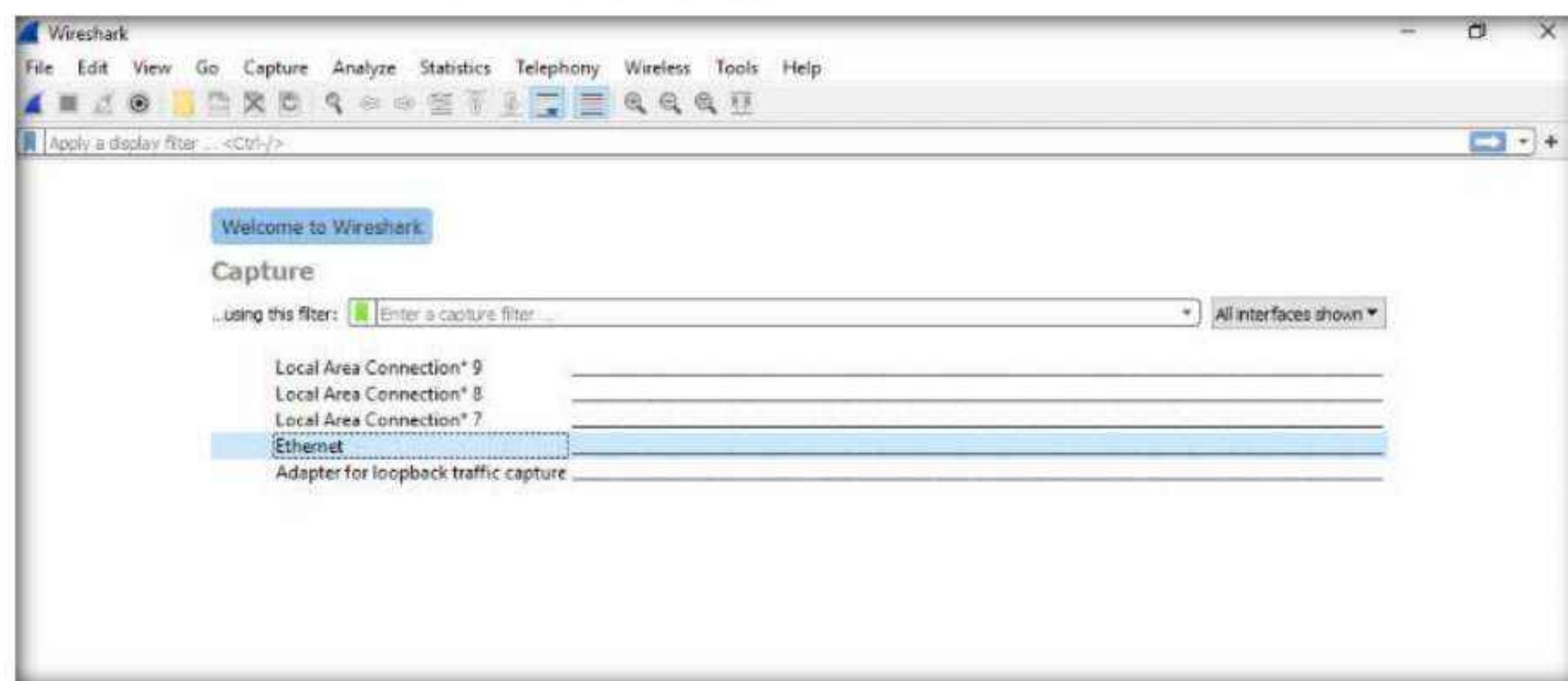


3. Click **Search** icon (🔍) on the **Desktop**. Type **wireshark** in the search field, the **Wireshark** appears in the results, click **Wireshark** to launch it.

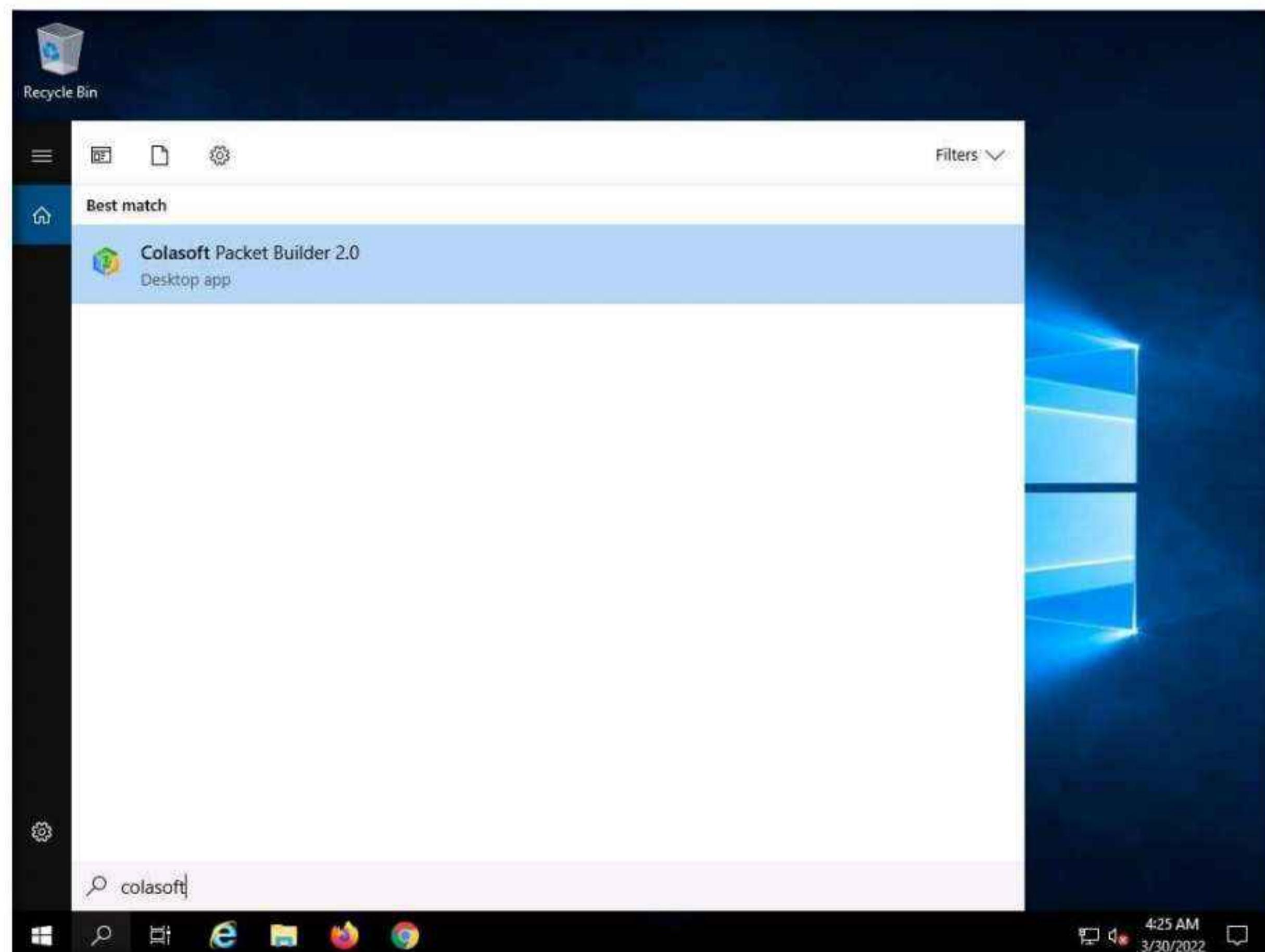


4. The **Wireshark Network Analyzer** main window appears; double-click the available ethernet or interface (here, **Ethernet**) to start the packet capture.

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



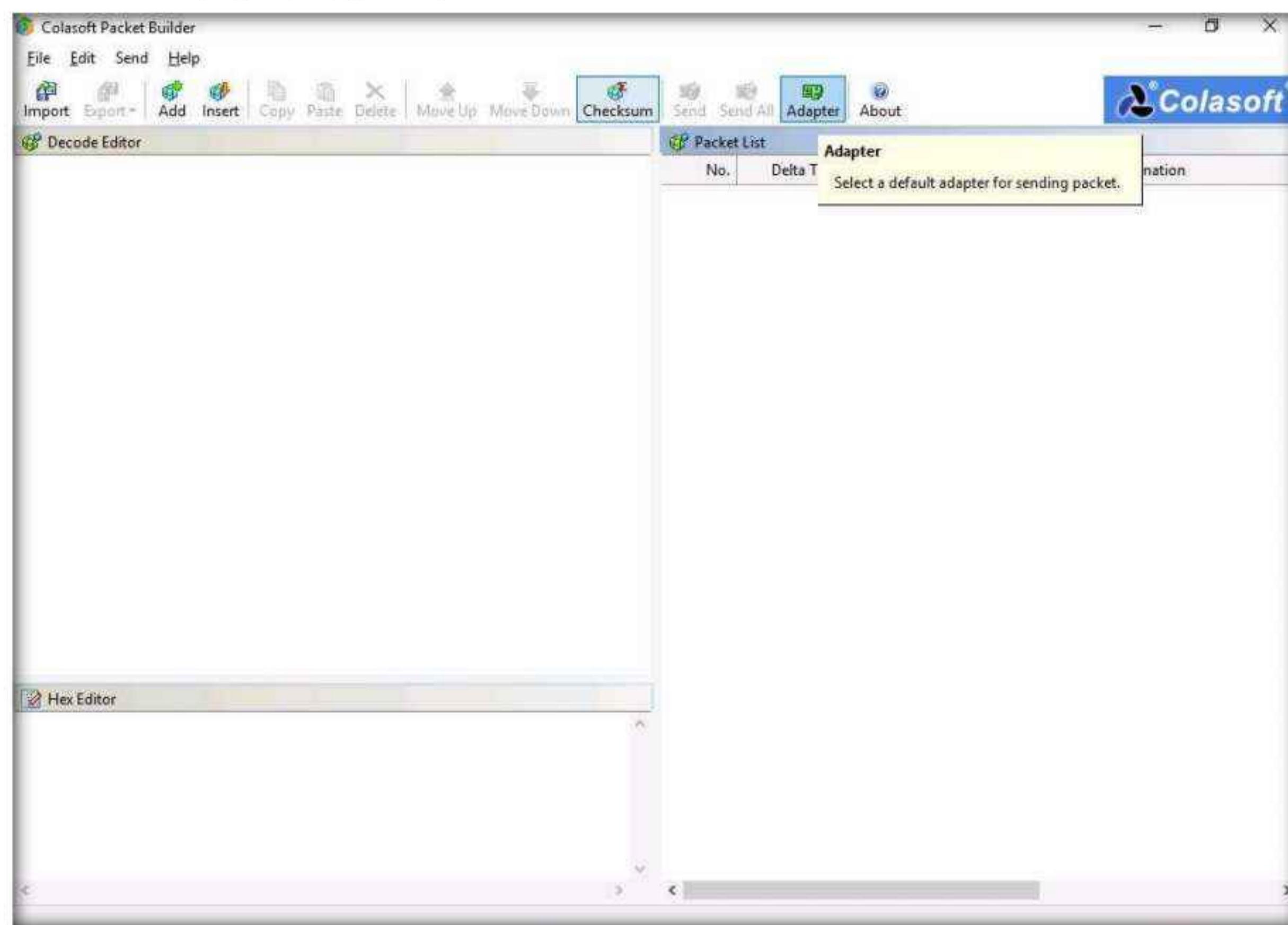
5. Minimize the **Wireshark** window, click **Search icon** (🔍) on the **Desktop**. Type **colasoft** in the search field, the **Colasoft Packet Builder 2.0** appears in the results, click **Colasoft Packet Builder 2.0** to launch it.



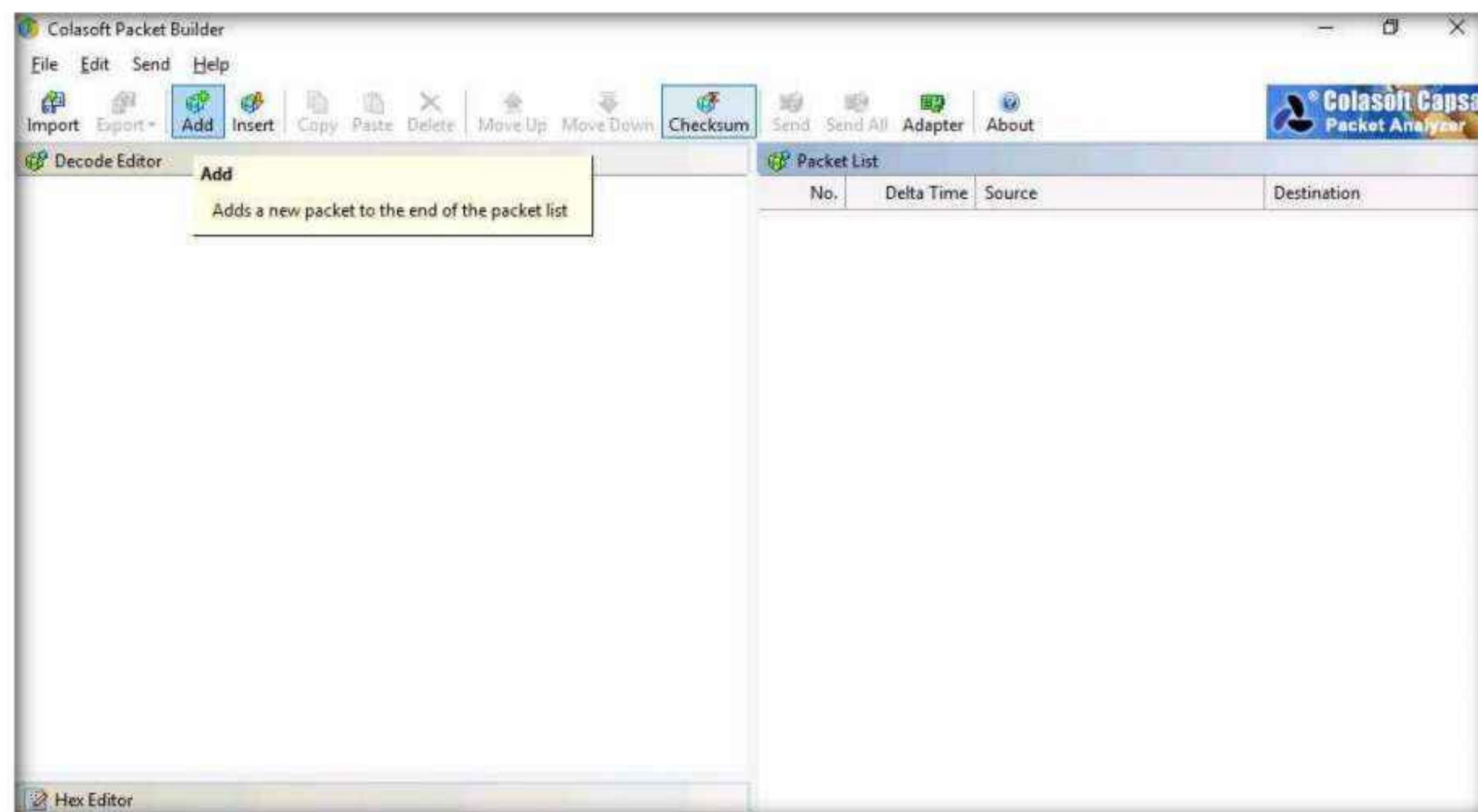
Module 03 – Scanning Networks

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

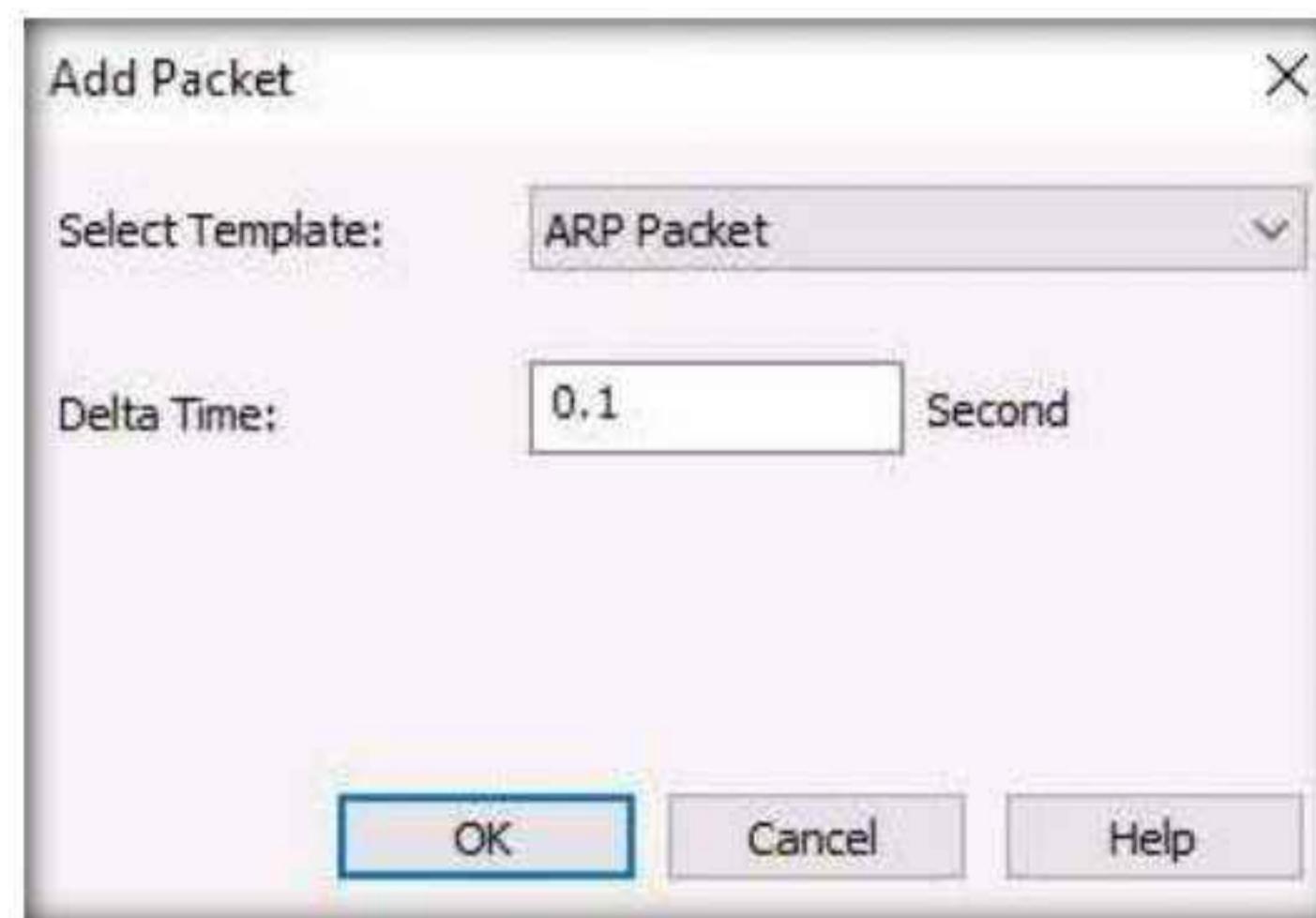
Note: If a pop-up appears, close the window.



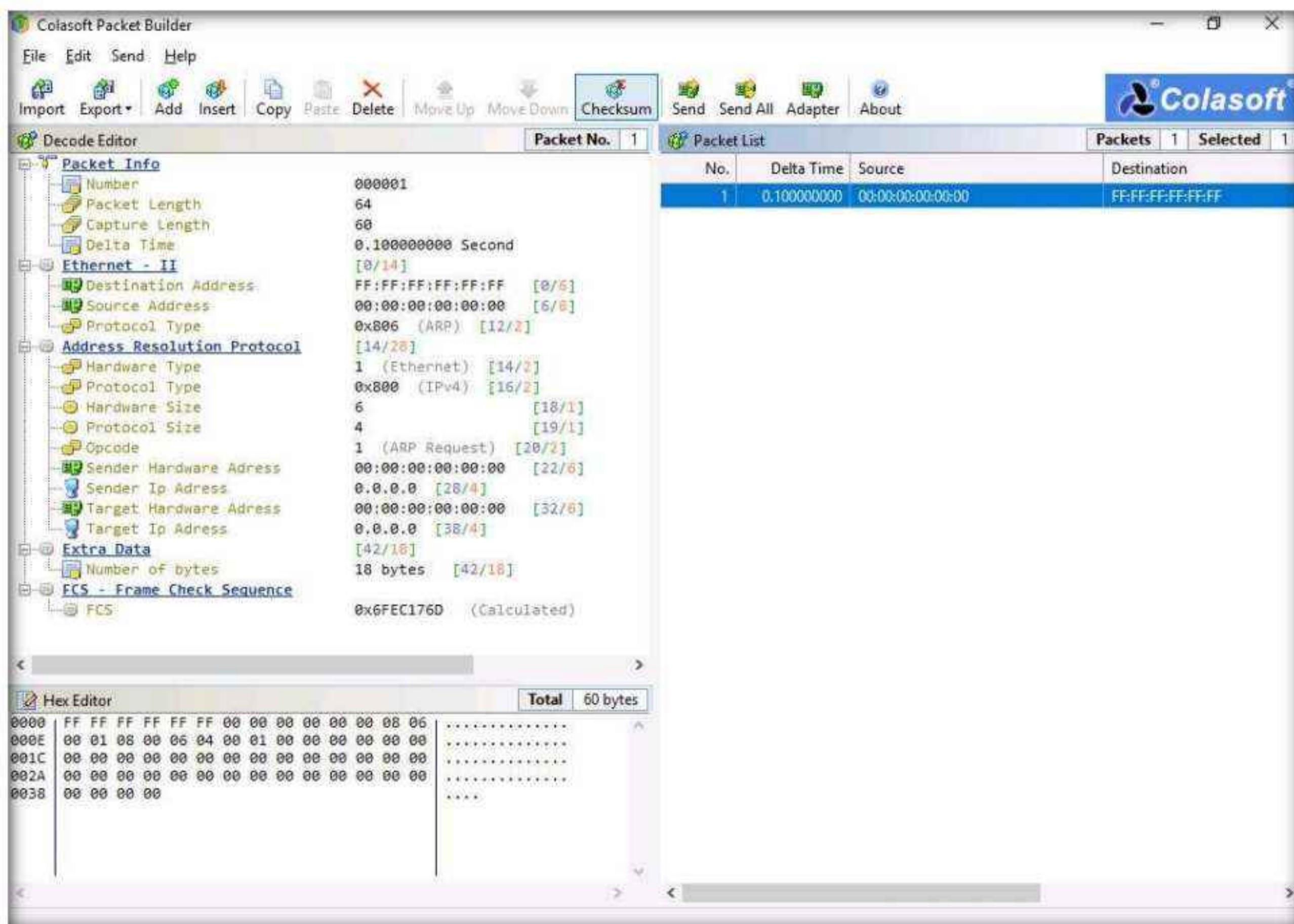
7. When the **Select Adapter** window appears, check the **Adapter** settings and click **OK**.
8. To add or create a packet, click the **Add** icon in the **Menu** bar.



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**.

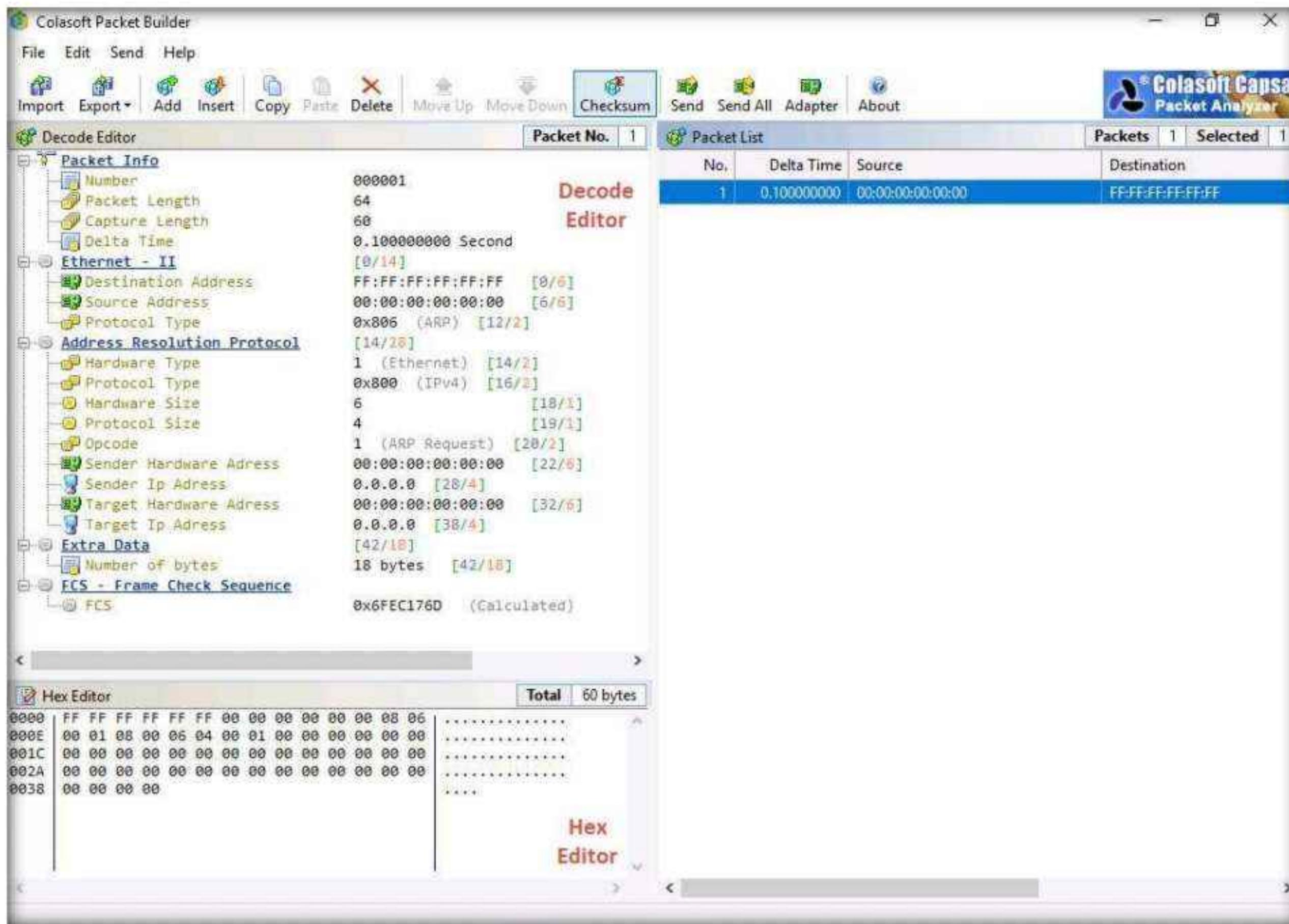


11. Colasoft Packet Builder allows you to edit the decoding information in the two editors, **Decode Editor** and **Hex Editor**, located in the left pane of the window.

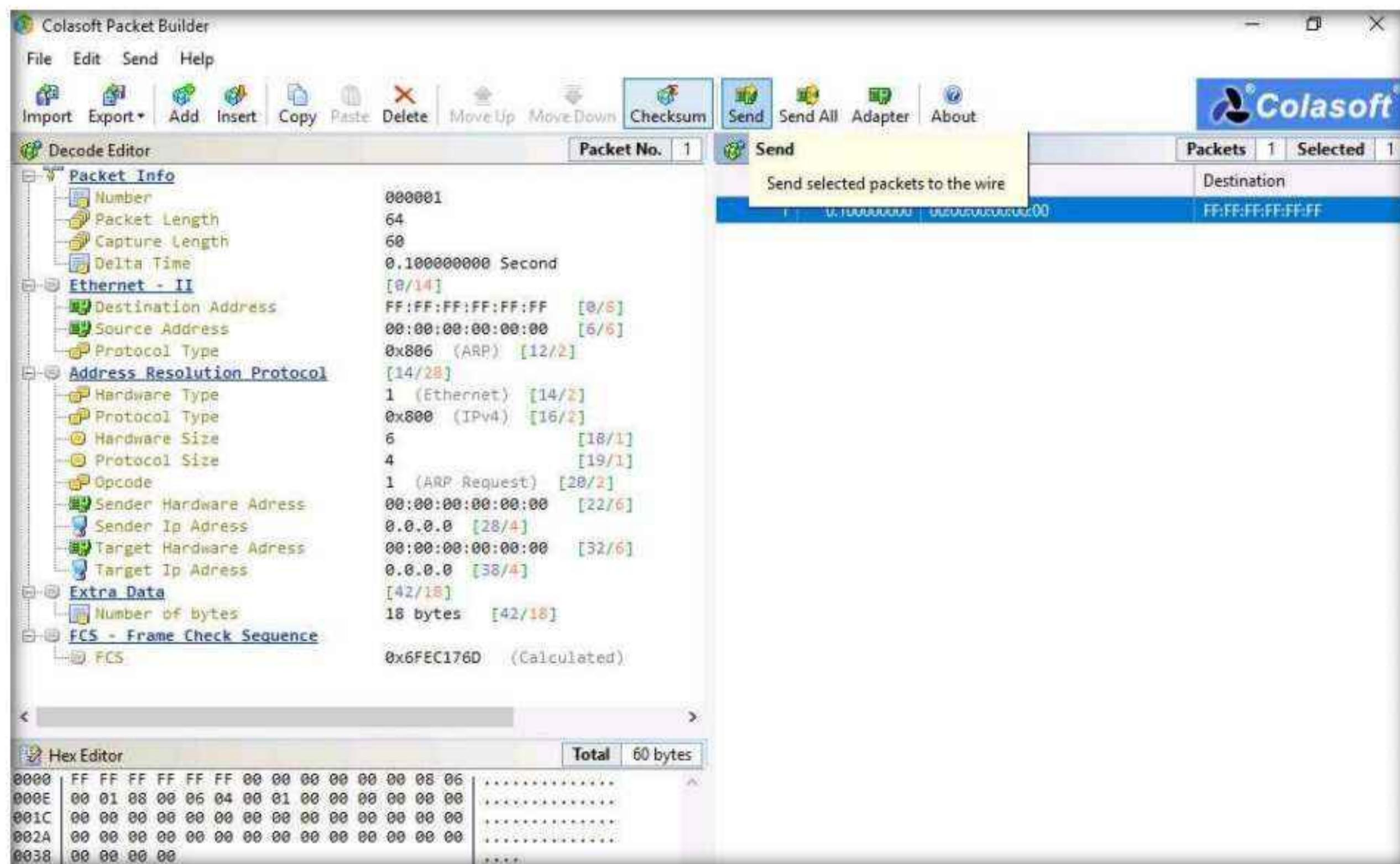
- The **Decode Editor** section allows you to edit the packet decoding information by double-clicking the item that you wish to decode.

Module 03 – Scanning Networks

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

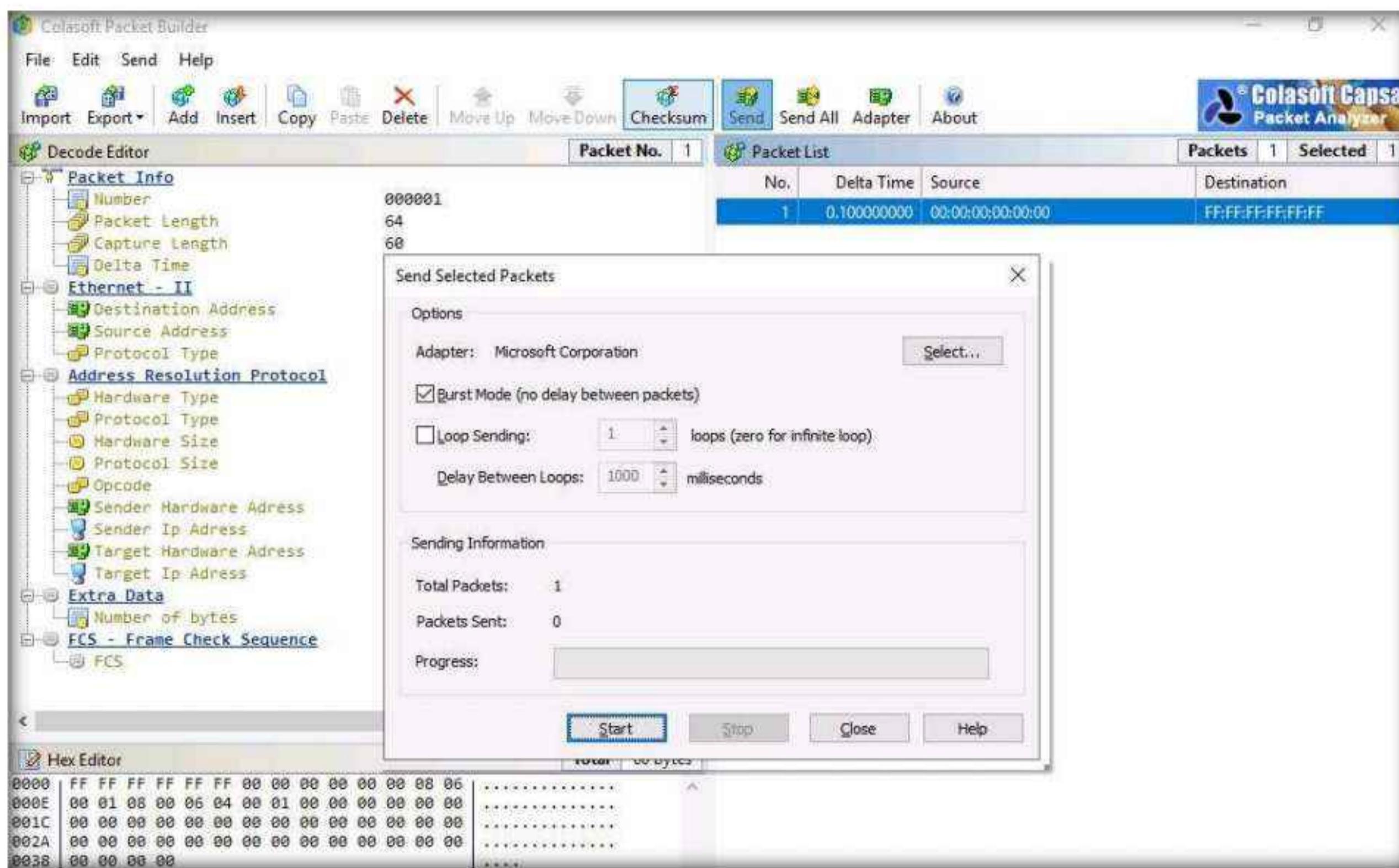


12. To send the packet, click **Send** from the **Menu** bar.

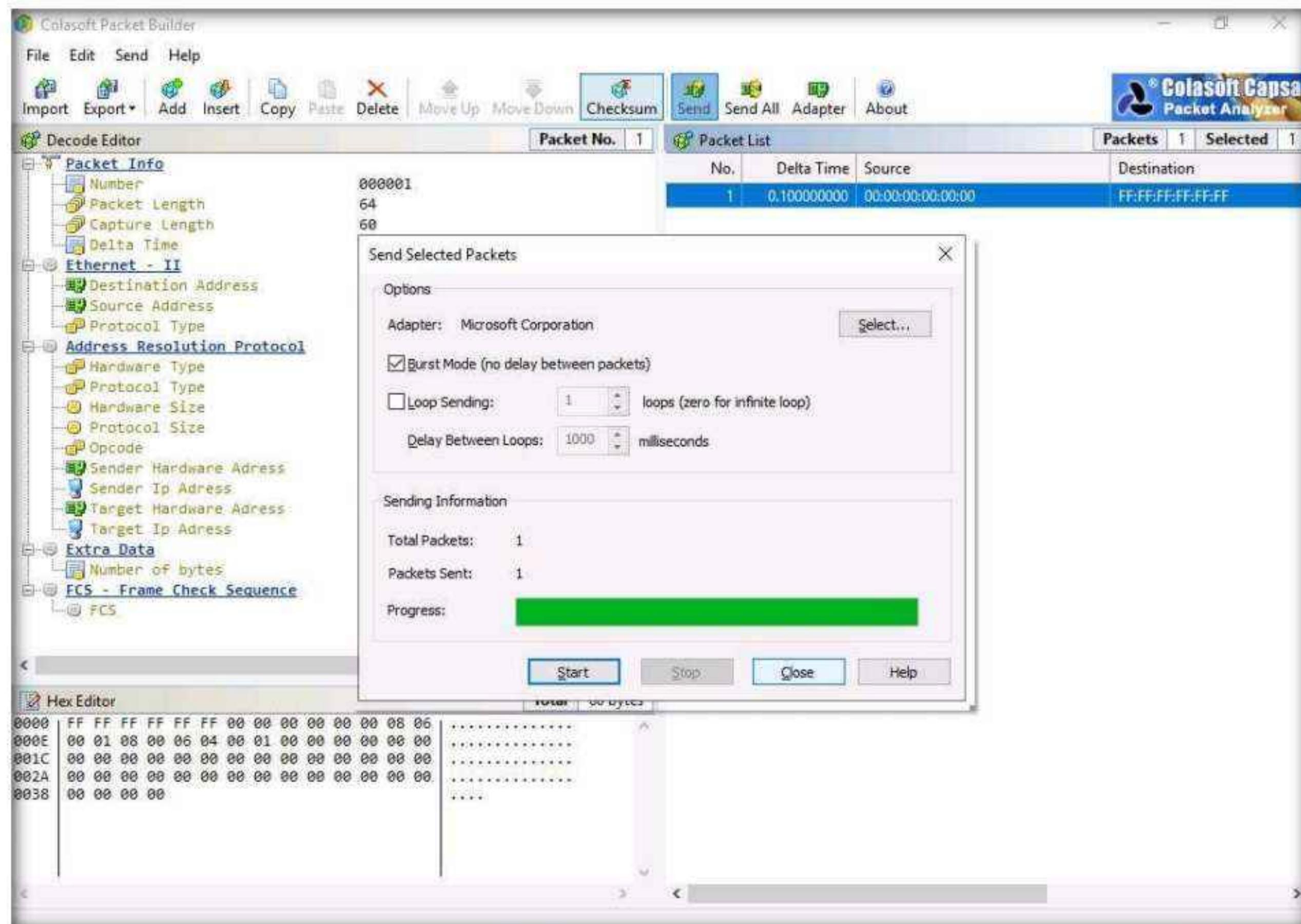


Module 03 – Scanning Networks

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

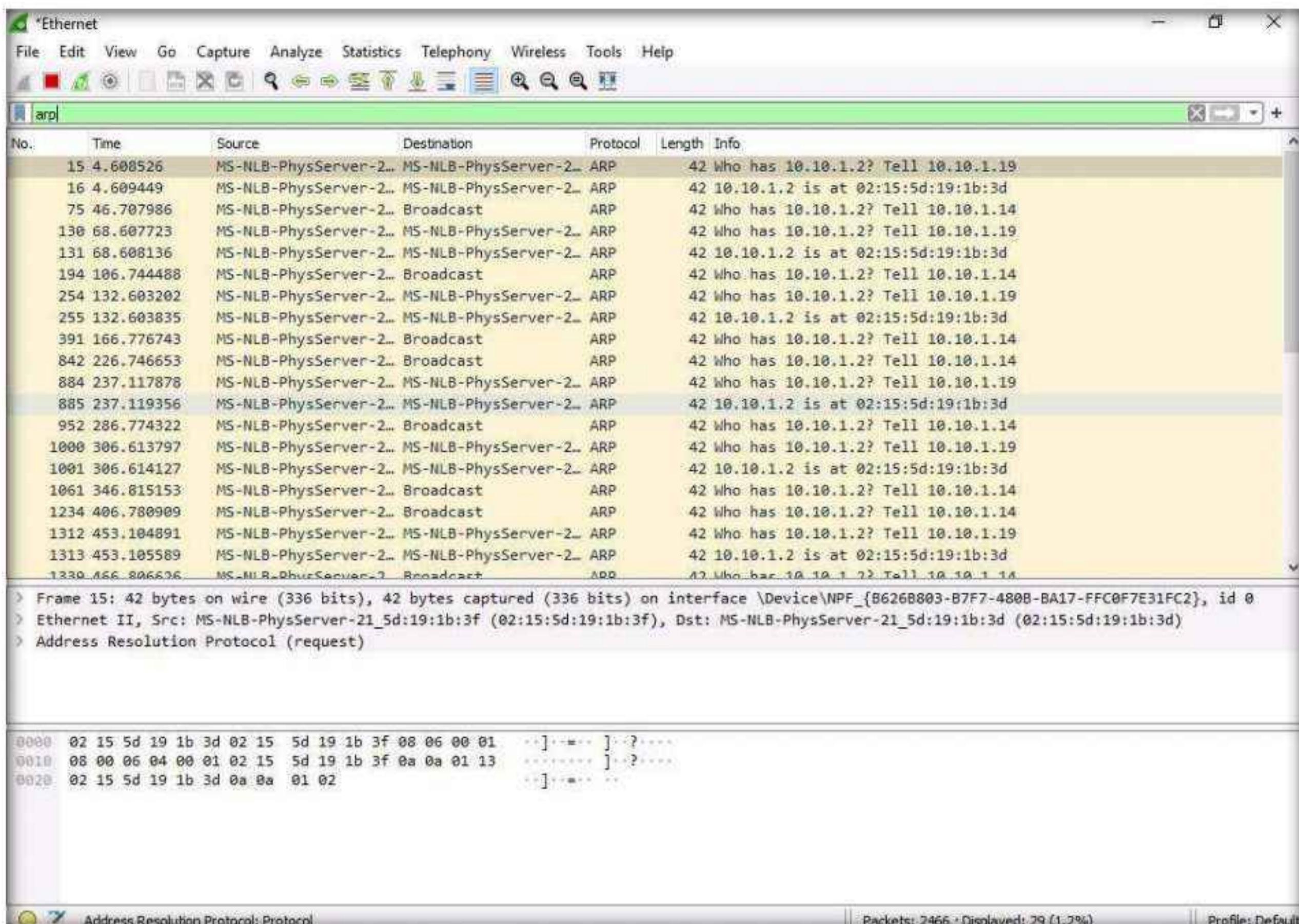


14. After the Progress bar completes, click **Close**.

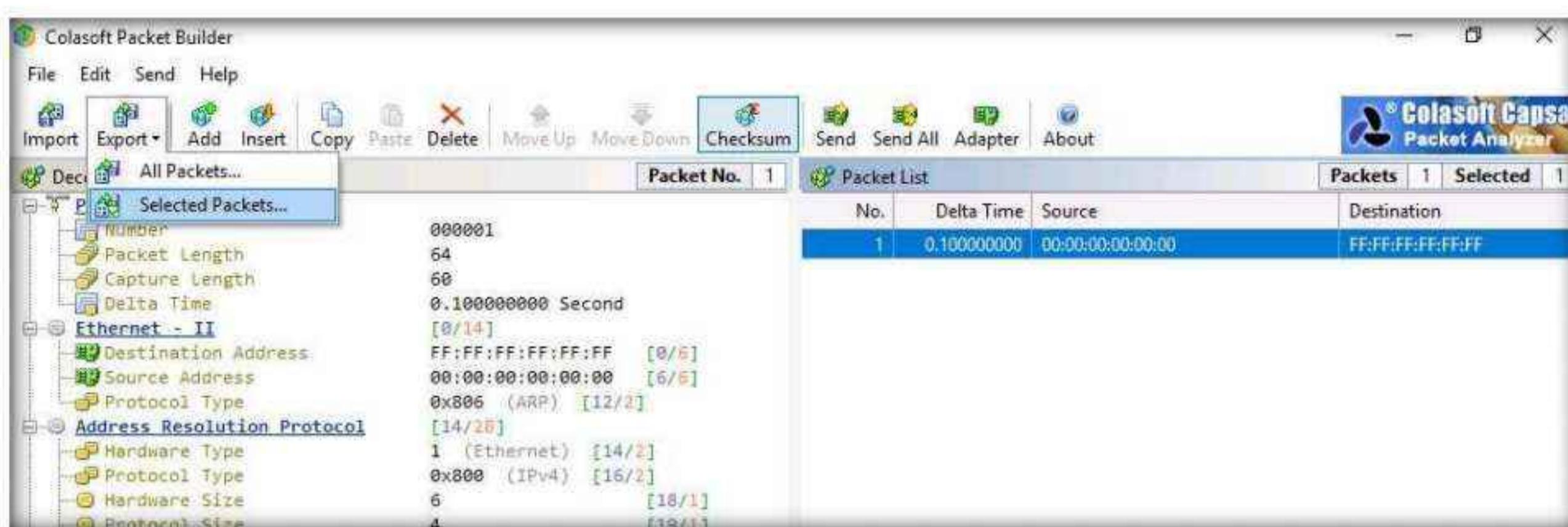


Module 03 – Scanning Networks

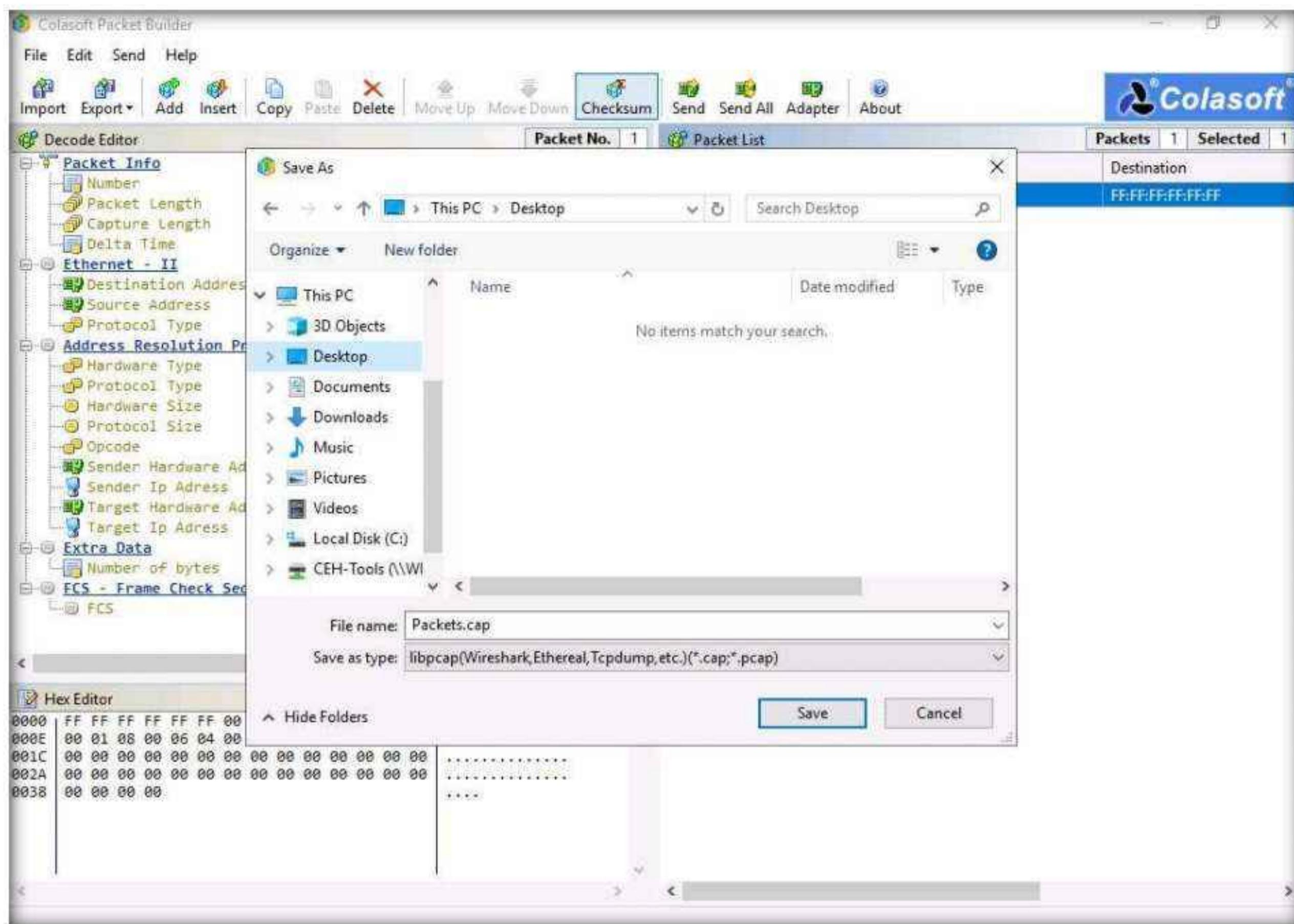
15. Now, when this ARP packet is broadcasted in the network, the active machines receive the packet, and a few start responding with an ARP reply. To evaluate which machine is responding to the ARP packet, you need to observe packets captured by the **Wireshark** tool.
 16. In the **Wireshark** window, 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.
23. Turn off the **Windows Server 2019** virtual machine.

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

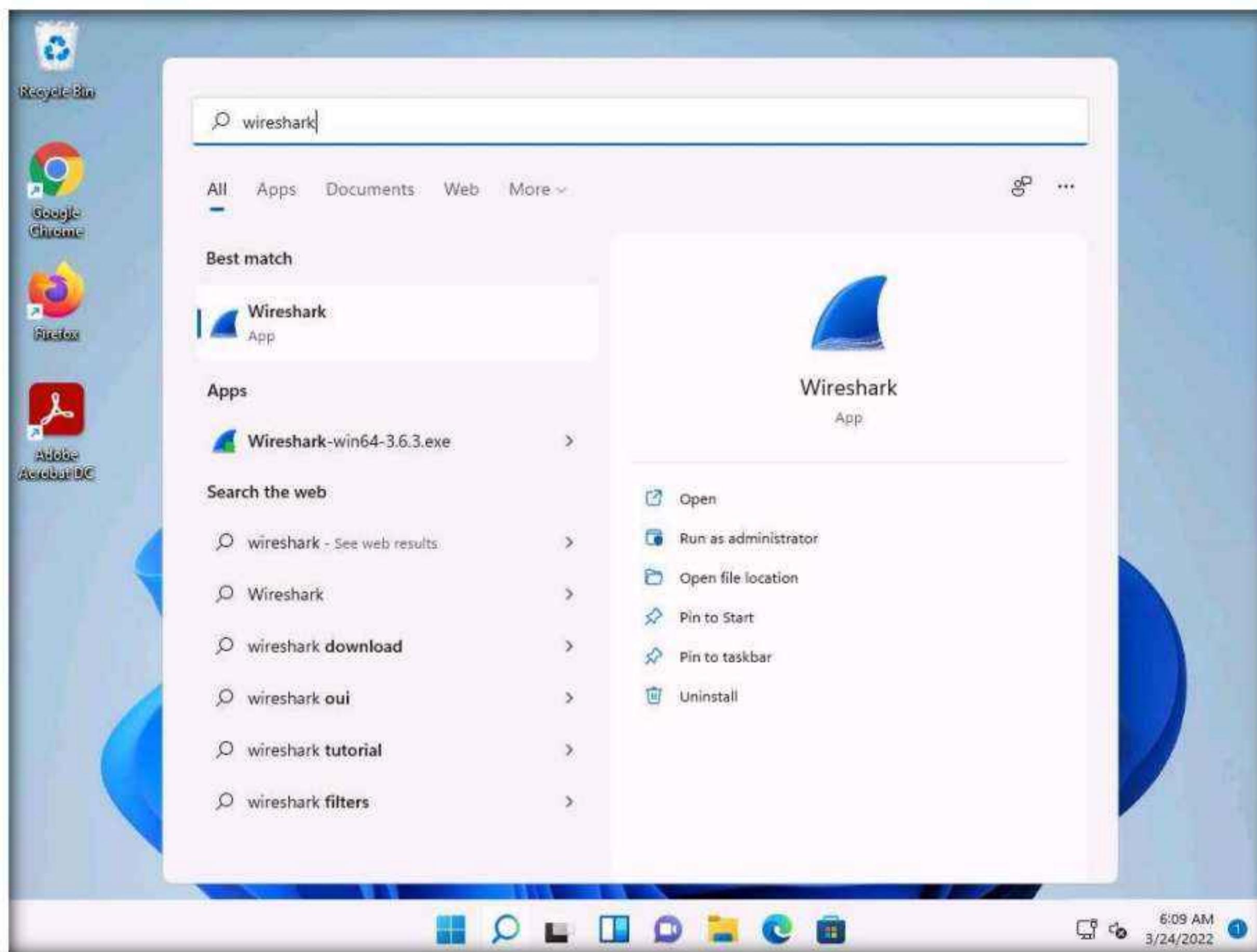
Hping3 is a scriptable program that uses the TCL language, whereby packets can be received and sent via a binary or string representation describing the packets.

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.

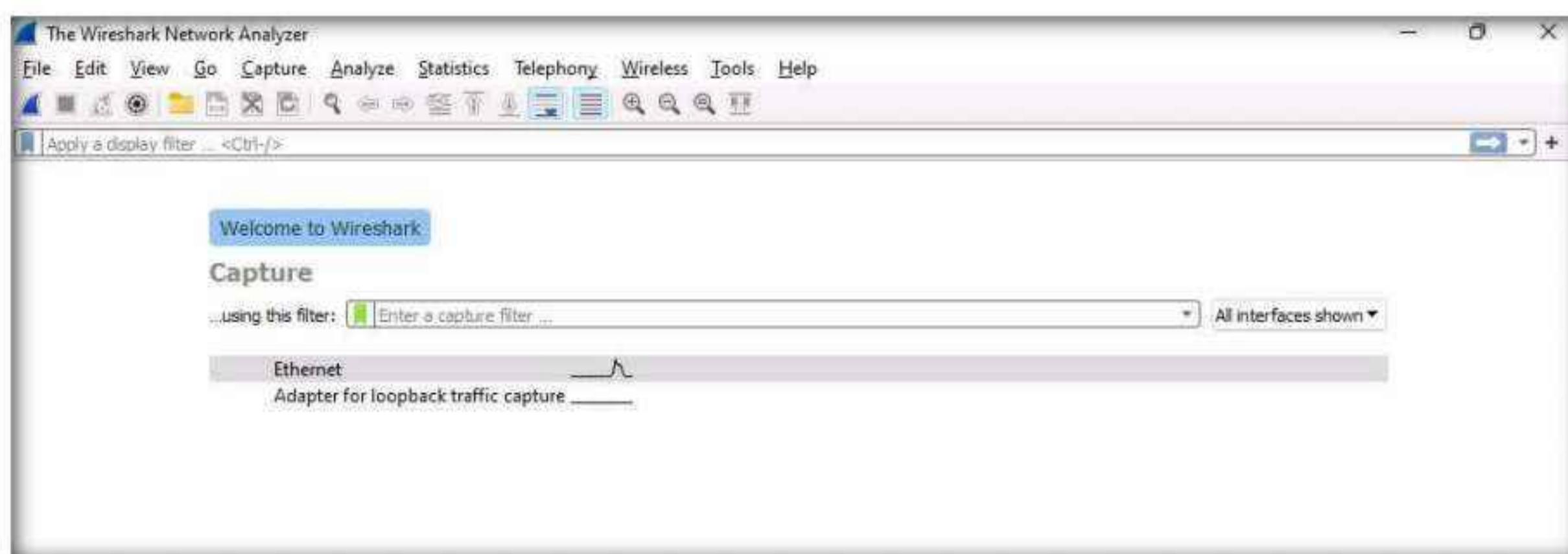
1. Switch to the **Windows 11** virtual machine.

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. Switch to the **Parrot Security** virtual 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.

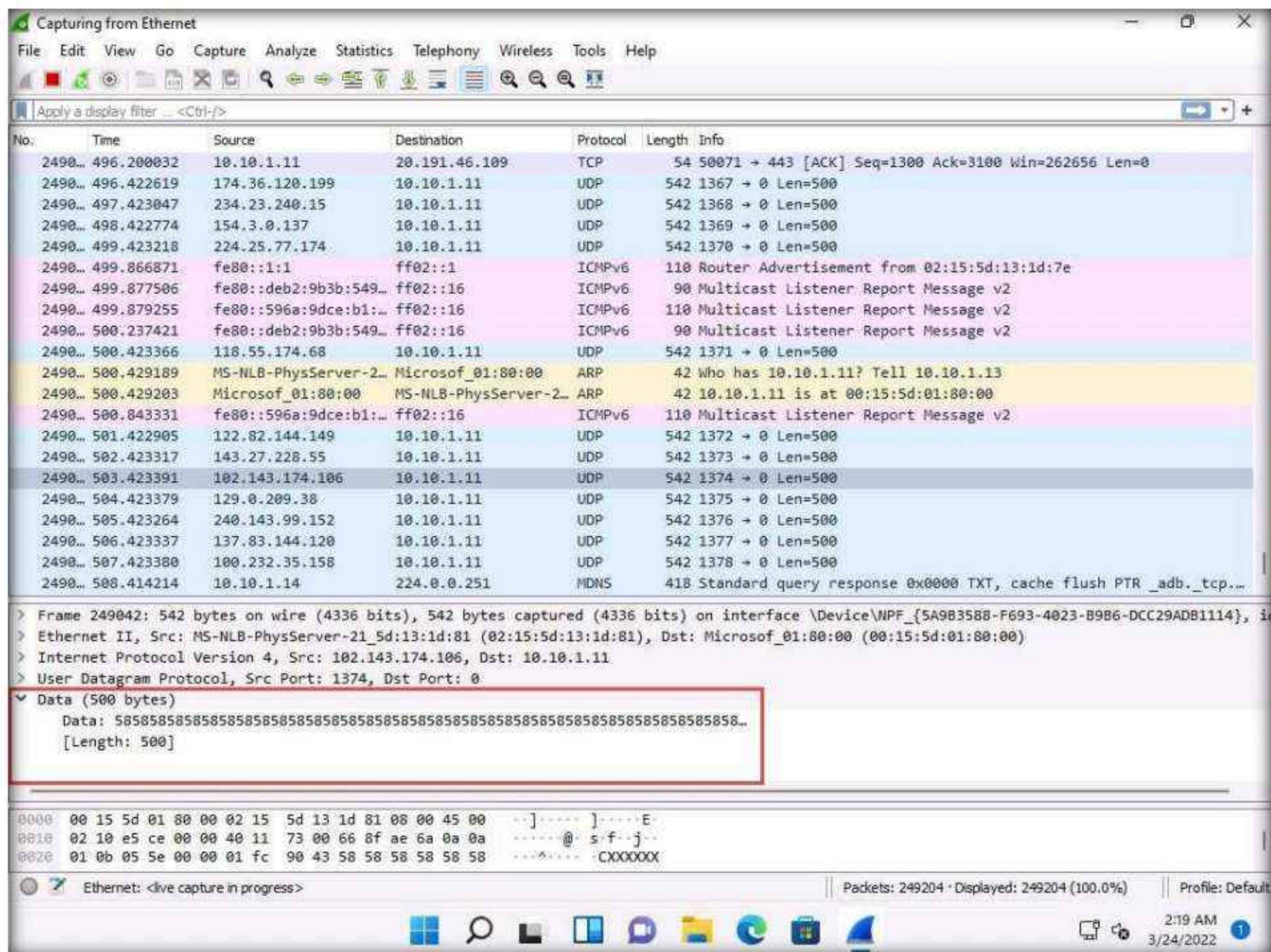


The screenshot shows a terminal window titled "hping3 10.10.1.11 --udp --rand-source --data 500 - Parrot Terminal". The terminal content is as follows:

```
[attacker@parrot]:~$ sudo su
[sudo] password for attacker:
[root@parrot]:~# 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
```

10. Switch to the **Windows 11** virtual 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**).

Module 03 – Scanning Networks



12. Switch to the **Parrot Security** virtual 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**.

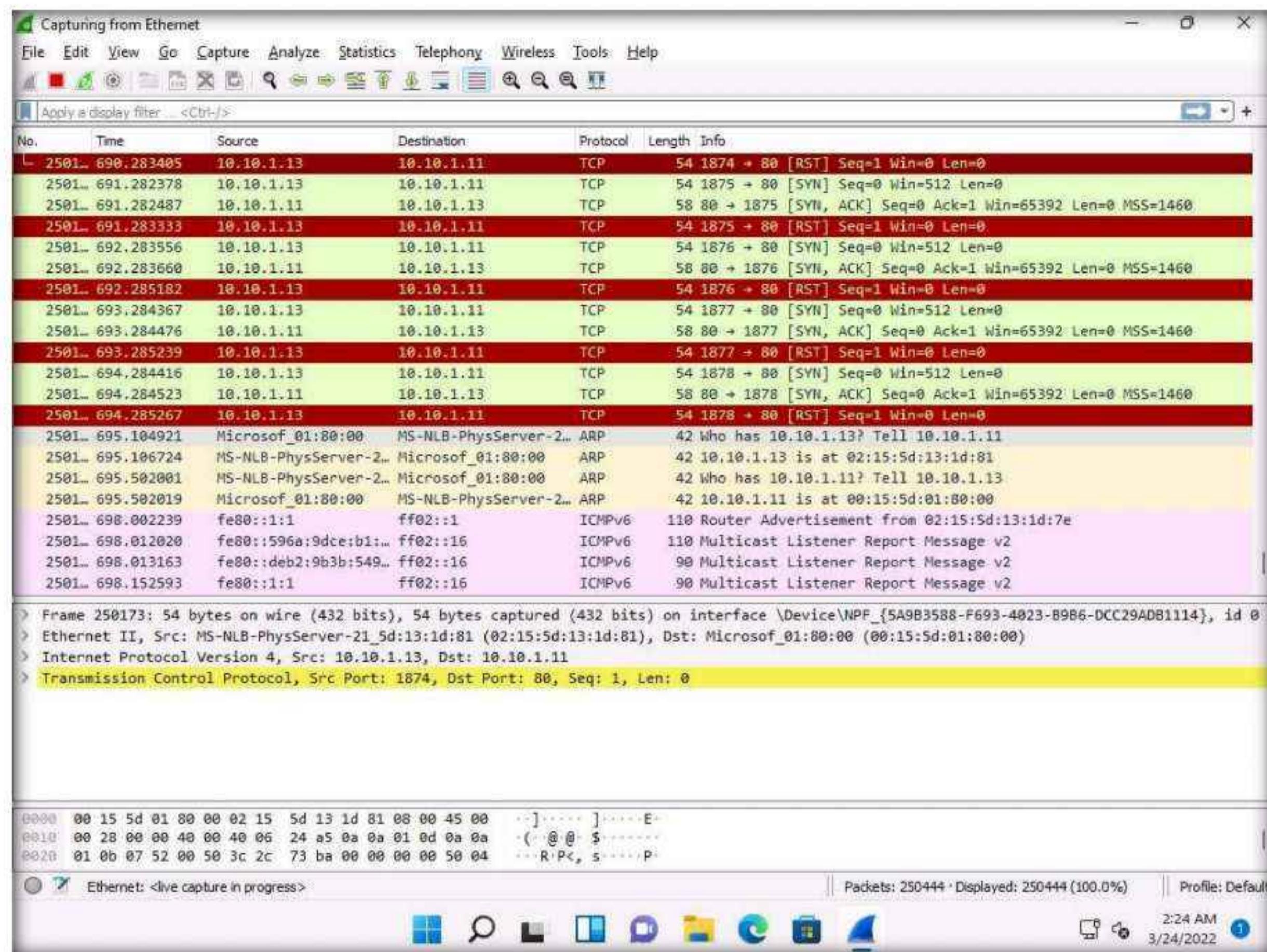
Note: Here, **-S** specifies the TCP SYN request on the target machine, **-p** specifies assigning the port to send the traffic, and **-c** is the count of the packets sent to the target machine.

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

```
[*]-[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]-[~]
#
```

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



15. Switch to the **Parrot Security** virtual 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.

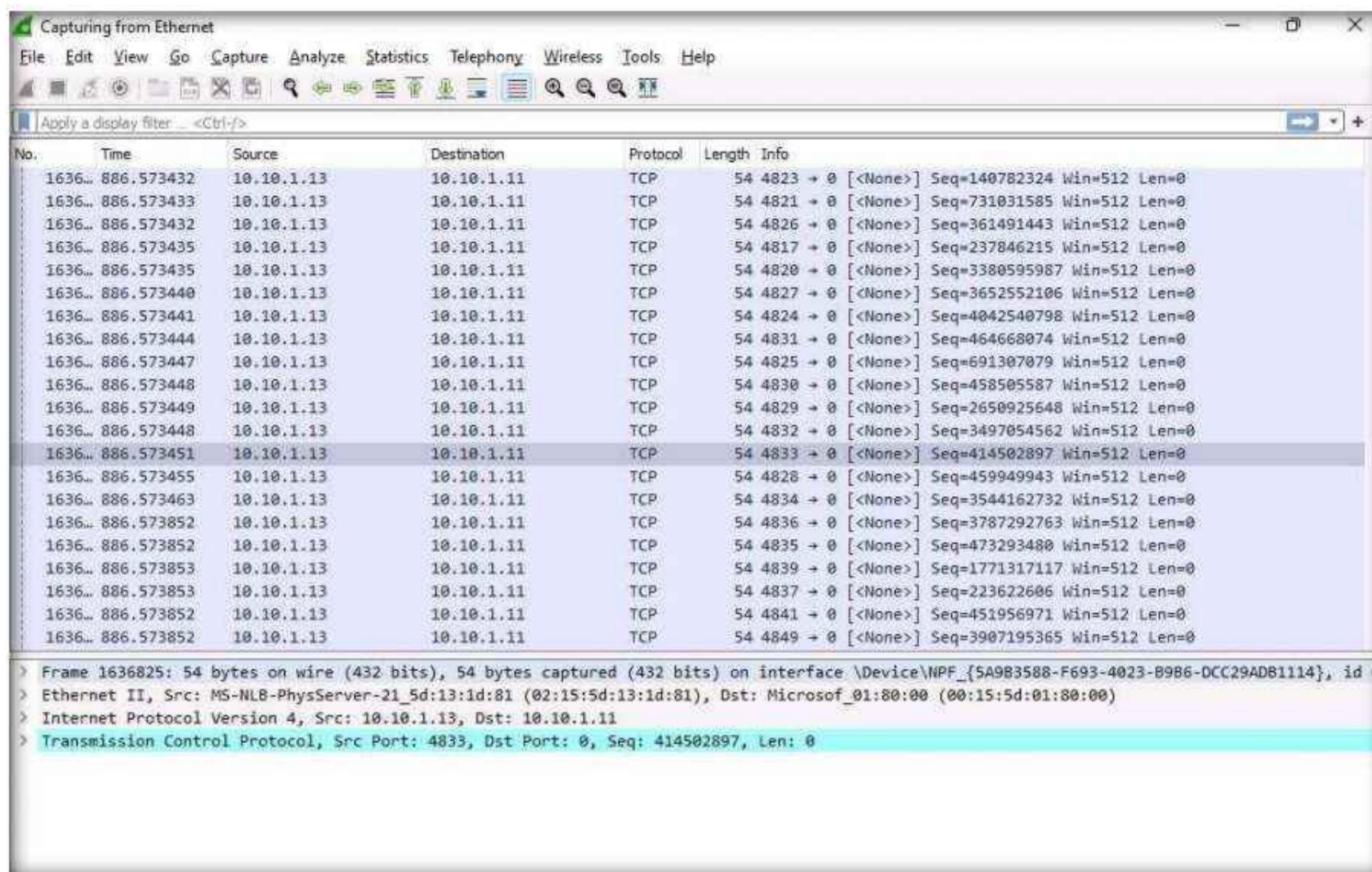
```
[root@parrot] ~
# hping3 10.10.1.11 --flood
HPING 10.10.1.11 (eth0 10.10.1.11): NO FLAGS are set, 40 headers + 0 data bytes
hpingle in flood mode, no replies will be shown
```

18. 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.

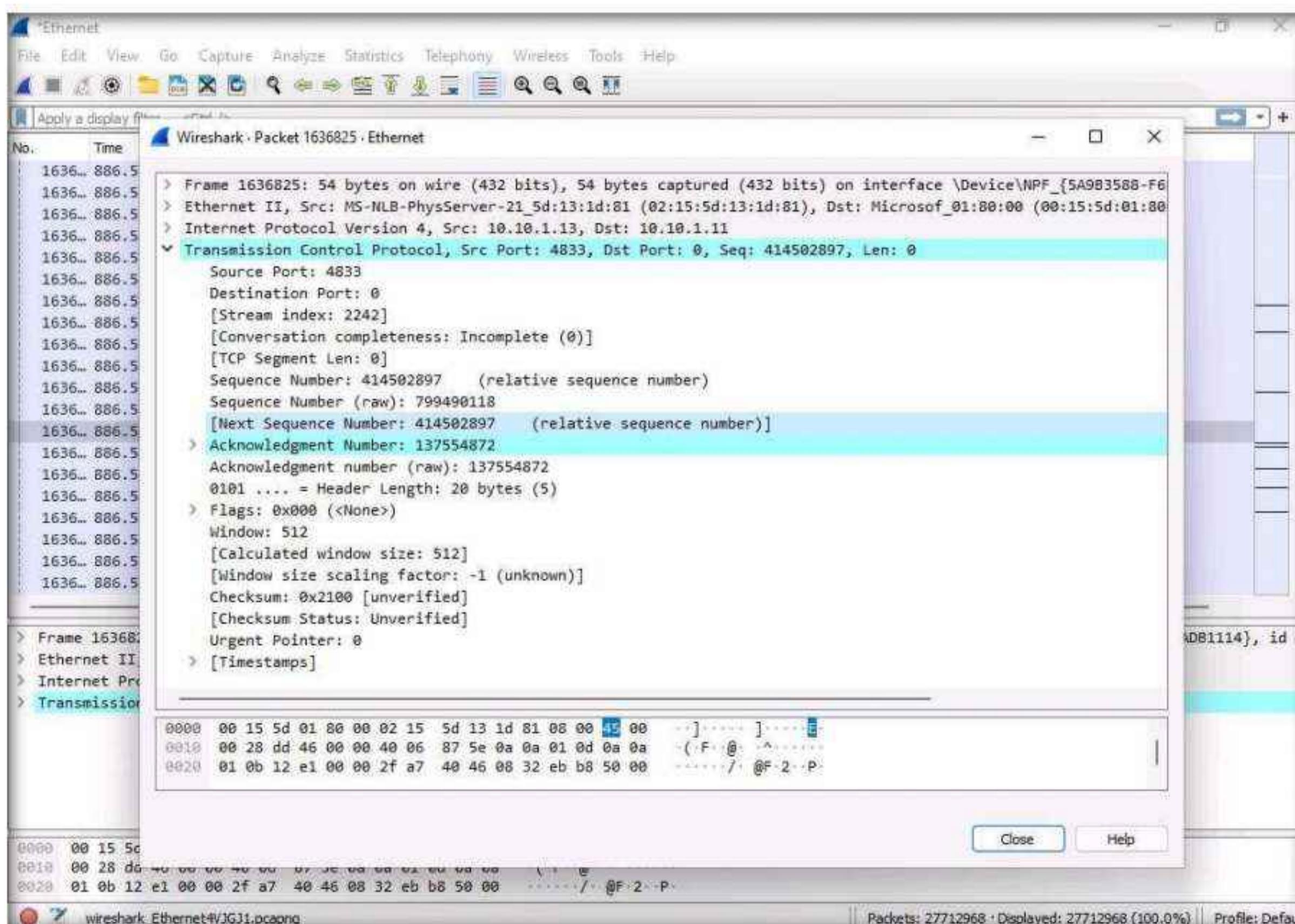
19. Observe the **Wireshark** window, which displays the TCP packet flooding from the host machine. The attacker employs TCP SYN flooding technique to perform a DoS attack on the target.

Module 03 – Scanning Networks

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.
25. Turn off the **Parrot Security** virtual machine.

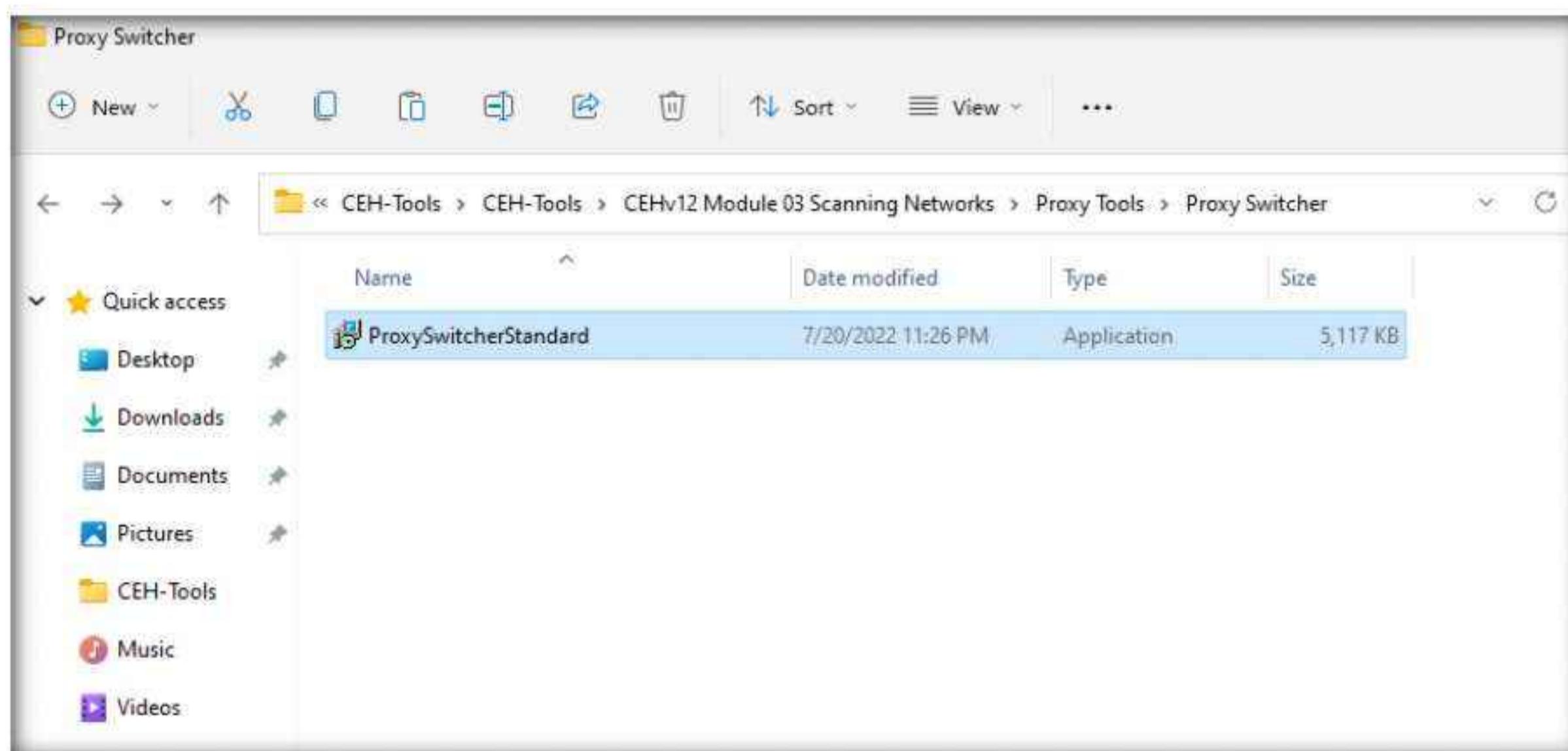
Task 4: Browse Anonymously using Proxy Switcher

Proxy Switcher allows you to surf the Internet anonymously without disclosing the IP address of your system, and helps to access various blocked sites in the organization. It avoids all types of limitations imposed by target sites.

Here, we will use Proxy Switcher to browse the Internet anonymously.

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

Note: If a **User Account Control** window appears, click **Yes**.



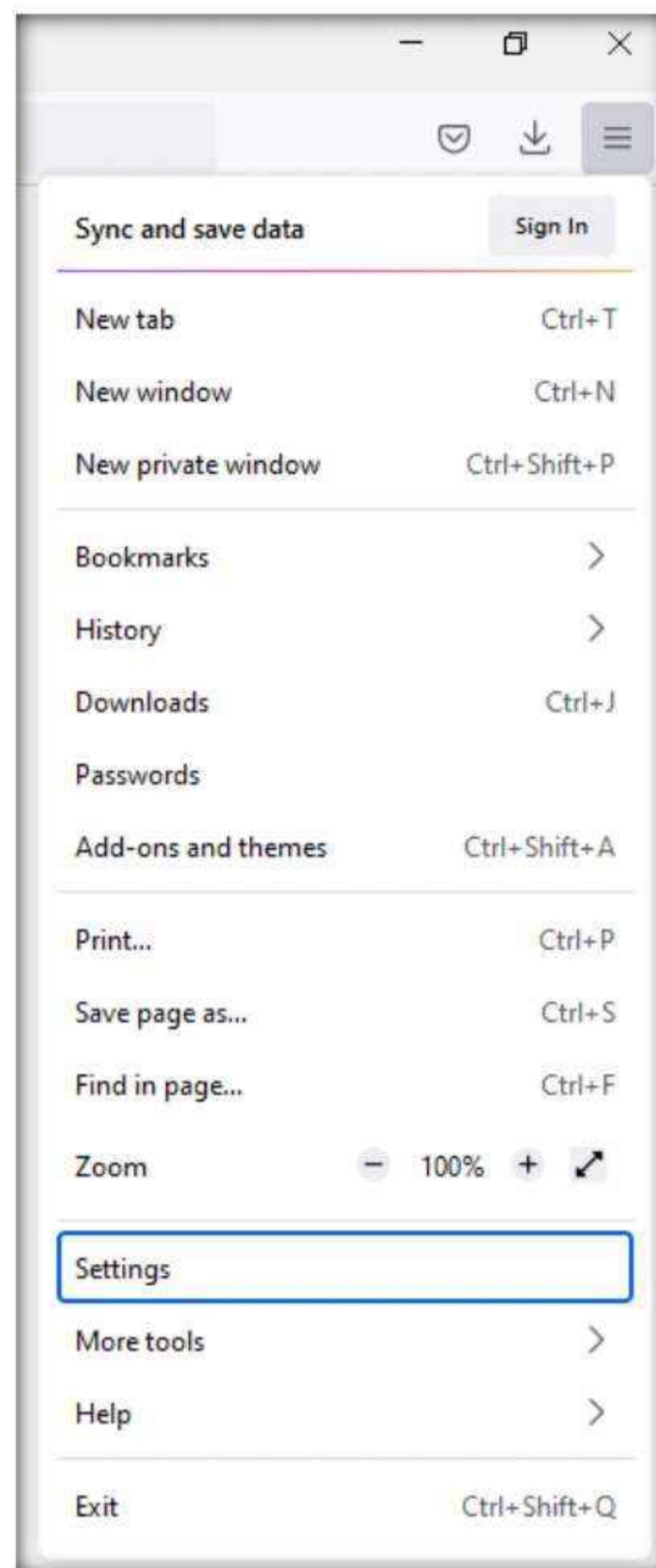
2. Follow the installation steps to install **Proxy Switcher** using all default settings.
3. Once the installation is complete, uncheck all options in the final step of the wizard, and click **Finish**.



4. Now, launch the **Firefox** browser.

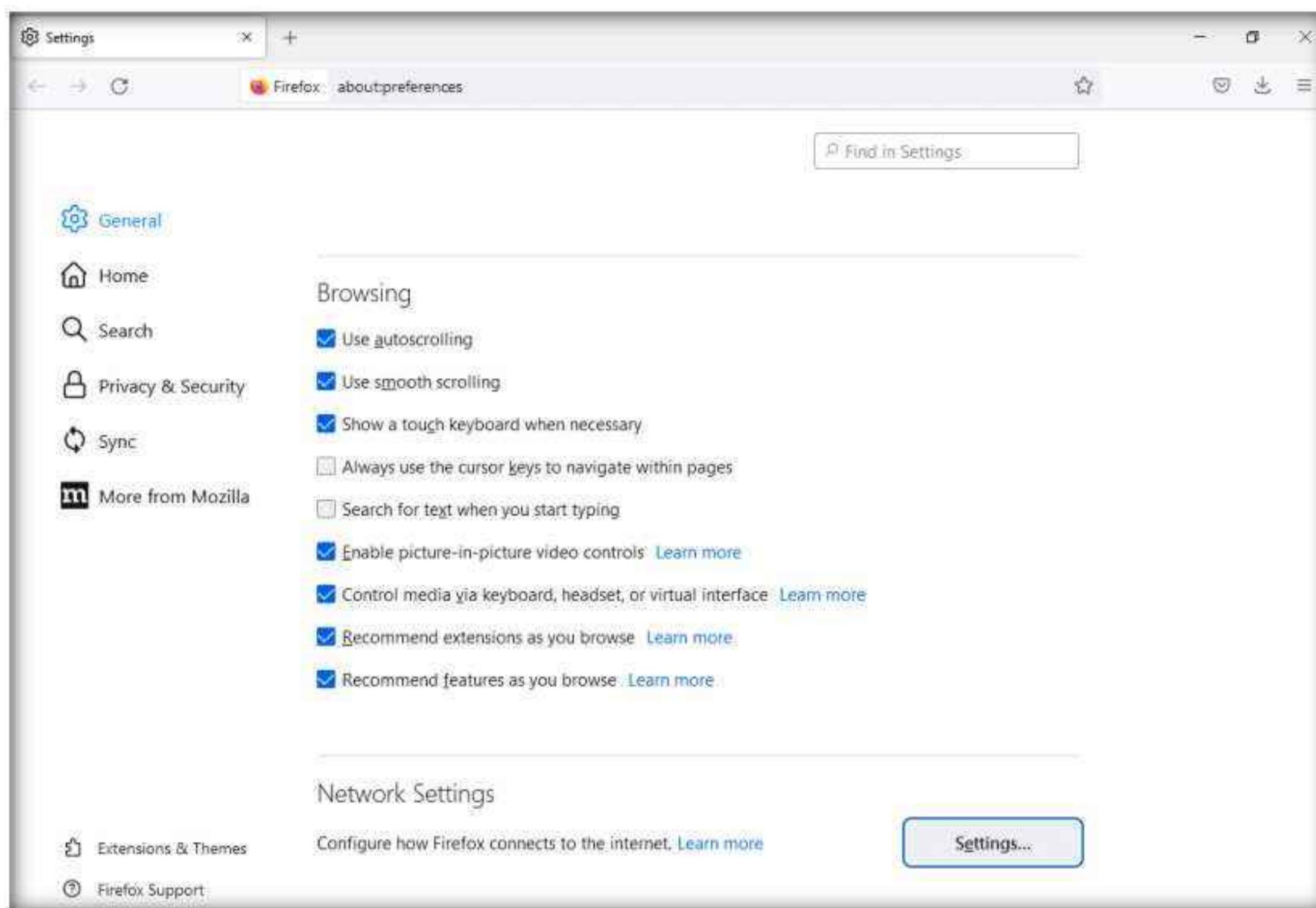
Note: If a **Default Browser** pop-up appears, click **Not now**.

5. Click the **Open menu** icon in the top-right corner of the browser window and click **Settings**.

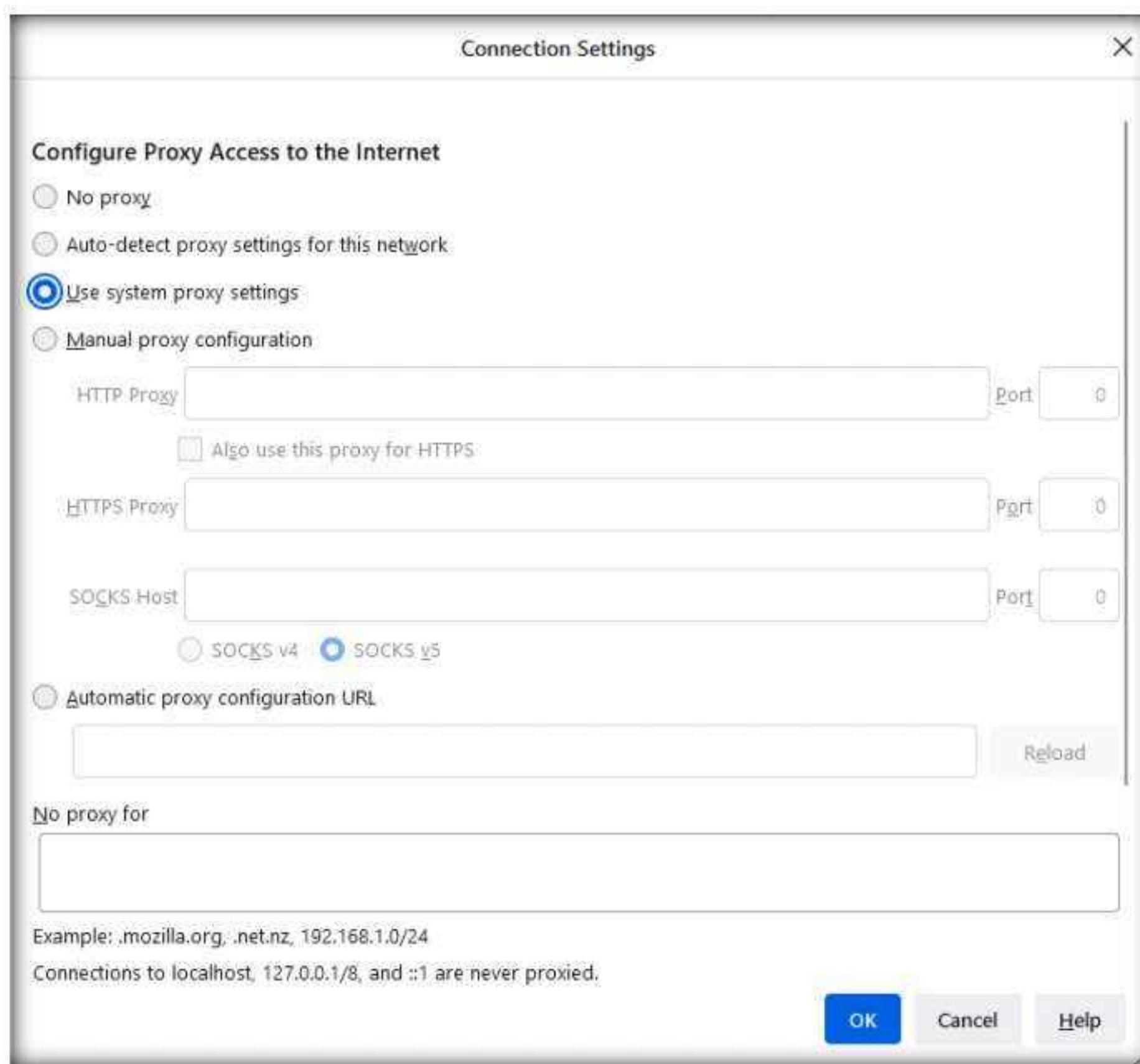


Module 03 – Scanning Networks

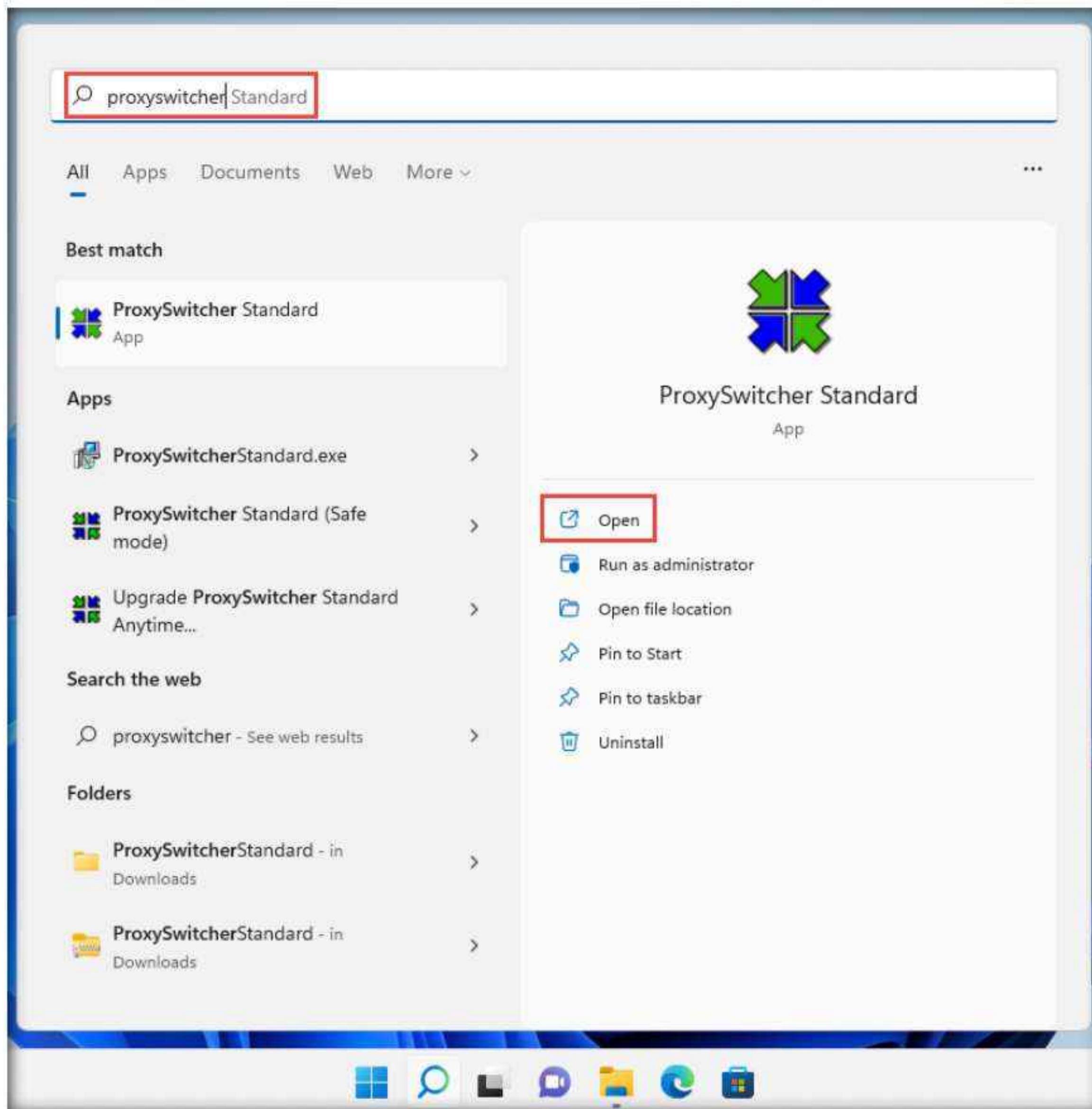
6. In the **Settings** wizard, scroll down to the end of the page and click **Settings...** under the **Network Settings** section.



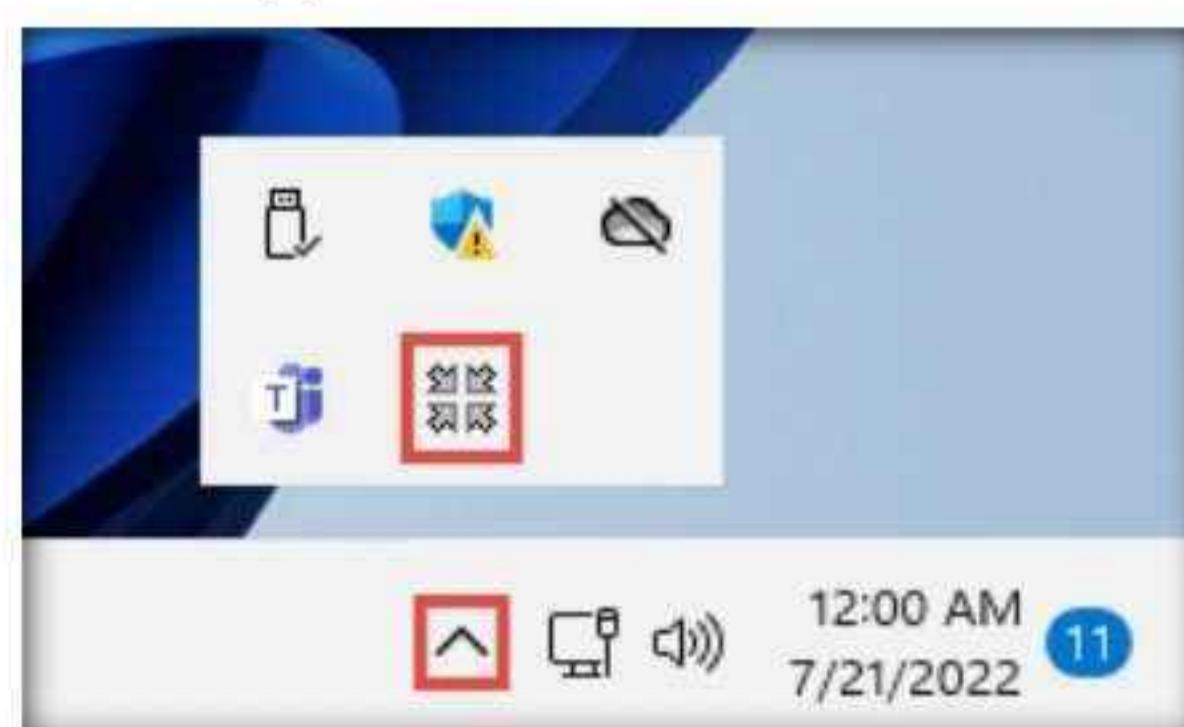
7. The **Connection Settings** window appears; under the **Configure Proxy Access to the Internet** section, ensure that the **Use system proxy settings** radio button is selected. Click **OK** and close the **Firefox** browser window.



8. Click **Search icon** (🔍) on the **Desktop**. Type **proxyswitcher** in the search field, the **ProxySwitcher Standard** appears in the result, click **open** to launch it.



9. The **ProxySwitcher Standard** loads, and its icon appears on **Taskbar**.
10. Click the **Taskbar** icon in the bottom-right corner of the desktop and the click **ProxySwitcher Standard** icon to launch the application.

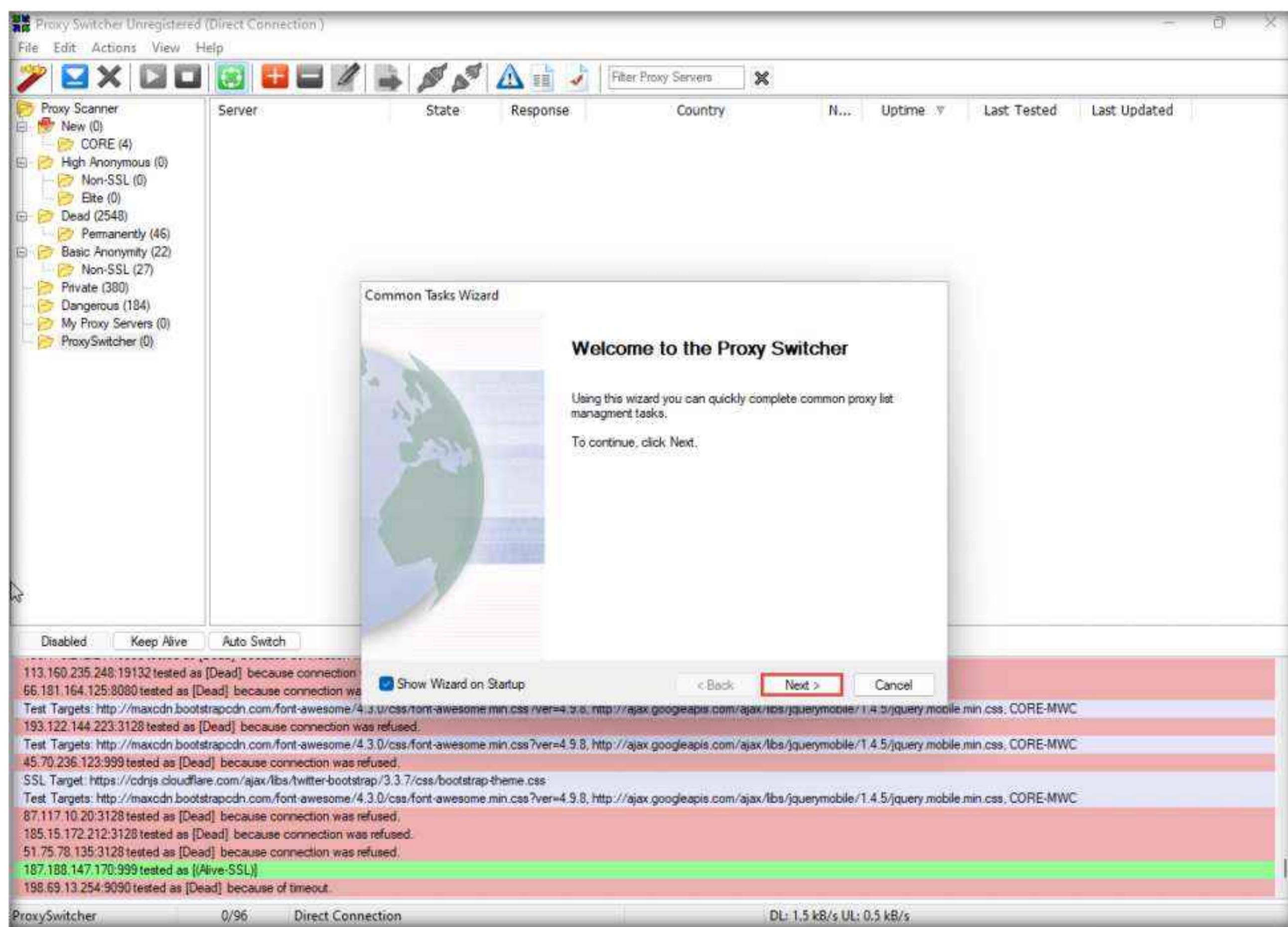


Module 03 – Scanning Networks

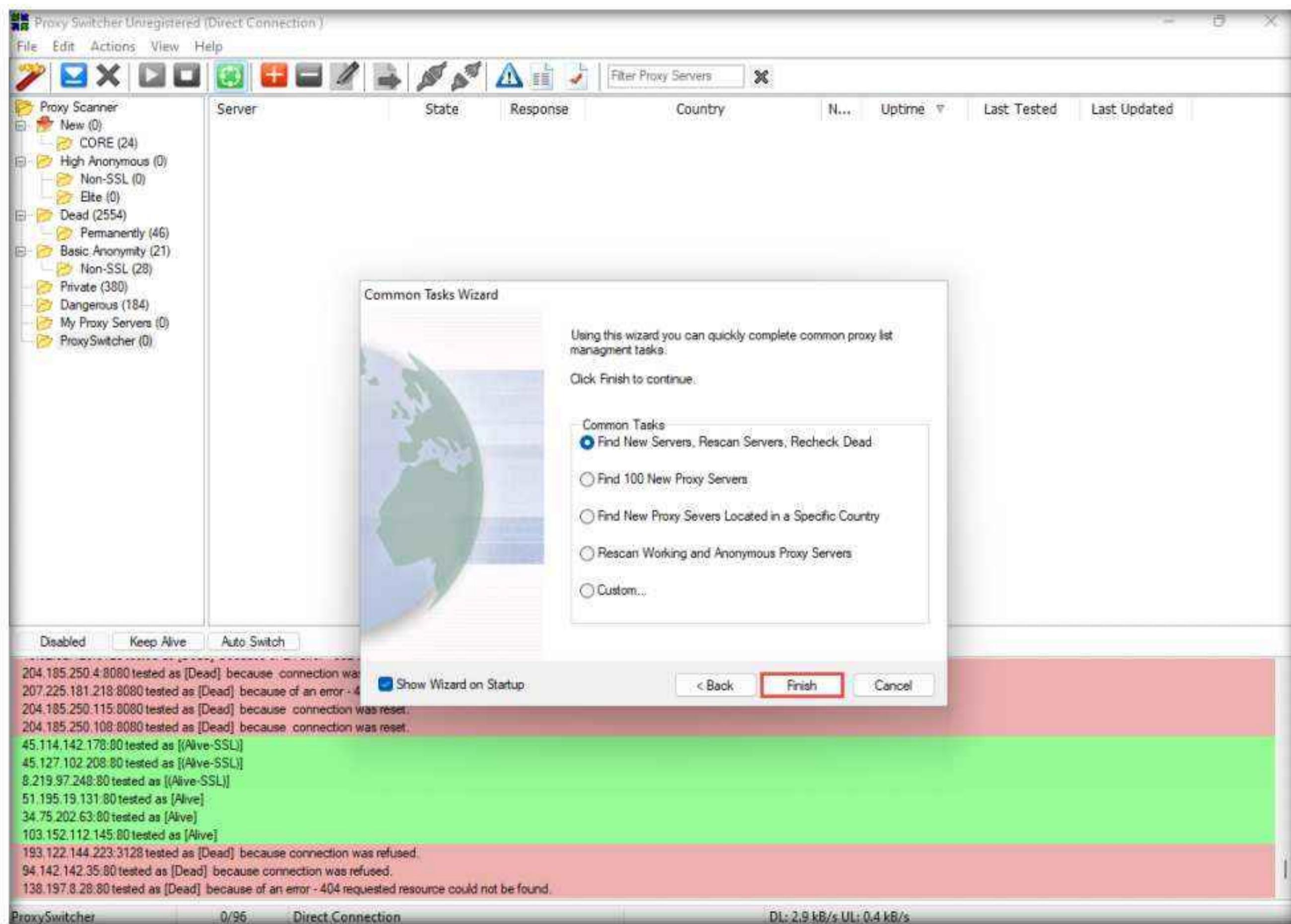
11. The **Please Register** window appears; click the **Start 15 Day Trial** button to proceed.



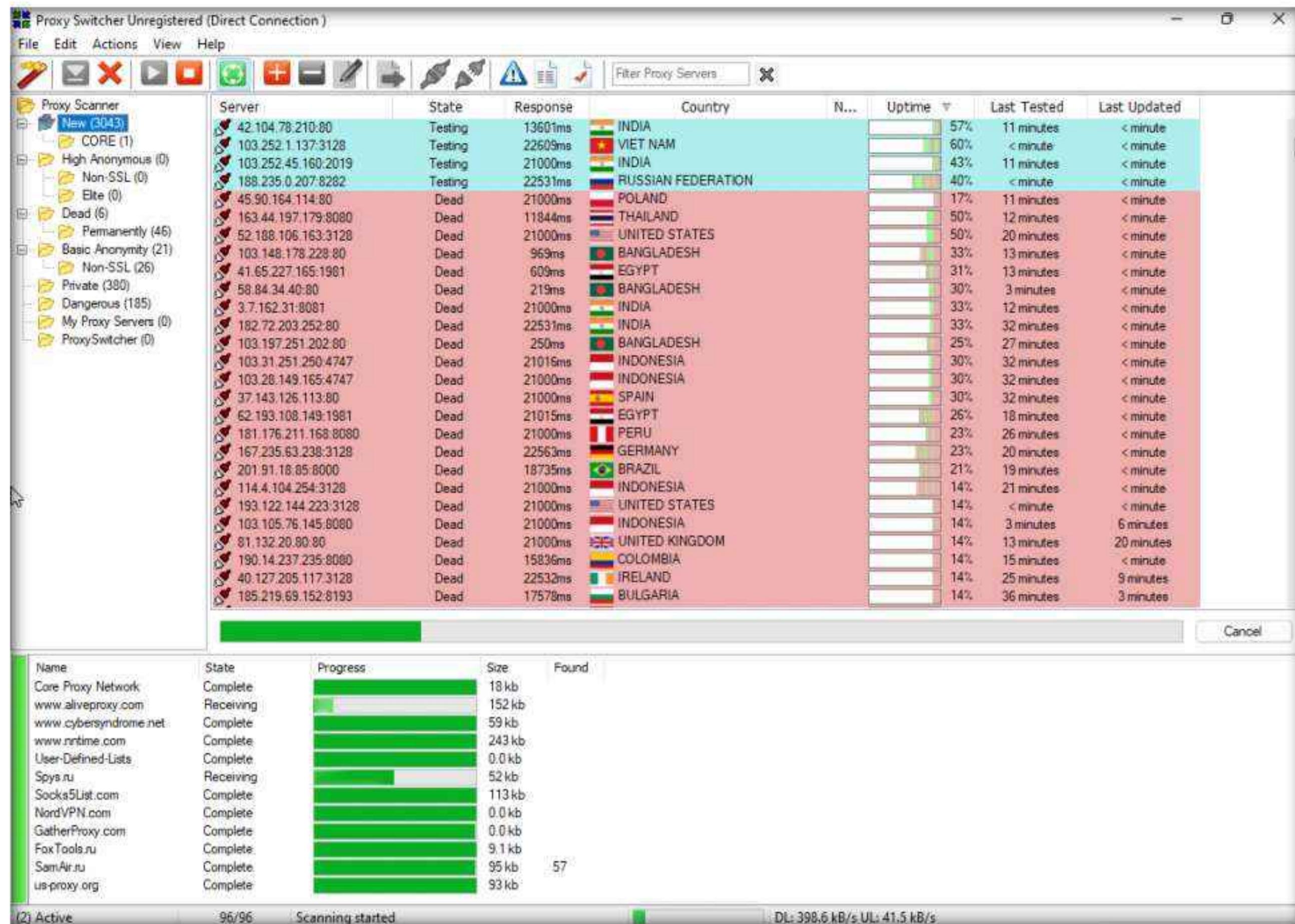
12. The **Common Tasks Wizard** window appears; under **Welcome to the Proxy Switcher**, click **Next**.



13. Ensure that the **Find New Server, Rescan Servers, Recheck Dead** radio button is selected under the **Common Tasks** section, and click **Finish**.



14. Proxy Switcher window appears, showing a list of proxy servers in the right pane, as shown in the following screenshot.



Note: The list of proxy servers might vary in your lab environment.

Note: It takes some time for the list to load

Module 03 – Scanning Networks

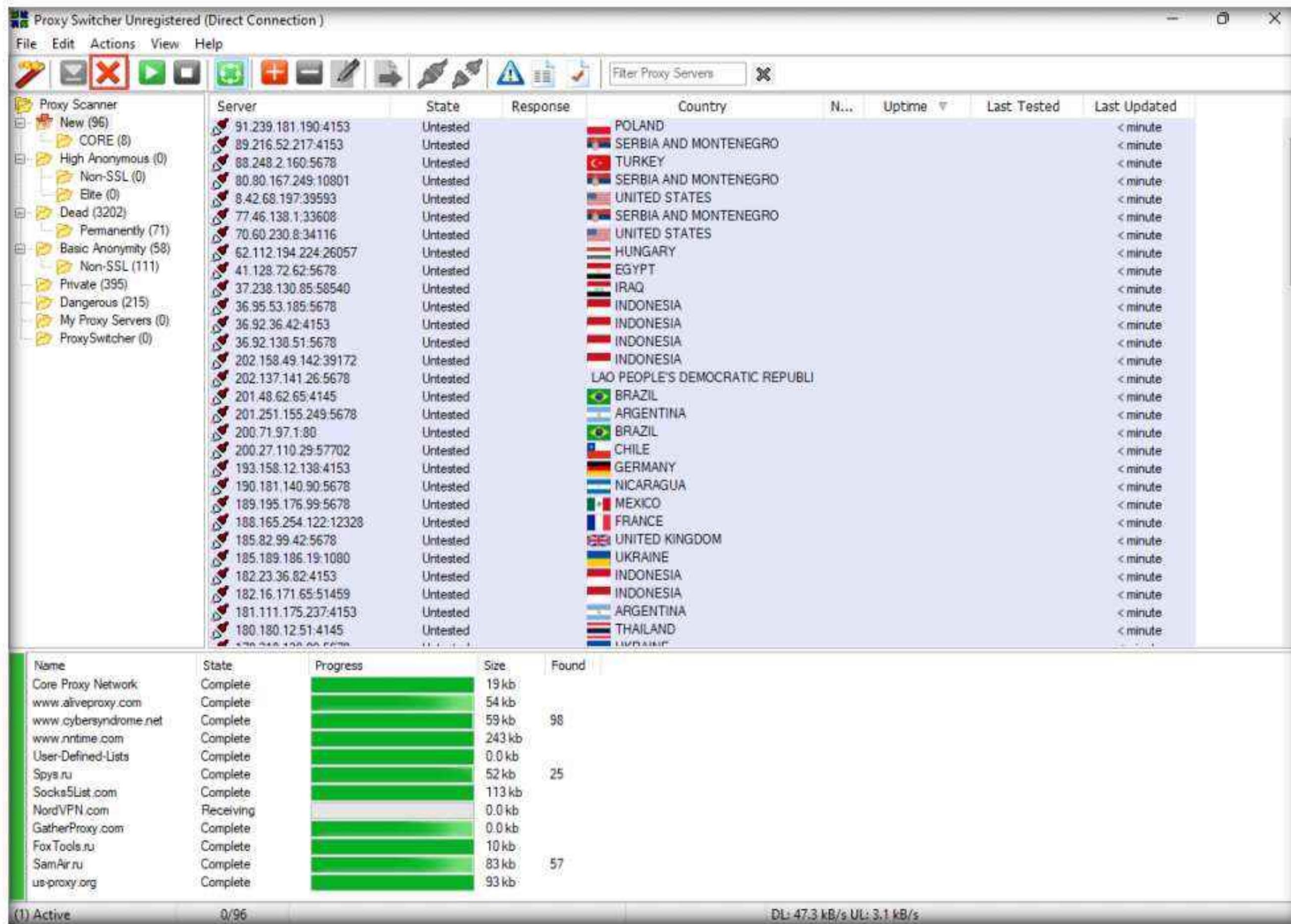
15. Observe the search bar below the server section; once it is completed, click the **Download Proxy Lists** icon () to download the proxy list.

The screenshot shows the 'Proxy Scanner' section of the software. On the left, there's a tree view of proxy categories: New (305), CORE (0), High Anonymous (0), Non-SSL (0), Elite (0), Dead (2915), Permanently (52), Basic Anonymity (52), Non-SSL (100), Private (394), Dangerous (215), My Proxy Servers (0), and ProxySwitcher (0). The main pane displays a table with columns: Server, State, Response, Country, N..., Uptime, Last Tested, and Last Updated. The table lists numerous proxy servers from various countries like United States, United Kingdom, Mexico, India, Russia, Thailand, Canada, Latvia, Egypt, Germany, Philippines, Singapore, Indonesia, and Nigeria. Below the table, a large red box highlights a list of proxy servers that have been marked as 'Dead' because connection was refused. At the bottom of the window, there are status indicators: '(0) Idle', '300/96', 'Scanning started', and network speeds 'DL: 48.0 kB/s UL: 3.6 kB/s'. A green progress bar at the bottom is labeled 'Cancel'.

16. Wait until all the proxy servers are downloaded. This can take a significant amount of time.

Module 03 – Scanning Networks

17. If you have enough downloaded proxy servers, you can click the Stop Download (icon to cancel the download.



Module 03 – Scanning Networks

18. Click the **Basic Anonymity** folder in the left-hand pane to display a list of alive proxy servers, as shown in the screenshot.

The screenshot shows the 'Proxy Switcher Unregistered (Direct Connection)' application window. On the left, a tree view displays proxy categories: New (96), CORE (1), High Anonymous (0), Non-SSL (0), Elite (0), Dead (3207), Permanently (71), and Basic Anonymity (59). The 'Basic Anonymity' node is selected and highlighted in red. The main pane is a table listing proxy servers with columns: Server, State, Response, Country, Uptime, Last Tested, and Last Updated. The table contains numerous entries from various countries like Ecuador, India, Mexico, France, Vietnam, United States, Germany, Russia, Brazil, Belarus, Egypt, and Uzbekistan. Below the table, a status bar shows 'Basic Anonymity 0/96' and network speeds 'DL: 1.4 kB/s UL: 0.2 kB/s'. A message bar at the bottom lists test results for various IP addresses, mostly marked as 'Alive'.

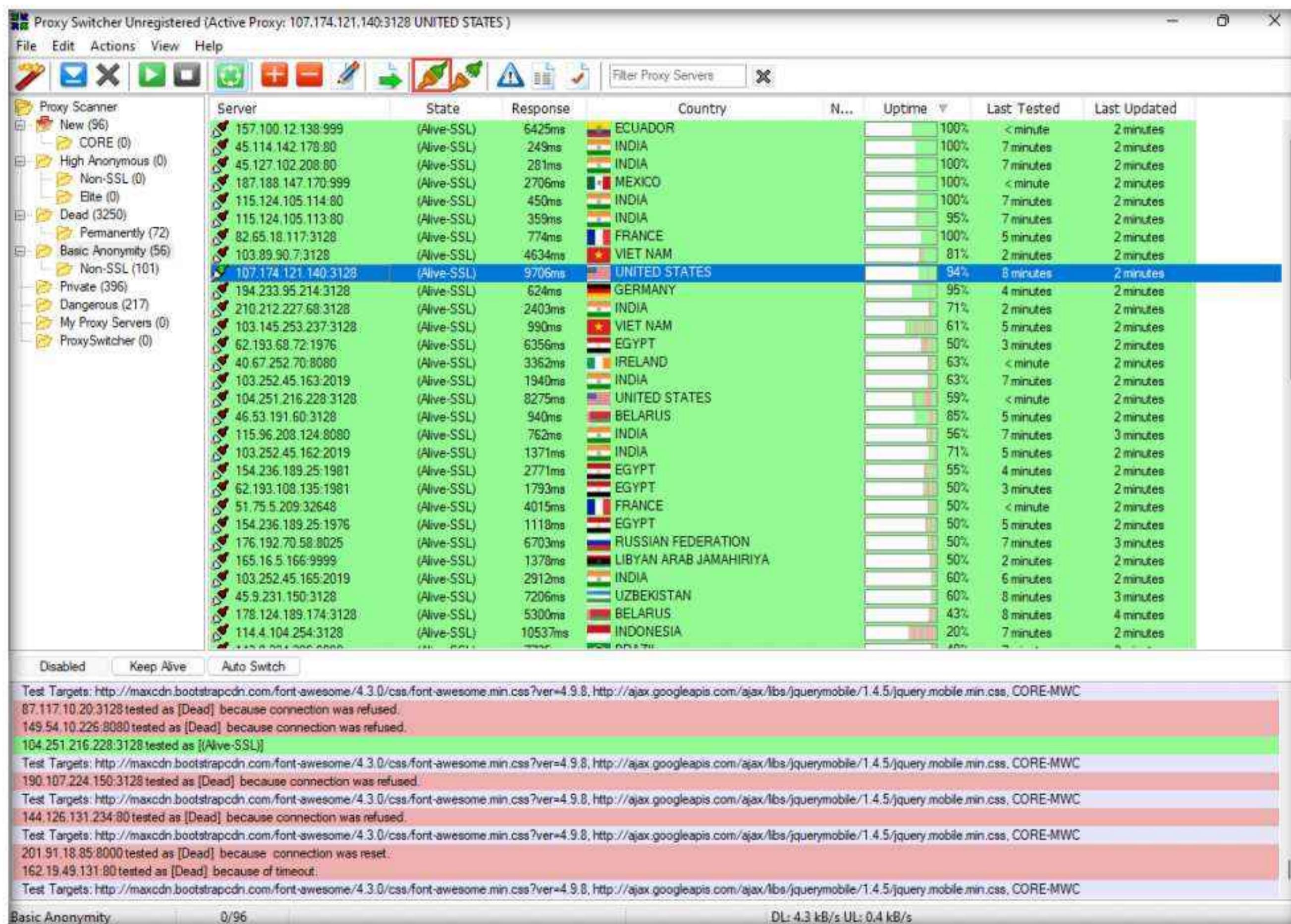
Server	State	Response	Country	Uptime	Last Tested	Last Updated
157.100.12.138:999	(Alive-SSL)	6134ms	ECUADOR	100%	4 minutes	< minute
45.114.142.179:80	(Alive-SSL)	249ms	INDIA	100%	2 minutes	1 minute
45.127.102.208:80	(Alive-SSL)	281ms	INDIA	100%	2 minutes	1 minute
115.124.105.114:80	(Alive-SSL)	450ms	INDIA	100%	2 minutes	1 minute
115.124.105.113:80	(Alive-SSL)	359ms	INDIA	95%	2 minutes	1 minute
187.188.147.170:999	(Alive-SSL)	5943ms	MEXICO	100%	4 minutes	< minute
82.65.18.117:3128	(Alive-SSL)	774ms	FRANCE	100%	< minute	1 minute
103.89.90.7:3128	(Alive-SSL)	2587ms	VIET NAM	80%	4 minutes	1 minute
107.174.121.140:3128	(Alive-SSL)	9706ms	UNITED STATES	94%	4 minutes	< minute
194.233.95.214:3128	(Alive-SSL)	1612ms	GERMANY	95%	4 minutes	1 minute
103.145.253.237:3128	(Alive-SSL)	990ms	VIET NAM	61%	< minute	1 minute
176.192.70.58:8029	(Alive-SSL)	5362ms	RUSSIAN FEDERATION	44%	4 minutes	1 minute
103.252.45.163:2019	(Alive-SSL)	1940ms	INDIA	63%	2 minutes	< minute
200.137.134.131:3128	(Alive-SSL)	2831ms	BRAZIL	67%	4 minutes	1 minute
210.212.227.68:3128	(Alive-SSL)	3122ms	INDIA	67%	4 minutes	1 minute
46.53.191.60:3128	(Alive-SSL)	940ms	BELARUS	85%	< minute	< minute
115.95.208.124:8080	(Alive-SSL)	762ms	INDIA	56%	2 minutes	1 minute
103.252.45.162:2019	(Alive-SSL)	1371ms	INDIA	71%	< minute	1 minute
62.193.68.72:1976	(Alive-SSL)	2062ms	EGYPT	42%	4 minutes	< minute
154.236.189.25:1981	(Alive-SSL)	1103ms	EGYPT	50%	4 minutes	1 minute
173.212.245.135:3128	(Alive-SSL)	3000ms	GERMANY	47%	4 minutes	< minute
154.236.189.25:1976	(Alive-SSL)	1118ms	EGYPT	50%	< minute	1 minute
176.192.70.58:8025	(Alive-SSL)	6703ms	RUSSIAN FEDERATION	50%	2 minutes	1 minute
62.193.108.135:1981	(Alive-SSL)	2303ms	EGYPT	50%	4 minutes	1 minute
103.252.45.165:2019	(Alive-SSL)	2912ms	INDIA	60%	2 minutes	1 minute
45.9.231.150:3128	(Alive-SSL)	7206ms	UZBEKISTAN	60%	3 minutes	1 minute
178.124.189.174:3128	(Alive-SSL)	5300ms	BELARUS	43%	4 minutes	< minute
51.75.5.209:32648	(Alive-SSL)	2421ms	FRANCE	43%	4 minutes	< minute
195.158.3.198:3128	(Alive-SSL)	4650ms	UZBEKISTAN	43%	4 minutes	1 minute
128.254.246.200:3128	(Alive-SSL)	5007	UNITED STATES	56%	+	+

Module 03 – Scanning Networks

19. Select one proxy server IP address in the right-hand pane. To switch to the selected proxy server, click the **Switch to Selected Proxy Server** () icon.

20. When the proxy server is connected, it will show the connection icon as .

Note: The proxy selected in this lab might vary in your lab **environment**.

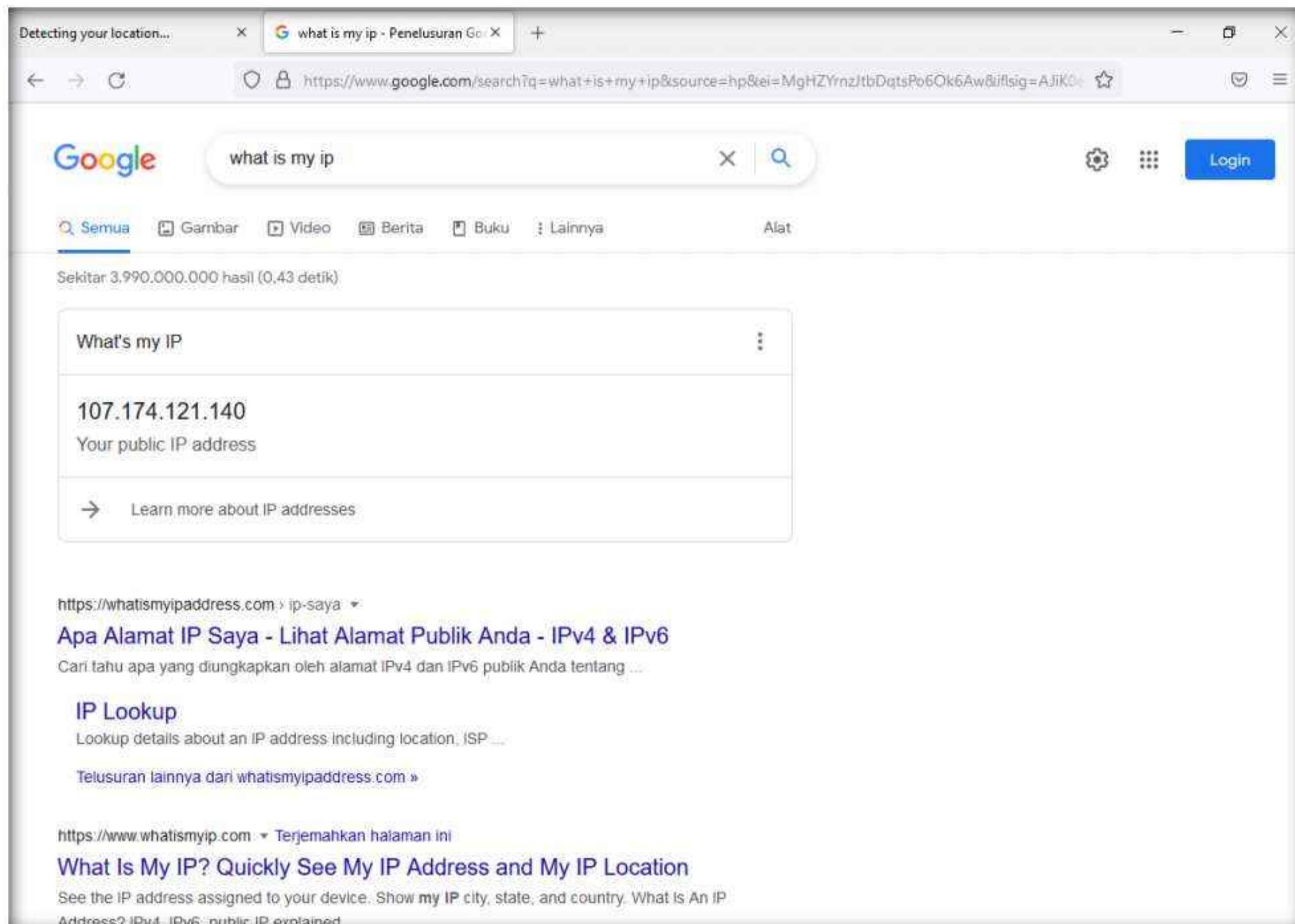


21. Launch the **Mozilla Firefox** web browser and enter the URL <http://www.proxyswitcher.com/check.php> to check the selected proxy-server connectivity. If the connection is successful, the following information is displayed in the browser:

Note: The information displayed above may differ in your lab environment.

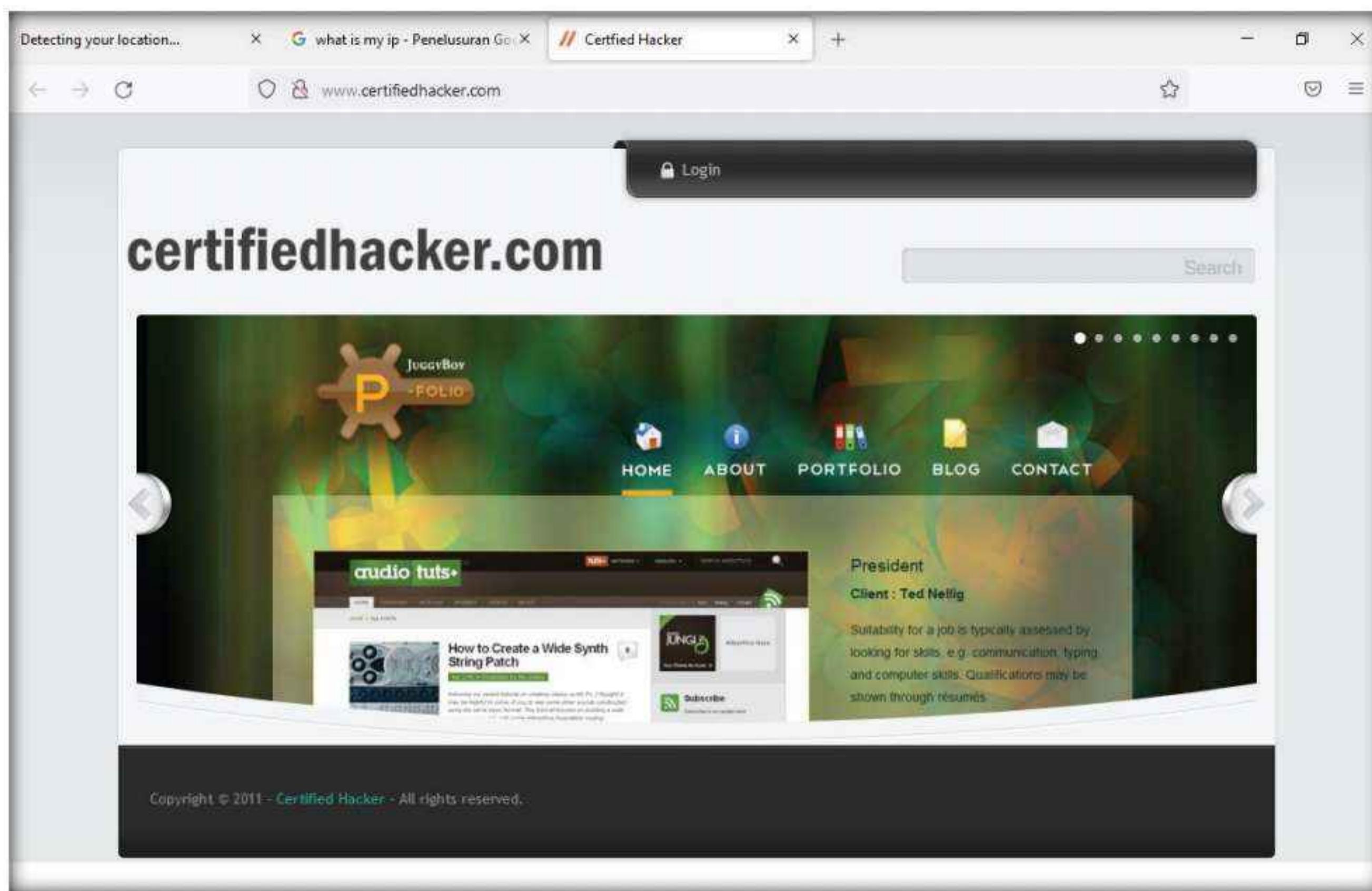


22. If the connection is unsuccessful, try selecting another proxy from **Proxy Switcher**, and repeat **Step 19**.
23. To ensure that the proxy is assigned, open a new tab and browse <https://www.google.com/>. In the search field, type **What is my ip** and press **Enter**.
24. If **About this page** webpage appears, check **I'm not a robot** checkbox and verify the CAPTCHA by selecting images as per the given guidelines.
25. The proxy IP address is displayed, which infers that the legitimate address is masked, and the proxy is in use.



Note: The displayed IP address might differ in your lab environment.

26. Open a new tab in your web browser and surf anonymously using this proxy.



27. This concludes the demonstration of anonymously surfing the Internet using Proxy Switcher.
28. Close all open windows and document all the acquired information.
29. Navigate to **Control Panel → Programs → Programs and Features** and uninstall the **Proxy Switcher** application.

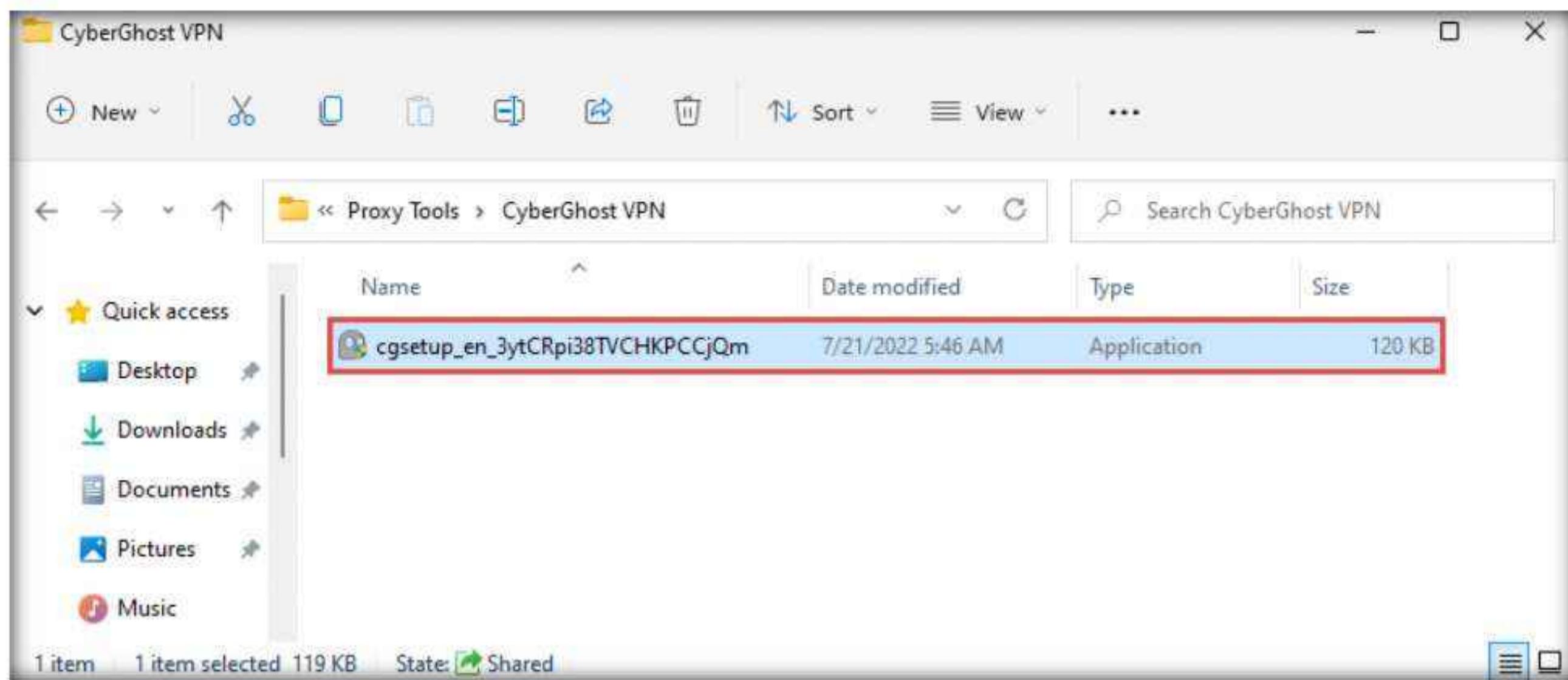
Task 5: Browse Anonymously using CyberGhost VPN

CyberGhost VPN hides the attacker's IP and replaces it with a selected IP, allowing him or her to surf anonymously and access blocked or censored content. It encrypts the connection and does not keep logs, thus securing data.

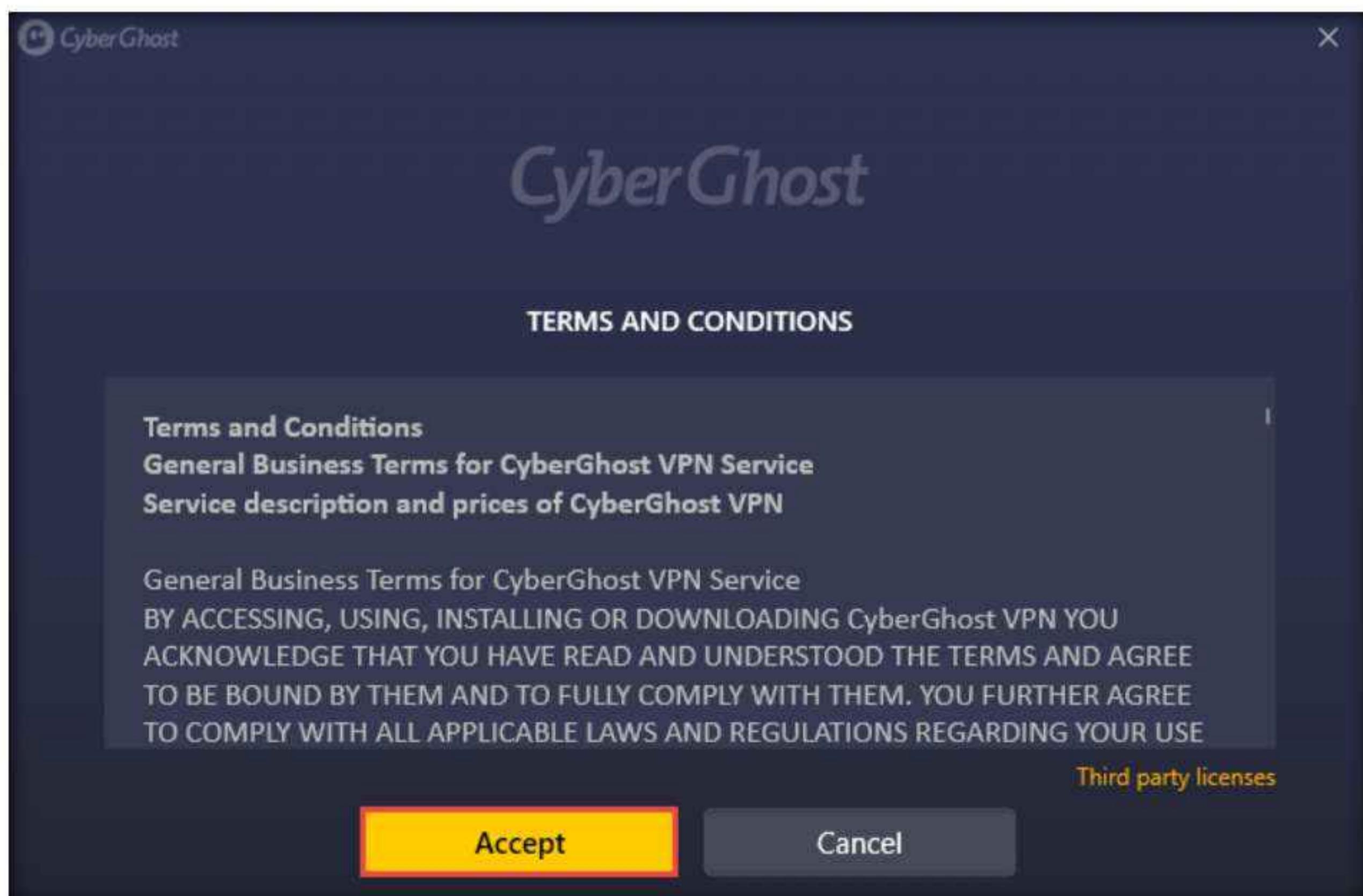
Here, we will use CyberGhost VPN to browse the Internet anonymously.

1. In the **Windows 11** virtual machine, navigate to **E:\CEH-Tools\CEHv12 Module 03 Scanning Networks\Proxy Tools\CyberGhost VPN** and double-click **cgsetup_en_3ytCRpi38TVCHKPCCQm.exe**.

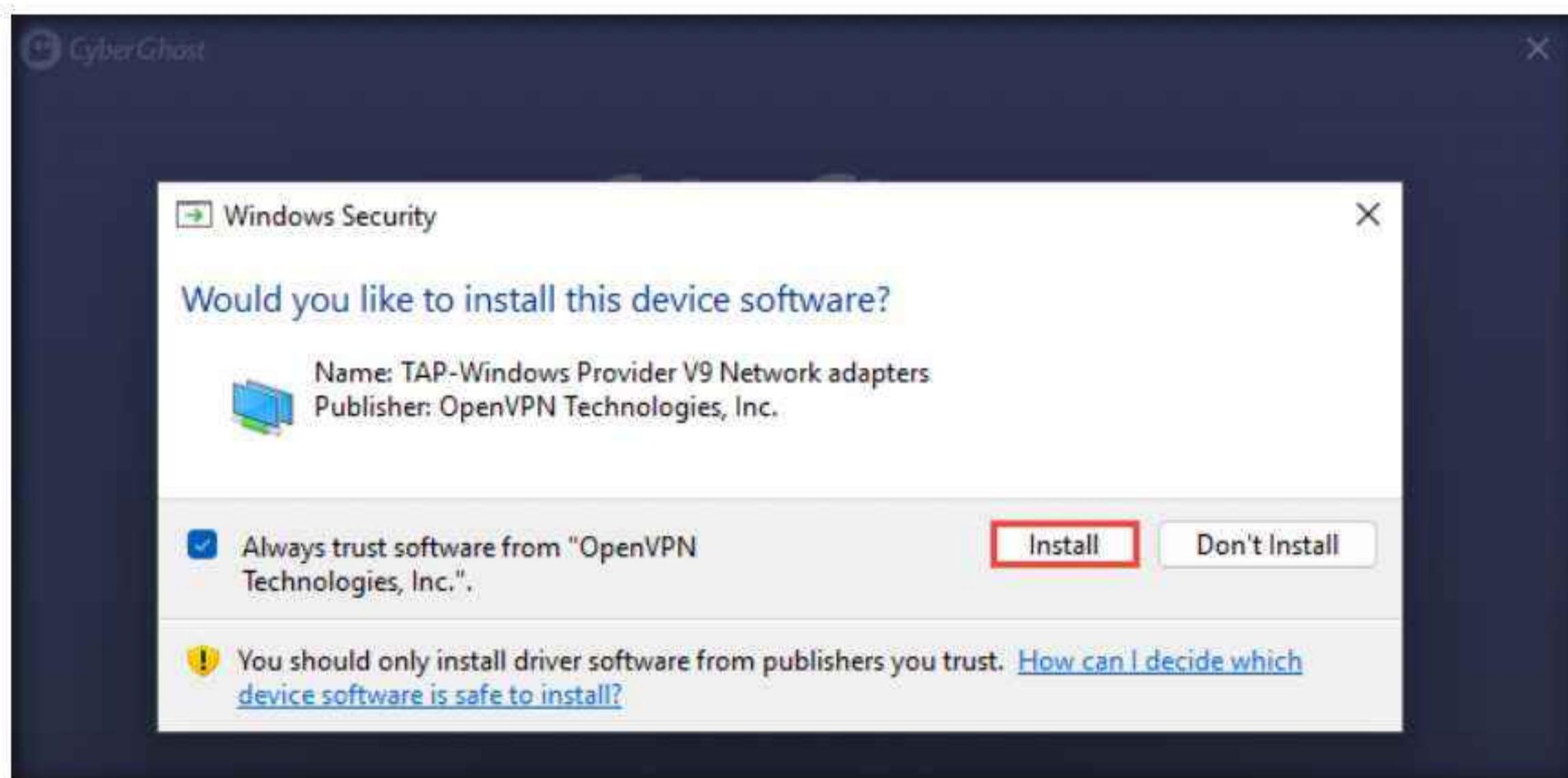
Note: If a **User Account Control** window appears, click **Yes**.



2. Downloading CyberGhost installer... appears; once the CyberGhost Setup window appears, click Accept.



3. Follow the installation steps to install **CyberGhost**.
4. In a **Windows Security** pop-up, click **Install**.



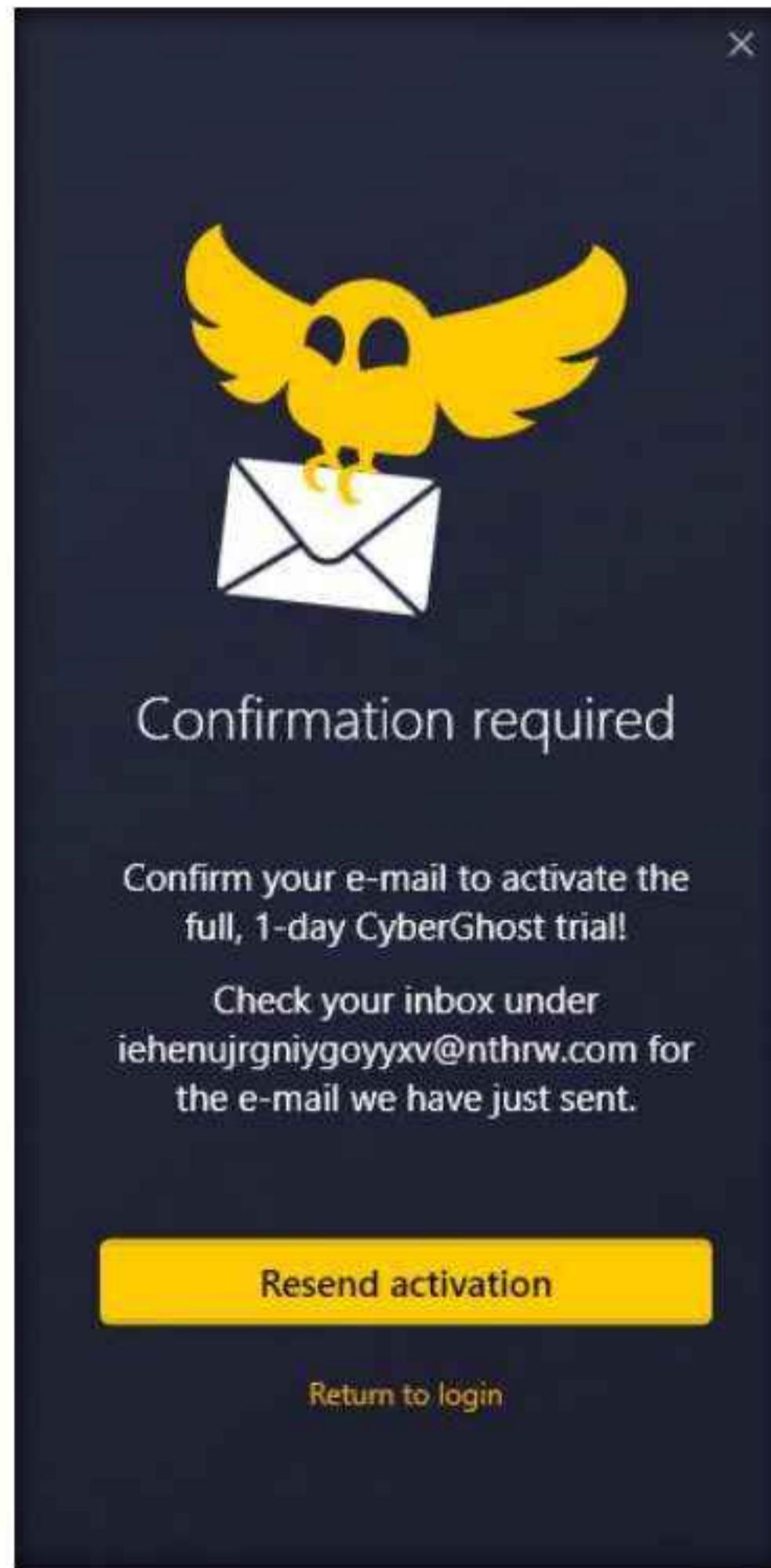
5. In the **Your privacy is our goal** pop-up, click **Agree and continue**.
6. Once the installation is complete, the **CyberGhost8** window appears, click on **Click here to create one** link to create an account.



7. Create an account using your personal details and click on **Sign Up**.



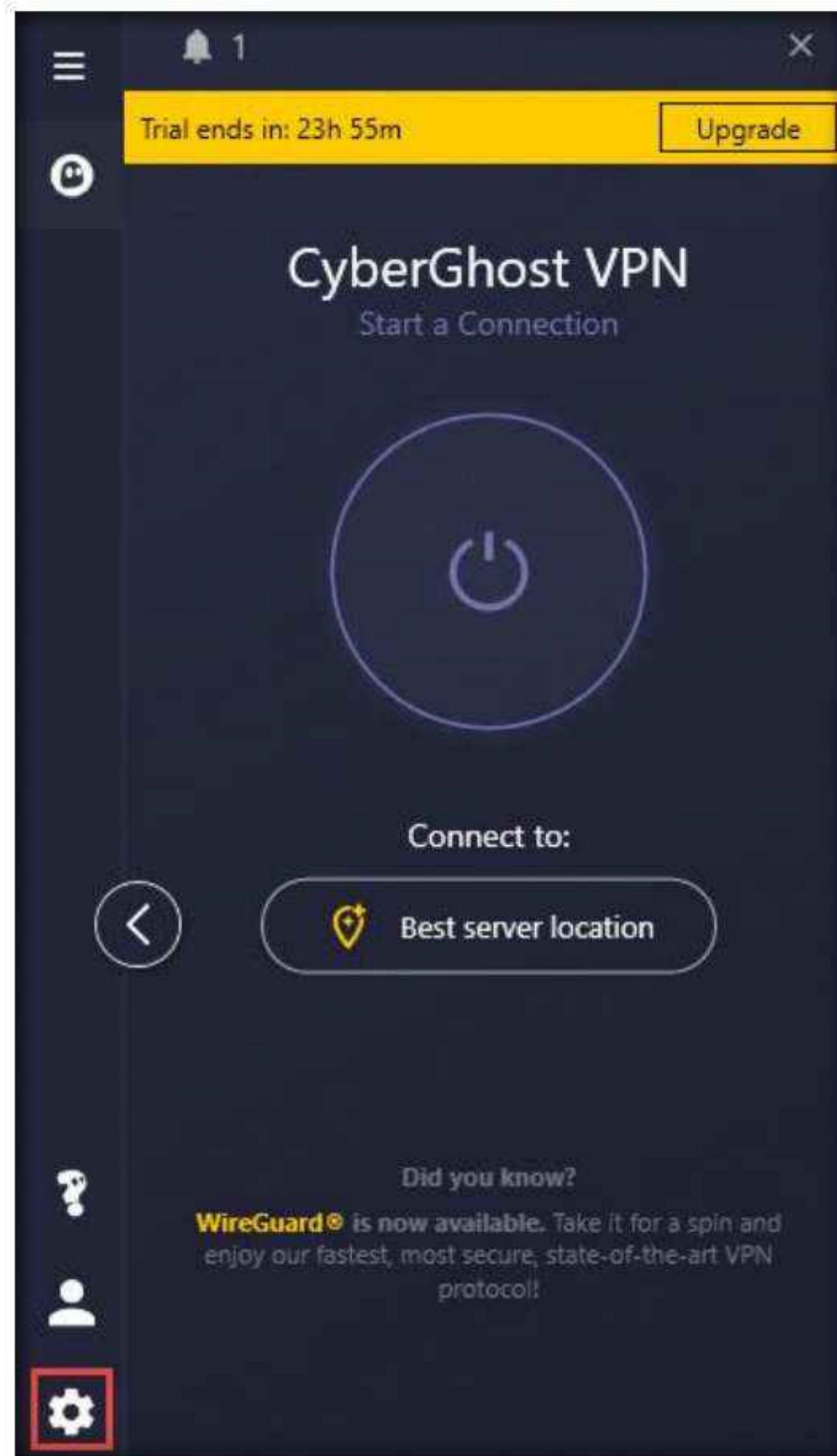
8. You will receive an activation email on your personal email. Open the email and click on **Activate Trial** to start your trial version of CyberGhost.



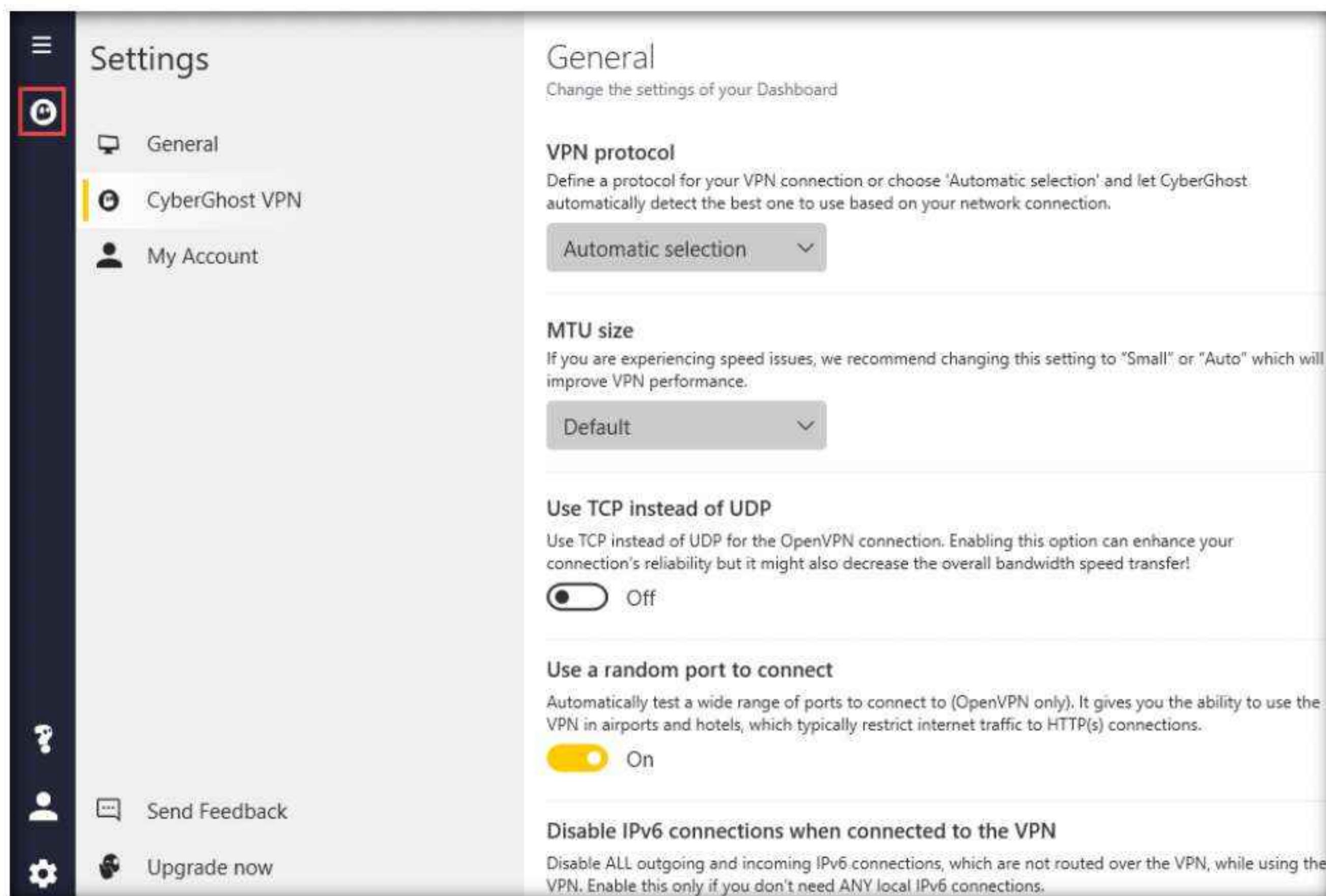
9. Now, switch to the CyberGhost page and click on **Start trial** button.



10. The CyberGhost VPN window appears, click the **Settings** icon.



11. The **Settings** window appears, click on **CyberGhost VPN** icon (VPN icon) under **Menu** icon



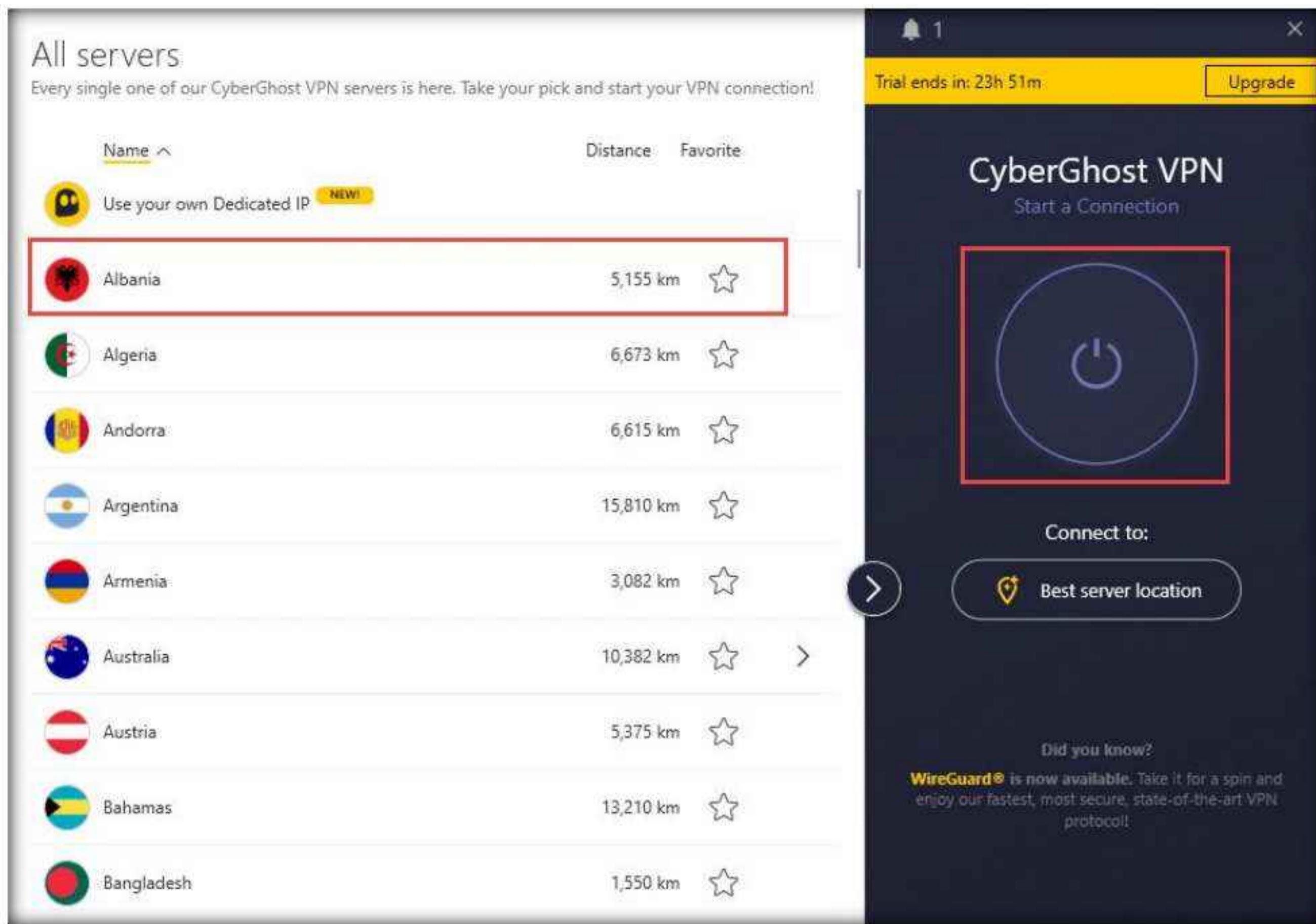
12. The **CyberGhost VPN** window appears; click on **All servers** from the left-hand pane.

Note: The list of the servers may vary in your lab environment

The screenshot shows the CyberGhost VPN application window. On the left, there's a sidebar with icons for Favorites, All servers (which is selected and highlighted with a red border), Dedicated IP, For Gaming, For torrenting, For streaming, Privacy settings, Smart rules, and Upgrade now. The main pane is titled "Favorites" and says "All your starred CyberGhost VPN servers are right here and ready to cater to your every need." It lists servers under "All servers" and "For streaming".

Name	Favorite
Use your own Dedicated IP <small>NEW!</small>	
France	★
Germany	★
United States	★
United Kingdom Optimized for: BBC iPlayer	★
United States Optimized for: Netflix US	★

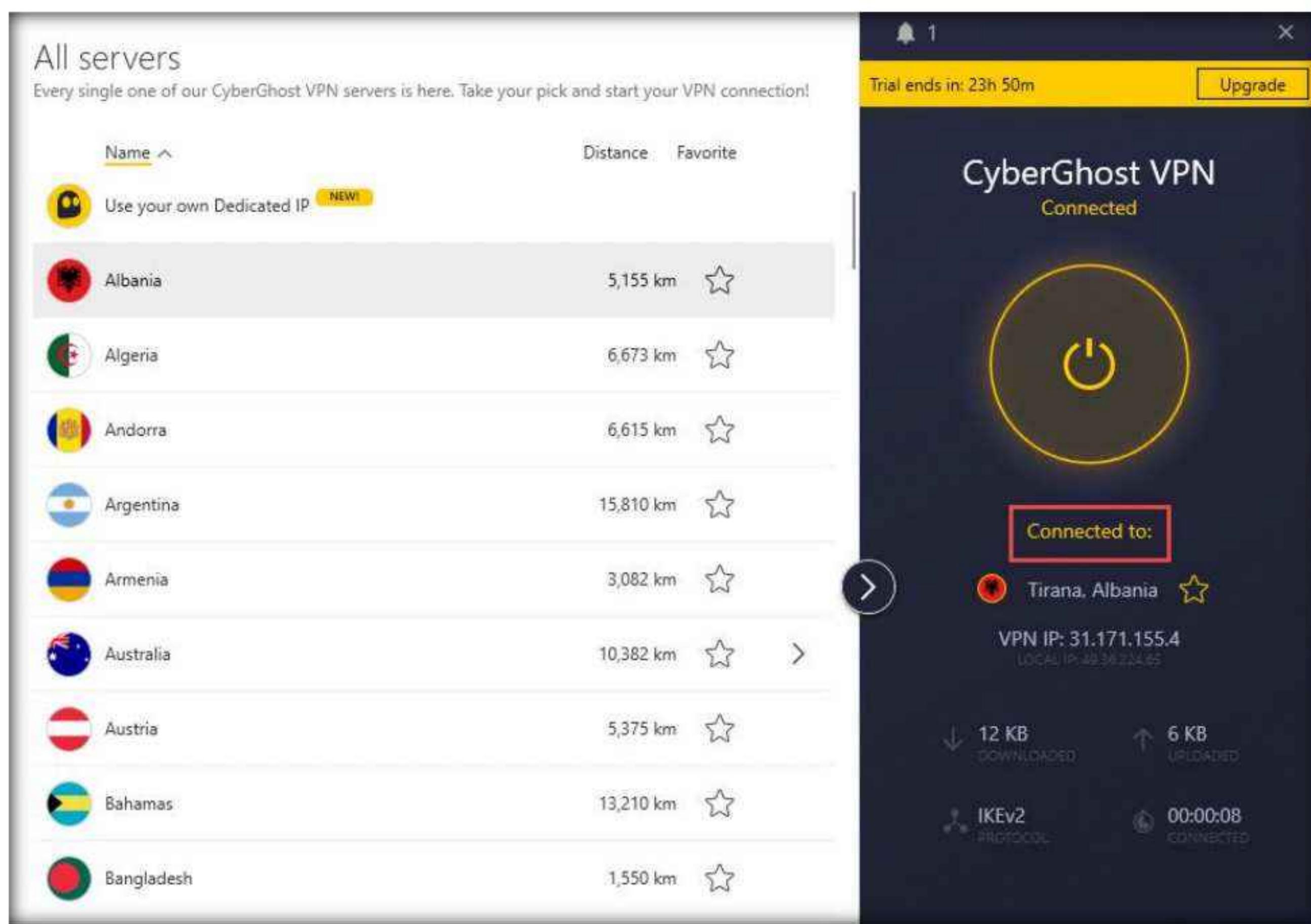
13. Click to select any proxy server from the available options in the **All servers** section (here, **Albania**) and click on the power icon () under **Start a Connection** as shown in the screenshot.



Note: If the CyberGhost window appears indicating that all free user slots are booked, then close the window and select another proxy server from the “all servers” list.

Module 03 – Scanning Networks

14. CyberGhost attempts to establish a connection to the proxy server. On successfully establishing a connection, **Connected** appears.



15. Minimize the **CyberGhost** window and launch the Mozilla Firefox web browser; type the URL <https://whatismyipaddress.com/location-feedback> in the address bar and press **Enter**.

Note: If a **Will you allow whatismyipaddress.com to access your location?** pop-up appears, click **Allow Location Access**.

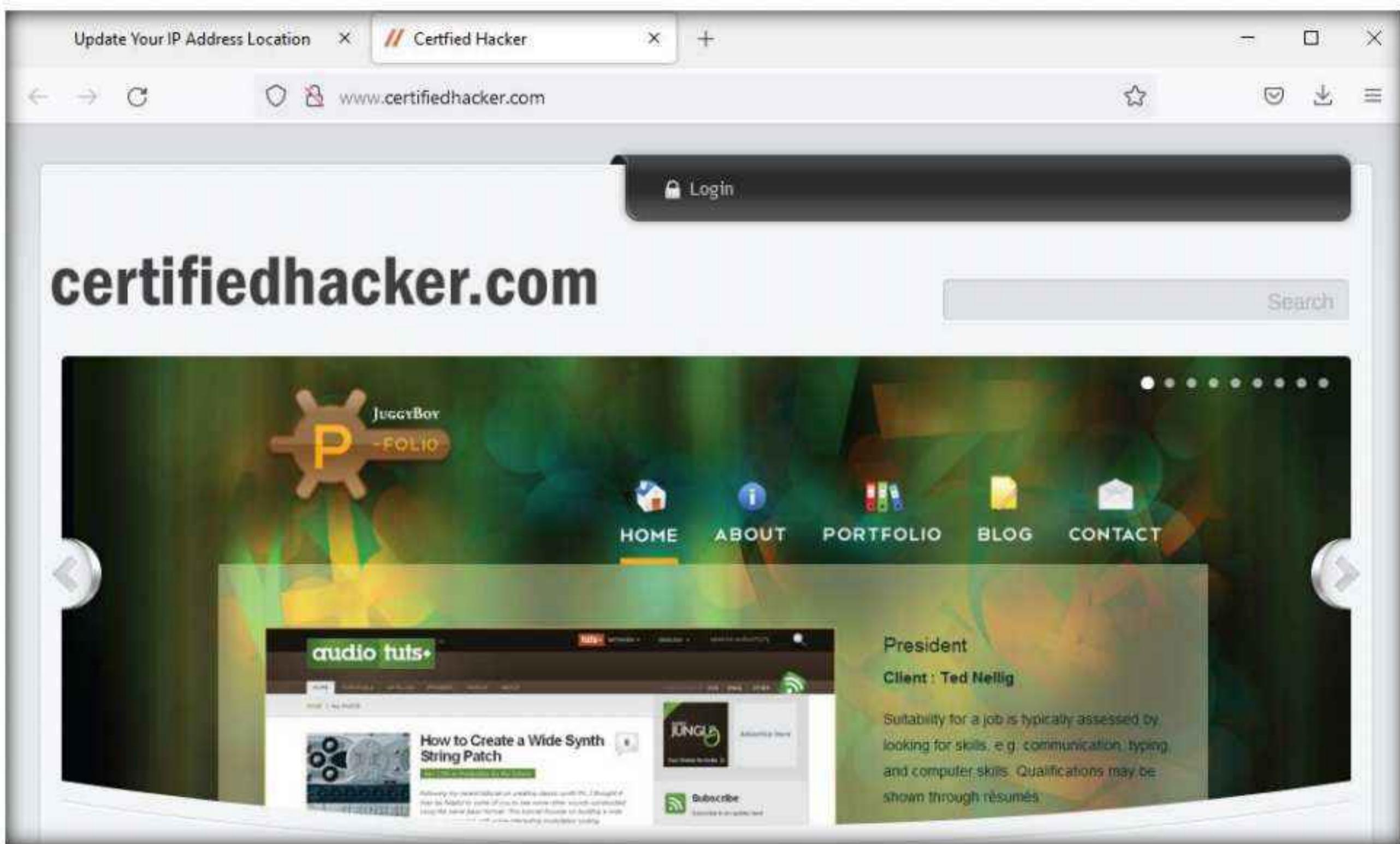
16. Scroll down to the **Geographical Details** section. Observe that the server IP address and location has changed to **31.171.155.4** and **Albania**.

The screenshot shows a web browser window for the site <https://whatismyipaddress.com/location-feedback>. At the top, there's a search bar with "Enter Keywords or IP Address..." and a "Search" button. Below the search bar are links for "ABOUT", "PRESS", "BLOG", and "CONTACT". The main content area features a map of Europe and the Middle East, with a red marker indicating the location as "Turkey". Below the map, the title "Geolocation Details for 31.171.155.4" is displayed, followed by a table comparing geolocation data from five sources (W3C, Provider A, Provider B, Provider C, Provider D). The table shows the following data:

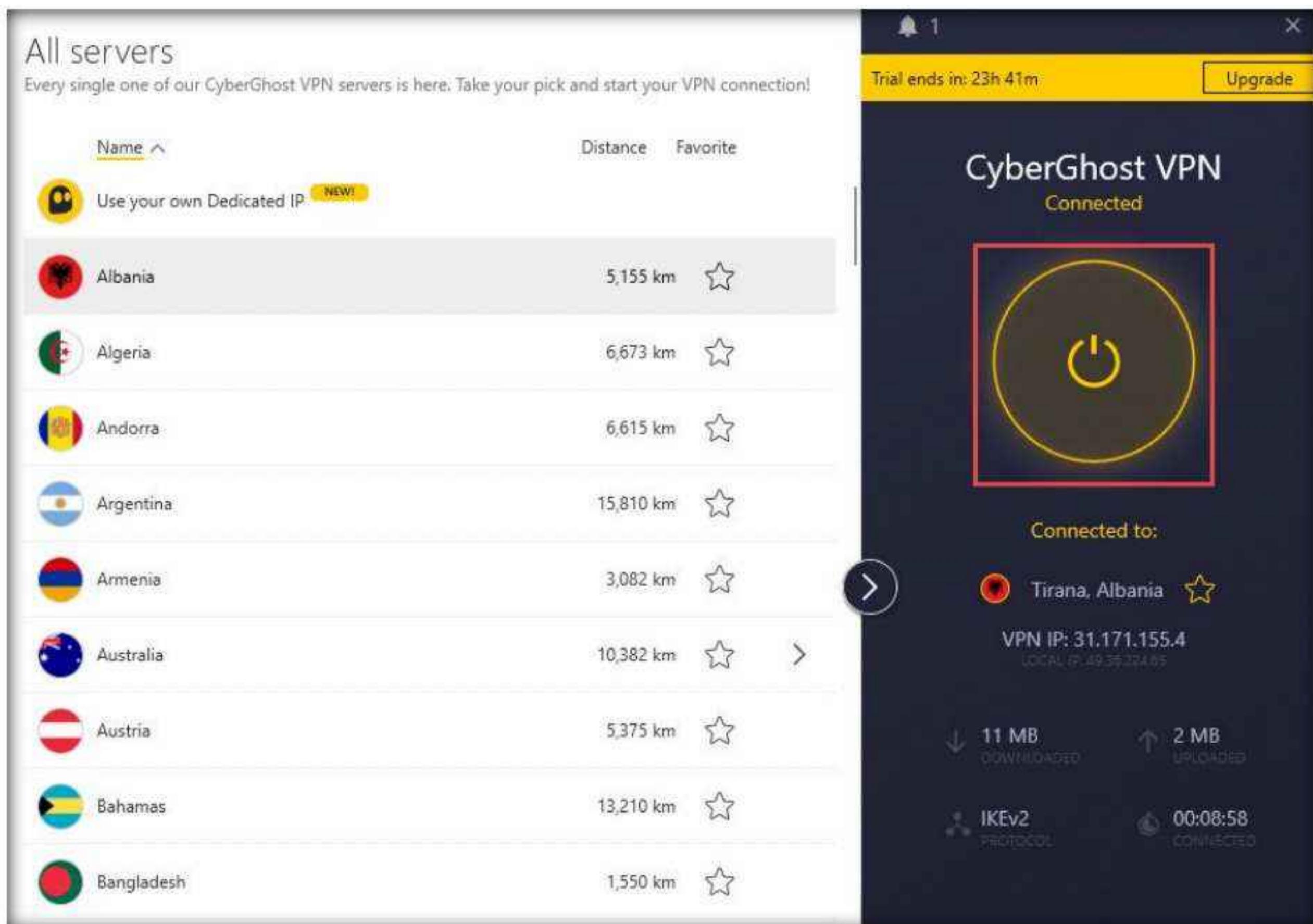
	W3C	Provider A	Provider B	Provider C	Provider D
Latitude	38.9637	0	41.3275	41.3275	
Longitude	35.2433	0	19.8189	19.8189	
Accuracy (m)	462936				
Country		0	albania	albania	
State/Region		0	tirane	tirane	
City		0	tirana	tirana	
Organization		0		keminet shpk	
ISP		0	keminet shpk		

At the bottom of the geolocation details section is a red "UPDATE GEOLOCATION" button and a small bell icon.

17. Open a new tab in the web browser and surf anonymously using this proxy.



18. Once you are done browsing, in the CyberGhost window, click the Power icon to disconnect the proxy, as shown in the screenshot.



19. This concludes the demonstration of anonymously surfing the Internet using CyberGhost.
20. Close all open windows and document all the acquired information.
21. Navigate to **Control Panel → Programs → Programs and Features** and uninstall the **CyberGhost 8** application.
22. Turn off the **Windows 11** virtual machine.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input checked="" type="checkbox"/> Yes	<input type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> CyberQ

Lab**5**

Perform Network Scanning using Various Scanning Tools

Ethical hackers and pen testers are aided in network scanning with the help of various scanning tools, which make scanning a target network an easy task.

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

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Android virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 20 Minutes

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.

Lab Tasks

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. Before beginning this task, turn on the **Windows 11, Windows Server 2022, Windows Server 2019, Ubuntu, Parrot Security** and **Android** virtual machines.
2. Switch to the **Parrot Security** virtual machine.
3. 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.

4. Click the **MATE Terminal** icon in the top of the **Desktop** to open a **Terminal** window.
 5. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.
 6. 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.
7. Now, type **cd** and press **Enter** to jump to the root directory.

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

```
[attacker@parrot] ~
$ sudo su
[sudo] password for attacker:
[root@parrot] ~
#cd
[root@parrot] ~
#service postgresql start
[root@parrot] ~
#
```

9. Now, type msfconsole and hit Enter to launch Metasploit.

```
[attacker@parrot] ~
$ sudo su
[sudo] password for attacker:
[root@parrot] ~
#cd
[root@parrot] ~
#service postgresql start
[root@parrot] ~
#msfconsole

Call trans opt: received. 2-19-98 13:24:18 REC:Loc

Trace program: running

      Wake up, Neo...
      the matrix has you
      follow the white rabbit.

      knock, knock, Neo.

      ODBC Module E
      Hacking Web
      Servers
```

10. 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.
11. Exit the Metasploit framework by typing **exit** and press **Enter**. Then, to initiate the database, type **msfdb init**, and press **Enter**.

The screenshot shows a terminal window titled "msfdb init - Parrot Terminal". The window contains the following text:

```
https://metasploit.com

      =[ metasploit v6.1.39-dev
+ --=[ 2214 exploits - 1171 auxiliary - 396 post
+ --=[ 618 payloads - 45 encoders - 11 nops
+ --=[ 9 evasion

Metasploit tip: View advanced module options with
advanced

msf6 > db_status
[*] postgresql selected, no connection
msf6 > exit
[-] [root@parrot]-
[-] #msfdb init
[i] Database already started
[+] Creating database user 'msf'
[+] Creating databases 'msf'
[+] Creating databases 'msf test'
[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'
[+] Creating initial database schema
[-] [root@parrot]-
[-] #
```

12. To restart the postgresql service, type **service postgresql restart** and press **Enter**. Now, start the Metasploit Framework again by typing **msfconsole** and pressing **Enter**.
13. 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.

The screenshot shows a terminal window titled "msfconsole - Parrot Terminal". The terminal is running on a Parrot OS desktop environment, indicated by the desktop icons in the background. The terminal window has a dark theme with green text. It displays the following command-line session:

```
[+] Creating databases 'msf'  
[+] Creating databases 'msf test'  
[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'  
[+] Creating initial database schema  
[root@parrot] ~ [~]  
└─# service postgresql restart  
[root@parrot] ~ [~]  
└─# msfconsole  
  
# cowsay++  
  
< metasploit >  
-----  
 \  'oo'  
  (_____)  
   ||--|| *  
  
       =[ metasploit v6.1.39-dev  
+ ..-= [ 2214 exploits - 1171 auxiliary - 396 post  
+ ..-= [ 618 payloads - 45 encoders - 11 nops  
+ ..-= [ 9 evasion  
  
Metasploit tip: Writing a custom module? After editing your  
module, why not try the reload command  
  
msf6 > db_status  
[*] Connected to msf. Connection type: postgresql.  
msf6 > █
```

14. 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.

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

The screenshot shows a terminal window titled "msfconsole - Parrot Terminal". The command entered was "nmap -Pn -sS -A -oX Test 10.10.1.0/24". The output shows the following details:

```
[*] Connected to msf. Connection type: postgresql.
[*] exec: nmap -Pn -sS -A -oX Test 10.10.1.0/24
[*] exec: nmap -Pn -sS -A -oX Test 10.10.1.0/24

Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-07 01:24 EDT
Nmap scan report for 10.10.1.1
Host is up (0.00049s latency).
All 1000 scanned ports on 10.10.1.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: E8:D1:48:1D:BC:B6 (Unknown)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1  0.49 ms  10.10.1.1

Nmap scan report for 10.10.1.9
Host is up (0.013s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 28:52:84:53:60:ec:72:72:ce:80:ba:db:35:74:b5:55 (ECDSA)
|   256 9a:1e:e9:21:07:9f:7c:25:95:c9:6a:b6:5e:fe:e4:51 (ED25519)
|_ 80/tcp    open  http     Apache httpd 2.4.52 ((Ubuntu))
|_ http-server-header: Apache/2.4.52 (Ubuntu)
|_ http-title: Apache2 Ubuntu Default Page: It works
MAC Address: 58:DD:75:00:1F:1E (Unknown)
Device type: general purpose
```

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

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

```

msfconsole - Parrot Terminal

File Edit View Search Terminal Help
TRACEROUTE
HOP RTT ADDRESS
1 4.41 ms 10.10.1.22

Nmap scan report for 10.10.1.13
Host is up (0.036s latency).
All 1000 scanned ports on 10.10.1.13 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

Post-scan script results:
| clock-skew:
  8h23m59s:
    10.10.1.11
    10.10.1.22

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 256 IP addresses (7 hosts up) scanned in 149.71 seconds
msf6 > db import Test
[*] Importing 'Nmap XML' data
[*] Import: Parsing with 'Nokogiri v1.13.4'
[*] 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 >

```

18. 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.

```

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:bc:b6
10.10.1.9    58:dd:75:00:1f:1e
10.10.1.11   b4:b5:78:89:75:64
10.10.1.13   Unknown
10.10.1.14   3e:7d:4f:2c:4b:d7
10.10.1.19   ac:90:49:48:6f:e6   www.moviesco pe.com
10.10.1.22   1c:5a:12:d9:10:bd

msf6 >

```

19. 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.

The screenshot shows a terminal window titled "msfconsole - Parrot Terminal". The command "services" has been entered, and the output displays a table of open services across multiple hosts. The columns are host, port, proto, name, state, and info. The table includes entries for various operating systems like Ubuntu, Windows, and Microsoft IIS, along with specific services like ssh, http, msrpc, netbios-ssn, and microsoft-ds.

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 workgroup: 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); device: 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

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

The screenshot shows the msfconsole interface on a Parrot OS terminal window. The command `search portscan` has been entered, and the results are displayed under the heading "Matching Modules". The output lists seven port scanning modules, each with a name, disclosure date, rank, check status, and a brief description. The modules are:

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/scanner/portscan/ftpbounce	normal	No		FTP Bounce Po...
1	auxiliary/scanner/natpmp/natpmp_portscanner	normal	No		NAT-PMP Exter...
2	auxiliary/scanner/sap/sap_router_portscanner	normal	No		SAPRouter Por...
3	auxiliary/scanner/portscan/xmas	normal	No		TCP "XMas" Po...
4	auxiliary/scanner/portscan/ack	normal	No		TCP ACK Firew...
5	auxiliary/scanner/portscan/tcp	normal	No		TCP Port Scan...
6	auxiliary/scanner/portscan/syn	normal	No		TCP SYN Port...
7	auxiliary/scanner/http/wordpress_pingback_access	normal	No		Wordpress Pin...

Interact with a module by name or index. For example info 7, use 7 or use auxiliary/scanner/http/wordpress_pingback_access

21. 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**.

22. 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:

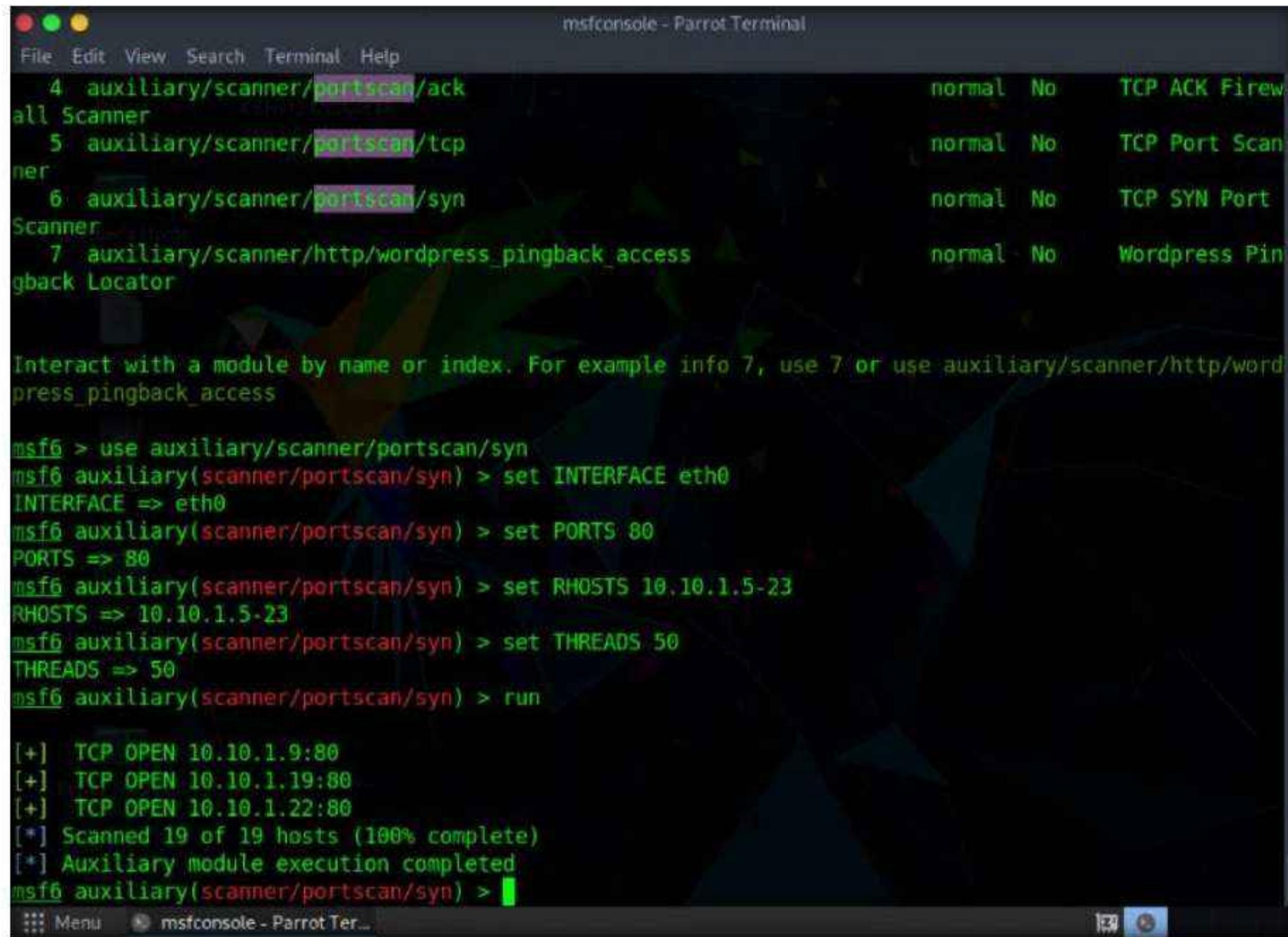
- **set INTERFACE eth0**
- **set PORTS 80**
- **set RHOSTS 10.10.1.5-23**
- **set THREADS 50**

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).

23. 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.

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



The screenshot shows the msfconsole interface on a Parrot OS terminal. The user has loaded the auxiliary/scanner/portscan/syn module. They have set the INTERFACE to eth0, the PORTS to 80, and the RHOSTS to 10.10.1.5-23. They have also set THREADS to 50. After running the module, the console displays the results of the scan, showing three open TCP ports: 10.10.1.9:80, 10.10.1.19:80, and 10.10.1.22:80. The message indicates that the scan was 100% complete and auxiliary module execution completed.

```

msfconsole - Parrot Terminal
File Edit View Search Terminal Help
 4 auxiliary/scanner/portscan/ack
 5 auxiliary/scanner/portscan/tcp
 6 auxiliary/scanner/portscan/syn
 7 auxiliary/scanner/http/wordpress_pingback_access
Scanner
normal No TCP ACK Firewall
normal No TCP Port Scan
normal No TCP SYN Port
normal No Wordpress Pin
gback Locator

Interact with a module by name or index. For example info 7, use 7 or use auxiliary/scanner/http/word
press_pingback_access

msf6 > use auxiliary/scanner/portscan/syn
msf6 auxiliary(scanner/portscan/syn) > set INTERFACE eth0
INTERFACE => eth0
msf6 auxiliary(scanner/portscan/syn) > set PORTS 80
PORTS => 80
msf6 auxiliary(scanner/portscan/syn) > set RHOSTS 10.10.1.5-23
RHOSTS => 10.10.1.5-23
msf6 auxiliary(scanner/portscan/syn) > set THREADS 50
THREADS => 50
msf6 auxiliary(scanner/portscan/syn) > run

[+] TCP OPEN 10.10.1.9:80
[+] TCP OPEN 10.10.1.19:80
[+] TCP OPEN 10.10.1.22:80
[*] Scanned 19 of 19 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/syn) >
  Menu msfconsole - Parrot Ter...

```

25. Now, we will perform a TCP scan for open ports on the target systems.
26. To load the **auxiliary/scanner/portscan/tcp** module, type **use auxiliary/scanner/portscan/tcp** and press **Enter**.
27. 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.
28. 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.
29. The results appear, displaying all open TCP ports in the target IP address (10.10.1.22).

The screenshot shows a terminal window titled "msfconsole - Parrot Terminal". The terminal displays the results of a port scan. The output includes:

```
[+] TCP OPEN 10.10.1.22:80
[*] Scanned 19 of 19 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/syn) > use auxiliary/scanner/portscan/tcp
msf6 auxiliary(scanner/portscan/tcp) > set RHOSTS 10.10.1.22
RHOSTS => 10.10.1.22
msf6 auxiliary(scanner/portscan/tcp) > run

[+] 10.10.1.22:          - 10.10.1.22:53 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:80 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:88 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:135 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:139 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:389 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:445 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:464 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:593 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:636 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:1801 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:2105 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:2103 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:2107 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:3269 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:3268 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:3389 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:5985 - TCP OPEN
[+] 10.10.1.22:          - 10.10.1.22:9389 - TCP OPEN
[*] 10.10.1.22:          - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/tcp) >
```

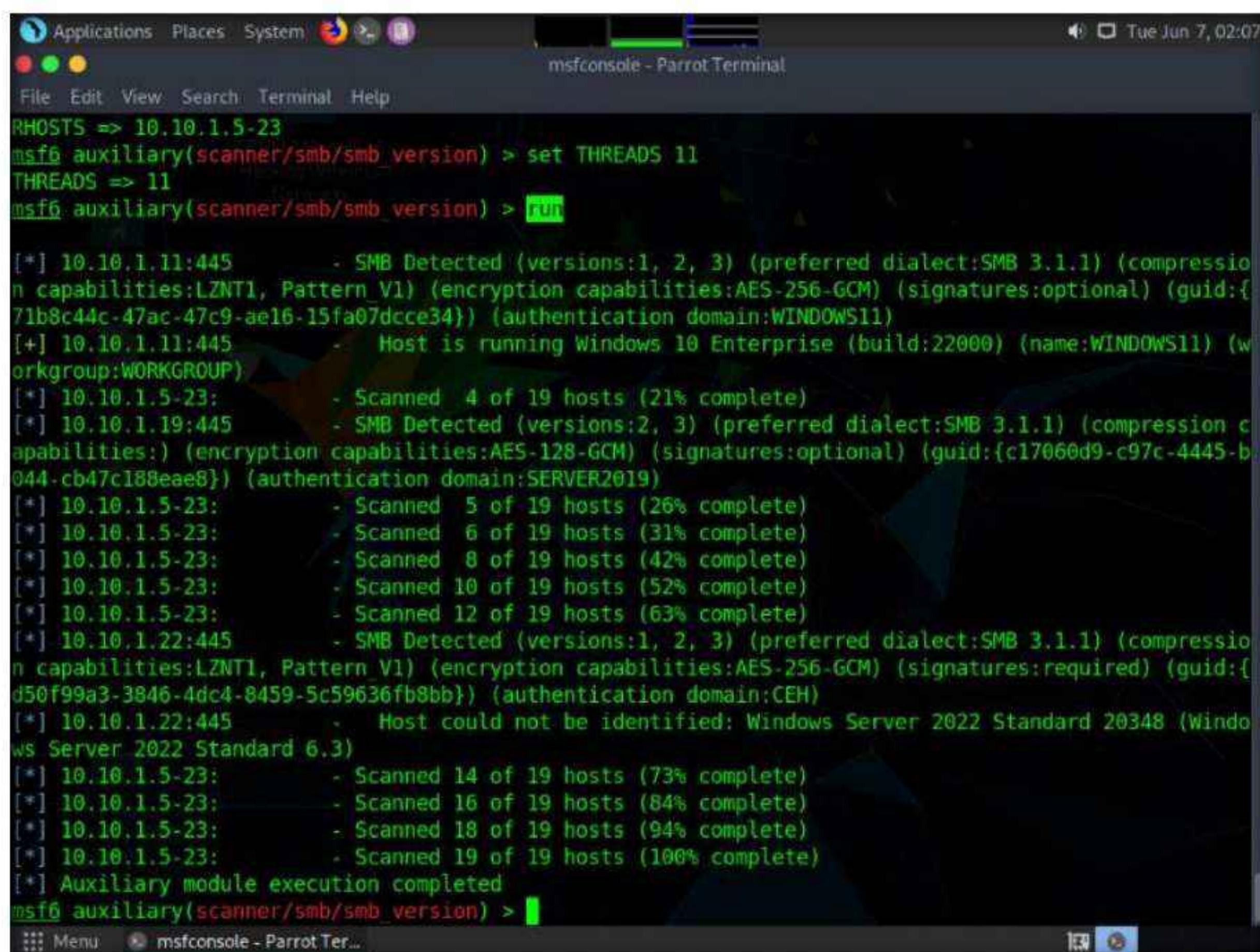
30. Now that we have determined the active hosts on the target network, we can further attempt to determine the OSes running on the target systems. As there are systems in our scan that have port 445 open, we will use the module scanner/smb/version to determine which version of Windows is running on a target and which Samba version is on a Linux host.
31. 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**.
32. 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**

Module 03 – Scanning Networks

```
[+] 10.10.1.22: - 10.10.1.22:3269 - TCP OPEN
[+] 10.10.1.22: - 10.10.1.22:3268 - TCP OPEN
[+] 10.10.1.22: - 10.10.1.22:3389 - TCP OPEN
[+] 10.10.1.22: - 10.10.1.22:5985 - TCP OPEN
[+] 10.10.1.22: - 10.10.1.22:9389 - TCP OPEN
[*] 10.10.1.22: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/tcp) > back
msf6 > use auxiliary/scanner/smb/smb_version
msf6 auxiliary(scanner/smb/smb_version) > set RHOSTS 10.10.1.5-23
RHOSTS => 10.10.1.5-23
msf6 auxiliary(scanner/smb/smb_version) > set THREADS 11
THREADS => 11
msf6 auxiliary(scanner/smb/smb_version) > [REDACTED]
::: Menu  msfconsole - Parrot Ter...[REDACTED]
```

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

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



The screenshot shows the msfconsole interface on a Parrot OS terminal window. The title bar reads "msfconsole - Parrot Terminal". The command "set RHOSTS 10.10.1.5-23" has been entered. The "auxiliary(scanner/smb/smb_version)" module is selected, and the "set THREADS 11" command has been run. The output shows the SMB version detection process for multiple hosts:

```
RHOSTS => 10.10.1.5-23
msf6 auxiliary(scanner/smb/smb_version) > set THREADS 11
THREADS => 11
msf6 auxiliary(scanner/smb/smb_version) > run

[*] 10.10.1.11:445 - SMB Detected (versions:1, 2, 3) (preferred dialect:SMB 3.1.1) (compression capabilities:LZNT1, Pattern V1) (encryption capabilities:AES-256-GCM) (signatures:optional) (guid:{71b8c44c-47ac-47c9-ael6-15fa07dcce34}) (authentication domain:WINDOWS11)
[+] 10.10.1.11:445 - Host is running Windows 10 Enterprise (build:22000) (name:WINDOWS11) (workgroup:WORKGROUP)
[*] 10.10.1.5-23: - Scanned 4 of 19 hosts (21% complete)
[*] 10.10.1.19:445 - SMB Detected (versions:2, 3) (preferred dialect:SMB 3.1.1) (compression capabilities:) (encryption capabilities:AES-128-GCM) (signatures:optional) (guid:{c17060d9-c97c-4445-b044-cb47c188eae8}) (authentication domain:SERVER2019)
[*] 10.10.1.5-23: - Scanned 5 of 19 hosts (26% complete)
[*] 10.10.1.5-23: - Scanned 6 of 19 hosts (31% complete)
[*] 10.10.1.5-23: - Scanned 8 of 19 hosts (42% complete)
[*] 10.10.1.5-23: - Scanned 10 of 19 hosts (52% complete)
[*] 10.10.1.5-23: - Scanned 12 of 19 hosts (63% complete)
[*] 10.10.1.22:445 - SMB Detected (versions:1, 2, 3) (preferred dialect:SMB 3.1.1) (compression capabilities:LZNT1, Pattern V1) (encryption capabilities:AES-256-GCM) (signatures:required) (guid:{d50f99a3-3846-4dc4-8459-5c59636fb8bb}) (authentication domain:CEH)
[*] 10.10.1.22:445 - Host could not be identified: Windows Server 2022 Standard 20348 (Windows Server 2022 Standard 6.3)
[*] 10.10.1.5-23: - Scanned 14 of 19 hosts (73% complete)
[*] 10.10.1.5-23: - Scanned 16 of 19 hosts (84% complete)
[*] 10.10.1.5-23: - Scanned 18 of 19 hosts (94% complete)
[*] 10.10.1.5-23: - Scanned 19 of 19 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_version) > [REDACTED]
::: Menu  msfconsole - Parrot Ter...[REDACTED]
```

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

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

37. 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.

38. Close all open windows and document all the acquired information.
39. Turn off the **Windows 11, Windows Server 2022, Windows Server 2019, Ubuntu, Parrot Security** and **Android** virtual machines.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required	
<input type="checkbox"/> Yes	<input checked="" type="checkbox"/> No
Platform Supported	
<input checked="" type="checkbox"/> Classroom	<input checked="" type="checkbox"/> CyberQ

CEH Lab Manual

Enumeration

Module 04

Enumeration

Enumeration is the process of extracting usernames, machine names, network resources, shares, and services from a system or network.

Lab Scenario

With the development of network technologies and applications, network attacks are greatly increasing in both number and severity. Attackers continuously search for service and application vulnerabilities on networks and servers. When they find a flaw or loophole in a service run over the Internet, they immediately exploit it to compromise the entire system. Any other data that they find may be further used to compromise additional network systems. Similarly, attackers seek out and use workstations with administrative privileges, and which run flawed applications, to execute arbitrary code or implant viruses in order to intensify damage to the network.

In the first step of the security assessment and penetration testing of your organization, you gather open-source information about your organization. In the second step, you collect information about open ports and services, OSes, and any configuration lapses.

The next step for an ethical hacker or penetration tester is to probe the target network further by performing enumeration. Using various techniques, you should extract more details about the network such as lists of computers, usernames, user groups, ports, OSes, machine names, network resources, and services.

The information gleaned from enumeration will help you to identify the vulnerabilities in your system's security that attackers would seek to exploit. Such information could also enable attackers to perform password attacks to gain unauthorized access to information system resources.

In the previous steps, you gathered necessary information about a target without contravening any legal boundaries. However, please note that enumeration activities may be illegal depending on an organization's policies and any laws that are in effect in your location. As an ethical hacker or penetration tester, you should always acquire proper authorization before performing enumeration.

Lab Objective

The objective of the lab is to extract information about the target organization that includes, but is not limited to:

- Machine names, their OSes, services, and ports
- Network resources
- Usernames and user groups
- Lists of shares on individual hosts on the network
- Policies and passwords

- Routing tables
- Audit and service settings
- SNMP and FQDN details

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Android virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 145 Minutes

Overview of Enumeration

Enumeration creates an active connection with the system and performs directed queries to gain more information about the target. It extracts lists of computers, usernames, user groups, ports, OSes, machine names, network resources, and services using various techniques. Enumeration techniques are conducted in an intranet environment.

Lab Tasks

Ethical hackers or penetration testers use several tools and techniques to enumerate the target network. Recommended labs that will assist you in learning various enumeration techniques include:

Lab No.	Lab Exercise Name	Core*	Self-study**	CyberQ ***
1	Perform NetBIOS Enumeration	√	√	√
	1.1 Perform NetBIOS Enumeration using Windows Command-Line Utilities	√		√
	1.2 Perform NetBIOS Enumeration using NetBIOS Enumerator		√	√
	1.3 Perform NetBIOS Enumeration using an NSE Script		√	√

Module 04 – Enumeration

2	Perform SNMP Enumeration	√	√	√
	2.1 Perform SNMP Enumeration using snmp-check		√	√
	2.2 Perform SNMP Enumeration using SoftPerfect Network Scanner		√	√
	2.3 Perform SNMP Enumeration using SnmpWalk	√		√
	2.4 Perform SNMP Enumeration using Nmap		√	√
3	Perform LDAP Enumeration	√	√	√
	3.1 Perform LDAP Enumeration using Active Directory Explorer (AD Explorer)	√		√
	3.2 Perform LDAP Enumeration using Python and Nmap		√	√
	3.3 Perform LDAP Enumeration using ldapsearch		√	√
4	Perform NFS Enumeration	√		√
	4.1 Perform NFS Enumeration using RPCScan and SuperEnum	√		√
5	Perform DNS Enumeration	√	√	√
	5.1 Perform DNS Enumeration using Zone Transfer	√		√
	5.2 Perform DNS Enumeration using DNSSEC Zone Walking		√	√
	5.3 Perform DNS Enumeration using Nmap		√	√
6	Perform SMTP Enumeration	√		√
	6.1 Perform SMTP Enumeration using Nmap	√		√
7	Perform RPC, SMB, and FTP Enumeration		√	√
	7.1 Perform SMB and RPC Enumeration using NetScanTools Pro		√	√
	7.2 Perform RPC, SMB, and FTP Enumeration using Nmap		√	√
8	Perform Enumeration using Various Enumeration Tools	√	√	√
	8.1 Enumerate Information using Global Network Inventory	√		√
	8.2 Enumerate Network Resources using Advanced IP Scanner		√	√
	8.3 Enumerate Information from Windows and Samba Hosts using Enum4linux		√	√

Remark

EC-Council has prepared a considered amount of lab exercises for student to practice during the 5-day class and at their free time to enhance their knowledge and skill.

***Core** - Lab exercise(s) marked under Core are recommended by EC-Council to be practised during the 5-day class.

****Self-study** - Lab exercise(s) marked under self-study is for students to practise at their free time. Steps to access the additional lab exercises can be found in the first page of CEHv12 volume 1 book.

*****CyberQ** - Lab exercise(s) marked under CyberQ are available in our CyberQ solution. CyberQ is a cloud-based virtual lab environment preconfigured with vulnerabilities, exploits, tools and scripts, and can be accessed from anywhere with an Internet connection. If you are interested to learn more about our CyberQ solution, please contact your training center or visit <https://www.cyberq.io/>.

Lab Analysis

Analyze and document the results related to this lab exercise. Give an opinion on your target's security posture.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Lab

1

Perform NetBIOS Enumeration

NetBIOS enumeration is a process of obtaining sensitive information about the target such as a list of computers belonging to a target domain, network shares, policies, etc.

Lab Scenario

As a professional ethical hacker or penetration tester, your first step in the enumeration of a Windows system is to exploit the NetBIOS API. NetBIOS enumeration allows you to collect information about the target such as a list of computers that belong to a target domain, shares on individual hosts in the target network, policies, passwords, etc. This data can be used to probe the machines further for detailed information about the network and host resources.

Lab Objectives

- Perform NetBIOS enumeration using Windows command-line utilities
- Perform NetBIOS enumeration using NetBIOS Enumerator
- Perform NetBIOS enumeration using an NSE Script

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 15 Minutes

Overview of NetBIOS Enumeration

NetBIOS stands for Network Basic Input Output System. Windows uses NetBIOS for file and printer sharing. A NetBIOS name is a unique computer name assigned to Windows systems, comprising a 16-character ASCII string that identifies the network device over TCP/IP. The first 15 characters are used for the device name, and the 16th is reserved for the service or name record type.

The NetBIOS service is easily targeted, as it is simple to exploit and runs on Windows systems even when not in use. NetBIOS enumeration allows attackers to read or write to a remote computer system (depending on the availability of shares) or launch a denial of service (DoS) attack.

Lab Tasks

Task 1: Perform NetBIOS Enumeration using Windows Command-Line Utilities

Nbtstat helps in troubleshooting NETBIOS name resolution problems. The nbtstat command removes and corrects preloaded entries using several case-sensitive switches. Nbtstat can be used to enumerate information such as NetBIOS over TCP/IP (NetBT) protocol statistics, NetBIOS name tables for both the local and remote computers, and the NetBIOS name cache.

Net use connects a computer to, or disconnects it from, a shared resource. It also displays information about computer connections.

Here, we will use the Nbtstat, and Net use Windows command-line utilities to perform NetBIOS enumeration on the target network.

Note: Here, we will use the **Windows Server 2019** (10.10.1.19) machine to target a **Windows 11** (10.10.1.11) machine.

1. Turn on the **Windows 11** and **Windows Server 2019** virtual machines.
2. Switch to the **Windows Server 2019** virtual machine. 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. Open a **Command Prompt** window.
4. Type **nbtstat -a [IP address of the remote machine]** (in this example, the target IP address is **10.10.1.11**) and press **Enter**.

Note: In this command, **-a** displays the NetBIOS name table of a remote computer.

5. The result appears, displaying the NetBIOS name table of a remote computer (in this case, the **WINDOWS11** machine), as shown in the screenshot.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.1457]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>nbtstat -a 10.10.1.11

Ethernet:
NodeIpAddress: [10.10.1.19] Scope Id: []

      NetBIOS Remote Machine Name Table

      Name          Type        Status
      -----
WORKGROUP    <00>  GROUP     Registered
WINDOWS11   <00>  UNIQUE    Registered
WINDOW511   <20>  UNIQUE    Registered
WORKGROUP    <1E>  GROUP     Registered
WORKGROUP    <1D>  UNIQUE    Registered
00_MSBROWSE_<01> GROUP     Registered

MAC Address = 1C-89-02-1A-0B-BD

C:\Users\Administrator>
```

6. In the same **Command Prompt** window, type **nbtstat -c** and press **Enter**.

Note: In this command, **-c** lists the contents of the NetBIOS name cache of the remote computer.

7. The result appears, displaying the contents of the NetBIOS name cache, the table of NetBIOS names, and their resolved IP addresses.

Note: It is possible to extract this information without creating a **null session** (an unauthenticated session).

```
C:\Users\Administrator>nbtstat -c

Ethernet:
NodeIpAddress: [10.10.1.19] Scope Id: []

      NetBIOS Remote Cache Name Table

      Name          Type        Host Address  Life [sec]
      -----
WINDOW511   <20>  UNIQUE    10.10.1.11    301

C:\Users\Administrator>
```

- Now, type **net use** and press **Enter**. The output displays information about the target such as connection status, shared folder/drive and network information, as shown in the screenshot.

```
C:\Users\Administrator>net use
New connections will be remembered.

Status      Local      Remote          Network
-----      ----      -----          -----
OK           Z:        \\WINDOWS11\CEH-Tools    Microsoft Windows Network
The command completed successfully.

C:\Users\Administrator>
```

- Using this information, the attackers can read or write to a remote computer system, depending on the availability of shares, or even launch a DoS attack.
- This concludes the demonstration of performing NetBIOS enumeration using Windows command-line utilities such as Nbtstat and Net use.
- Close all open windows and document all the acquired information.

Task 2: Perform NetBIOS Enumeration using NetBIOS Enumerator

NetBIOS Enumerator is a tool that enables the use of remote network support and several other techniques such as SMB (Server Message Block). It is used to enumerate details such as NetBIOS names, usernames, domain names, and MAC addresses for a given range of IP addresses.

Here, we will use the NetBIOS Enumerator to perform NetBIOS enumeration on the target network.

Note: Here, we will use the **Windows 11** machine to target **Windows Server 2019** and **Windows Server 2022** machines.

- Turn on **Windows Server 2022** virtual machine.

Note: Ensure that the **Windows 11** and **Windows Server 2019** virtual machines are running.

- Switch to the **Windows 11** virtual machine. By default, **Admin** user profile is selected, type **Pa\$\$w0rd** in the **Password** field and press **Enter** to login.

Note: If **Welcome to Windows** wizard appears, click **Continue** and in **Sign in with Microsoft** wizard, click **Cancel**.

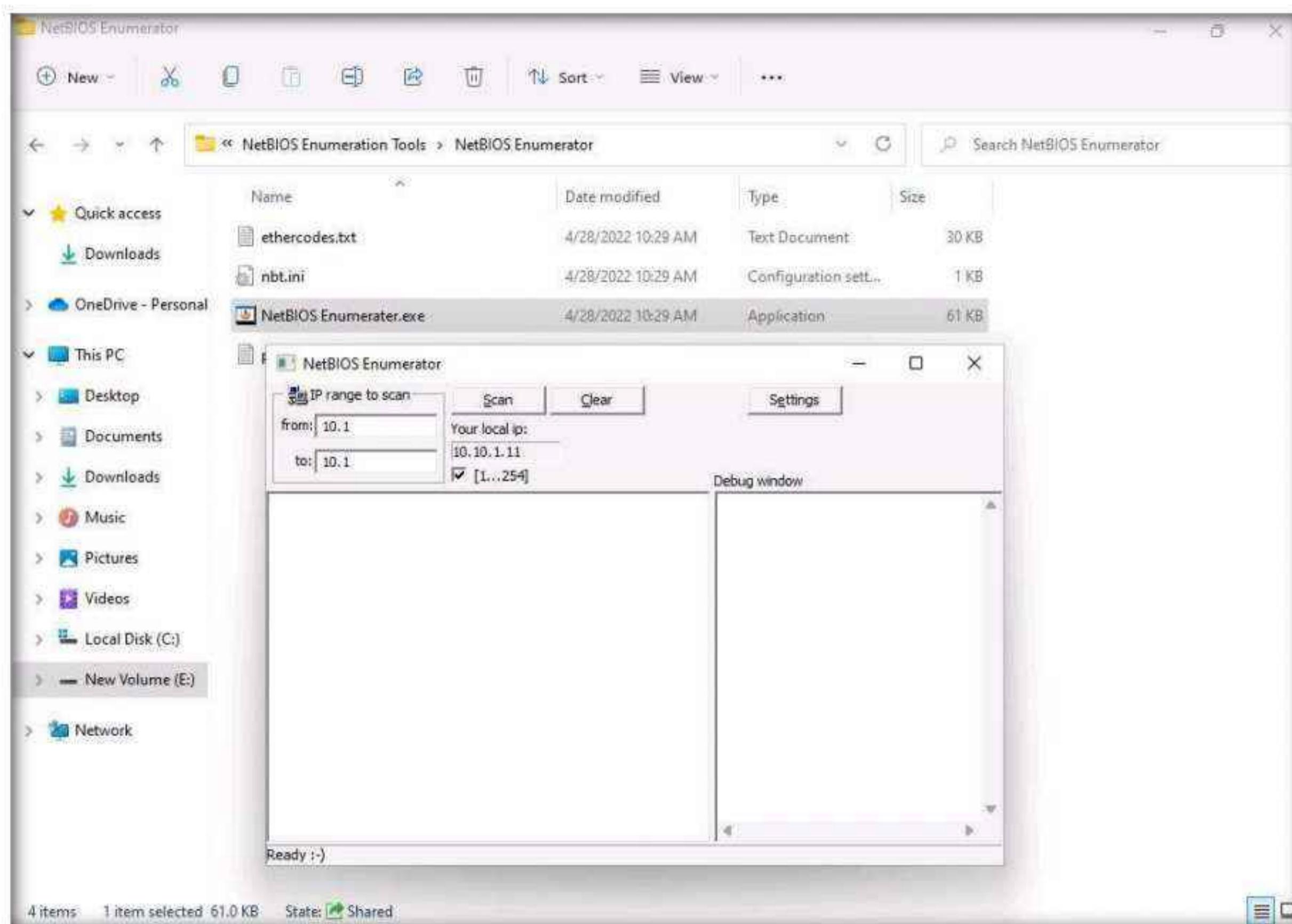
Note: Networks screen appears, click **Yes** to allow your PC to be discoverable by other PCs and devices on the network.

Module 04 – Enumeration

3. In the **Windows 11** machine, navigate to **E:\CEH-Tools\CEHv12 Module 04 Enumeration\NetBIOS Enumeration Tools\NetBIOS Enumerator** and double-click **NetBIOS Enumerator.exe**.

Note: If the **Open - File Security Warning** pop-up appears, click **Run**.

4. The **NetBIOS Enumerator** main window appears, as shown in the screenshot.



5. Under **IP range to scan**, enter an **IP range** in the **from** and **to** fields and click the **Scan** button to initiate the scan (In this example, we are targeting the IP range **10.10.1.15-10.10.1.100**).

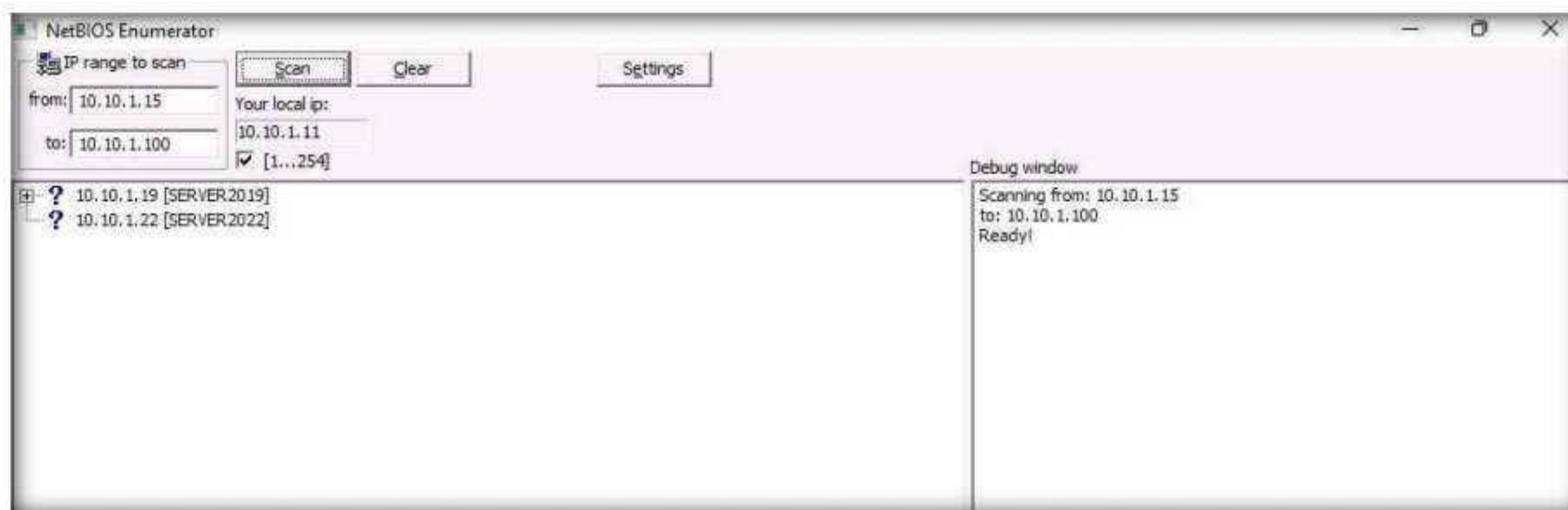
Note: Ensure that the IP address in **to** field is between 10.10.1.100 to 10.10.1.250. If the IP address is less than 10.10.1.100, the tool might crash.



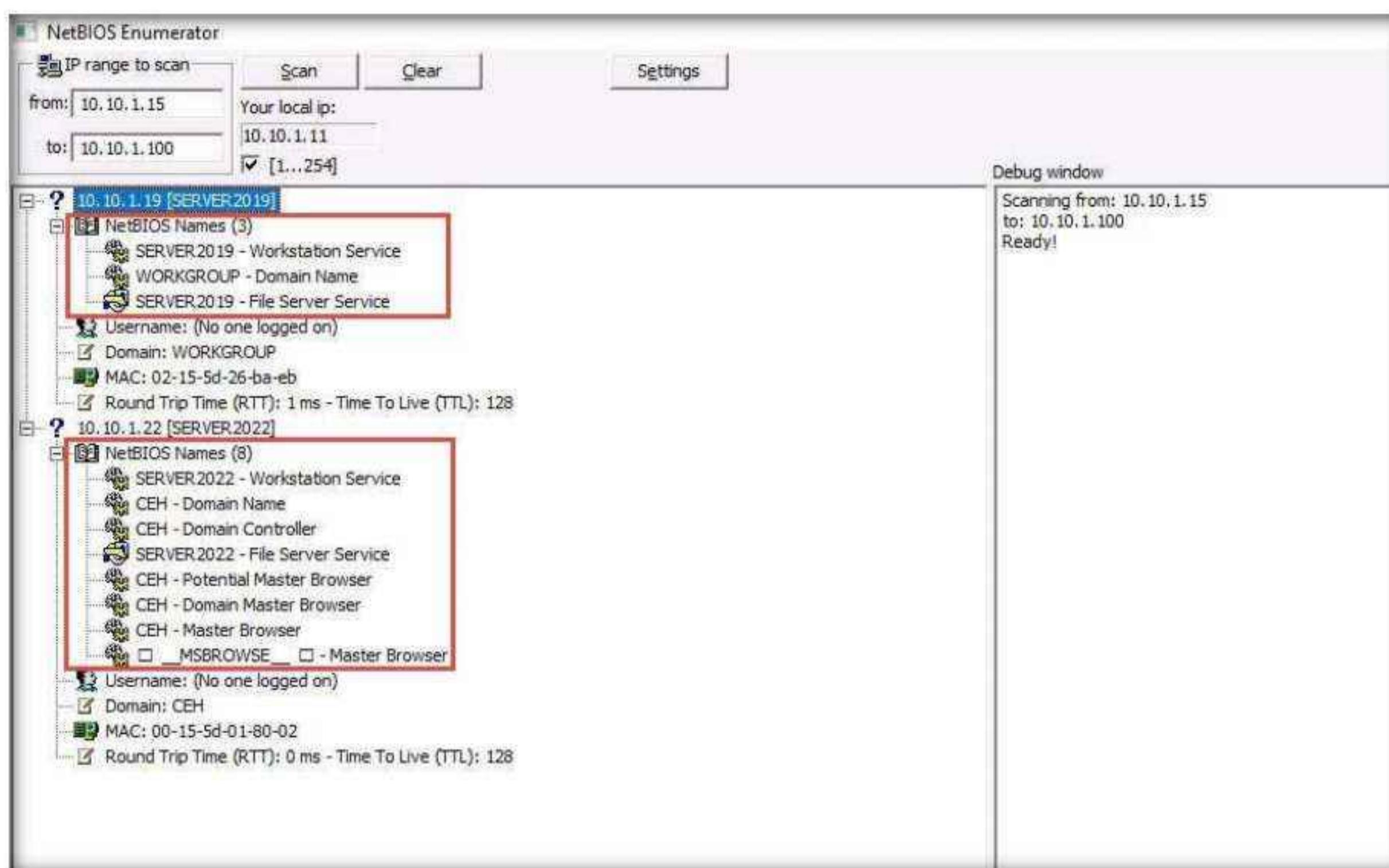
Module 04 – Enumeration

6. NetBIOS Enumerator scans for the provided IP address range. On completion, the scan results are displayed in the left pane, as shown in the screenshot.
7. The **Debug window** section in the right pane shows the scanning range of IP addresses and displays **Ready!** after the scan is finished.

Note: It takes approximately 5 minutes for the scan to finish.



8. Click on the expand icon (+) to the left of the **10.10.1.19** and **10.10.1.22** IP addresses in the left pane of the window. Then click on the expand icon to the left of **NetBIOS Names** to display NetBIOS details of the target IP address, as shown in the screenshot.



9. This concludes the demonstration of performing NetBIOS enumeration using NetBIOS Enumerator. This enumerated NetBIOS information can be used to strategize an attack on the target.
10. Close all open windows and document all the acquired information.
11. Turn off the **Windows 11** and **Windows Server 2019** virtual machines.

Task 3: Perform NetBIOS Enumeration using an NSE Script

NSE allows users to write (and share) simple scripts to automate a wide variety of networking tasks. NSE scripts can be used for discovering NetBIOS shares on the network. Using the nbstat NSE script, for example, you can retrieve the target's NetBIOS names and MAC addresses. Moreover, increasing verbosity allows you to extract all names related to the system.

Here, we will run the nbstat script to enumerate information such as the name of the computer and the logged-in user.

Note: Ensure that the **Windows Server 2022** virtual machine is running.

1. Turn on the **Parrot Security** virtual 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 **nmap -sV -v --script nbstat.nse [Target IP Address]** (in this example, the target IP address is **10.10.1.22**) and press **Enter**.

Note: **-sV** detects the service versions, **-v** enables the verbose output (that is, includes all hosts and ports in the output), and **--script nbstat.nse** performs the NetBIOS enumeration.

The screenshot shows a terminal window titled "Parrot Terminal". The terminal window has a dark background with light-colored text. At the top, there is a menu bar with options: File, Edit, View, Search, Terminal, Help. Below the menu bar, the terminal prompt is shown: "[attacker@parrot]~[~]". The user then types the command: "\$sudo su". After a password entry, the user becomes root: "[root@parrot]~[~/home/attacker]". Finally, the user runs the command: "#nmap -sV -v --script nbstat.nse 10.10.1.22". The terminal window displays the results of the nmap scan, which include information about the target host's NetBIOS shares and MAC addresses.

- The scan results appear, displaying the open ports and services, along with their versions. Displayed under the **Host script results** section are details about the target system such as the NetBIOS name, NetBIOS user, and NetBIOS MAC address, as shown in the screenshot.

```
Applications Places System nmap -sV -v --script nbstat.nse 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
MAC Address: 84:86:4C:A3:0B:66 (Unknown)
Service Info: Host: SERVER2022; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
nbstat: NetBIOS name: SERVER2022, NetBIOS user: <unknown>, NetBIOS MAC: 84:86:4c:a3:0b:66 (unknown)
Names:
SERVER2022<00>          Flags: <unique><active>
CEH<00>                  Flags: <group><active>
CEH<1c>                  Flags: <group><active>
SERVER2022<20>          Flags: <unique><active>
CEH<1e>                  Flags: <group><active>
CEH<1b>                  Flags: <unique><active>
CEH<1d>                  Flags: <unique><active>
\x01\x02 MSBROWSE \x02<01>  Flags: <group><active>
Statistics:
84 86 4c a3 0b 66 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

NSE: Script Post-scanning.
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 56.29 seconds
Raw packets sent: 1001 (44.028KB) | Rcvd: 1001 (40.096KB)
[root@parrot]# !/home/attacker]
```

8. In the terminal window, type **nmap -sU -p 137 --script nbstat.nse** [Target IP Address] (in this case, the target IP address is **10.10.1.22**) and press **Enter**.

Note: -sU performs a UDP scan, -p specifies the port to be scanned, and --script nbstat.nse performs the NetBIOS enumeration.

A screenshot of a Parrot OS terminal window titled "clear - Parrot Terminal". The window shows a command-line interface with the following text:
[root@parrot]~[/home/attacker]
#nmap -sU -p 137 --script nbstat.nse 10.10.1.22

- The scan results appear, displaying the open NetBIOS port (137) and, under the **Host script results** section, NetBIOS details such as NetBIOS name, NetBIOS user, and NetBIOS MAC of the target system, as shown in the screenshot.

```

nmap -sU -p 137 --script nbstat.nse 10.10.1.22 - Parrot Terminal
[+] root@parrot:[~/home/attacker]
# nmap -sU -p 137 --script nbstat.nse 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-10 01:15 EDT
Nmap scan report for 10.10.1.22
Host is up (0.0014s latency).

PORT      STATE SERVICE
137/udp    open  netbios-ns
MAC Address: 84:86:4C:A3:0B:66 (Unknown)

Host script results:
nbstat: NetBIOS name: SERVER2022, NetBIOS user: <unknown>, NetBIOS MAC: 84:86:4c:a3:0b:66 (unknown)
Names:
  SERVER2022<00>          Flags: <unique><active>
  CEH<00>                  Flags: <group><active>
  CEH<1c>                  Flags: <group><active>
  SERVER2022<20>          Flags: <unique><active>
  CEH<1e>                  Flags: <group><active>
  CEH<1b>                  Flags: <unique><active>
  CEH<1d>                  Flags: <unique><active>
  \x01\x02_MSBRWSE_\x02<01> Flags: <group><active>

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
[+] root@parrot:[~/home/attacker]
#

```

10. This concludes the demonstration of performing NetBIOS enumeration using an NSE script.
11. Other tools may also be used to perform NetBIOS enumeration on the target network such as **Global Network Inventory** (<http://www.magnetosoft.com>), **Advanced IP Scanner** (<https://www.advanced-ip-scanner.com>), **Hyena** (<https://www.systemtools.com>), and **Nsauditor Network Security Auditor** (<https://www.nsauditor.com>).
12. Close all open windows and document all the acquired information.
13. Turn off the **Parrot Security** and **Windows Server 2022** virtual machines.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required

Yes

No

Platform Supported

Classroom

CyberQ

Lab

2

Perform SNMP Enumeration

SNMP enumeration uses SNMP to obtain a list of user accounts and devices on a target system.

Lab Scenario

As a professional ethical hacker or penetration tester, your next step is to carry out SNMP enumeration to extract information about network resources (such as hosts, routers, devices, and shares) and network information (such as ARP tables, routing tables, device-specific information, and traffic statistics).

Using this information, you can further scan the target for underlying vulnerabilities, build a hacking strategy, and launch attacks.

Lab Objectives

- Perform SNMP enumeration using snmp-check
- Perform SNMP enumeration using SoftPerfect Network Scanner
- Perform SNMP enumeration using SnmpWalk
- Perform SNMP enumeration using Nmap

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Ubuntu virtual machine
- Android virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 20 Minutes

Overview of SNMP Enumeration

SNMP (Simple Network Management Protocol) is an application layer protocol that runs on UDP (User Datagram Protocol) and maintains and manages routers, hubs, and switches on an IP network. SNMP agents run on networking devices on Windows and UNIX networks.

SNMP enumeration uses SNMP to create a list of the user accounts and devices on a target computer. SNMP employs two types of software components for communication: the SNMP agent and SNMP management station. The SNMP agent is located on the networking device, and the SNMP management station communicates with the agent.

Lab Tasks

Task 1: Perform SNMP Enumeration using snmp-check

snmp-check is a tool that enumerates SNMP devices, displaying the output in a simple and reader-friendly format. The default community used is “public.” As an ethical hacker or penetration tester, it is imperative that you find the default community strings for the target device and patch them up.

Here, we will use the snmp-check tool to perform SNMP enumeration on the target IP address

Note: We will use the **Parrot Security** (10.10.1.13) machine to target the **Windows Server 2022** (10.10.1.22) machine.

1. Turn on the **Parrot Security** and **Windows Server 2022** virtual machines.
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.

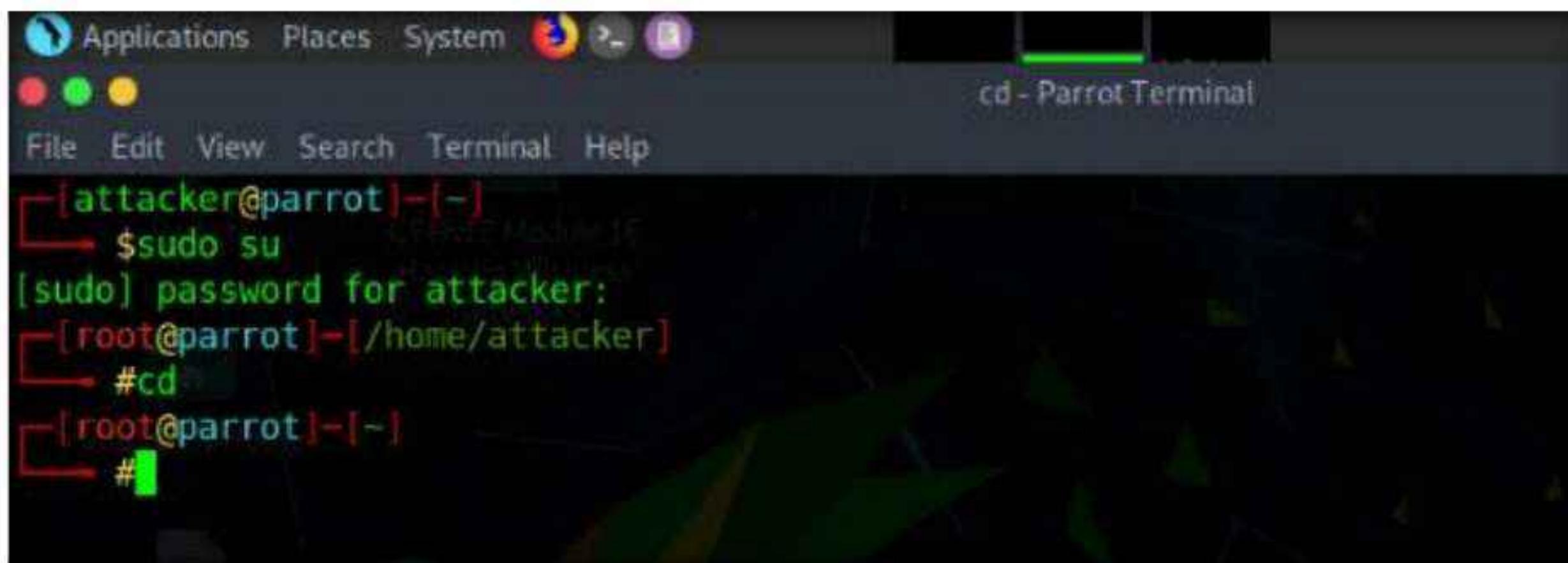
Note: Before starting SNMP enumeration, we must first discover whether the SNMP port is open. SNMP uses port 161 by default; to check whether this port is opened, we will first run Nmap port scan.

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.



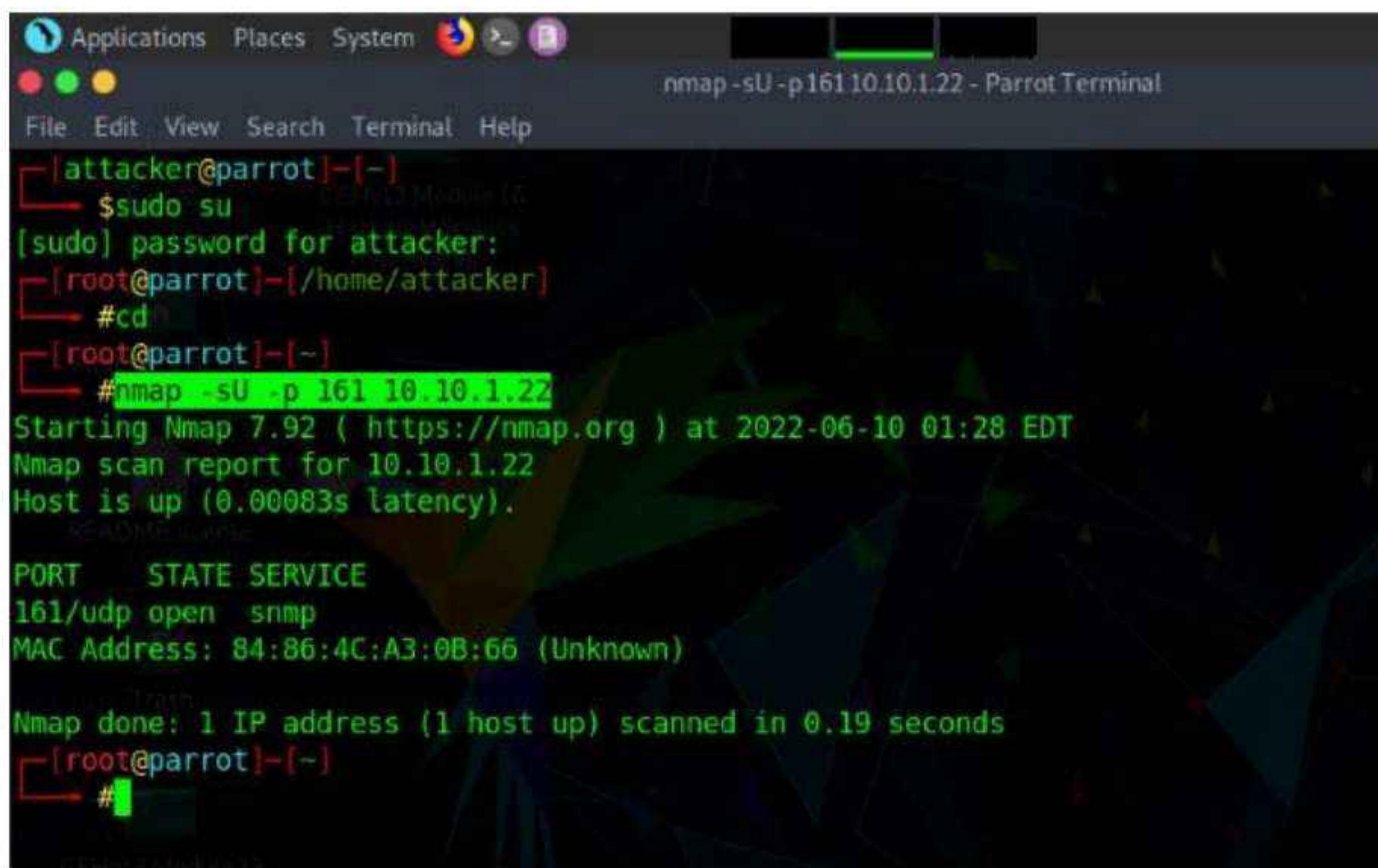
The screenshot shows a terminal window titled "cd - Parrot Terminal". The terminal session is as follows:

```
[attacker@parrot] ~
└─$ sudo su
[sudo] password for attacker:
[root@parrot] ~
└─# cd
[root@parrot] ~
└─#
```

- In the **Parrot Terminal** window, type **nmap -sU -p 161 [Target IP address]** (in this example, the target IP address is **10.10.1.22**) and press **Enter**.

Note: **-sU** performs a UDP scan and **-p** specifies the port to be scanned.

- The results appear, displaying that port 161 is **open** and being used by SNMP, as shown in the screenshot.



The screenshot shows a terminal window titled "nmap -sU -p 161 10.10.1.22 - Parrot Terminal". The terminal session is as follows:

```
[attacker@parrot] ~
└─$ sudo su
[sudo] password for attacker:
[root@parrot] ~
└─# cd
[root@parrot] ~
└─# nmap -sU -p 161 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-10 01:28 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00083s latency).

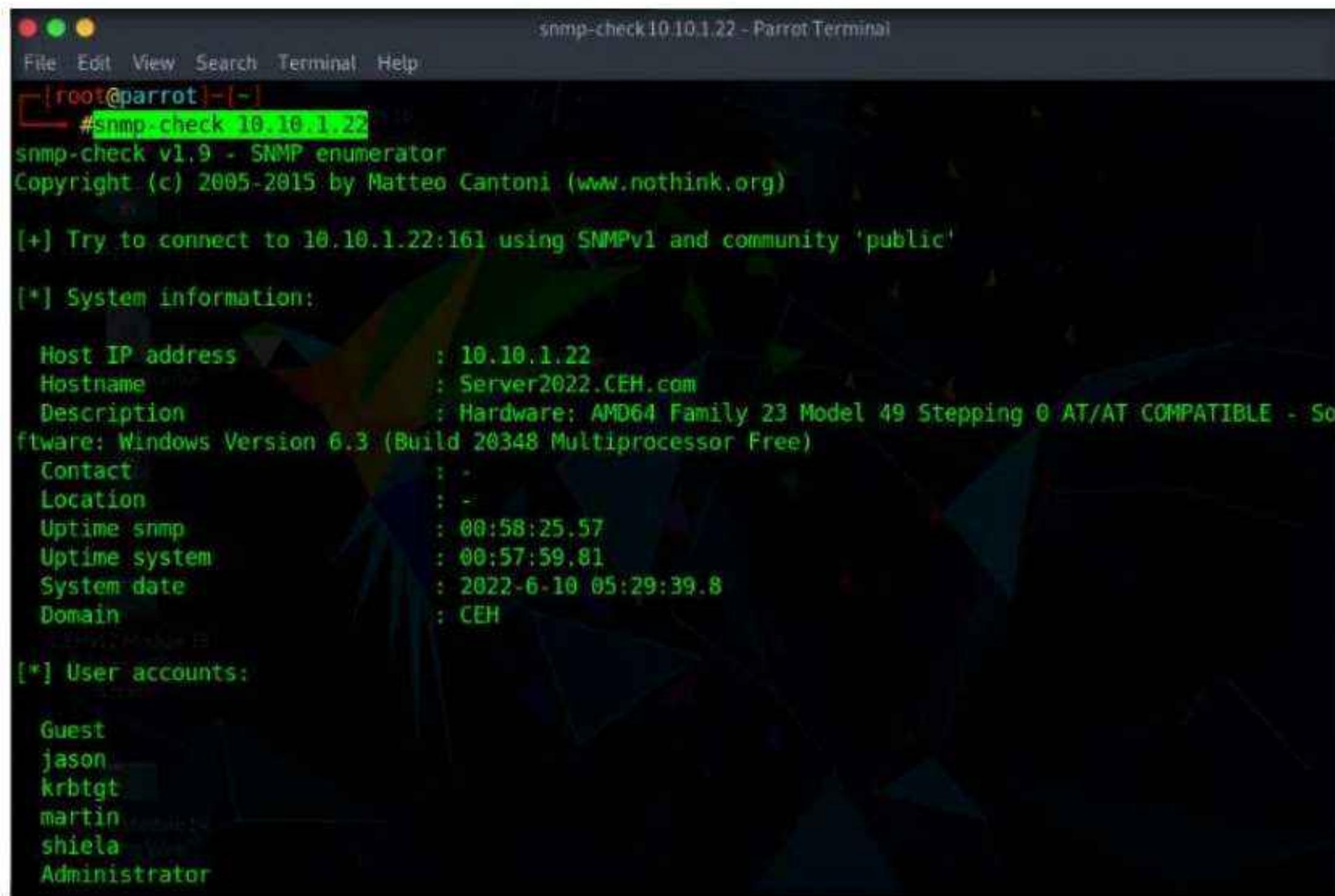
PORT      STATE SERVICE
161/udp    open  snmp
MAC Address: 84:86:4C:A3:0B:66 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
[root@parrot] ~
└─#
```

- We have established that the SNMP service is running on the target machine. Now, we shall exploit it to obtain information about the target system.
- In the **Parrot Terminal** window, type **snmp-check [Target IP Address]** (in this example, the target IP address is **10.10.1.22**) and press **Enter**.
- The result appears as shown in the screenshot. It reveals that the extracted SNMP port 161 is being used by the default “public” community string.

Note: If the target machine does not have a valid account, no output will be displayed.

12. The snmp-check command enumerates the target machine, listing sensitive information such as **System information** and **User accounts**.



```
snmp-check 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot] ~
#snmp check 10.10.1.22
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 10.10.1.22:161 using SNMPv1 and community 'public'

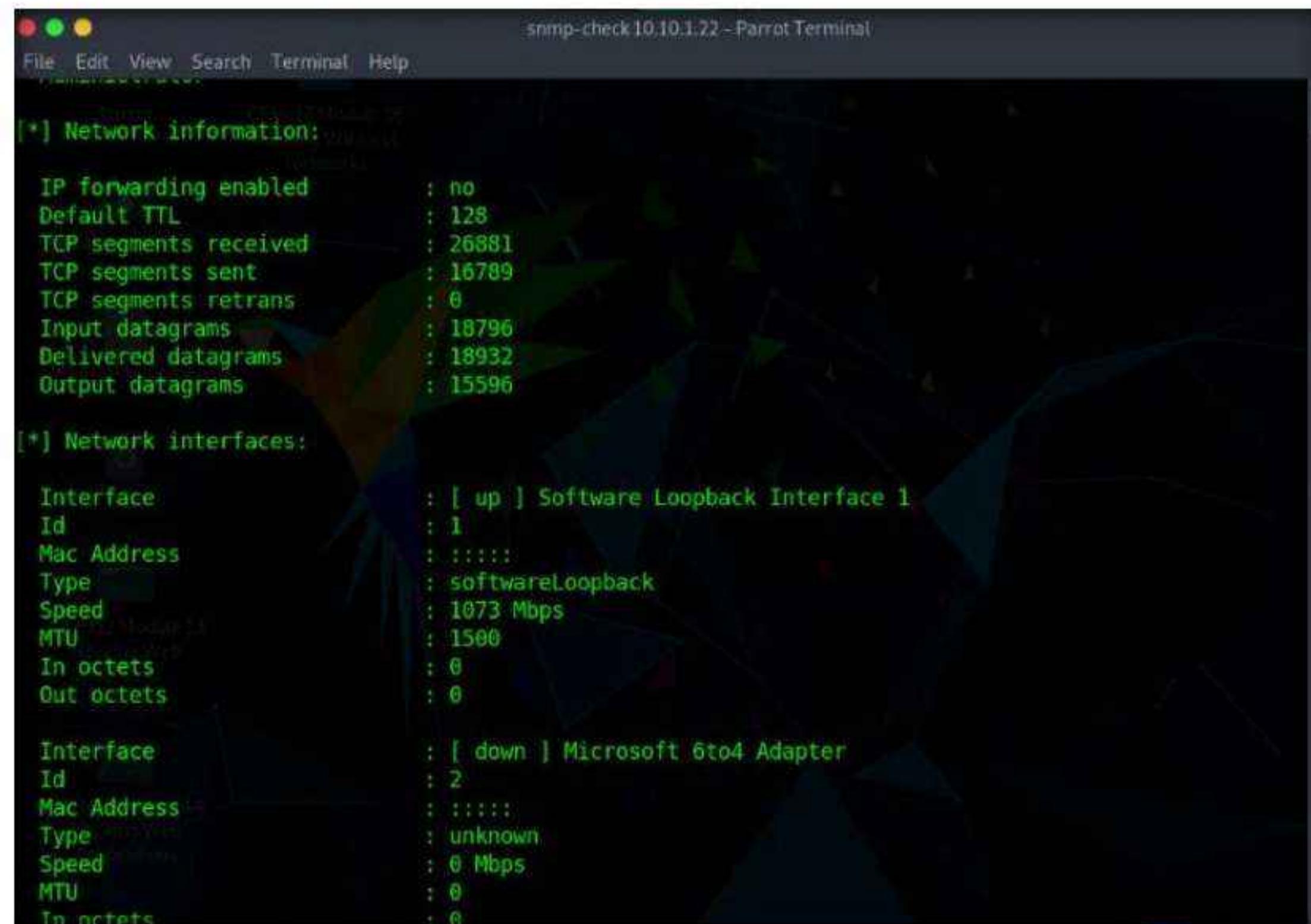
[*] System information:

Host IP address : 10.10.1.22
Hostname : Server2022.CEH.com
Description : Hardware: AMD64 Family 23 Model 49 Stepping 0 AT/AT COMPATIBLE - So
ftware: Windows Version 6.3 (Build 20348 Multiprocessor Free)
Contact :
Location :
Uptime snmp : 00:58:25.57
Uptime system : 00:57:59.81
System date : 2022-6-10 05:29:39.8
Domain : CEH

[*] User accounts:

Guest
jason
krbtgt
martin
shiela
Administrator
```

13. Scroll down to view detailed information regarding the target network under the following sections: **Network information**, **Network interfaces**, **Network IP and Routing information**, and **TCP connections and listening ports**.



```
snmp-check 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[*] Network information:

IP forwarding enabled : no
Default TTL : 128
TCP segments received : 26881
TCP segments sent : 16789
TCP segments retrans : 0
Input datagrams : 18796
Delivered datagrams : 18932
Output datagrams : 15596

[*] Network interfaces:

Interface : [ up ] Software Loopback Interface 1
Id : 1
Mac Address :
Type :
Speed :
MTU :
In octets :
Out octets :

Interface : [ down ] Microsoft 6to4 Adapter
Id : 2
Mac Address :
Type :
Speed : 0 Mbps
MTU :
In octets :
```

Module 04 – Enumeration

```
File Edit View Search Terminal Help
[*] Network IP: 10.10.1.22 - Parrot Terminal
[*] Routing information:
Destination      Next hop      Mask          Metric
0.0.0.0          10.10.1.1    0.0.0.0        271
10.10.1.0         10.10.1.22  255.255.255.0 271
10.10.1.22        10.10.1.22  255.255.255.255 271
10.10.1.255       10.10.1.22  255.255.255.255 271
127.0.0.0          127.0.0.1   255.0.0.0      331
127.0.0.1          127.0.0.1   255.255.255.255 331
127.255.255.255  127.0.0.1   255.255.255.255 331
224.0.0.0          127.0.0.1   240.0.0.0      331
255.255.255.255  127.0.0.1   255.255.255.255 331
[*] TCP connections and listening ports:
Local address     Local port    Remote address  Remote port  State
0.0.0.0            80           0.0.0.0        0           listen
0.0.0.0            88           0.0.0.0        0           listen
0.0.0.0            135          0.0.0.0        0           listen
```

14. Similarly, scrolling down reveals further sensitive information on **Processes, Storage information, File system information, Device information, Share**, etc.

```
File Edit View Search Terminal Help
[*] Processes:
Id      Status      Name          Path          Parameters
1       running     System Idle Process
4       running     System
172     running     Registry
360     running     svchost.exe   C:\Windows\system32\ -k RPCSS -p
396     running     smss.exe
464     running     svchost.exe   C:\Windows\system32\ -k DcomLaunch
512     running     csrss.exe
584     running     wininit.exe
592     running     csrss.exe
656     running     winlogon.exe
728     running     services.exe
748     running     lsass.exe   C:\Windows\system32\
772     running     dwm.exe
```

Module 04 – Enumeration

```
Applications Places System snmp-check10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
Fri Jun 10, 01:37

[*] Storage information:
Description : ["C:\\\\ Label: Serial Number 62d6615e"]
Device id   : [#<SNMP::Integer:0x00005650443275e0 @value=1>]
Filesystem type : ["unknown"]
Device unit  : [#<SNMP::Integer:0x0000565044322d60 @value=4096>]
Memory size  : 74.39 GB
Memory used   : 23.26 GB

Description : ["Virtual Memory"]
Device id   : [#<SNMP::Integer:0x0000565044317028 @value=2>]
Filesystem type : ["unknown"]
Device unit  : [#<SNMP::Integer:0x0000565044312e10 @value=65536>]
Memory size  : 9.25 GB
Memory used   : 2.02 GB

Description : ["Physical Memory"]
Device id   : [#<SNMP::Integer:0x000056504424f3c0 @value=3>]
Filesystem type : ["unknown"]
Device unit  : [#<SNMP::Integer:0x000056504430aff8 @value=65536>]
Memory size  : 8.00 GB
Memory used   : 1.86 GB

[*] File system information:
Index       : 1
Mount point  :
Remote mount point  :

[*] File system information:
Index       : 1
Mount point  :
Remote mount point  :
Access      : 1
Bootable    : 0

[*] Device information:
Id          Type        Status      Descr
1           unknown     running     Microsoft XPS Document Writer v4
2           unknown     running     Microsoft Print To PDF
3           unknown     running     Unknown Processor Type
4           unknown     running     Unknown Processor Type
5           unknown     running     Unknown Processor Type
6           unknown     running     Unknown Processor Type
7           unknown     running     Unknown Processor Type
8           unknown     running     Unknown Processor Type
9           unknown     running     Unknown Processor Type
10          unknown    unknown     Software Loopback Interface 1
11          unknown    unknown     Microsoft 6to4 Adapter
12          unknown    unknown     WAN Miniport (GRE)
13          unknown    unknown     Microsoft IP-HTTPS Platform Adapt
14          unknown    unknown     WAN Miniport (PPTP)
15          unknown    unknown     WAN Miniport (L2TP)
16          unknown    unknown     Microsoft Kernel Debug Network Ad
```

```

File Edit View Search Terminal Help
CurrentNonAnonymousUsers : 0
TotalAnonymousUsers : 0
TotalNonAnonymousUsers : 24
MaxAnonymousUsers : 0
MaxNonAnonymousUsers : 1
CurrentConnections : 0
MaxConnections : 0
ConnectionAttempts : 3
LogonAttempts : 24
Gets : 60
Posts : 0
Heads : 21
Others : 3
CGIRequests : 0
BGIRequests : 0
NotFoundErrors : 0

[*] Share:

Name : SYSVOL
Path : C:\Windows\SYSVOL\sysvol
Comment : Logon server share

Name : NETLOGON
Path : C:\Windows\SYSVOL\sysvol\CEH.com\SCRIPTS
Comment : Logon server share

[root@parrot]# 

```

15. Attackers can further use this information to discover vulnerabilities in the target machine and further exploit them to launch attacks.
16. This concludes the demonstration of performing SNMP enumeration using the `snmp-check`.
17. Close all open windows and document all the acquired information.

Task 2: Perform SNMP Enumeration using SoftPerfect Network Scanner

SoftPerfect Network Scanner can ping computers, scan ports, discover shared folders, and retrieve practically any information about network devices via WMI (Windows Management Instrumentation), SNMP, HTTP, SSH, and PowerShell.

The program also scans for remote services, registries, files, and performance counters. It can check for a user-defined port and report if one is open, and is able to resolve hostnames as well as auto-detect your local and external IP range. SoftPerfect Network Scanner offers flexible filtering and display options, and can export the NetScan results to a variety of formats, from XML to JSON. In addition, it supports remote shutdown and Wake-On-LAN.

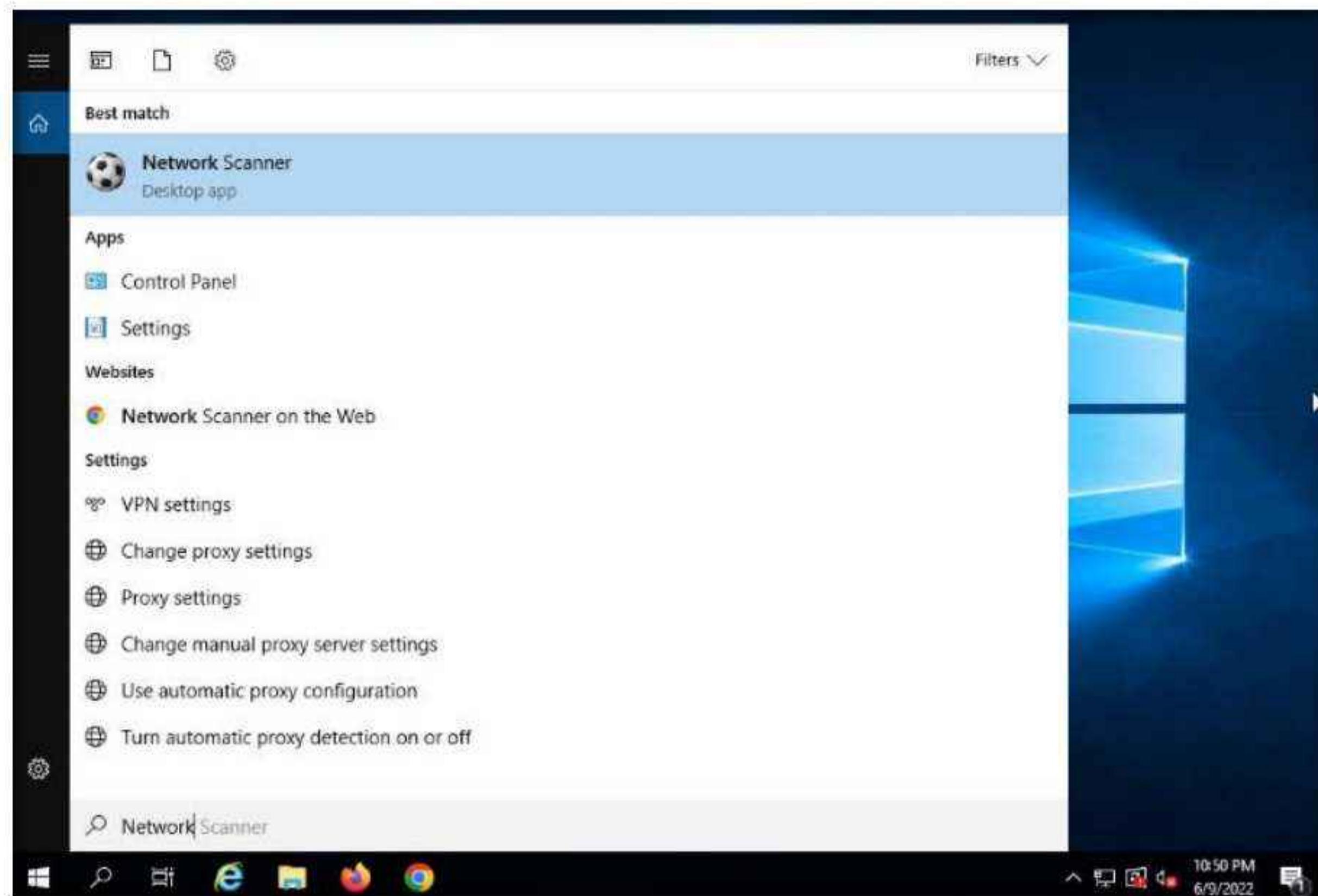
Here, we will use the SoftPerfect Network Scanner to perform SNMP enumeration on a target system.

Note: Ensure that the **Parrot Security** and **Windows Server 2022** virtual machines are running.

1. Turn on the **Windows 11**, **Windows Server 2019**, **Ubuntu** and **Android** virtual machines.
2. Switch to the **Windows Server 2019** virtual machine. Click **Ctrl+Alt+Del**, then login into **Administrator** user profile using **Pa\$\$w0rd** as password.

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 **network** in the search field, the **Network Scanner** appears in the results, select **Network Scanner** to launch it.

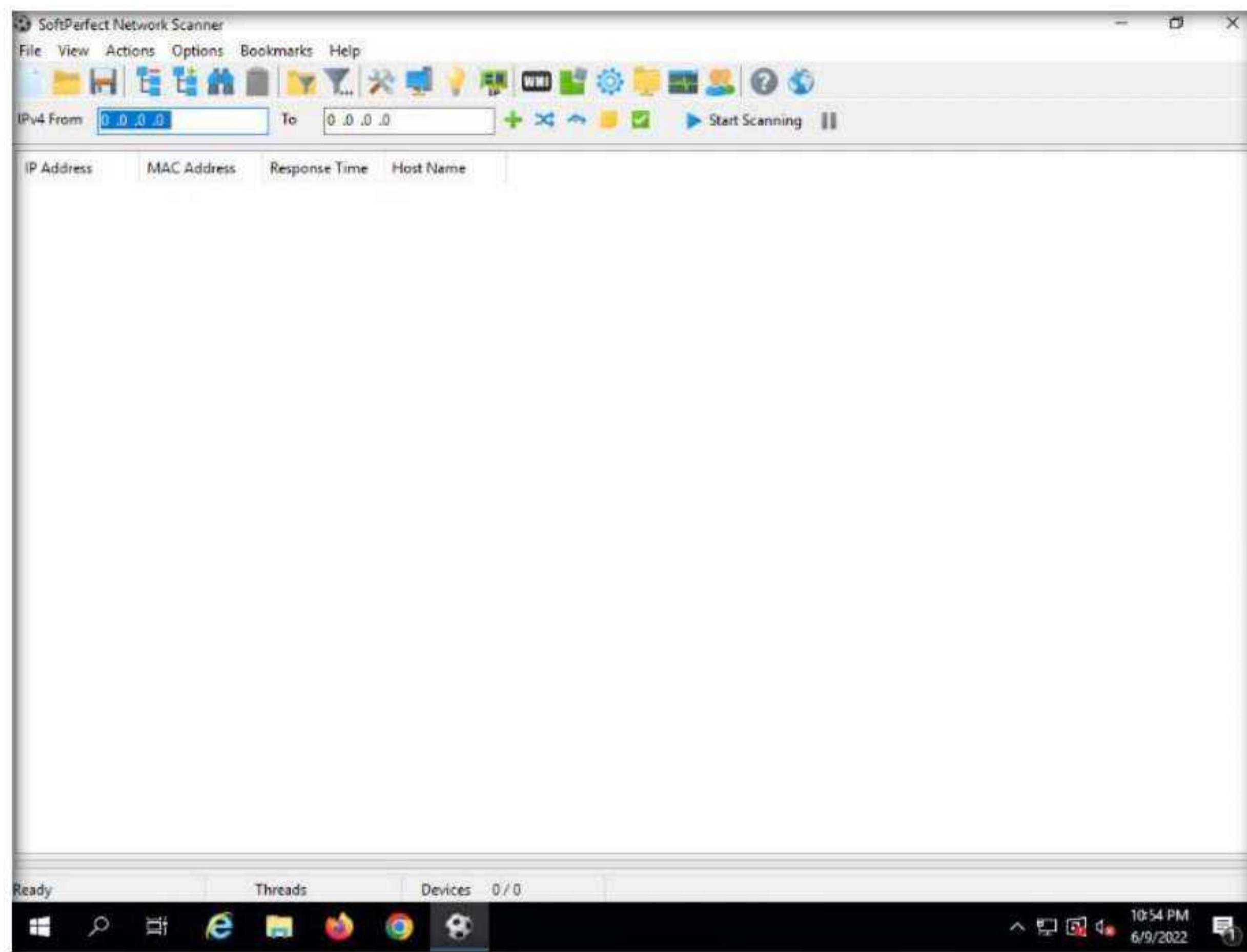


Note: If a User Account Control pop-up appears, click **Yes**.

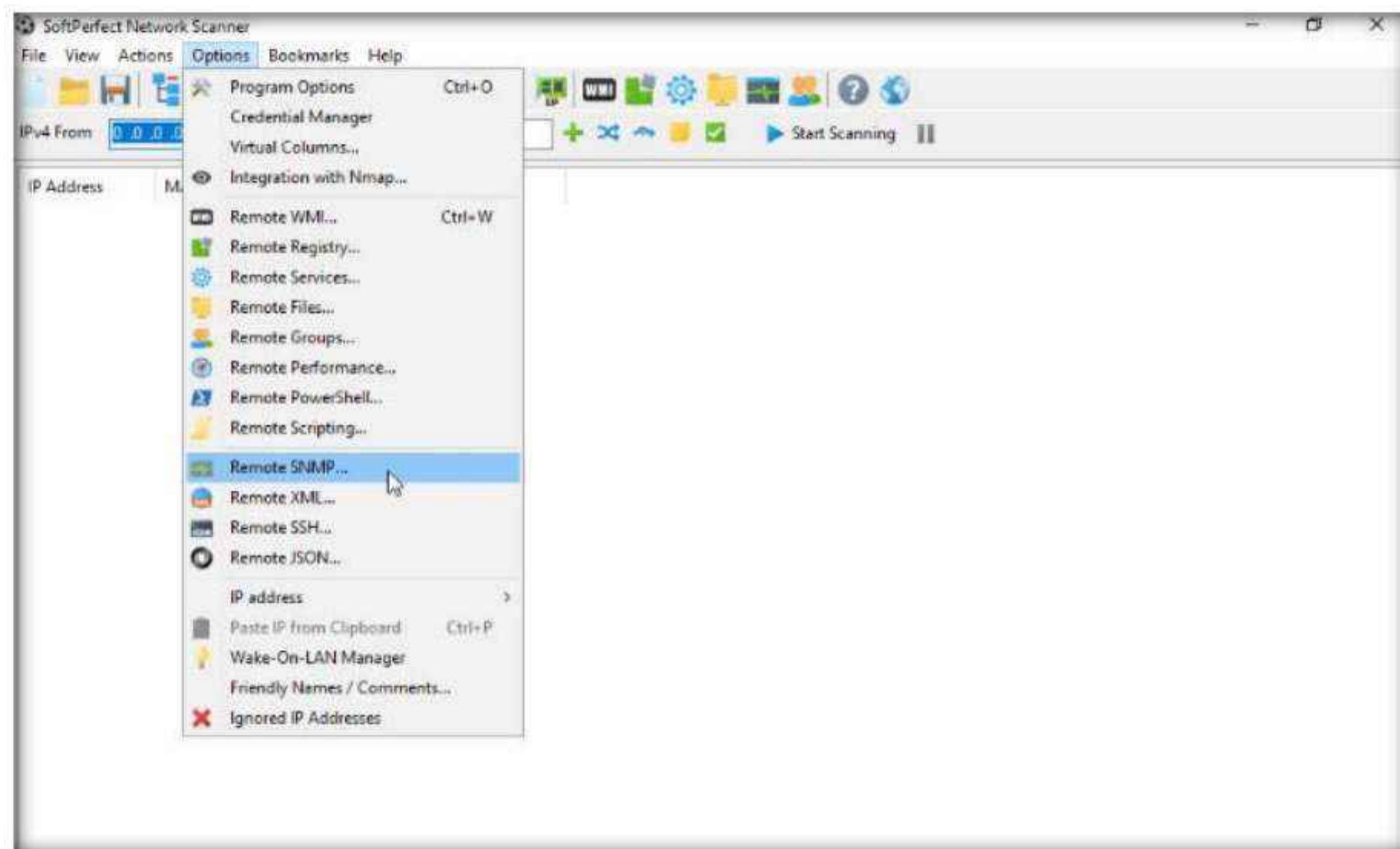
4. When the **Welcome to the Network Scanner!** wizard appears, click **Continue**.



5. The **SoftPerfect Network Scanner** GUI window will appear, as shown in the screenshot.

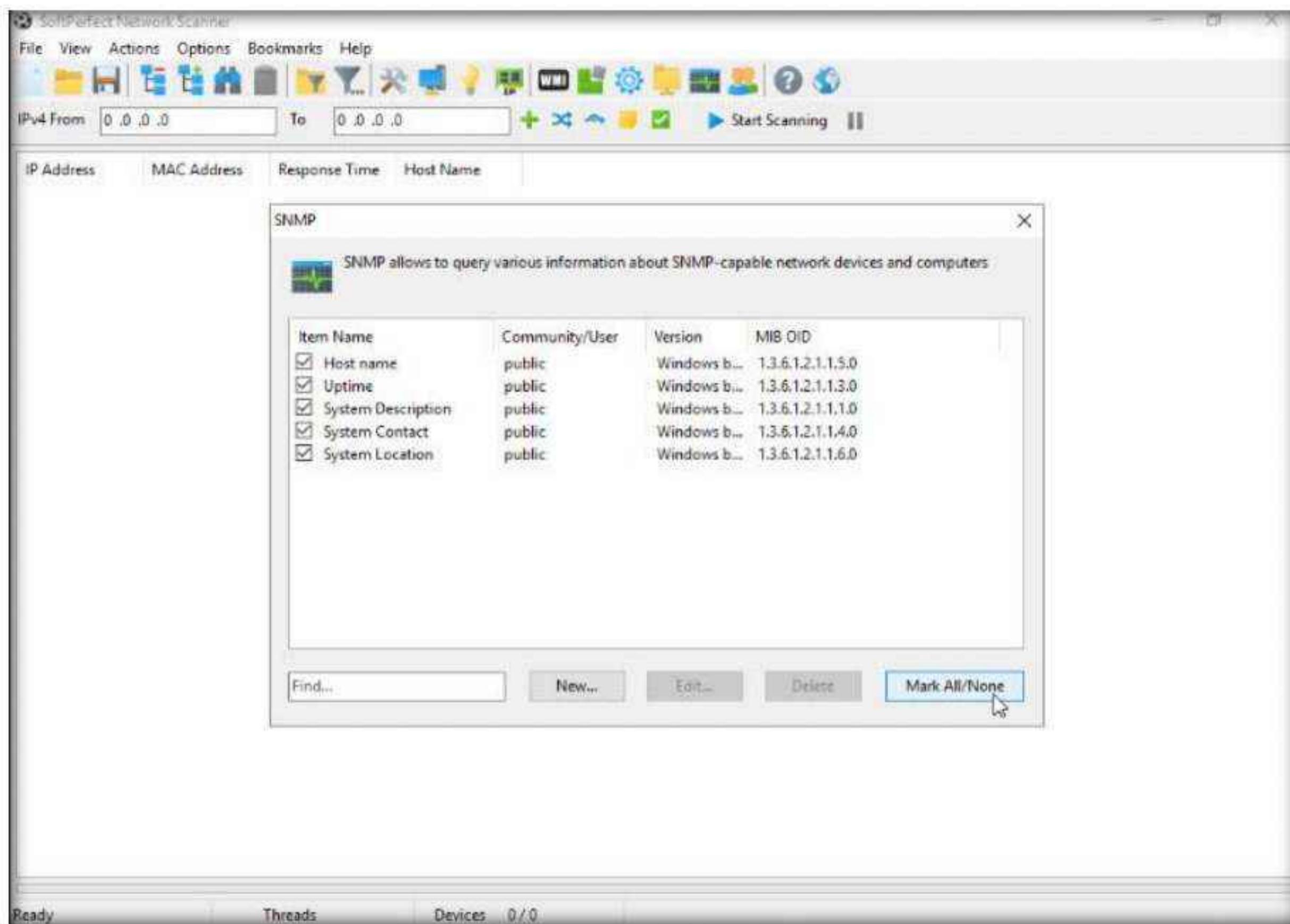


6. Click on the **Options** menu, and select **Remote SNMP...** from the drop down list. The **SNMP** pop-up window will appear.

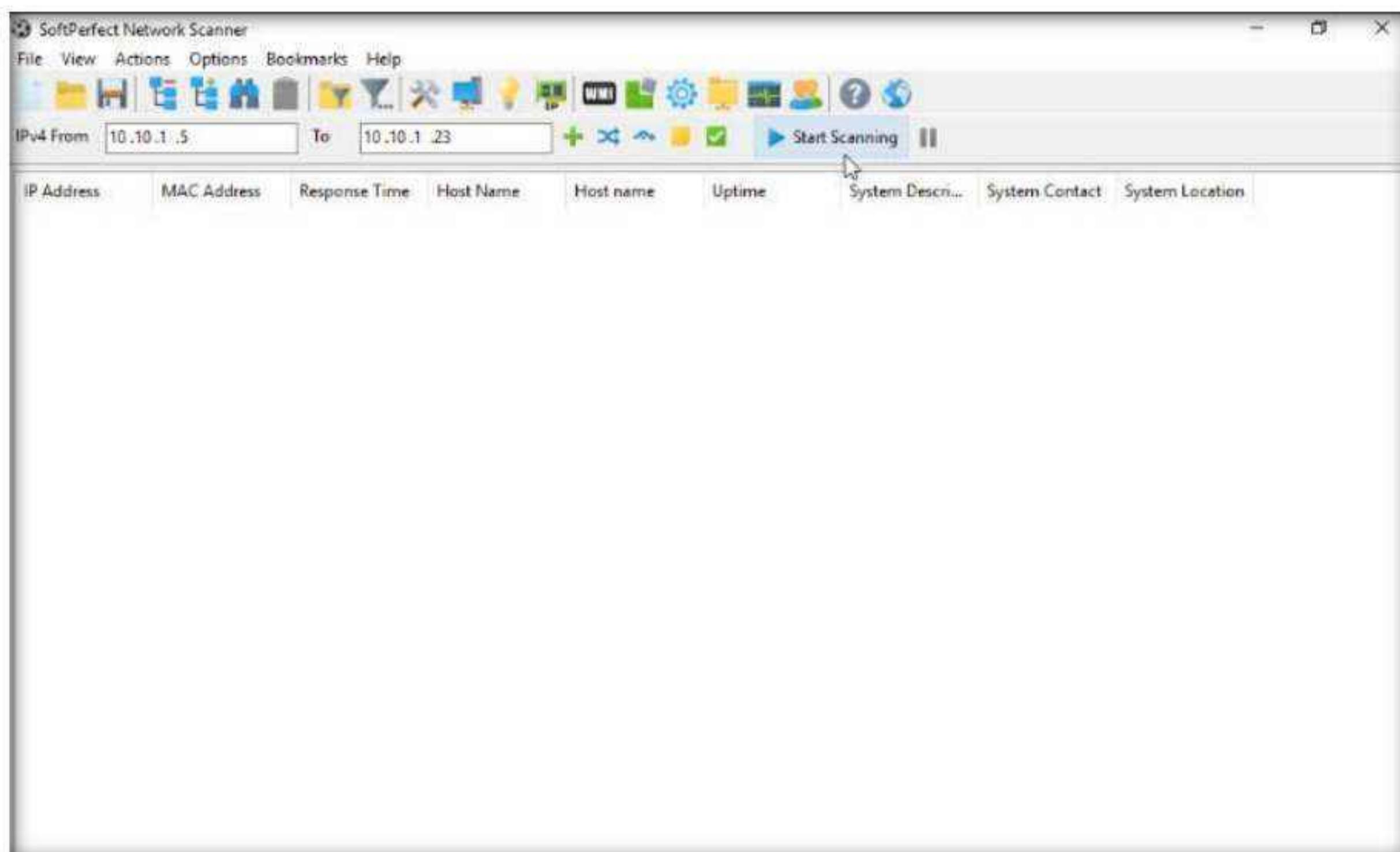


Module 04 – Enumeration

7. Click the **Mark All/None** button to select all the items available for SNMP scanning and close the window.



8. To scan your network, enter an IP range in the **IPv4 From** and **To** fields (in this example, the target IP address range is **10.10.1.5-10.10.1.23**), and click the **Start Scanning** button.

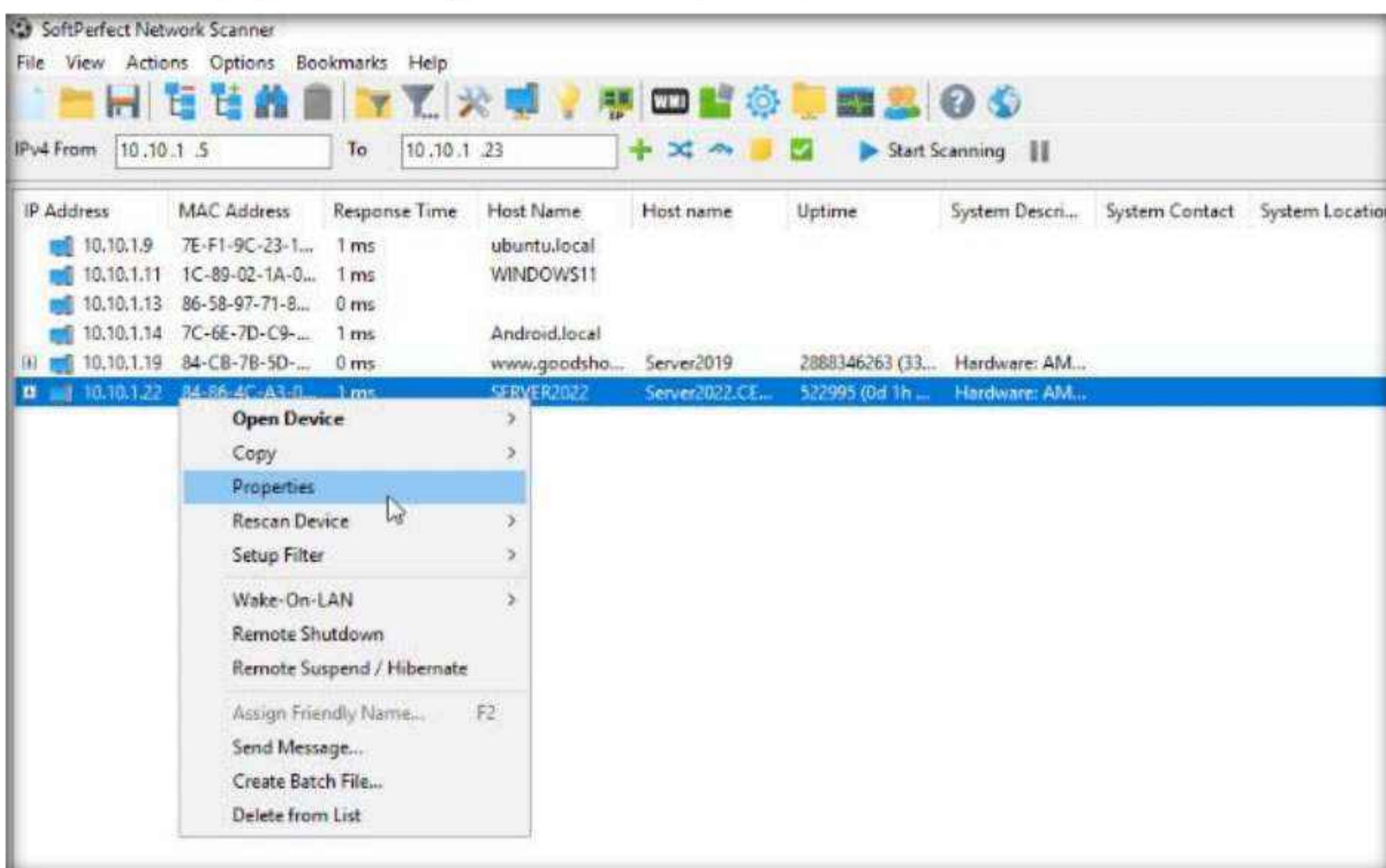


Module 04 – Enumeration

9. The **status bar** at the lower-right corner of the GUI displays the status of the scan.
10. The scan results appear, displaying the active hosts in the target IP address range, as shown in the screenshot.

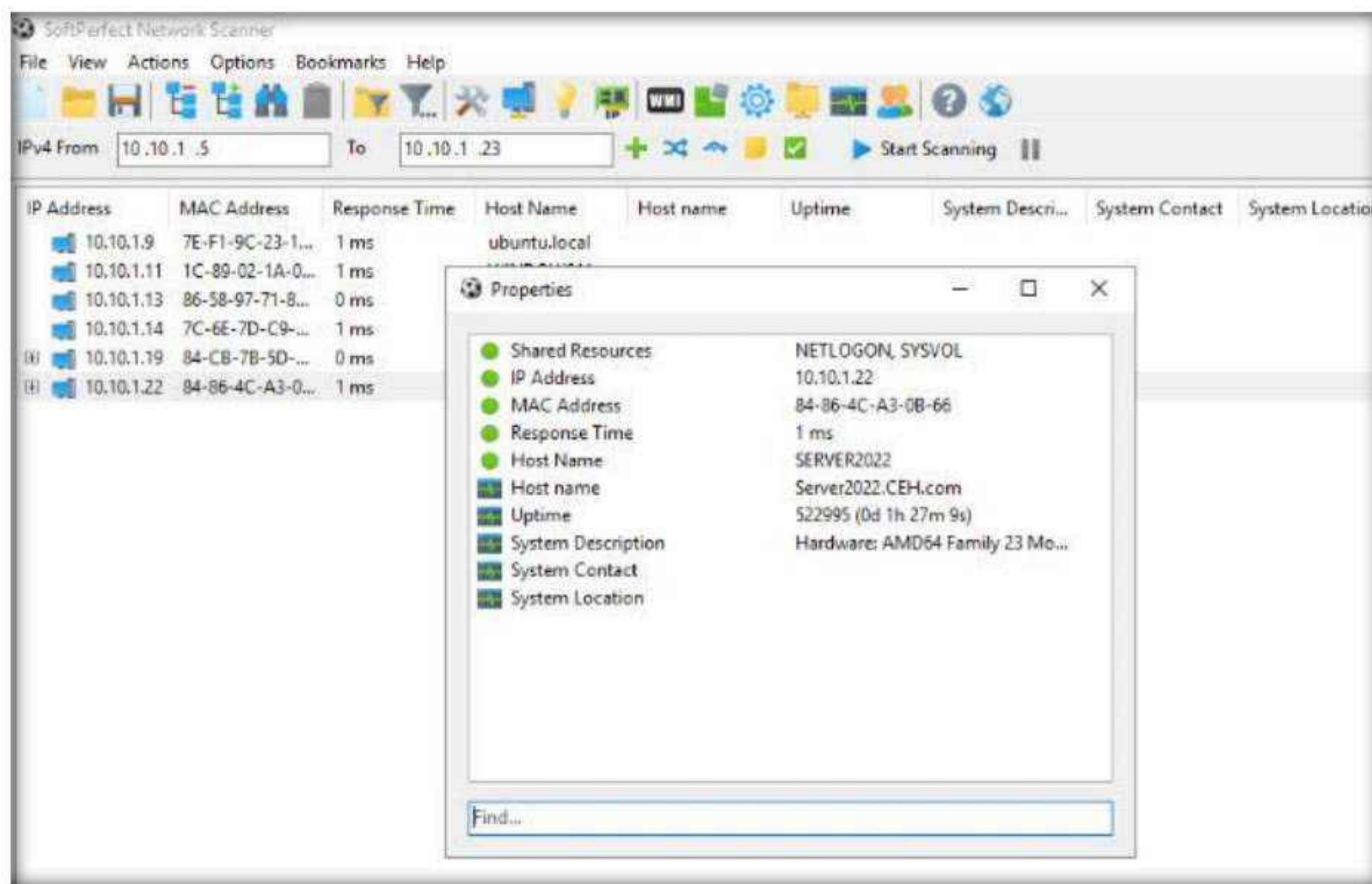
IP Address	MAC Address	Response Time	Host Name	Host name	Uptime	System Descr...	System Contact	System Location
10.10.1.9	7E-F1-9C-23-1...	1 ms	ubuntu.local					
10.10.1.11	1C-89-02-1A-0...	1 ms	WINDOWS11					
10.10.1.13	86-58-97-71-8...	0 ms						
10.10.1.14	7C-6E-7D-C9-...	1 ms	Android.local					
(I) 10.10.1.19	84-CB-7B-5D-...	0 ms	www.goodsho...	Server2019	2888346263 (33...)	Hardware: AM...		
(I) 10.10.1.22	84-86-4C-A3-0...	1 ms	SERVER2022	Server2022.CE...	522995 (0d 1h ...)	Hardware: AM...		

11. To view the properties of an individual IP address, right-click a particular IP address (in this example, **10.10.1.22**) and select **Properties**, as shown in the screenshot.



Module 04 – Enumeration

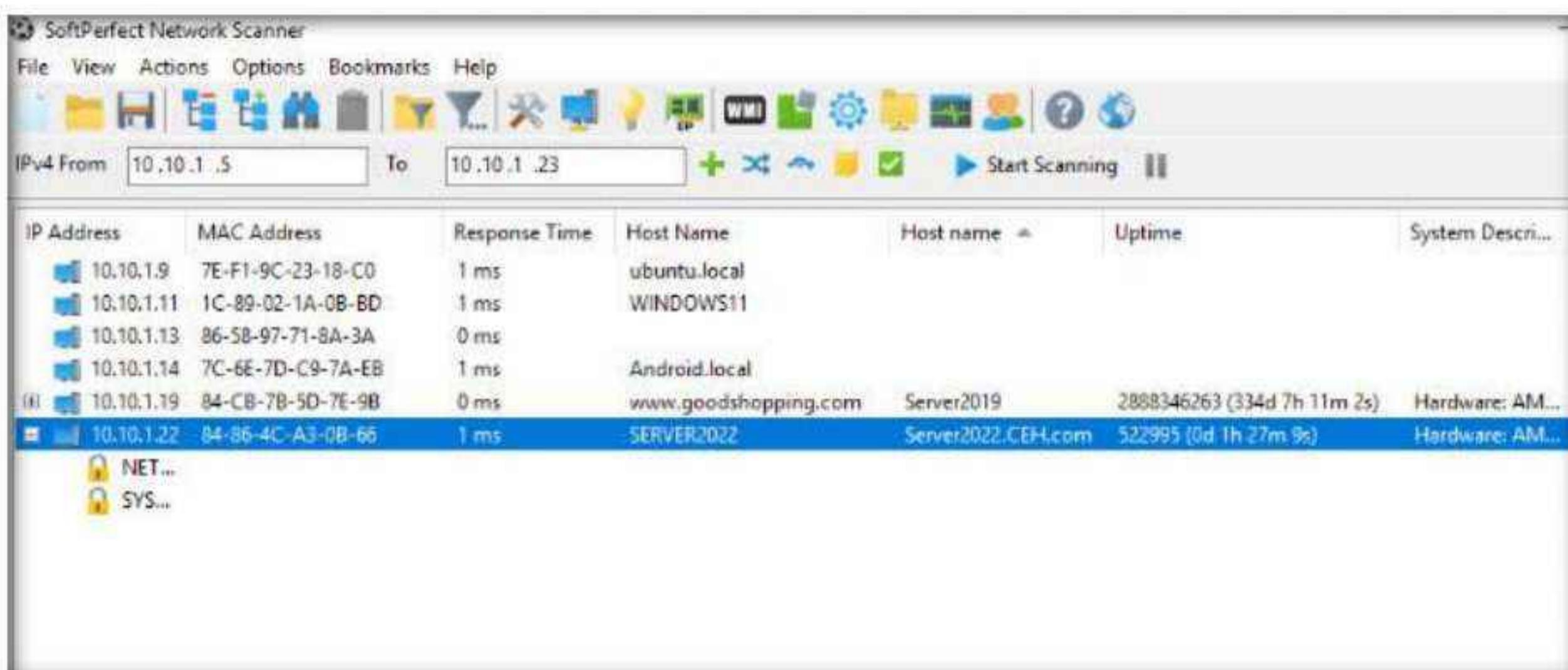
12. The **Properties** window appears, displaying the **Shared Resources, IP Address, MAC Address, Response Time, Host Name, Uptime, and System Description** of the machine corresponding to the selected IP address.



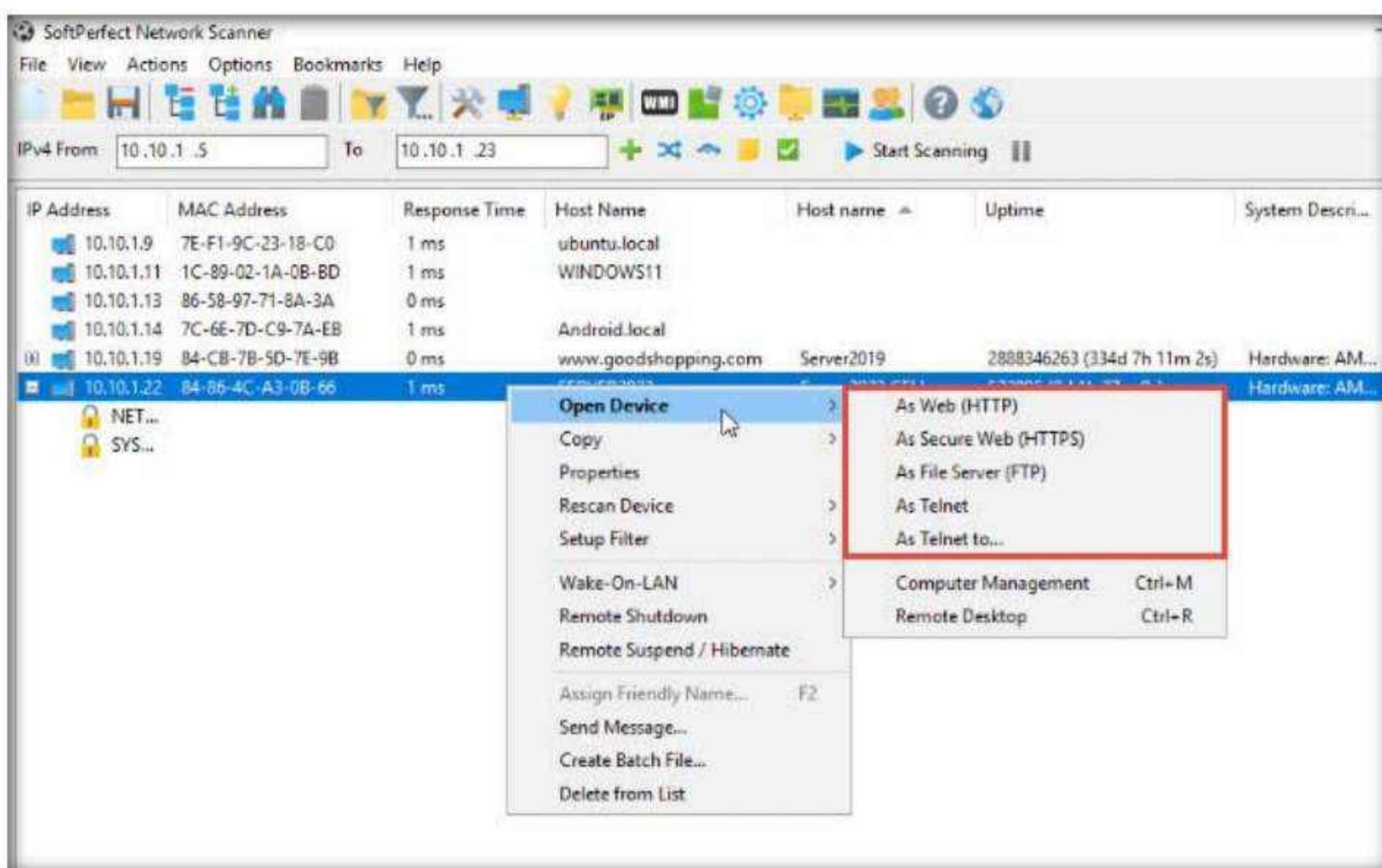
13. Close the **Properties** window.

14. To view the shared folders, note the scanned hosts that have a + node before them. Expand the node to view all the shared folders.

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



15. Right-click the selected host, and click **Open Device**. A drop-down list appears, containing options that allow you to connect to the remote machine over HTTP, HTTPS, FTP, and Telnet.



Note: If the selected host is not secure enough, you may use these options to connect to the remote machines. You may also be able to perform activities such as sending a message and shutting down a computer remotely. These features are applicable only if the selected machine has a poor security configuration.

- This concludes the demonstration of performing SNMP enumeration using the SoftPerfect Network Scanner.
- You can also use other SNMP enumeration tools such as **Network Performance Monitor** (<https://www.solarwinds.com>), **OpUtils** (<https://www.manageengine.com>), **PRTG Network Monitor** (<https://www.paessler.com>), and **Engineer's Toolset** (<https://www.solarwinds.com>) to perform SNMP enumeration on the target network.
- Close all open windows and document all the acquired information.
- Turn off the **Windows 11**, **Windows Server 2019**, **Ubuntu** and **Android** virtual machines.

Task 3: Perform SNMP Enumeration using SnmpWalk

SnmpWalk is a command-line tool that scans numerous SNMP nodes instantly and identifies a set of variables that are available for accessing the target network. It is issued to the root node so that the information from all the sub nodes such as routers and switches can be fetched.

Here, we will use SnmpWalk to perform SNMP enumeration on a target system.

Note: Ensure that the **Windows Server 2022** virtual machine is running.

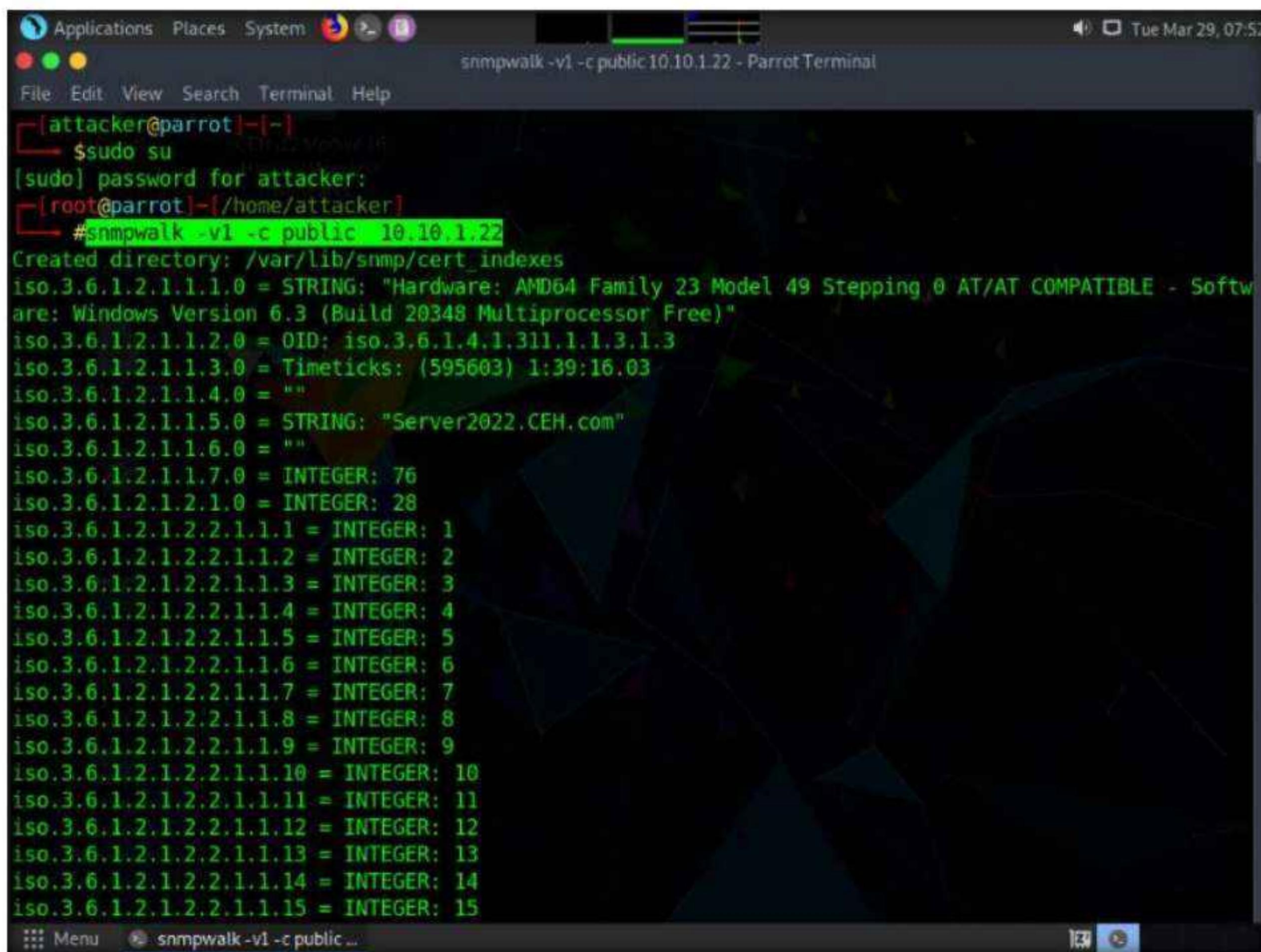
1. Switch to the **Parrot Security** virtual machine.
2. Click the **MATE Terminal** icon at the top of the **Desktop** to open a **Terminal** window.
3. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.
4. In the **[sudo] password for attacker** field, type **toor** as a password and press **Enter**.

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

5. Type **snmpwalk -v1 -c public [target IP]** and press **Enter** (here, the target IP address is **10.10.1.22**).

Note: **-v1**: specifies the SNMP version number (1 or 2c or 3) and **-c**: sets a community string.

6. The result displays all the OIDs, variables and other associated information.



The screenshot shows a terminal window titled "snmpwalk -v1 -c public 10.10.1.22 - Parrot Terminal". The terminal window has a dark background with light-colored text. It displays the command entered and the resulting output from the snmpwalk command. The output includes various SNMP variables and their values, such as hardware descriptions and system identifiers.

```
[attacker@parrot:~]$
[sudo] password for attacker:
[root@parrot:~/home/attacker]
#snmpwalk -v1 -c public 10.10.1.22
Created directory: /var/lib/snmp/cert_indexes
iso.3.6.1.2.1.1.1.0 = STRING: "Hardware: AMD64 Family 23 Model 49 Stepping 0 AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 20348 Multiprocessor Free)"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.311.1.1.3.1.3
iso.3.6.1.2.1.1.3.0 = Timeticks: (595603) 1:39:16.03
iso.3.6.1.2.1.1.4.0 =
iso.3.6.1.2.1.1.5.0 = STRING: "Server2022.CEH.com"
iso.3.6.1.2.1.1.6.0 =
iso.3.6.1.2.1.1.7.0 = INTEGER: 76
iso.3.6.1.2.1.2.1.0 = INTEGER: 28
iso.3.6.1.2.1.2.2.1.1.1 = INTEGER: 1
iso.3.6.1.2.1.2.2.1.1.2 = INTEGER: 2
iso.3.6.1.2.1.2.2.1.1.3 = INTEGER: 3
iso.3.6.1.2.1.2.2.1.1.4 = INTEGER: 4
iso.3.6.1.2.1.2.2.1.1.5 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.1.6 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.1.7 = INTEGER: 7
iso.3.6.1.2.1.2.2.1.1.8 = INTEGER: 8
iso.3.6.1.2.1.2.2.1.1.9 = INTEGER: 9
iso.3.6.1.2.1.2.2.1.1.10 = INTEGER: 10
iso.3.6.1.2.1.2.2.1.1.11 = INTEGER: 11
iso.3.6.1.2.1.2.2.1.1.12 = INTEGER: 12
iso.3.6.1.2.1.2.2.1.1.13 = INTEGER: 13
iso.3.6.1.2.1.2.2.1.1.14 = INTEGER: 14
iso.3.6.1.2.1.2.2.1.1.15 = INTEGER: 15
```

- Type **snmpwalk -v2c -c public [Target IP Address]** and press Enter to perform SNMPv2 enumeration on the target machine.

Note: **-v:** specifies the SNMP version (here, 2c is selected) and **-c:** sets a community string.

```

Applications Places System snmpwalk -v2c -c public 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[~/home/attacker]
#snmpwalk -v2c -c public 10.10.1.22
iso.3.6.1.2.1.1.1.0 = STRING: "Hardware: Intel64 Family 6 Model 85 Stepping 7 AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 20348 Multiprocessor Free)"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.311.1.1.3.1.3
iso.3.6.1.2.1.1.3.0 = Timeticks: (2890168050) 334 days, 12:14:40.50
iso.3.6.1.2.1.1.4.0 =
iso.3.6.1.2.1.1.5.0 = STRING: "Server2022.CEH.com"
iso.3.6.1.2.1.1.6.0 =
iso.3.6.1.2.1.1.7.0 = INTEGER: 76
iso.3.6.1.2.1.2.1.0 = INTEGER: 24
iso.3.6.1.2.1.2.2.1.1.1 = INTEGER: 1
iso.3.6.1.2.1.2.2.1.1.2 = INTEGER: 2
iso.3.6.1.2.1.2.2.1.1.3 = INTEGER: 3
iso.3.6.1.2.1.2.2.1.1.4 = INTEGER: 4
iso.3.6.1.2.1.2.2.1.1.5 = INTEGER: 5
iso.3.6.1.2.1.2.2.1.1.6 = INTEGER: 6
iso.3.6.1.2.1.2.2.1.1.7 = INTEGER: 7
iso.3.6.1.2.1.2.2.1.1.8 = INTEGER: 8
iso.3.6.1.2.1.2.2.1.1.9 = INTEGER: 9
iso.3.6.1.2.1.2.2.1.1.10 = INTEGER: 10
iso.3.6.1.2.1.2.2.1.1.11 = INTEGER: 11
iso.3.6.1.2.1.2.2.1.1.12 = INTEGER: 12
iso.3.6.1.2.1.2.2.1.1.13 = INTEGER: 13
iso.3.6.1.2.1.2.2.1.1.14 = INTEGER: 14
iso.3.6.1.2.1.2.2.1.1.15 = INTEGER: 15
iso.3.6.1.2.1.2.2.1.1.16 = INTEGER: 16
iso.3.6.1.2.1.2.2.1.1.17 = INTEGER: 17
iso.3.6.1.2.1.2.2.1.1.18 = INTEGER: 18
iso.3.6.1.2.1.2.2.1.1.19 = INTEGER: 19

```

- The result displays data transmitted from the SNMP agent to the SNMP server, including information on server, user credentials, and other parameters.
- This concludes the demonstration of performing SNMP enumeration using the SnmpWalk.
- Close all open windows and document all the acquired information.

Task 4: Perform SNMP Enumeration using Nmap

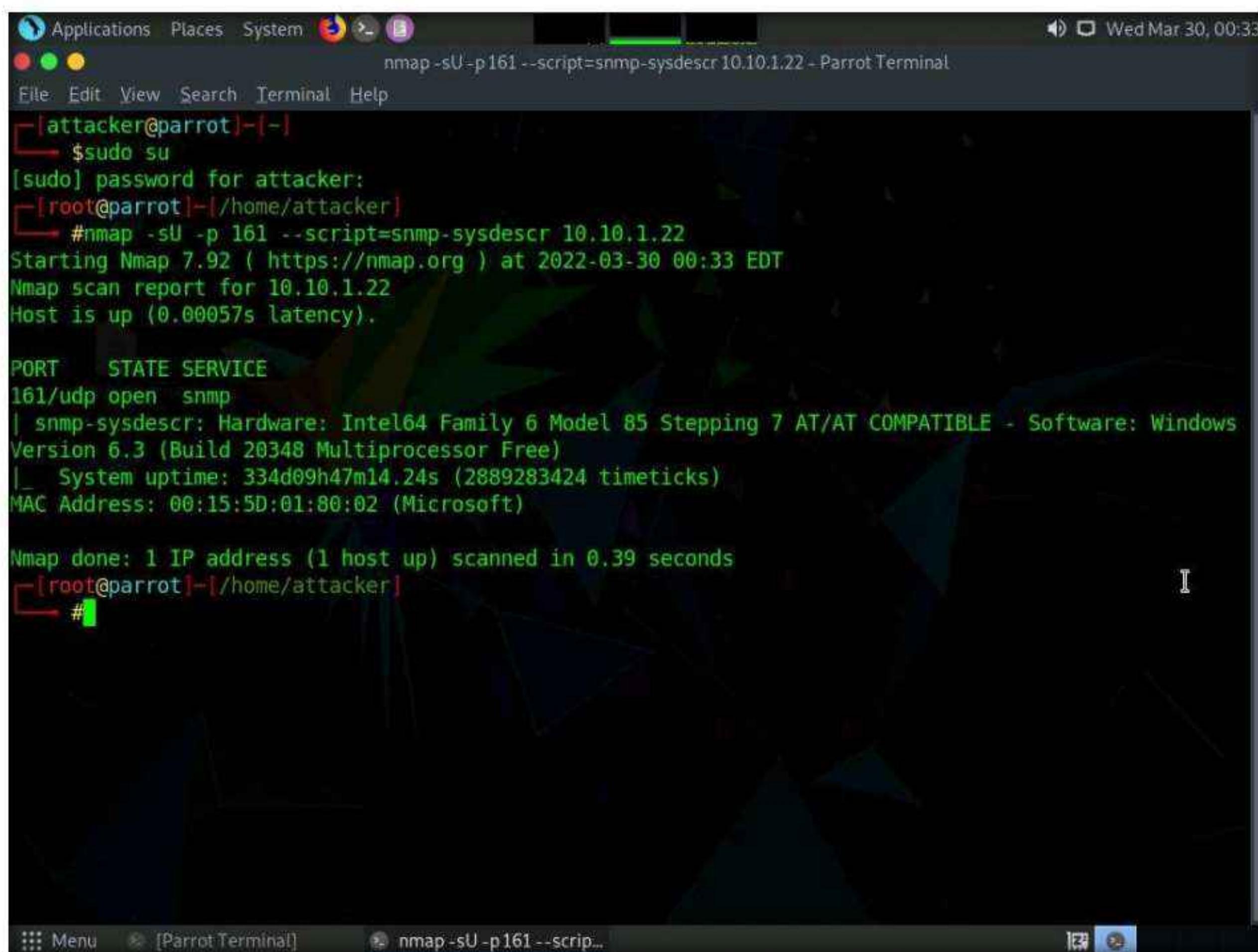
The Nmap snmp script is used against an SNMP remote server to retrieve information related to the hosted SNMP services.

Here, we will use various Nmap scripts to perform SNMP enumeration on the target system.

Note: Here, we will perform SNMP enumeration on a target machine **Windows Server 2022** (10.10.1.22).

1. In the **Parrot Security** machine, click the **MATE Terminal** icon at the top-left corner of **Desktop** to launch a **Terminal** window.
2. 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. In the terminal, type **nmap -sU -p 161 --script=snmp-sysdescr [target IP Address]** and press **Enter** (here, the target IP address is **10.10.1.22**).
Note: **-sU:** specifies a UDP scan, **-p:** specifies the port to be scanned, and **--script:** is an argument used to execute a given script (here, **snmp-sysdescr**).
5. The result appears displaying information regarding SNMP server type and operating system details, as shown in the screenshot below.

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



The screenshot shows a terminal window titled "nmap -sU -p 161 --script=snmp-sysdescr 10.10.1.22 - Parrot Terminal". The terminal content is as follows:

```
[attacker@parrot:~]$ sudo su
[sudo] password for attacker:
[root@parrot:~]# nmap -sU -p 161 --script=snmp-sysdescr 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-30 00:33 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00057s latency).

PORT      STATE SERVICE
161/udp    open  snmp
| snmp-sysdescr: Hardware: Intel64 Family 6 Model 85 Stepping 7 AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 20348 Multiprocessor Free)
|_ System uptime: 334d09h47m14.24s (2889283424 timeticks)
MAC Address: 00:15:5D:01:80:02 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds
[root@parrot:~]#
```

6. Type **nmap -sU -p 161 --script=snmp-processes [target IP Address]** and press **Enter** (here, the target IP address is **10.10.1.22**).

Note: **-sU:** specifies UDP scan, **-p:** specifies the port to be scanned, and **--script:** is an argument used to execute a given script (here, **snmp-processes**).

7. The result appears displaying a list of all the running SNMP processes along with the associated ports on the target machine (here, **Windows Server 2022**), as shown in the screenshot below.

The screenshot shows a terminal window titled "nmap -sU -p 161 --script=snmp-processes 10.10.1.22 - Parrot Terminal". The command entered was "# nmap -sU -p 161 --script=snmp-processes 10.10.1.22". The output shows the host is up with 0.00069s latency. It lists various running processes with their names, paths, and parameters. The processes include System Idle Process, System, Registry, smss.exe, svchost.exe (with path C:\Windows\system32\ and parameters -k DcomLaunch -p -s LSM), svchost.exe (with path C:\Windows\system32\ and parameters -k LocalService -s W32Time), csrss.exe, and another svchost.exe (with path C:\Windows\System32\ and parameters -k LocalService -s W32Time).

```
nmap -sU -p 161 --script=snmp-processes 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[~/home/attacker]
# nmap -sU -p 161 --script=snmp-processes 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-30 00:36 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00069s latency).

PORT      STATE SERVICE
161/udp    open  snmp
snmp-processes:
 1:
    Name: System Idle Process
 4:
    Name: System
 100:
    Name: Registry
 380:
    Name: smss.exe
 460:
    Name: svchost.exe
    Path: C:\Windows\system32\
    Params: -k DcomLaunch -p -s LSM
 500:
    Name: svchost.exe
    Path: C:\Windows\system32\
    Params: -k LocalService -s W32Time
 508:
    Name: csrss.exe
 596:
    Name: svchost.exe
    Path: C:\Windows\System32\
```

- Type **nmap -sU -p 161 --script=snmp-win32-software [target IP Address]** and press **Enter** (here, the target IP address is **10.10.1.22**).

Note: **-sU**: specifies UDP scan, **-p**: specifies the port to be scanned, and **--script**: argument used to execute a given script (here, the script is **snmp-win32-software**).

- The result appears displaying a list of all the applications running on the target machine (here, **Windows Server 2022**), as shown in the screenshot.

```

nmap -sU -p 161 --script=snmp-win32-software 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[/home/attacker]
#nmap -sU -p 161 --script=snmp-win32-software 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-30 00:38 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00058s latency).

PORT      STATE SERVICE
161/udp    open  snmp
| snmp-win32-software:
|   Adobe Acrobat DC (64-bit); 2022-02-01T04:01:22
|   Google Chrome; 2022-02-01T04:01:24
|   Java 8 Update 321 (64-bit); 2022-02-03T04:36:12
|   Java Auto Updater; 2022-02-03T04:36:36
|   Microsoft Edge; 2022-02-06T22:25:50
|   Microsoft Edge Update; 2022-02-01T04:01:24
|   Microsoft Visual C++ 2008 Redistributable - x64 9.0.30729.17; 2022-02-02T01:21:42
|   Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.17; 2022-02-02T01:21:56
|   Microsoft Visual C++ 2010 x64 Redistributable - 10.0.40219; 2022-02-02T01:22:14
|   Microsoft Visual C++ 2010 x86 Redistributable - 10.0.40219; 2022-02-02T01:22:24
|   Microsoft Visual C++ 2012 Redistributable (x64) - 11.0.61030; 2022-02-02T01:22:50
|   Microsoft Visual C++ 2012 Redistributable (x86) - 11.0.61030; 2022-02-02T01:23:00
|   Microsoft Visual C++ 2012 x64 Additional Runtime - 11.0.61030; 2022-02-02T01:22:50
|   Microsoft Visual C++ 2012 x64 Minimum Runtime - 11.0.61030; 2022-02-02T01:22:50
|   Microsoft Visual C++ 2012 x86 Additional Runtime - 11.0.61030; 2022-02-02T01:23:00
|   Microsoft Visual C++ 2012 x86 Minimum Runtime - 11.0.61030; 2022-02-02T01:23:00
|   Microsoft Visual C++ 2013 Redistributable (x64) - 12.0.30501; 2022-02-02T01:23:08
|   Microsoft Visual C++ 2013 Redistributable (x86) - 12.0.30501; 2022-02-02T01:23:16
|   Microsoft Visual C++ 2013 x64 Additional Runtime - 12.0.21005; 2022-02-02T01:23:08
|   Microsoft Visual C++ 2013 x64 Minimum Runtime - 12.0.21005; 2022-02-02T01:23:08
|   Microsoft Visual C++ 2013 x86 Additional Runtime - 12.0.21005; 2022-02-02T01:23:16

```

- Type **nmap -sU -p 161 --script=snmp-interfaces [target IP Address]** and press **Enter** (here the target IP address is **10.10.1.22**).

Note: **-sU** specifies a UDP scan, **-p** specifies the port to be scanned, and **--script** is an argument allows us to run a given script (here, **snmp-interfaces**).

- The result appears displaying information about the Operating system, network interfaces, and applications that are installed on the target machine (here, **Windows Server 2022**), as shown in the screenshot below.

Note: The list of interfaces might differ when you perform the task.

```
nmap -sU -p 161 --script=snmp-interfaces 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~[~/home/attacker]
# nmap -sU -p 161 --script=snmp-interfaces 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-30 00:43 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00050s latency).

PORT      STATE SERVICE
161/udp    open  snmp
| snmp-interfaces:
|   Software Loopback Interface 1\x00
|     IP address: 127.0.0.1 Netmask: 255.0.0.0
|     Type: softwareLoopback Speed: 1 Gbps
|     Status: up
|     Traffic stats: 0.00 Kb sent, 0.00 Kb received
|   Microsoft 6to4 Adapter\x00
|     Type: tunnel Speed: 0 Kbps
|     Traffic stats: 0.00 Kb sent, 0.00 Kb received
|   WAN Miniport (IKEv2)\x00
|     Type: tunnel Speed: 0 Kbps
|     Status: down
|     Traffic stats: 0.00 Kb sent, 0.00 Kb received
|   WAN Miniport (PPTP)\x00
|     Type: tunnel Speed: 0 Kbps
|     Status: down
|     Traffic stats: 0.00 Kb sent, 0.00 Kb received
|   Microsoft IP-HTTPS Platform Adapter\x00
|     Type: tunnel Speed: 0 Kbps
|     Traffic stats: 0.00 Kb sent, 0.00 Kb received
|   WAN Miniport (Network Monitor)\x00
|     Type: ethernetCsmacd Speed: 0 Kbps
```

12. This concludes the demonstration of performing SNMP enumeration using Nmap.
13. Close all open windows and document all the acquired information.
14. Turn off the **Parrot Security** and **Windows Server 2022** virtual machines.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required

Yes

No

Platform Supported

Classroom

CyberQ

Lab**3**

Perform LDAP Enumeration

This method of enumeration uses LDAP to generate a list of distributed directory services on the target system.

Lab Scenario

As a professional ethical hacker or penetration tester, the next step after SNMP enumeration is to perform LDAP enumeration to access directory listings within Active Directory or other directory services. Directory services provide hierarchically and logically structured information about the components of a network, from lists of printers to corporate email directories. In this sense, they are similar to a company's org chart.

LDAP enumeration allows you to gather information about usernames, addresses, departmental details, server names, etc.

Lab Objectives

- Perform LDAP enumeration using Active Directory Explorer (AD Explorer)
- Perform LDAP enumeration using Python and Nmap
- Perform LDAP enumeration using ldapsearch

Lab Environment

To carry out this lab, you need:

- Windows 11 virtual machine
- Windows Server 2022 virtual machine
- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 25 Minutes

Overview of LDAP Enumeration

LDAP (Lightweight Directory Access Protocol) is an Internet protocol for accessing distributed directory services over a network. LDAP uses DNS (Domain Name System) for quick lookups and fast resolution of queries. A client starts an LDAP session by connecting to a DSA (Directory System Agent), typically on TCP port 389, and sends an operation request to the DSA, which then responds. BER (Basic Encoding Rules) is used to transmit information between the client and the server. One can anonymously query the LDAP service for sensitive information such as usernames, addresses, departmental details, and server names.

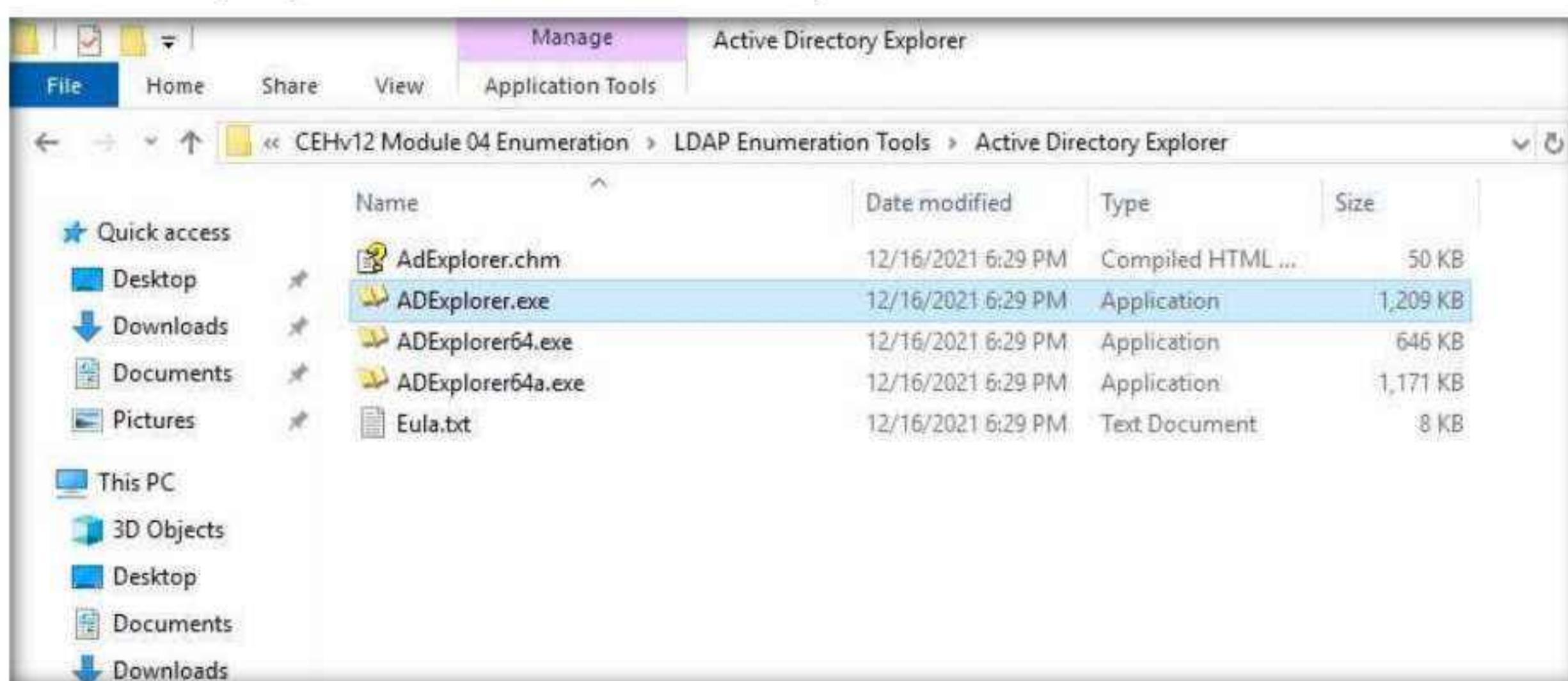
Lab Tasks

Task 1: Perform LDAP Enumeration using Active Directory Explorer (AD Explorer)

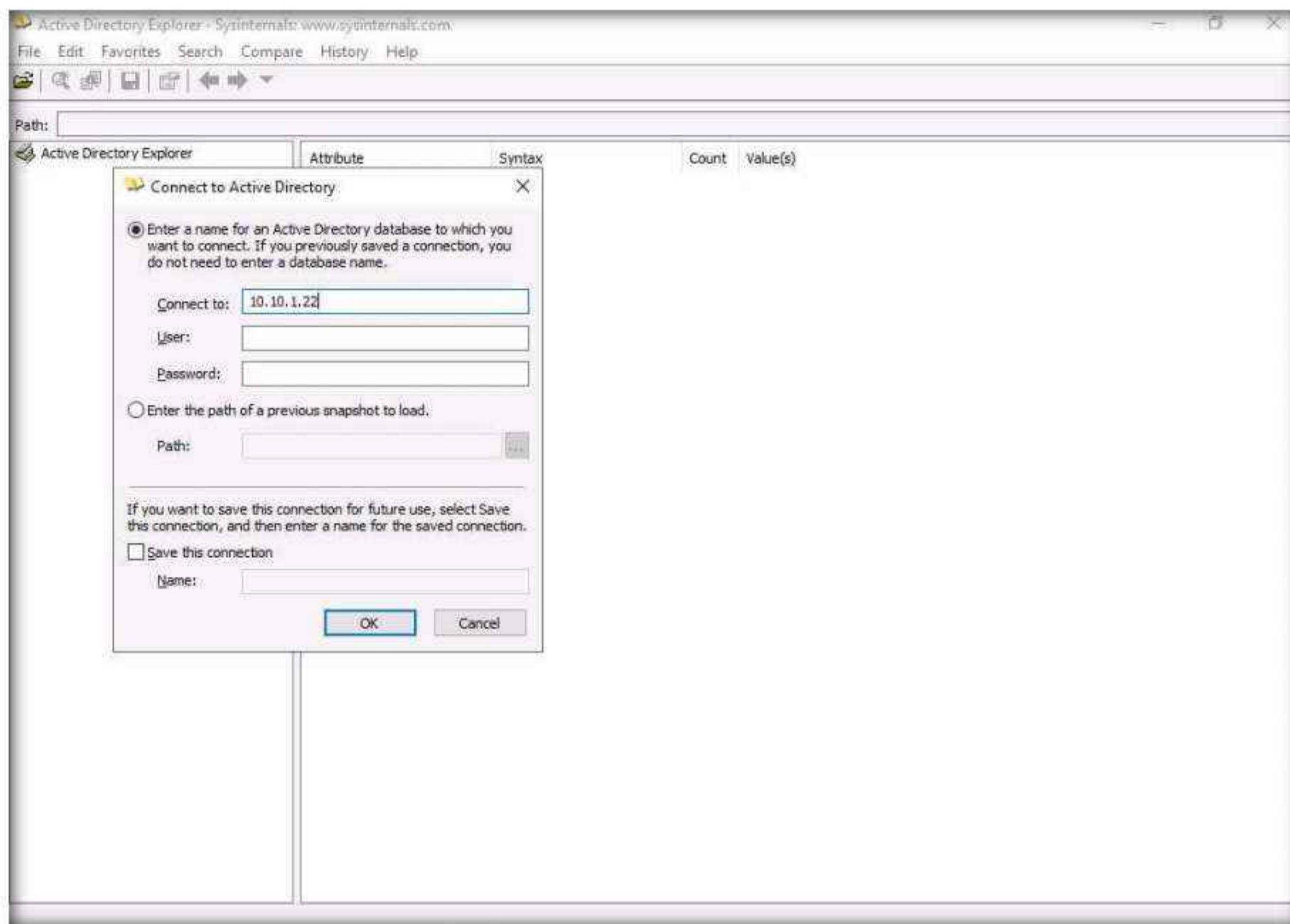
Active Directory Explorer (AD Explorer) is an advanced Active Directory (AD) viewer and editor. It can be used to navigate an AD database easily, define favorite locations, view object properties and attributes without having to open dialog boxes, edit permissions, view an object's schema, and execute sophisticated searches that can be saved and re-executed.

Here, we will use the AD Explorer to perform LDAP enumeration on an AD domain and modify the domain user accounts.

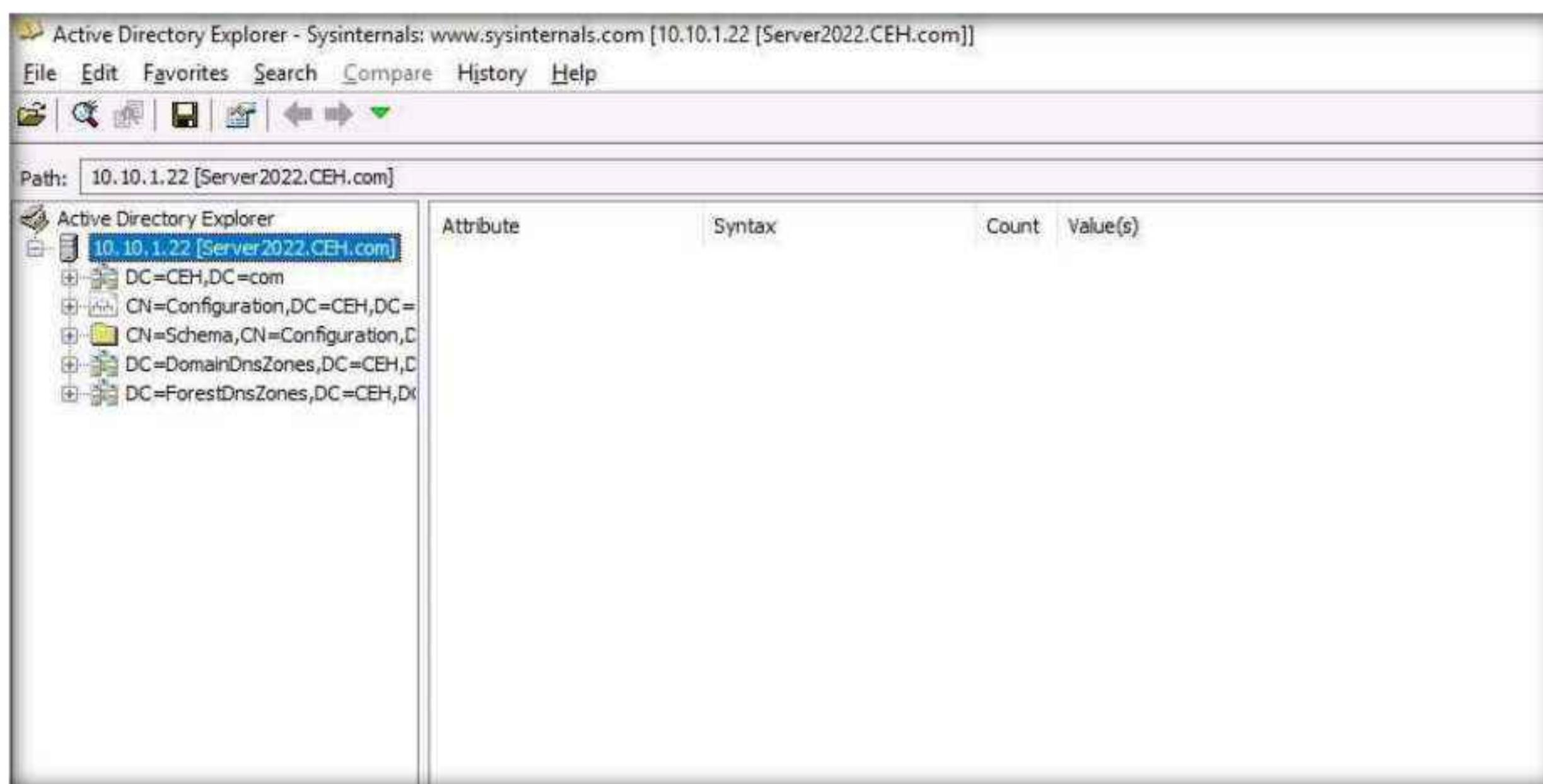
1. Turn on the **Windows 11**, **Windows Server 2022** and **Windows Server 2019** virtual machines.
2. Switch to the **Windows Server 2019** virtual machine. Click **Ctrl+Alt+Del** to activate the machine. By default, **Administrator** user profile is selected, type **Pa\$\$w0rd** in the **Password** field and press **Enter** to login.
3. Navigate to **Z:\CEHv12 Module 04 Enumeration\LDAP Enumeration Tools\Active Directory Explorer** and double-click **ADExplorer.exe**.



4. The **Active Directory Explorer License Agreement** window appears; click **Agree**.
5. The **Connect to Active Directory** pop-up appears; type the IP address of the target in the **Connect to** field (in this example, we are targeting the **Windows Server 2022** machine: **10.10.1.22**) and click **OK**.

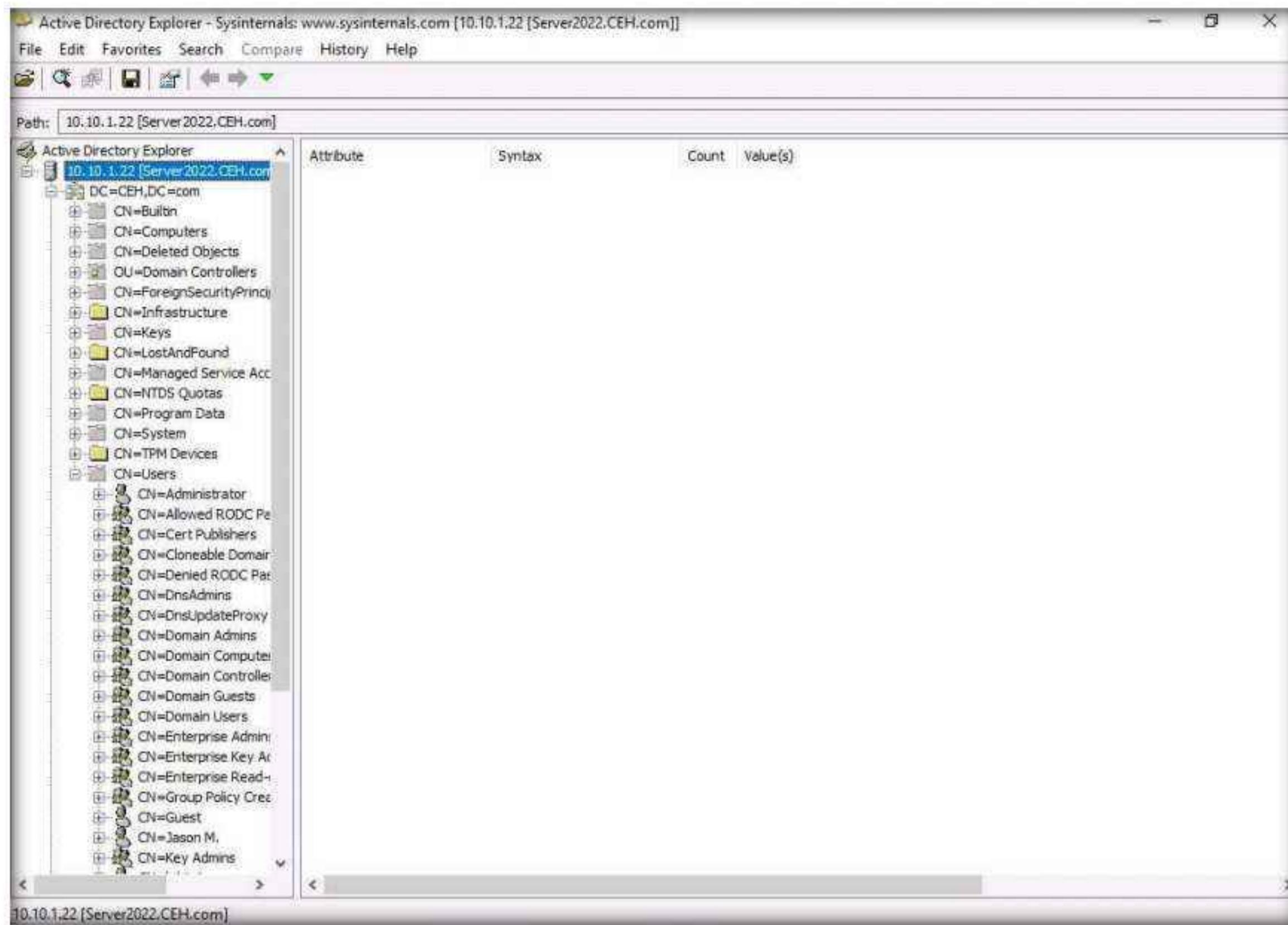


6. The **Active Directory Explorer** displays the active directory structure in the left pane, as shown in the screenshot.

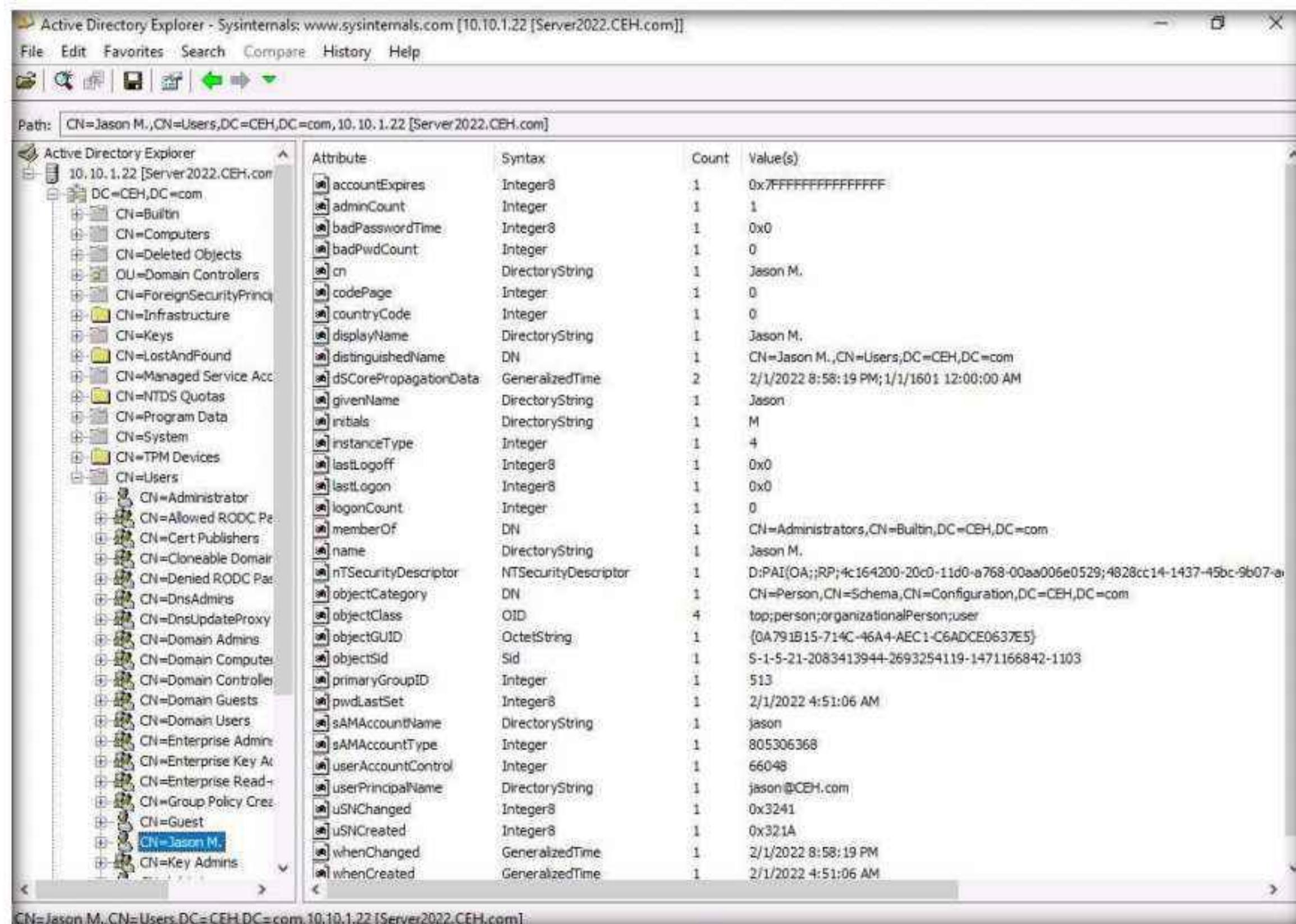


Module 04 – Enumeration

7. Now, expand **DC=CEH**, **DC=com**, and **CN=Users** by clicking “+” to explore domain user details.

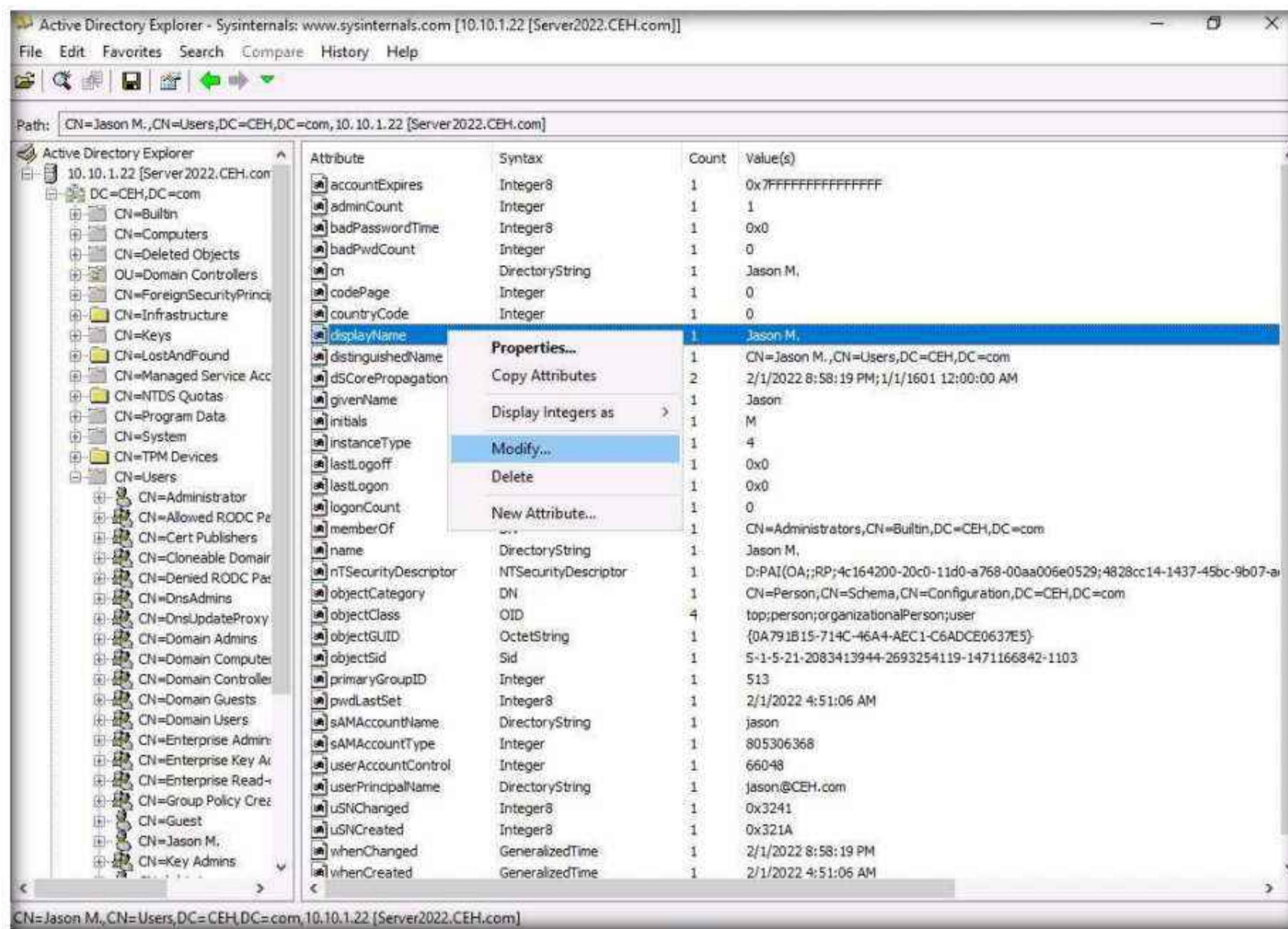


8. Click any **username** (in the left pane) to display its properties in the right pane.

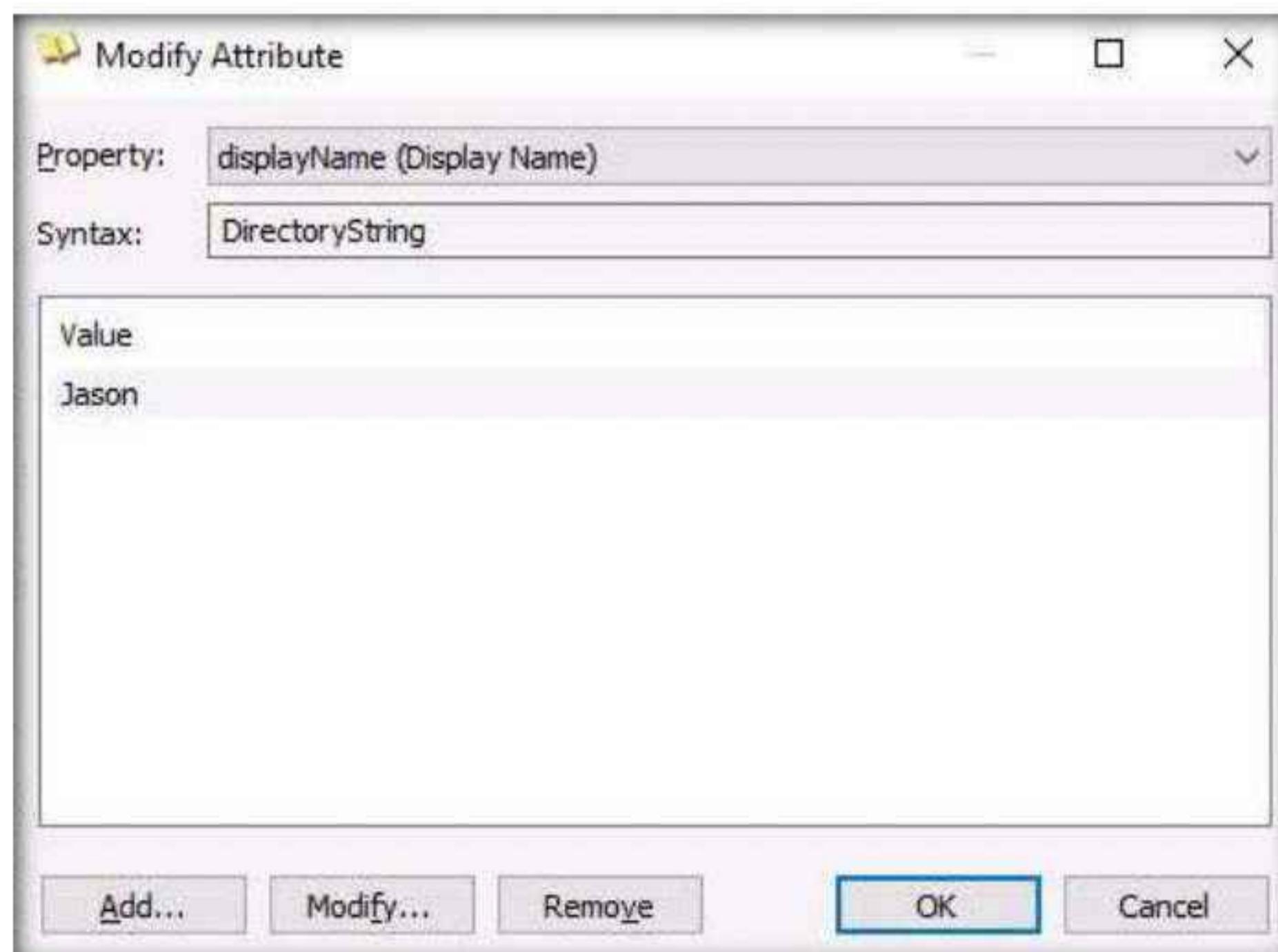


Module 04 – Enumeration

9. Right-click any attribute in the right pane (in this case, **displayName**) and click **Modify...** from the context menu to modify the user's profile.



10. The **Modify Attribute** window appears. First, select the username under the **Value** section, and then click the **Modify...** button. The **Edit Value** pop-up appears. Rename the username in the **Value data** field and click **OK** to save the changes.



11. You can read and modify other user profile attributes in the same way.
12. This concludes the demonstration of performing LDAP enumeration using AD Explorer.
13. You can also use other LDAP enumeration tools such as **Softerra LDAP Administrator** (<https://www.ldapadministrator.com>), **LDAP Admin Tool** (<https://www.ldapsoft.com>), **LDAP Account Manager** (<https://wwwldap-account-manager.org>), and **LDAP Search** (<https://securityxploded.com>) to perform LDAP enumeration on the target.
14. Close all open windows and document all the acquired information.
15. Turn off the **Windows 11** and **Windows Server 2019** virtual machines.

Task 2: Perform LDAP Enumeration using Python and Nmap

LDAP enumeration can be performed using both manual and automated methods. Using various Python commands LDAP enumeration is performed on the target host to obtain information such as domains, naming context, directory objects, etc. Using NSE script can be used to perform queries to brute force LDAP authentication using the built-in username and password lists.

Here, we will use Nmap and python commands to extract details on the LDAP server and connection.

Note: Ensure that the **Windows Server 2022** virtual machine is running.

1. Turn on the **Parrot Security** virtual 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-left corner 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 **Parrot Terminal** window, type **nmap -sU -p 389 [Target IP address]** (here, the target IP address is **10.10.1.22**) and press **Enter**.

Note: **-sU:** performs a UDP scan and **-p:** specifies the port to be scanned.

7. The results appear, displaying that the port 389 is **open** and being used by LDAP, as shown in the screenshot below.

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

The screenshot shows a terminal window titled "nmap -sU -p 389 10.10.1.22 - Parrot Terminal". The session starts with the user "attacker" at the root prompt, entering "sudo su" and providing the password. The user then runs "nmap -sU -p 389 10.10.1.22" to scan the target host. The output shows that port 389/udp is open and is identified as "ldap". The MAC address of the target host is listed as "00:15:5D:01:80:02 (Microsoft)". The scan summary indicates it took 0.21 seconds to scan one host. The user then exits the terminal.

```
nmap -sU -p 389 10.10.1.22 - Parrot Terminal
[attacker@parrot]~[-]
$ sudo su
[sudo] password for attacker:
[root@parrot]~[/home/attacker]
# nmap -sU -p 389 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-29 06:07 EDT
Nmap scan report for 10.10.1.22
Host is up (0.00091s latency).

PORT      STATE SERVICE
389/udp    open  ldap
MAC Address: 00:15:5D:01:80:02 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.21 seconds
[root@parrot]~[/home/attacker]
#
```

8. Now, we will use NSE script to perform username enumeration on the target machine **Windows Server 2022** (10.10.1.22).

9. Type **nmap -p 389 --script ldap-brute --script-args**

ldap.base=""cn=users,dc=CEH,dc=com"" [Target IP Address] (here, the target IP address is **10.10.1.22**) and press **Enter**.

Note: **-p:** specifies the port to be scanned, **ldap-brute:** to perform brute-force LDAP authentication. **ldap.base:** if set, the script will use it as a base for the password guessing attempts.

10. Nmap attempts to brute-force LDAP authentication and displays the usernames that are found, as shown in the screenshot below.

```

nmap -p389 --script ldap-brute --script-args ldap.base="cn=users,dc=CEH,dc=com" 10.10.1.22 - Parrot Terminal
File Edit View Search Terminal Help

PORT      STATE SERVICE
389/udp    open  ldap
MAC Address: 00:15:5D:01:80:02 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.21 seconds
[+] root@parrot:[~/home/attacker]
[-] # nmap -p 389 --script ldap-brute --script-args ldap.base='cn=users,dc=CEH,dc=com' 10.10.1.22
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-29 06:09 EDT
Nmap scan report for 10.10.1.22
Host is up (0.0014s latency).

PORT      STATE SERVICE
389/tcp    open  ldap
| ldap-brute:
|   cn=root,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=admin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=administrator,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=webadmin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=sysadmin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=netadmin,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=guest,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=user,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=web,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
|   cn=test,cn=users,dc=CEH,dc=com:<empty> => Valid credentials
MAC Address: 00:15:5D:01:80:02 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
[+] root@parrot:[~/home/attacker]
[-] #

```

11. Close the terminal window. Now, we will perform manual LDAP Enumeration using Python.
12. Click the **MATE Terminal** icon at the top-left corner of the **Desktop** to open a **Terminal** window.
13. A **Parrot Terminal** window appears. In the terminal window, type **sudo su** and press **Enter** to run the programs as a root user.
14. 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.

15. Type **python3** and press **Enter** to open a python3 shell.

```

python3 - Parrot Terminal
File Edit View Search Terminal Help
[attacker@parrot:~]
[-] $sudo su
[sudo] password for attacker:
[root@parrot:~]
[-] #python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 

```

16. Type **import ldap3** and press Enter to import LDAP.

```
[attacker@parrot](-)
└─$ sudo su
[sudo] password for attacker:
[root@parrot](-[/home/attacker]
└─# python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import ldap3
>>>
```

17. Now, we will connect to the target LDAP server without credentials using python.

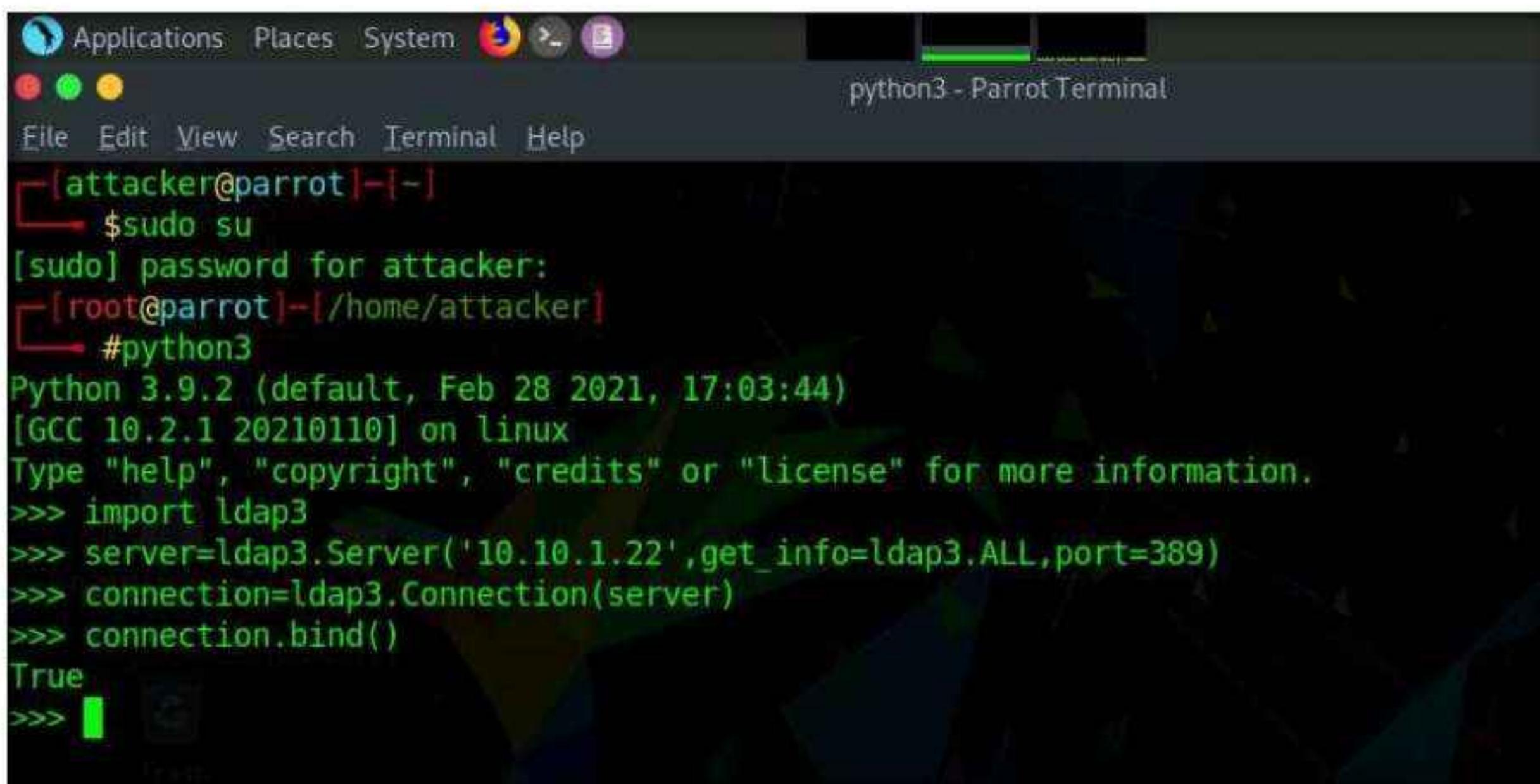
18. Type **server=ldap3.Server('[Target IP Address]', get_info=ldap3.ALL, port=[Target Port])** and press Enter to provide the target IP address and port number (here, the target IP address is **10.10.1.22**, and the port number is **389**).

```
[attacker@parrot](-)
└─$ sudo su
[sudo] password for attacker:
[root@parrot](-[/home/attacker]
└─# python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import ldap3
>>> server=ldap3.Server('10.10.1.22',get_info=ldap3.ALL,port=389)
>>>
```

19. In the python3 shell, type **connection=ldap3.Connection(server)** and press Enter.

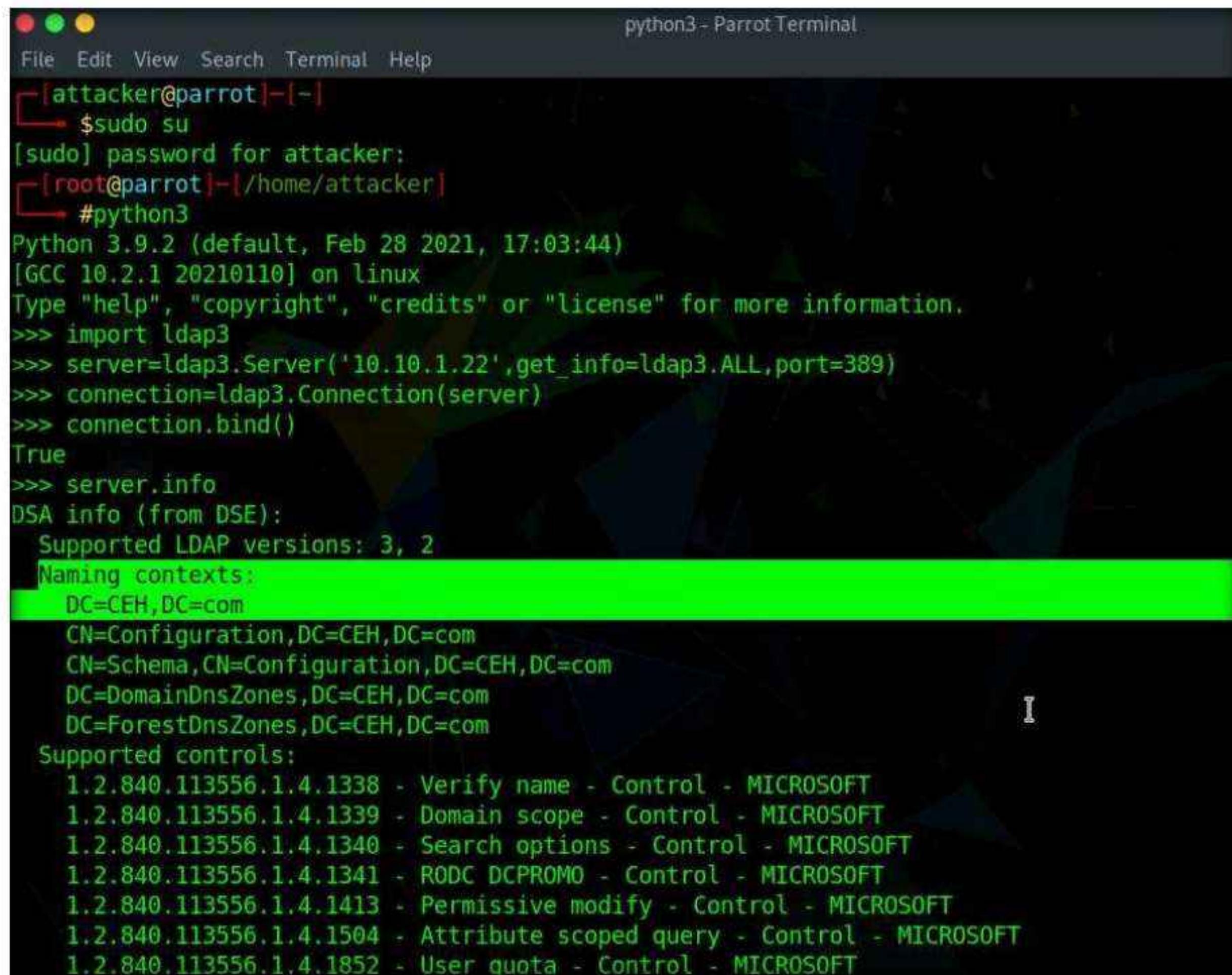
```
[attacker@parrot](-)
└─$ sudo su
[sudo] password for attacker:
[root@parrot](-[/home/attacker]
└─# python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import ldap3
>>> server=ldap3.Server('10.10.1.22',get_info=ldap3.ALL,port=389)
>>> connection=ldap3.Connection(server)
>>>
```

20. Type **connection.bind()** and press **Enter** to bind the connection. We will receive response as **True** which means the connection is established successfully



```
[attacker@parrot]~[-]
└─$ sudo su
[sudo] password for attacker:
[root@parrot]~[~/home/attacker]
└─# python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import ldap3
>>> server=ldap3.Server('10.10.1.22',get_info=ldap3.ALL,port=389)
>>> connection=ldap3.Connection(server)
>>> connection.bind()
True
>>>
```

21. Type **server.info** and press **Enter** to gather information such as naming context or domain name, as shown in the screenshot below.



```
[attacker@parrot]~[-]
└─$ sudo su
[sudo] password for attacker:
[root@parrot]~[~/home/attacker]
└─# python3
Python 3.9.2 (default, Feb 28 2021, 17:03:44)
[GCC 10.2.1 20210110] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import ldap3
>>> server=ldap3.Server('10.10.1.22',get_info=ldap3.ALL,port=389)
>>> connection=ldap3.Connection(server)
>>> connection.bind()
True
>>> server.info
DSA info (from DSE):
  Supported LDAP versions: 3, 2
  Naming contexts:
    DC=CEH,DC=com
      CN=Configuration,DC=CEH,DC=com
      CN=Schema,CN=Configuration,DC=CEH,DC=com
      DC=DomainDnsZones,DC=CEH,DC=com
      DC=ForestDnsZones,DC=CEH,DC=com
  Supported controls:
    1.2.840.113556.1.4.1338 - Verify name - Control - MICROSOFT
    1.2.840.113556.1.4.1339 - Domain scope - Control - MICROSOFT
    1.2.840.113556.1.4.1340 - Search options - Control - MICROSOFT
    1.2.840.113556.1.4.1341 - RODC DCPROMO - Control - MICROSOFT
    1.2.840.113556.1.4.1413 - Permissive modify - Control - MICROSOFT
    1.2.840.113556.1.4.1504 - Attribute scoped query - Control - MICROSOFT
    1.2.840.113556.1.4.1852 - User quota - Control - MICROSOFT
```

22. After receiving the naming context, we can make more queries to the server to extract more information.
 23. In the terminal window, type
`connection.search(search_base='DC=CEH,DC=com',search_filter='(&(objectclass*))',search_scope='SUBTREE', attributes='*')` and press **Enter**.
 24. Type `connection.entries` and press **Enter** to retrieve all the directory objects.

Applications Places System > python3 - Parrot Terminal

```
>>> connection.search(search_base='DC=CEH,DC=com', search_filter='(&(objectclass=*))', search_scope='SUBTREE', attributes='*')
True
>>> connection.entries
[DN: DC=CEH,DC=com - STATUS: Read - READ TIME: 2022-03-29T06:50:11.036562
 auditingPolicy:
 creationTime: 132930309893191915
 dSASignature: b'\x01\x00\x00\x00(\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x9e\x89\xc2D\xf5!\x9fM\x9cd\xd8X\x91dB\xbf'
 dSCorePropagationData: 16010101000000.0Z
 dc: CEH
 distinguishedName: DC=CEH,DC=com
 fSMORoleOwner: CN=NTDS Settings,CN=SERVER2022,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=CEH,DC=com
 forceLogoff: -9223372036854775808
 gPLink: [LDAP://CN={31B2F340-016D-1102-945F-00C04FB984F9},CN=Policies,CN=System,DC=CEH,DC=com;0]
 instanceType: 5
 isCriticalSystemObject: TRUE
 lockOutObservationWindow: -18000000000
 lockoutDuration: -18000000000
 lockoutThreshold: 0
 masteredBy: CN=NTDS Settings,CN=SERVER2022,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=CEH,DC=com
 maxPwdAge: -9223372036854775808
 minPwdAge: 0
 minPwdLength: 0
 modifiedCount: 1
 modifiedCountAtLastProm: 0
 ms-DS-MachineAccountQuota: 10
 msDS-AllUsersTrustQuota: 1000]
```

25. In the python3 shell, type
`connection.search(search_base='DC=CEH,DC=com',search_filter='(&(objectclass=person)',search_scope='SUBTREE', attributes='userpassword')` and press Enter. True response indicates that the query is successfully executed.

26. Type `connection.entries` and press Enter to dump the entire LDAP information.

```

python3 - Parrot Terminal
File Edit View Search Terminal Help
READ TIME: 2022-03-29T06:50:11.088000
cn: Windows Virtual Machine
dSCorePropagationData: 20220329095440.0Z
16010101000001.0Z
distinguishedName: CN=Windows Virtual Machine,CN=SERVER2022,OU=Domain Controllers,DC=CEH,DC=com
instanceType: 4
name: Windows Virtual Machine
objectCategory: CN=Service-Connection-Point,CN=Schema,CN=Configuration,DC=CEH,DC=com
objectClass: top
leaf
REALNAME
connectionPoint
serviceConnectionPoint
objectGUID: b'(Q\xd5\xf8\x1b\xla\x8fH\x86\xab\xf3"\x1e%\x84'
showInAdvancedViewOnly: TRUE
uSNCreated: 20493
uSNChanged: 20493
whenChanged: 20220202061319.0Z
whenCreated: 20220202061319.0Z
]
>>> connection.search(search_base='DC=CEH,DC=com', search_filter='(&(objectclass=person))', search_scope='SUBTREE', attributes='userpassword')
True
>>> connection.entries
[DN: CN=Guest,CN=Users,DC=CEH,DC=com - STATUS: Read - READ TIME: 2022-03-29T06:57:00.575720
, DN: CN=SERVER2022,OU=Domain Controllers,DC=CEH,DC=com - STATUS: Read - READ TIME: 2022-03-29T06:57:00.575775
, DN: CN=Martin J.,CN=Users,DC=CEH,DC=com - STATUS: Read - READ TIME: 2022-03-29T06:57:00.575816
, DN: CN=Shiela D.,CN=Users,DC=CEH,DC=com - STATUS: Read - READ TIME: 2022-03-29T06:57:00.575853
]
>>>

```

27. Using this information attackers can launch web application attacks and they can also gain access to the target machine.
28. This concludes the demonstration of LDAP enumeration using Nmap and Python.
29. Close all open windows and document all the acquired information.

Task 3: Perform LDAP Enumeration using ldapsearch

ldapsearch is a shell-accessible interface to the `ldap_search_ext(3)` library call. `ldapsearch` opens a connection to an LDAP server, binds the connection, and performs a search using the specified parameters. The filter should conform to the string representation for search filters as defined in RFC 4515. If not provided, the default filter, `(objectClass=*)`, is used.

Here, we will use `ldapsearch` to perform LDAP enumeration on the target system.

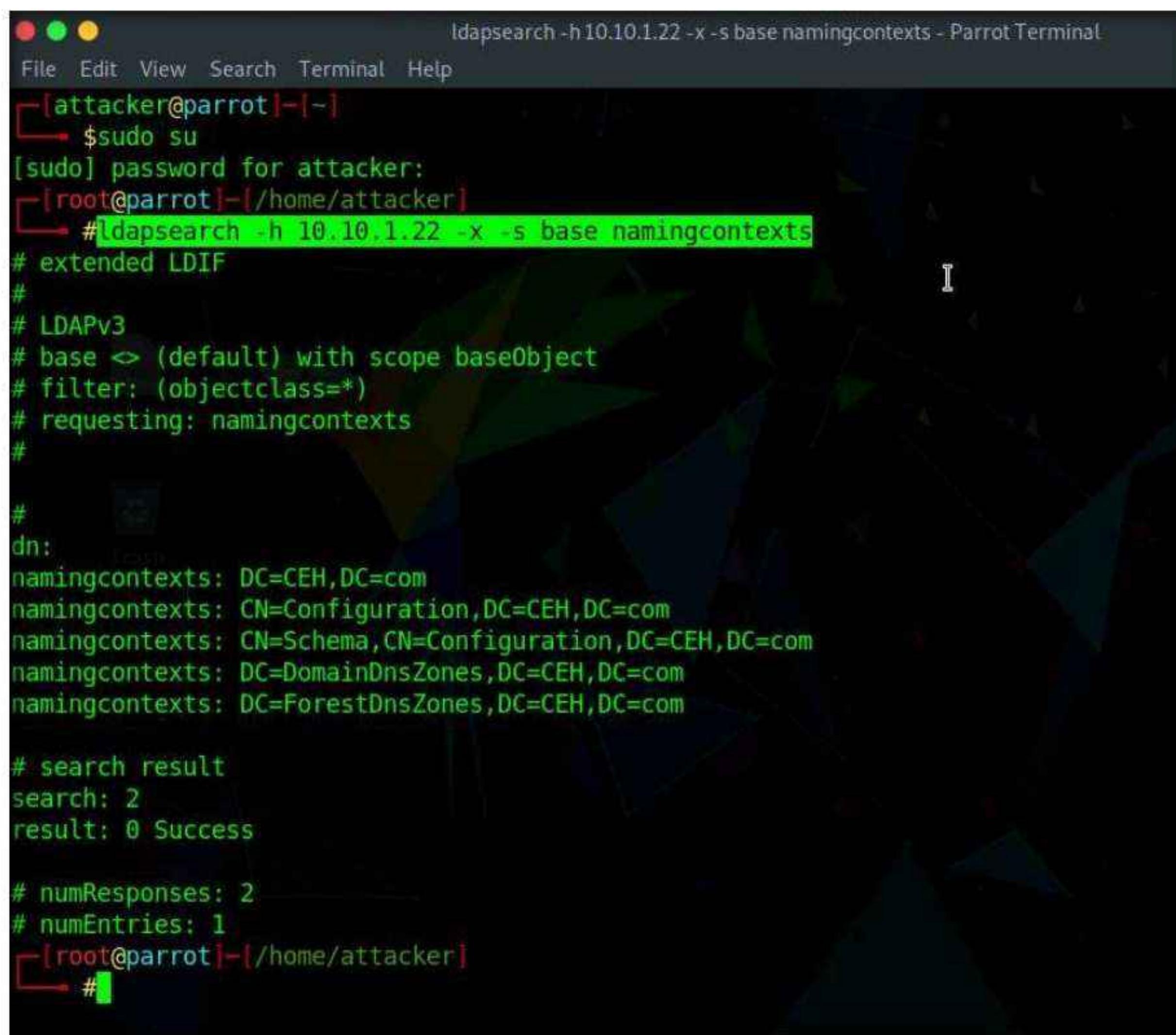
Note: Ensure that the **Windows Server 2022** virtual machine is running.

1. In **Parrot Security** virtual machine, click the **MATE Terminal** icon at the top-left corner 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. In the terminal, type **ldapsearch -h [Target IP Address] -x -s base namingcontexts** and press **Enter** (here, the target IP address is **10.10.1.22**), to gather details related to the naming contexts.

Note: **-x:** specifies simple authentication, **-h:** specifies the host, and **-s:** specifies the scope.



```
File Edit View Search Terminal Help
[attacker@parrot:~]
$ sudo su
[sudo] password for attacker:
[root@parrot:~]
# ldapsearch -h 10.10.1.22 -x -s base namingcontexts
# extended LDIF
#
# LDAPv3
# base <> (default) with scope baseObject
# filter: (objectclass=*)
# requesting: namingcontexts
#
#
dn:
namingcontexts: DC=CEH,DC=com
namingcontexts: CN=Configuration,DC=CEH,DC=com
namingcontexts: CN=Schema,CN=Configuration,DC=CEH,DC=com
namingcontexts: DC=DomainDnsZones,DC=CEH,DC=com
namingcontexts: DC=ForestDnsZones,DC=CEH,DC=com

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
[root@parrot:~]
#
```

5. Type **ldapsearch -h [Target IP Address] -x -b "DC=CEH,DC=com"** and press **Enter** (here, the target IP address is **10.10.1.22**), to obtain more information about the primary domain.

Note: **-x:** specifies simple authentication, **-h:** specifies the host, and **-b:** specifies the base DN for search.

```
root@parrot:~/home/attacker# ldapsearch - h 10.10.1.22 -x -b "DC=CEH,DC=com"
# extended LDIF
#
# LDAPv3
# base <DC=CEH,DC=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# CEH.com
dn: DC=CEH,DC=com
objectClass: top
objectClass: domain
objectClass: domainDNS
distinguishedName: DC=CEH,DC=com
instanceType: 5
whenCreated: 20220201120107.0Z
whenChanged: 20220329095440.0Z
subRefs: DC=ForestDnsZones,DC=CEH,DC=com
subRefs: DC=DomainDnsZones,DC=CEH,DC=com
subRefs: CN=Configuration,DC=CEH,DC=com
uSNCreated: 4099
dSASignature:: AQAAACgAAAAAAAAAAAAAAAAnonCRPUhn02cZNhYkWRCvw==
uSNChanged: 41013
name: CEH
objectGUID:: uw6KhEWuwkCFNdTAaGEoIQ==
replUpToDateVector:: AgAAAAAAAAADAAAAAAAJ6JwKT1IZ9NnGTYWJFkOr8GgAAAAAAAJ2GD
BgDAAAAlNyj/YudE23x0j6xuRTZwigAAAAAAASo5TGAMAAAB3ndbWP6QMT4P9ccK6EhZMB5AAAA
AAAABrdREYAwAAAA==
```

6. Type **ldapsearch -x -h [Target IP Address] -b "DC=CEH,DC=com" "objectclass=*" and press **Enter** (here, the target IP address is **10.10.1.22**), to retrieve information related to all the objects in the directory tree.**

Note: **-x:** specifies simple authentication, **-h:** specifies the host, and **-b:** specifies the base DN for search.

```
File Edit View Search Terminal Help
[root@parrot]# /home/attacker
[root@parrot]# #ldapsearch -x -h 10.10.1.22 -b "DC=CEH,DC=com" "objectClass=*"
# extended LDIF
#
# LDAPv3
# base <DC=CEH,DC=com> with scope subtree
# filter: objectClass=*
# requesting: ALL
#
# CEH.com
dn: DC=CEH,DC=com
objectClass: top
objectClass: domain
objectClass: domainDNS
distinguishedName: DC=CEH,DC=com
instanceType: 5
whenCreated: 20220201120107.0Z
whenChanged: 20220329095440.0Z
subRefs: DC=ForestDnsZones,DC=CEH,DC=com
subRefs: DC=DomainDnsZones,DC=CEH,DC=com
subRefs: CN=Configuration,DC=CEH,DC=com
uSNCreated: 4099
dSASignature:: AQAAACgAAAAAAAAAAAAAAAAnonCRPUhn02cZNhYkWRCvw==
uSNChanged: 41013
name: CEH
objectGUID:: uw6KhEWuwkCFNdTAaGEoIQ==
repUpToDateVector:: AgAAAAAAAAADAAAAAAAJ6Jwkt1IZ9NnGTYWJFkQr8GgAAAAAAAJ2GD
BgdAAAAAmNyj/YudE23x0j6xuRTZwigAAAAAAASo5TGAMAAAB3ndbWP6QMT4P9ccK6EhZMB5AAAA
AAAAABrdREYAwAAAA==
```

7. Attackers use `ldapsearch` for enumerating AD users. It allows attackers to establish connection with an LDAP server to carry out different searches using specific filters.
8. This concludes the demonstration of performing LDAP enumeration using `ldapsearch`.
9. Close all open windows and document all the acquired information.
10. Turn off the **Windows Server 2022** and **Parrot Security** virtual machines.

Lab Analysis

Analyze and document the results of this lab exercise.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS RELATED TO THIS LAB.

Internet Connection Required

Yes

No

Platform Supported

Classroom

CyberQ

Lab

4

Perform NFS Enumeration

NFS enumeration is a method by which exported directories and shared data on target systems is extracted.

Lab Scenario

As a professional ethical hacker or penetration tester, the next step after LDAP enumeration is to perform NFS enumeration to identify exported directories and extract a list of clients connected to the server, along with their IP addresses and shared data associated with them.

After gathering this information, it is possible to spoof target IP addresses to gain full access to the shared files on the server.

Lab Objectives

- Perform NFS enumeration using RPCScan and SuperEnum

Lab Environment

To carry out this lab, you need:

- Windows Server 2019 virtual machine
- Parrot Security virtual machine
- Web browsers with an Internet connection
- Administrator privileges to run the tools

Lab Duration

Time: 10 Minutes

Overview of NFS Enumeration

NFS (Network File System) is a type of file system that enables computer users to access, view, store, and update files over a remote server. This remote data can be accessed by the client computer in the same way that it is accessed on the local system.