

EC-Council

C | C T


Certified Cybersecurity Technician

Module - 16

Network Troubleshooting

This page is intentionally left blank.

Module Objectives



- 01** Understanding the Importance of Network Troubleshooting
- 02** Understanding Basic Network Issues
- 03** Understanding How to Troubleshoot Network Issues
- 04** Overview of Troubleshooting Network Issues using Various Tools and Utilities

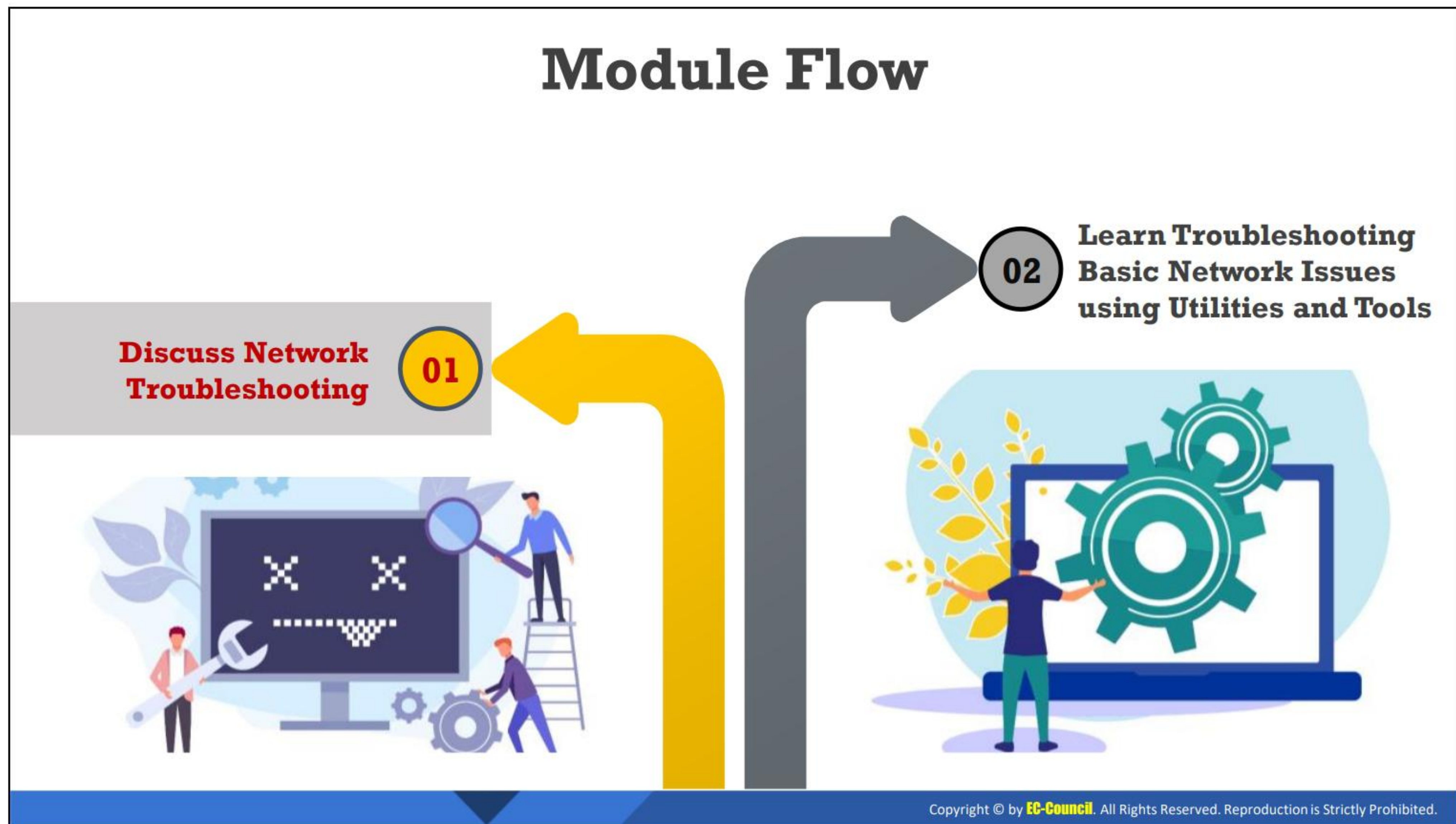
Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Objectives

Most Internet users encounter certain failures or defects when using a network or while working on a system. These interruptions must be resolved as soon as possible to prevent any possible damage. Therefore, it is necessary to identify the problem first and then fix it. Troubleshooting a network is more effective than attempting various random methods as it focuses on targeting individual network components and testing each component; the process can also be documented for future use. This module discusses various network issues and the techniques and tools used for network troubleshooting.

At the end of this module, you will be able to do the following:

- Understand the importance of network troubleshooting
- Understand basic network issues
- Explain how to troubleshoot network issues
- Use various tools and utilities to troubleshoot network issues









Discuss Network Troubleshooting

This section discusses common network issues and various troubleshooting techniques to resolve them.

Troubleshooting

❑ Troubleshooting is the process of finding **issues** in a computer network and diagnosing them

Typical Network Issues

 Physical Connection Issues <ul style="list-style-type: none">➤ Network connectivity issues due to faulty or loose connection of cables	 Connectivity Issues <ul style="list-style-type: none">➤ Network failure or faulty configuration of ports or interfaces in LAN and WAN, which may affect the connectivity to the host server	 Configuration Issues <ul style="list-style-type: none">➤ Misconfiguration of DHCP and DNS settings, causing routing issues that result in communication failure
 Software Issues <ul style="list-style-type: none">➤ Incompatible software and version mismatch, leading to disruptions of the transmission of IP data packets between the source and destination	 Traffic Overload issues <ul style="list-style-type: none">➤ Network behavior changes when traffic exceeds the capacity of the network devices	 Network IP Issues <ul style="list-style-type: none">➤ Improper IP settings, subnet mask, and routing at the source, resulting in the interruption of communication with the destination IP

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting

Troubleshooting is the process executed by network engineers/administrators to find issues in a computer network and diagnose them to enhance the network performance. It comprises systematic measures and processes to identify, diagnose, and resolve network issues to recover normal network operations on end nodes. Troubleshooting is primarily performed by network engineers/administrators. The process of troubleshooting can be performed manually or by using automated tools (network diagnostic software).

Some of the processes involved in network troubleshooting are as follows:

- Finding and resolving the faulty network connection of a system
- Configuring network devices (routers, switches, etc.)
- Installing physical equipment (cables, Wi-Fi devices, etc.)
- Updating firmware devices
- Removing viruses using antivirus software
- Adding, configuring, and reinstalling devices

Typical Network Issues

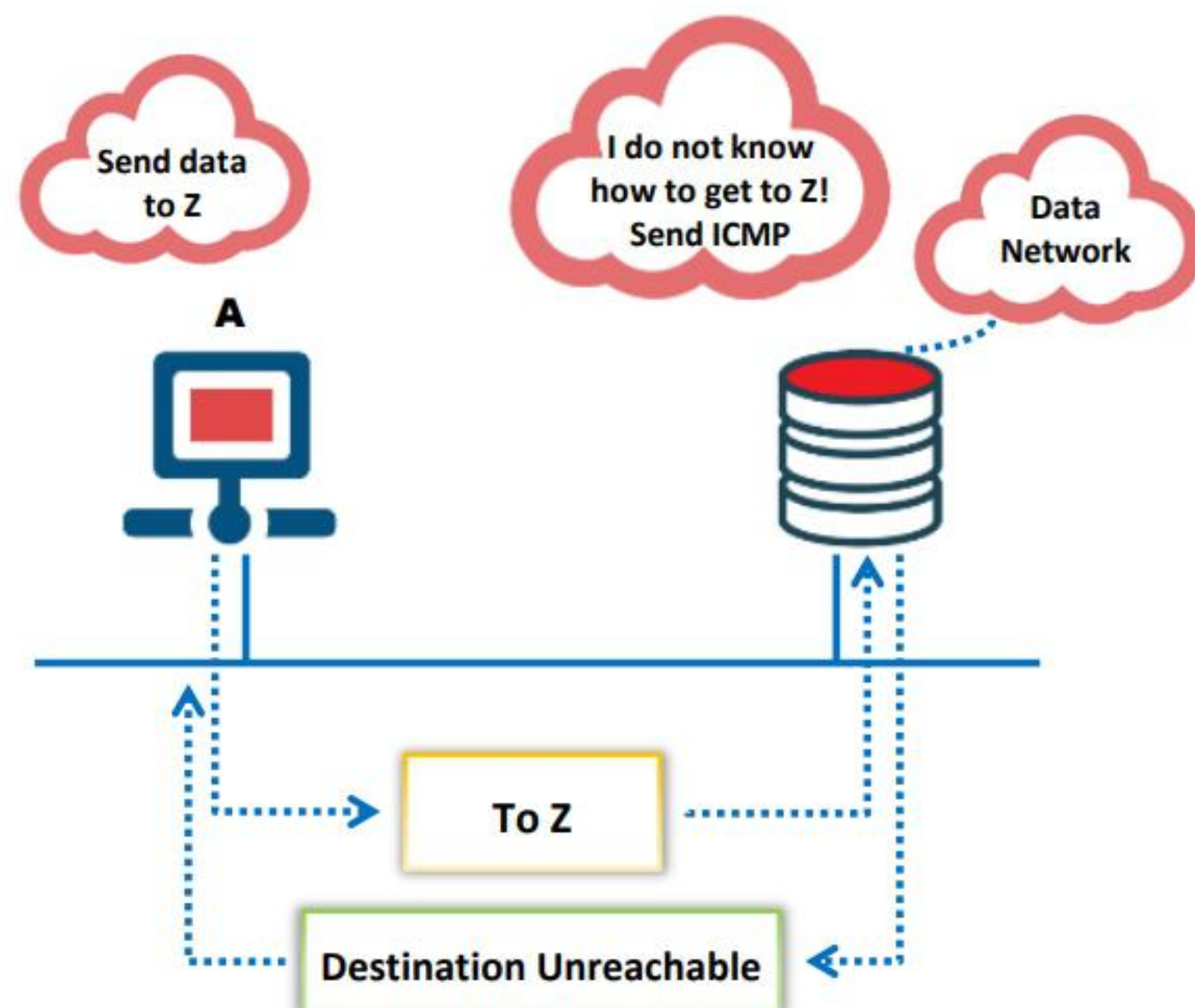
The issues that generally occur in a network are as follows:

- **Physical connection issues:** These issues occur due to faulty, shortened, or damaged cables in a network. For example, cut cables, incorrect cable connections, and connector problems may result in poor network connectivity.

- **Connectivity issues:** These issues are caused by faulty/physical shutting down of the port or interface on which devices are connected in a LAN or WAN. For example, the malfunction of a hub, router, or switch may affect the network connectivity with the host server.
- **Configuration issues:** These issues occur due to the faulty configuration of DHCP and DNS settings, routing issues, IP looping, etc. and may result in communication failure.
- **Software issues:** These issues are caused by software compatibility issues and mismatched software versions. These may result in the interruption of the transmission of IP data packets between the source and destination.
- **Traffic overload issues:** These issues occur due to traffic that exceeds the normal capacity of a network device. For example, transfers of an excessive amount of information due to a poor network setup may result in abnormal network behavior or slow connectivity.
- **Network IP issues:** These issues are caused by the faulty configuration of IP addresses, subnet masks, and routing at the source and may result in interruption in reaching the destination IP through the network.

Basic Network Issues: Unreachable Networks

- ❑ An ICMP destination **unreachable message** is sent in the case of the following:
 - ✓ The host or port is unreachable
 - ✓ The network is unreachable



Echo Request/Reply When host is up

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Martin>ping 10.10.10.16

Pinging 10.10.10.16 with 32 bytes of data:
Reply from 10.10.10.16: bytes=32 time=1ms TTL=128
Reply from 10.10.10.16: bytes=32 time<1ms TTL=128
Reply from 10.10.10.16: bytes=32 time<1ms TTL=128
Reply from 10.10.10.16: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.10.16:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\Martin>
```

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters		
Data.....		

Echo = Type 8; Echo Reply = Type 0

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Basic Network Issues: Unreachable Networks

Network communication depends on certain basic conditions being met.

- Sending and receiving devices must have the TCP/IP protocol stack properly configured:
 - The IP address and subnet mask must be properly configured.
 - A default gateway must also be configured if datagrams are to travel outside of the local network.
- A router must also have the TCP/IP protocol stack properly configured on its interfaces, and it must use an appropriate routing protocol.
- If these conditions are not met, then network communication cannot occur.
- The following are examples of problems:
 - The sending device may address the datagram to a non-existent IP address.
 - The destination device is not connected to its network.
 - The router's connecting interface is down.
 - The router does not have the information necessary to find the destination network.
- An ICMP destination unreachable message is sent in the case of the following issues:
 - Host or port unreachable
 - Network unreachable

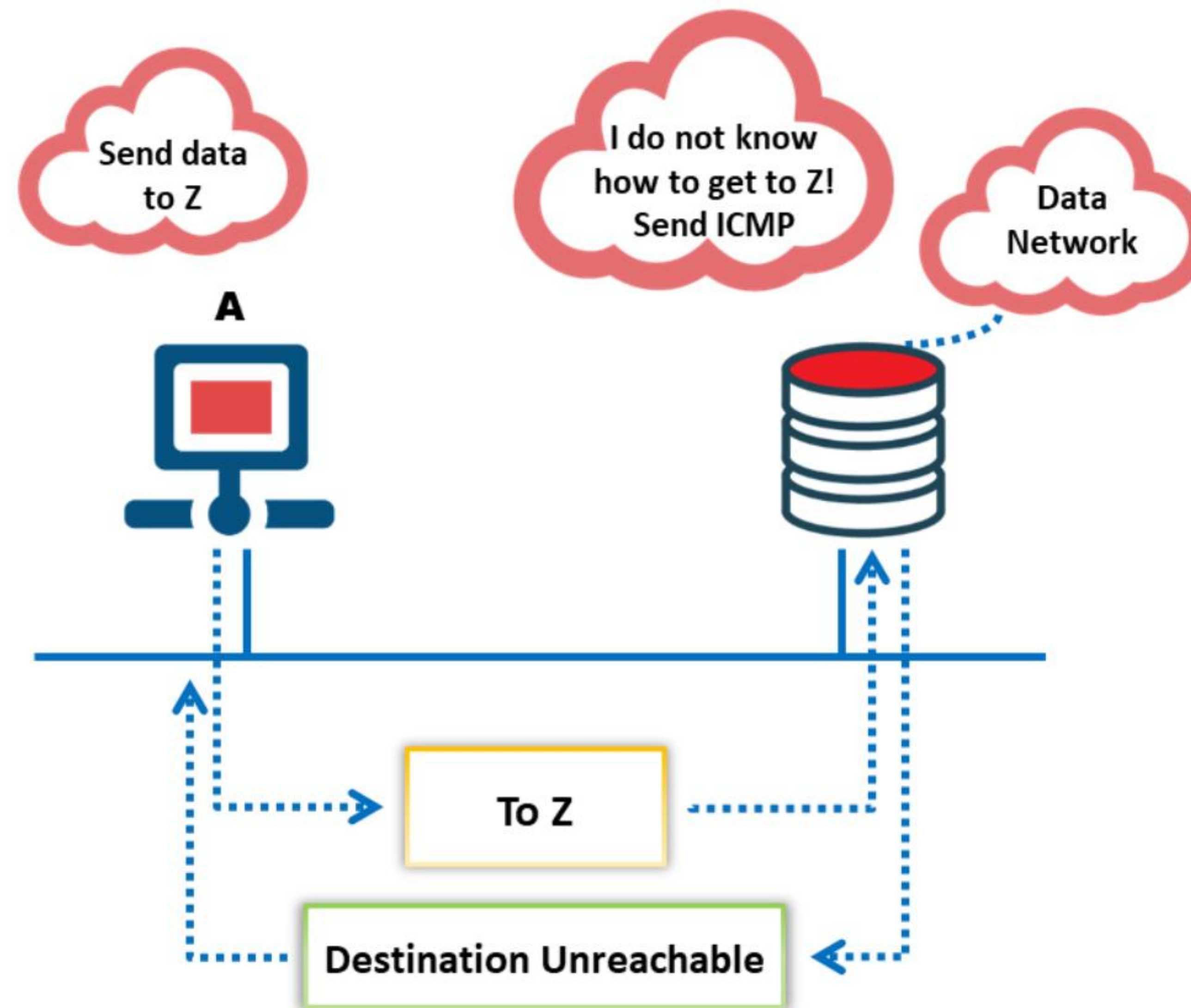


Figure 16.1: Unreachable Destination

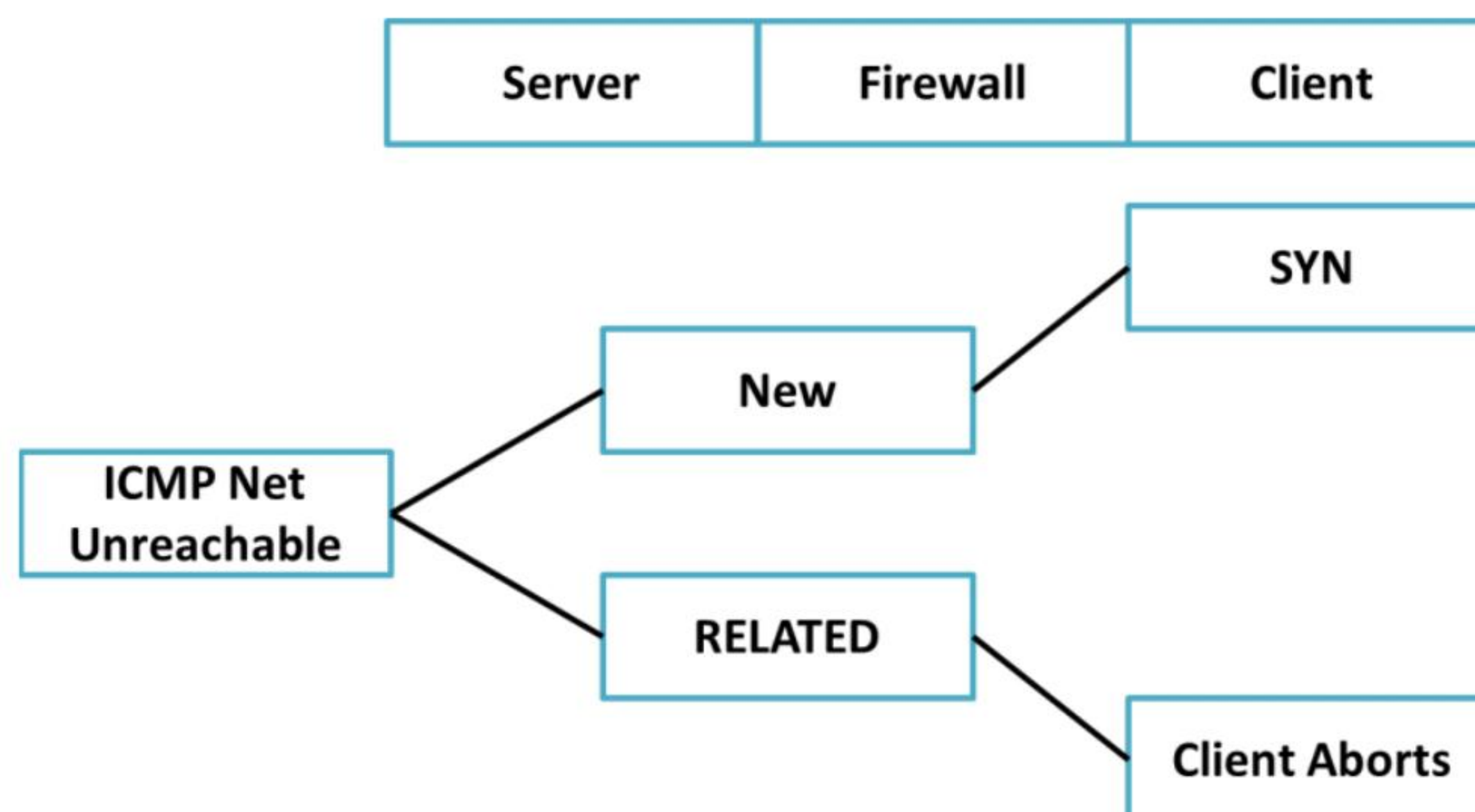


Figure 16.2: Unreachable-Destination Problems arising when Specific Conditions are not Met

ICMP Echo (Request) and Echo Reply

An ICMP echo request is a query-based ICMP message that helps in diagnosing a network. By using this message, a system in the network can be tested to verify whether it is ready for data transmission. To do so, the first system that wishes to communicate with another system sends an ICMP echo request message. If the second system is ready for the communication, it responds with an ICMP echo reply message.

Format of ICMP Echo Requests and Replies

The figure shows the fields in an ICMP echo request/reply.

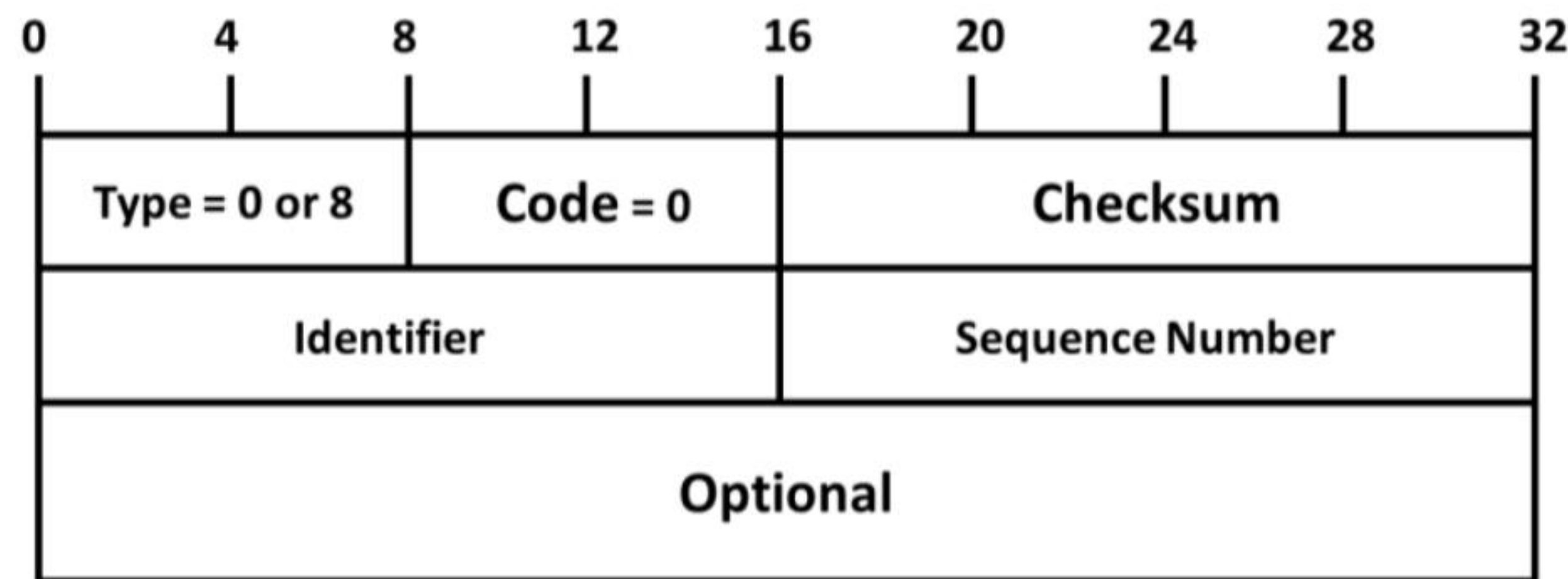
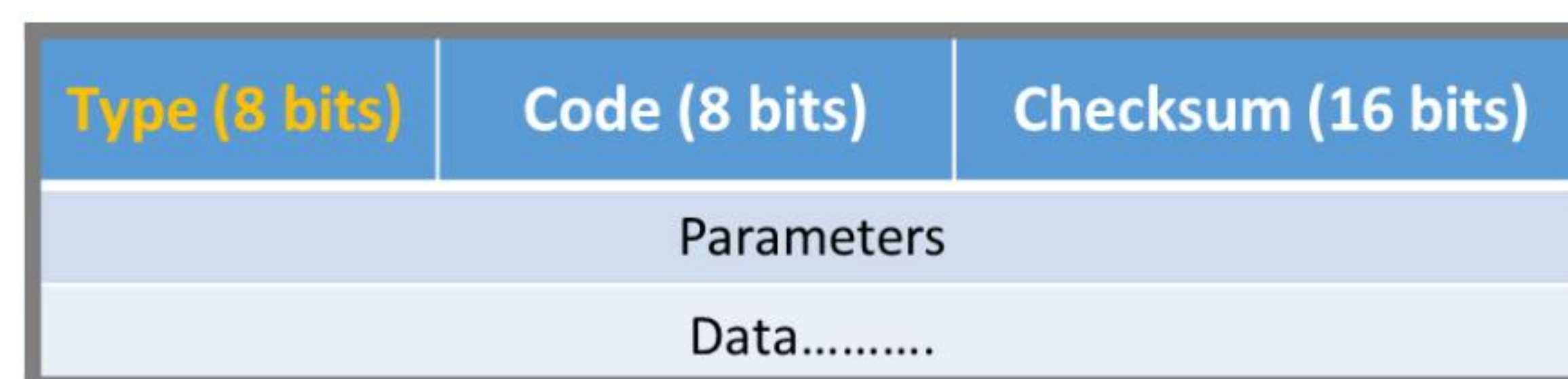


Figure 16.3: Format of an ICMP Echo Request/Reply

Each field in the ICMP echo request/reply format is described below.

- **Type (1 byte):** This field defines the type of the ICMP message. For an ICMP echo request, its value is 8, and for an ICMP echo reply, its value is 0.
- **Code (1 byte):** This field does not define anything for these message types; therefore, it is set to 0.
- **Checksum (2 bytes):** This field defines a checksum for the ICMP header.
- **Identifier (2 bytes):** This field is used to compare echo request and echo reply messages.
- **Sequence number (2 bytes):** This field contains a number for comparing echo request and echo reply messages.
- **Optional (variable):** This is an optional field that contains information about the message.

ICMP echo requests and replies can be sent through the ping command. This command provides important information such as how many echo request messages were pinged, the message size, and how many responses were received.



Echo = Type 8; Echo Reply = Type 0

Figure 16.4: ICMP Echo and Response

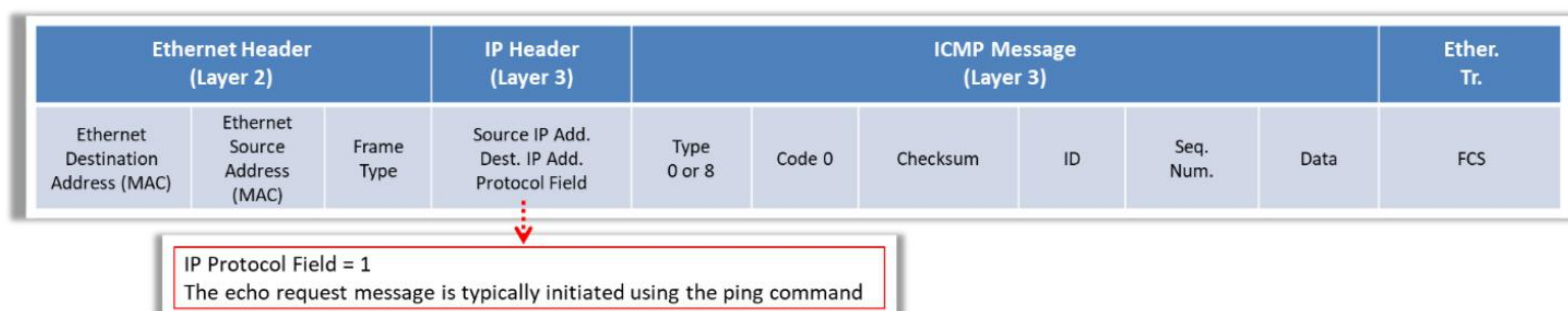


Figure 16.5: ICMP Frame


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Martin>ping 10.10.10.16

Pinging 10.10.10.16 with 32 bytes of data:
Reply from 10.10.10.16: bytes=32 time=1ms TTL=128
Reply from 10.10.10.16: bytes=32 time<1ms TTL=128
Reply from 10.10.10.16: bytes=32 time<1ms TTL=128
Reply from 10.10.10.16: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.10.16:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\Martin>
```

Figure 16.6: Echo Request/Reply When host is up

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Martin>ping 10.0.10.16

Pinging 10.0.10.16 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.0.10.16:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\Users\Martin>
```

Figure 16.7: Echo Request/Reply When host is down

Basic Network Issues: Destination Unreachable Message

- If a datagram cannot be forwarded to its destination, **ICMP** returns a destination unreachable message, indicating to the sender that the datagram could not be properly forwarded
- A destination unreachable message may also be sent when packet fragmentation is required to forward a packet:
 - Fragmentation is usually necessary when a datagram is forwarded from a Token Ring network to an Ethernet network
 - If the datagram does not allow fragmentation, the packet cannot be forwarded; consequently, a destination unreachable message is sent
- **Destination unreachable messages** may also be generated if IP-related services such as FTP or web services are unavailable



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.10586]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping 10.10.10.16

Pinging 10.10.10.16 with 32 bytes of data:
Reply from 10.10.10.10: Destination host unreachable.
Reply from 10.10.10.10: Destination host unreachable.
Reply from 10.10.10.10: Destination host unreachable.
Reply from 10.10.10.10: Destination host unreachable.

Ping statistics for 10.10.10.16:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Windows\system32>
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Basic Network Issues: Destination Unreachable Message

We have already discussed the concept of unreachable networks in previous sections. As we know, IP is a connectionless protocol that does not consider the information being sent. If a host that IP attempts to send information to is unavailable, this has to be notified to IP. This notification can be accomplished using ICMP destination unreachable message.

If a datagram cannot be forwarded to its destination, ICMP returns a destination unreachable message indicating to the sender that the datagram could not be properly forwarded.

A destination unreachable message may also be sent when packet fragmentation is required to forward a packet:

- Fragmentation is usually necessary when a datagram is forwarded from a Token Ring network to an Ethernet network.
- If the datagram does not allow fragmentation, the packet cannot be forwarded; consequently, a destination unreachable message is sent.

Destination unreachable messages may also be generated if IP-related services such as FTP or web services are unavailable.

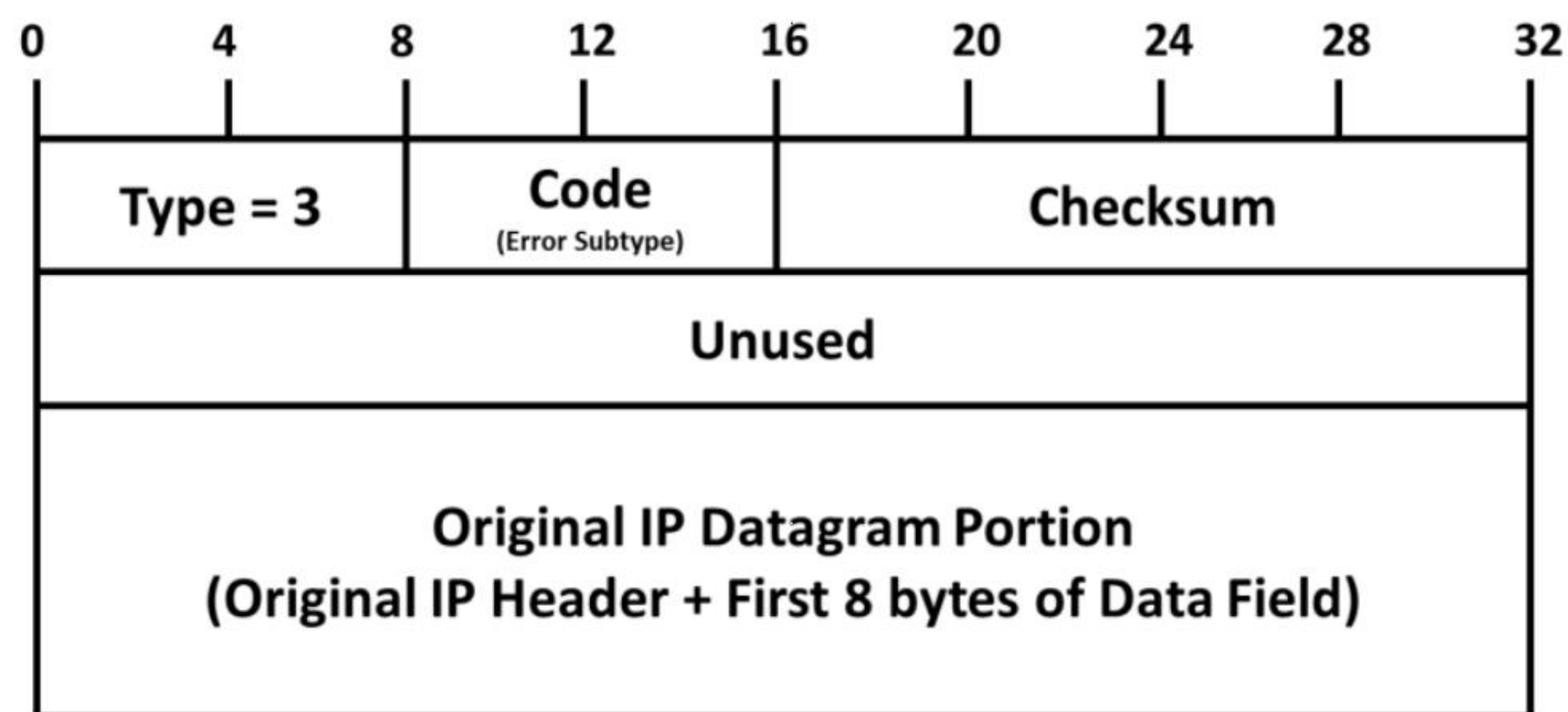


Figure 16.8: Format of an ICMP Destination Unreachable Message

Each field in the ICMP destination unreachable message is described below.

- **Type (1 byte):** This field defines the type of the ICMP message; for a destination unreachable message, its value is 3.
- **Code (1 byte):** This field defines the reason behind the error, and a series of numbers represent various types of errors; for example, code 0 represents a network unreachable error, and code 1 represents a host unreachable error.
- **Checksum (2 bytes):** This field defines a checksum for the ICMP header.
- **Unused (4 bytes):** This field is left blank.
- **Original datagram portion (variable):** This field defines the IP header of the datagram and the first 8 bytes of the datagram that prompted this error message to be sent.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.10586]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping 10.10.10.16

Pinging 10.10.10.16 with 32 bytes of data:
Reply from 10.10.10.10: Destination host unreachable.
Reply from 10.10.10.10: Destination host unreachable.
Reply from 10.10.10.10: Destination host unreachable.
Reply from 10.10.10.10: Destination host unreachable.

Ping statistics for 10.10.10.16:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Windows\system32>
```

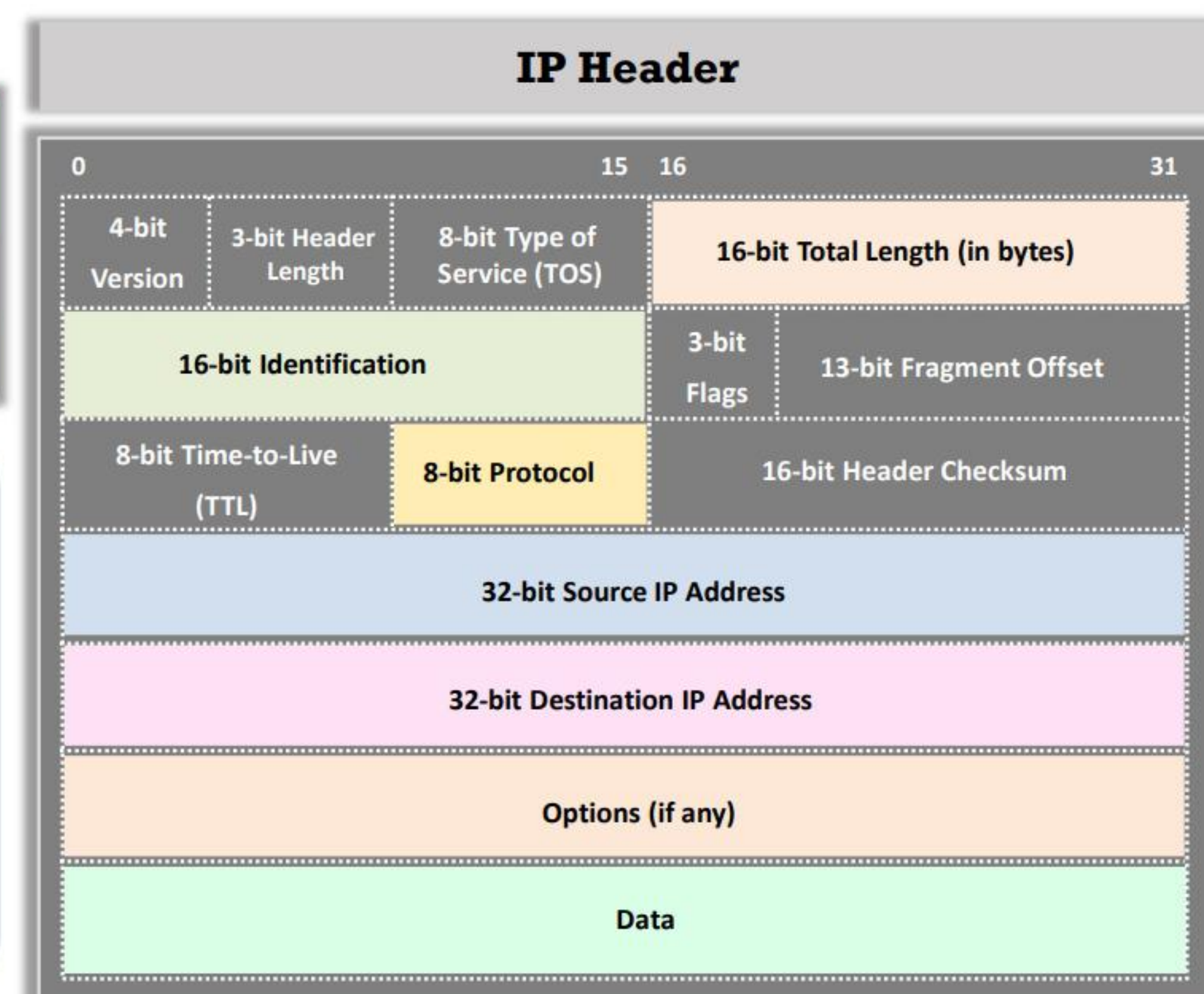
Figure 16.9: Destination Unreachable Message

Basic Network Issues: Time Exceeded Message

ICMP Time Exceeded
Type = 11

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters		
Data.....		

- A TTL value is defined in each datagram (IP packet)
- As each router processes the datagram, it decreases the **TTL value by one**
- When the TTL of the datagram value reaches zero, the packet is **discarded**
- ICMP uses a time exceeded message to notify the source device that the TTL of the datagram has been exceeded



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Basic Network Issues: Time Exceeded Message

In huge networks with hundreds of interconnected devices, the packet delay is a common problem. This delay might be caused by too many routers to choose the shortest path, router issues, router loop, etc.

The router loop problem arises in the following kind of scenario:

- Let there be two networks exchanging information.
- The first network sends a packet to router R1, and R1 must choose the shortest path to reach the second network.
- R1 chooses router R2 as the shortest path and sends the packet to it.
- R2 chooses router R3 as the shortest path and sends the packet to it.
- R3 chooses router R1 as the shortest path and sends the packet to it.
- Likewise, the packet loops around these routers indefinitely, causing the router loop problem.

A router loop is a serious problem that causes packets to loop around a network continuously. To avoid this kind of overhead, the IP header of a packet contains a time to live (TTL) field that sets the number of hops the packet can travel. Each time the packet reaches a router, its TTL value reduces by 1, and the process continuous until TTL = 0. At this moment, the packet loses its lifetime and expires. The device at which the packet expired, sends an ICMP time exceeded message to the source machine that sent the packet.

The figure shows the TTL expiry scenario.

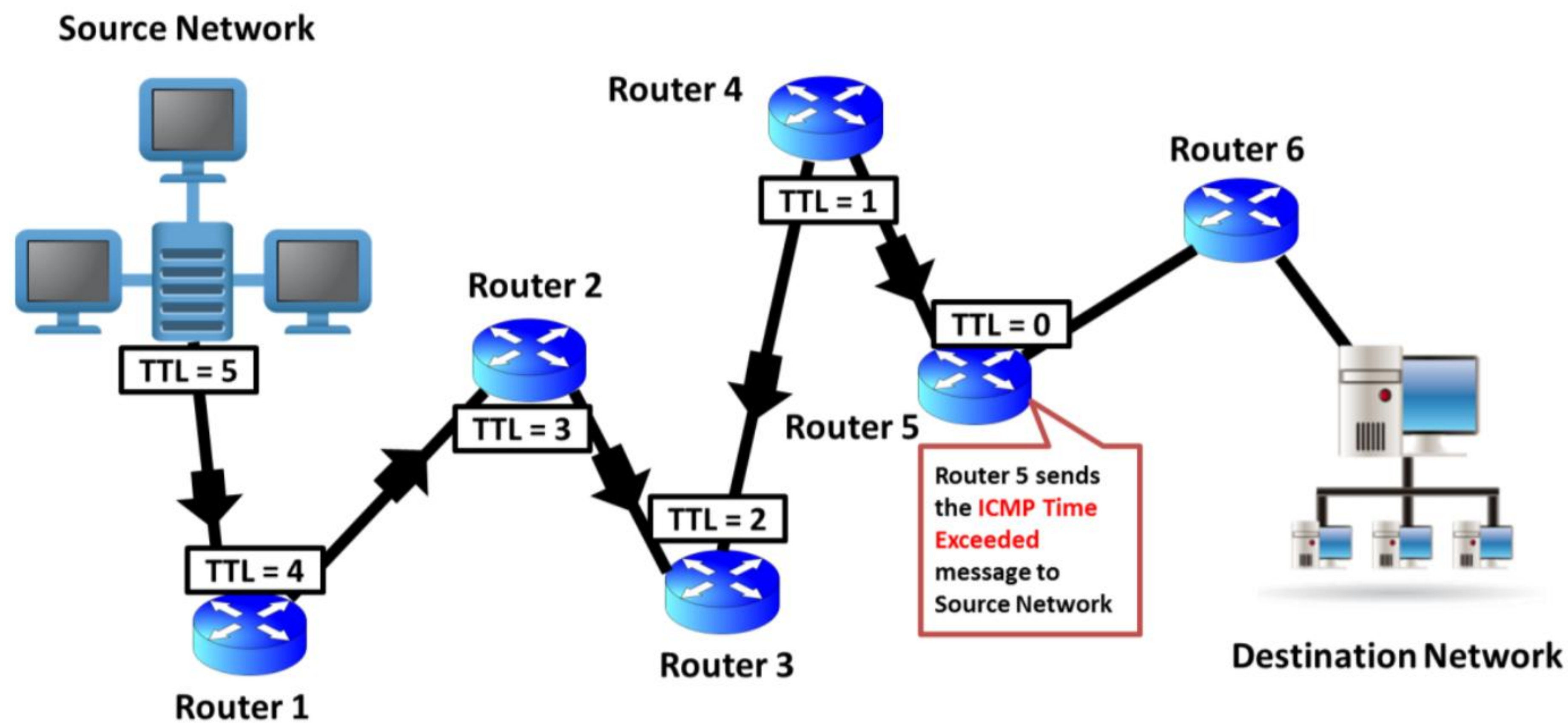


Figure 16.10: TTL Expiry Scenario

There is another scenario that leads to packet expiry and creates an error message. In certain situations, the IP packet is fragmented into small parts; these fragments choose different routing paths to reach the destination. It is the duty of the destination machine to join all these fragments into a full packet after the arrival of all the packets. If a fragment took the shortest path and reached the destination while the others are yet to reach, the destination host must wait till it gathers all the fragments. This may cause the destination host to wait for a long or even indefinite amount of time if any fragments were lost. To avoid such a scenario, the destination host sets a timer when it collects the first fragment and waits for the others. If this timer expires, the destination host discards the fragments that it received and sends an ICMP time exceeded message to the source host.

The ICMP time exceeded message contains the following fields.

- **Type (1 byte):** This field defines the type of ICMP message; for a time exceeded message, it is set to 11.
- **Code (1 byte):** This field defines the reason behind the error, and a series of numbers represent various types of errors; for example, code 0 represents the expiration of TTL, and code 1 represents fragment reassembly timeout.
- **Checksum (2 bytes):** This field defines a checksum for the ICMP header.
- **Unused (4 bytes):** This field is not used and left blank.
- **Original datagram portion:** This field contains the IP header and first 8 bits of the IP packet that was discarded because of the time exceeded error.

ICMP Time Exceeded

Type = 11

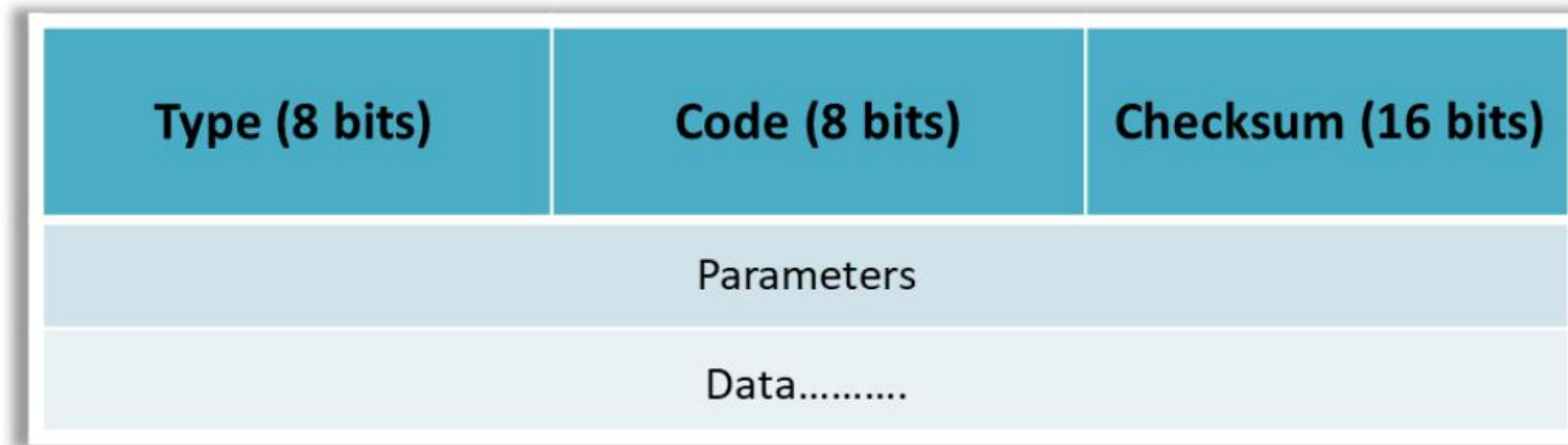


Figure 16.11: ICMP Time Exceeded Frame

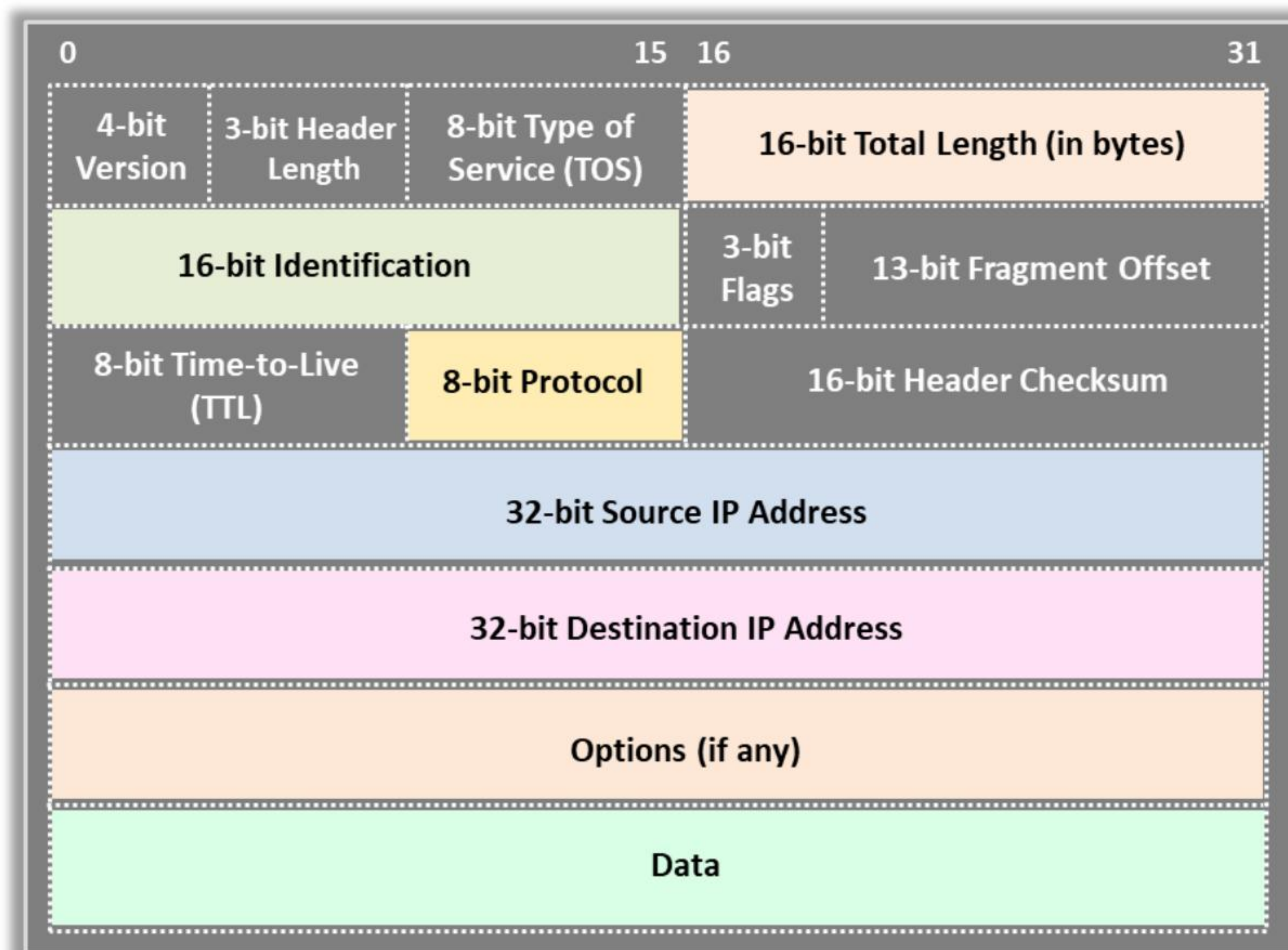


Figure 16.12: IP Header

Basic Network Issues: IP Parameter Problem

- ❑ Devices that **process** datagrams may not be able to forward a **datagram** owing to some type of **error** in the header
- ❑ This error does not relate to the state of the destination **host** or network but still prevents the datagram from being **processed** and **delivered**
- ❑ An ICMP **type-12 parameter** problem message is sent to the **source** of the **datagram**

ICMP Parameter Problem
Type = 12

0	8	16	31
Type (3)	Code (0-12)		Checksum
Unused (must be zero)			
Internet Header + First 64 Bits of Datagram			

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Basic Network Issues: IP Parameter Problem

Devices that process datagrams may not be able to forward a datagram owing to some type of error in the header. This error does not relate to the state of the destination host or network but still prevents the datagram from being processed and delivered. An ICMP type-12 parameter problem message is sent to the source of the datagram.

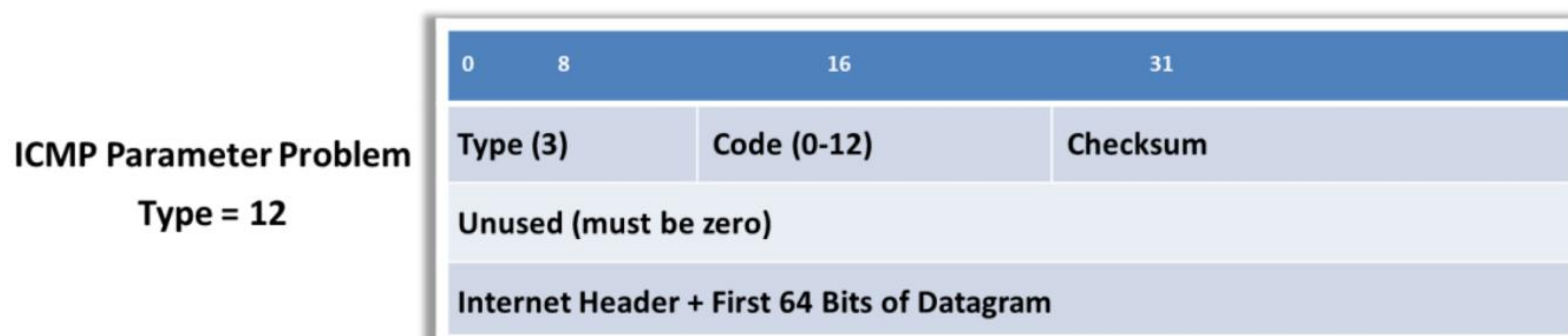


Figure 16.13: IP Parameter Problem

The parameter problem occurs when invalid data exist in the fields of an IP header. In this case, the device that traced this invalid information from the IP header sends an ICMP parameter problem message to the source machine that sent the packet. This message contains a pointer that points to the field that caused the error, instead of error codes.

The figure shows the format of the ICMP parameter problem message.

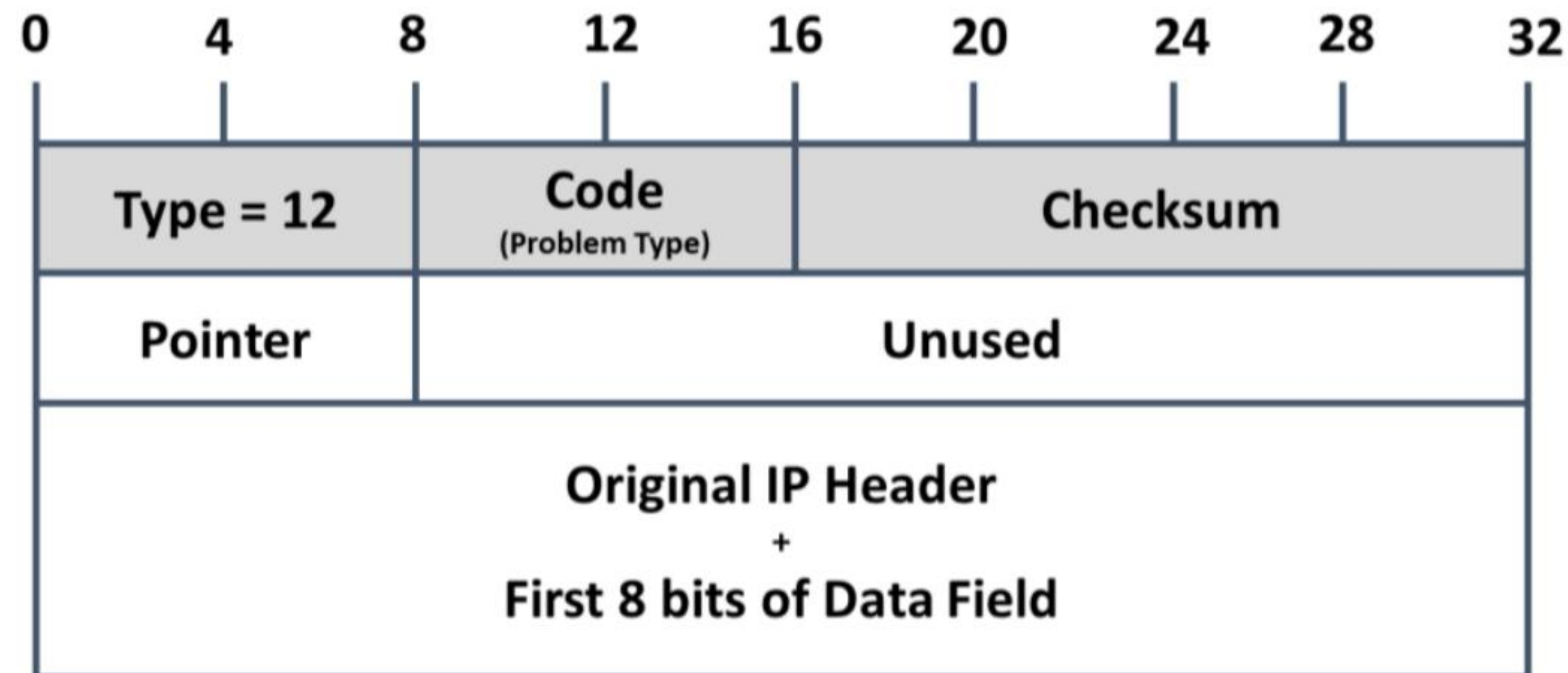


Figure 16.14: Format of an ICMP parameter problem message

Each field of an ICMP parameter problem message is discussed below.

- **Type (1 byte):** This field defines the type of the ICMP message; for parameter problem messages, it is set to 12.
- **Code (1 byte):** This field defines the reason behind the error.
- **Checksum (2 bytes):** This field defines a checksum for the ICMP header.
- **Pointer (1 byte):** This field defines the cause of this message by pointing to the field that is responsible.
- **Unused (3 bytes):** This field is not used and left blank.
- **Original datagram portion:** This field contains the IP header and first 8 bits of the IP packet that was discarded because of the parameter problem.

Code and Pointer fields of the ICMP Parameter Problem Message:

We have already discussed that the pointer field is used to point out the field that caused the parameter problem. Generally, the pointer to the field that caused this error message contains the code value 0. However, to provide a meaningful reason for the generation of the parameter problem message, two more code values are introduced, as described in the table below.

Code	Message	Details
0	Pointer for the field	It is a common way of showing the error field through the pointer.
1	Missing mandatory field	When a required field is missed, this code is indicated. The pointer cannot point an empty field.
2	Bad length	Invalid size of the datagram packet indicates that the error is in whole packet but not with a particular field. Here there is nothing to do with a pointer to a field.

Table 16.1: Pointer and Code fields in an ICMP parameter problem message

Basic Network Issues: ICMP Control Messages



Unlike error messages, control messages are not the result of lost packets or error conditions that occur during packet transmission



Instead, they are used to **inform hosts** of conditions such as the following:

- ✓ Network congestion
- ✓ Existence of a better gateway to a remote network

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Basic Network Issues: ICMP Control Messages

Certain values in the type field are used to identify control messages. The code field provides more information about the message. Unlike error messages, control messages are not the result of lost packets or error conditions that occur during packet transmission. Instead, they are used to indicate conditions such as the following.

Instead, they are used to indicate conditions such as the following:

- **Network congestion** When a router begins buffering packets because of failure to process and transmit them as quickly as they are absorbed, an ICMP Source Quench message is generated, which causes the rate of packet transmission to slow down. Generating too many Source Quench messages can cause further network congestion; therefore, they are used sparsely.

ICMP collects the source IP from the dropped packet and notifies the source by transmitting a Source Quench message. Then, the transmission rate is decreased by the source.

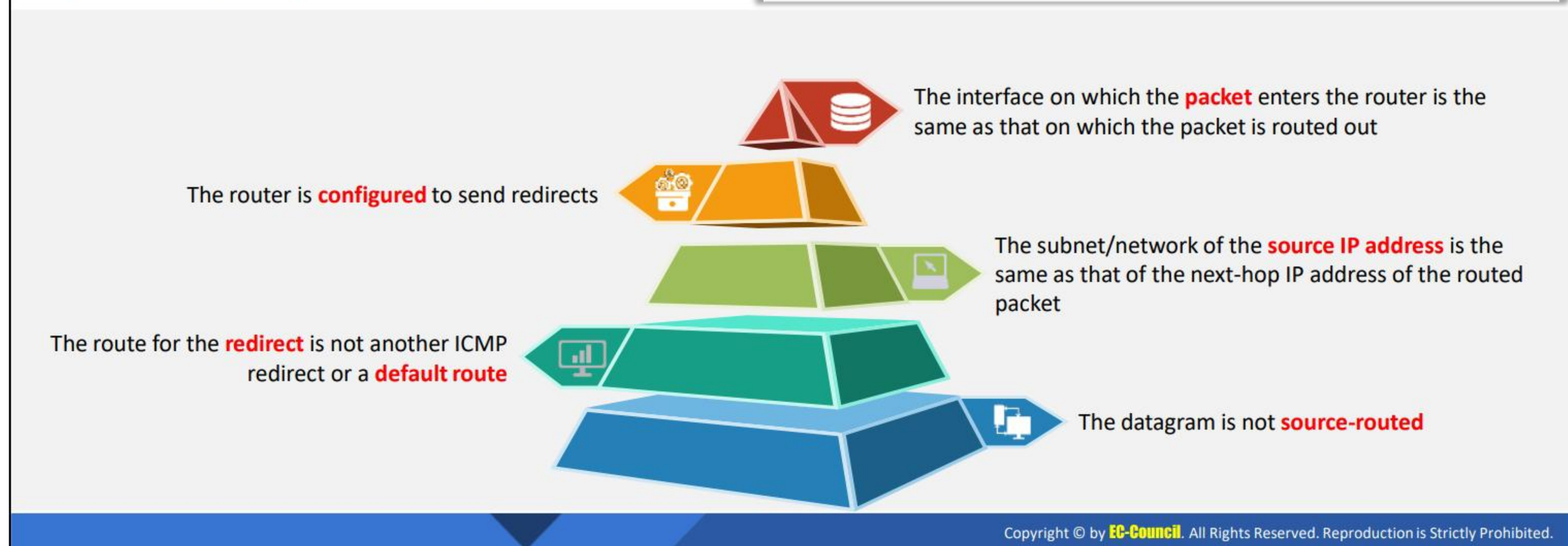
- **Existence of a better gateway to a remote network:** The ICMP Redirect message notifies a remote host to transmit packets through a different route. Redirects need to be sent by gateways only.

Basic Network Issues: ICMP Redirects

ICMP Redirects
 Type = 5, Code = 0 to 3

The default gateway sends the ICMP **redirect/change** request messages only if the following **conditions** are met:

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters		
Data.....		



Basic Network Issues: ICMP Redirects

Routers are used for transmitting IP packets from one network to the other; they contain a routing table that provides the route to a network. Therefore, whenever an IP packet reaches a host that it does not belong to, the host sends the packet to a router near it. The router takes the responsibility of sending the packet to the correct host (destination).

ICMP redirect messages are used by routers to inform the host that another router is available in the same network that can deal with the packet more easily. Let us consider an example to understand this message more clearly.

Consider three hosts in a network A, B, and C and three local routers X, Y, and Z. If host A wishes to send an IP packet to host C, it sends the packet to X. Based on the routing table information, X comes to know that Z is the nearest router to host C and an easy way to route. X informs host A about this to avoid further communication through this route to host C and sends an ICMP redirect message to host A. Along with this message, X sends the IP packet to Z so that the packet is forwarded to its destination.

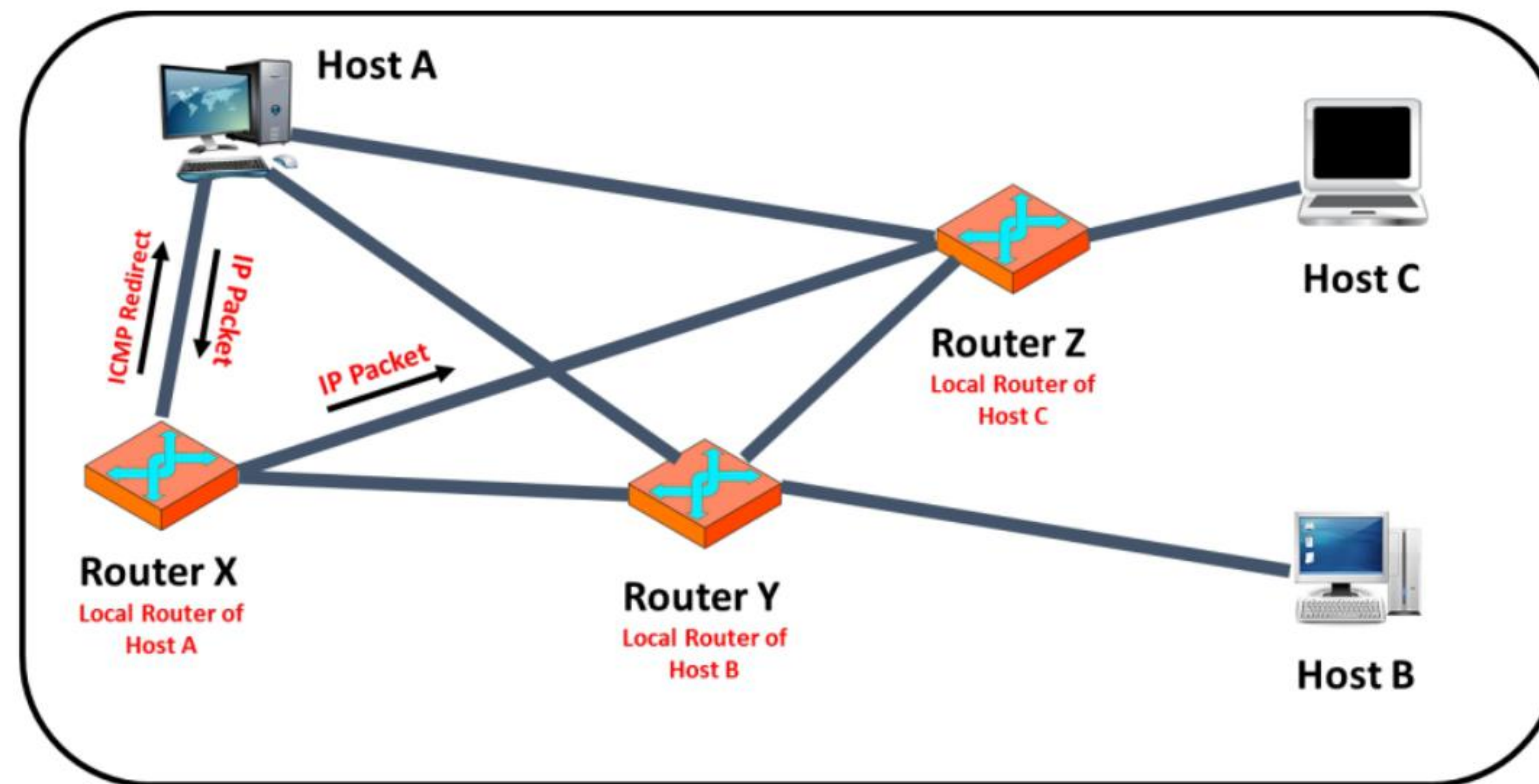


Figure 16.15: Schematic of an ICMP redirect message

ICMP Redirect Message Format

The following are the fields in an ICMP redirect message:

- **Type (1 Byte):** This field defines the ICMP message type; for ICMP redirect messages, it is set to 5.
- **Code (1 byte):** This field defines the reason behind this message.
- **Checksum (2 bytes):** This field defines a checksum for the ICMP header.
- **Internet address (4 bytes):** This field defines the address of the router that is local to the destination for further communication.
- **Original datagram portion (variable):** This field contains the IP header and first 8 bits of the IP packet that reached the wrong router.

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters		
Data.....		

Figure 16.16: Format of an ICMP Redirect Message

ICMP Redirects of Type 5 and Code 0–3

The default gateway sends ICMP redirect/change request messages only if the following conditions are met:

- The router is configured to send redirects.
- The route for the redirect is not another ICMP redirect or a default route.
- The interface on which the packet enters the router is the same as that on which the packet is routed out.
- The subnet/network of the source IP address is the same as that of the next-hop IP address of the routed packet.
- The datagram is not source-routed.



Troubleshooting Network Issues: IP Problems

Steps for troubleshooting IP related issues

- 1 Using tools, locate the devices that raised the issue in the **path of communication**
- 2 Check the physical connections between the source and the destination
- 3 **LAN connectivity** faults can raise network connectivity issues
- 4 At each **intermediate hop**, check whether the router is working
- 5 Ensure the proper configuration settings of the devices

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: IP Problems

Many different types of IP problems can be encountered. The problem may be a misconfiguration or malfunction in the client or target machine, or there may be an issue with the DNS or a network issue between the client and source machine. Additionally, there may be an issue with physical connections. IP problems can be troubleshooted using various tools such as the ping utility, traceroute, ipconfig, and netstat. These tools help in identifying the area in the network that has the problem. Once the problem area is isolated, more advanced tools can be used to resolve the problem.

Steps for troubleshooting IP related issues:






- Using tools, locate the devices that raised the issue in the path of communication
- Check the physical connections between the source and the destination
- LAN connectivity faults can raise network connectivity issues
- At each intermediate hop, check whether the router is working
- Ensure the proper configuration settings of the devices

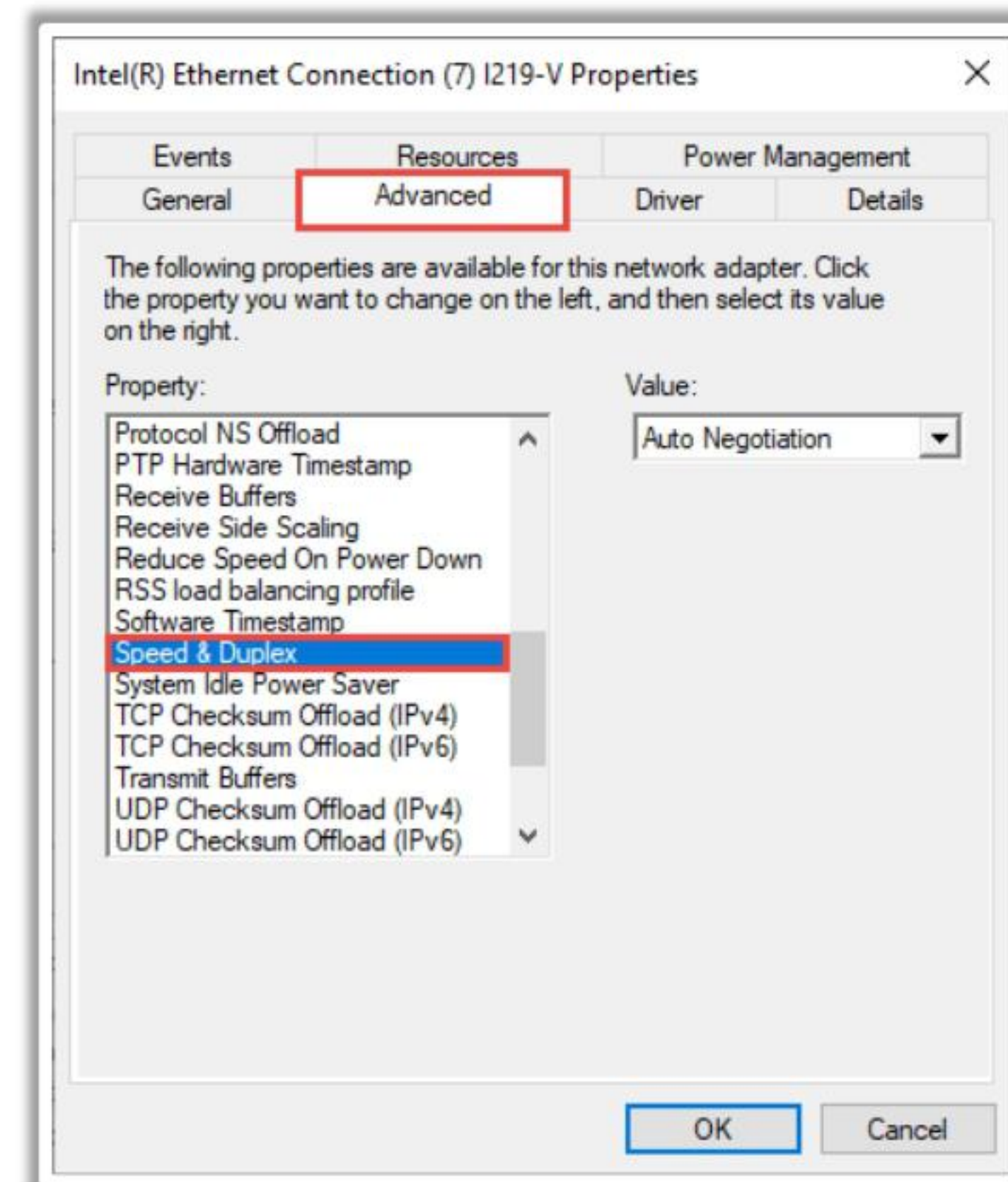
Troubleshooting Network Issues: Network Cable is Unplugged



The network issue called “network cable is unplugged” indicates that the computer is not detecting an **Ethernet connection** correctly

The following are potential solutions to this issue:

-  Restart the computer
-  Check the Ethernet network cable
-  Change the Ethernet Adapter’s Duplex Settings
-  Disable the Ethernet Network Adapter
-  Update the Network Adapter Driver



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: Network Cable is Unplugged

The network issue called “network cable is unplugged” indicates that the computer is not detecting an Ethernet connection correctly. This error can be viewed on the desktop or notification screen. This error is also visible in the status of the Ethernet network connection. However, this error is not specific to Ethernet; the error can also be observed if a Wi-Fi connection is being used and would result in failure to connect to the Internet. Once a user encounters this error on their system, it is likely to occur every few hours/days.

The reason behind this error varies according to the computer configuration. Some computers face this issue owing to outdated or faulty drivers, whereas other computers show this error because of a conflict between the computer/application and the connection speed.

The following are potential solutions to this issue.

Restart the Computer

Restart the computer by powering down, wait for a few seconds, and then turn on the computer. Laptop users must take the following extra step: remove the battery and power supply cable and, if possible, wait for 10 min. After 10 minutes, reattach the battery, power supply cable, and start Windows again.

Check the Ethernet Network Cable

Check to ensure that both ends of the Ethernet cable are tight. Connect one end of the cable to the computer and the other end to the main network device, e.g., a router. If this did not help, then check whether there is any fault in the cable. Instead of purchasing a new cable, use the cable with different computers or temporarily swap the Ethernet cable with a good Ethernet cable.

Change the Ethernet Adapter's Duplex Settings

The duplex settings in Windows manage the direction of network communication. For the best performance, the duplex values are automatically set by Windows. Most users have reported that duplex settings caused the “network cable is unplugged” error and that their concern was resolved by changing the duplex settings. To change the duplex settings, follow the steps below.

- **Step 1:** Right-click on the **Start Menu button** and select the **Device Manager**.

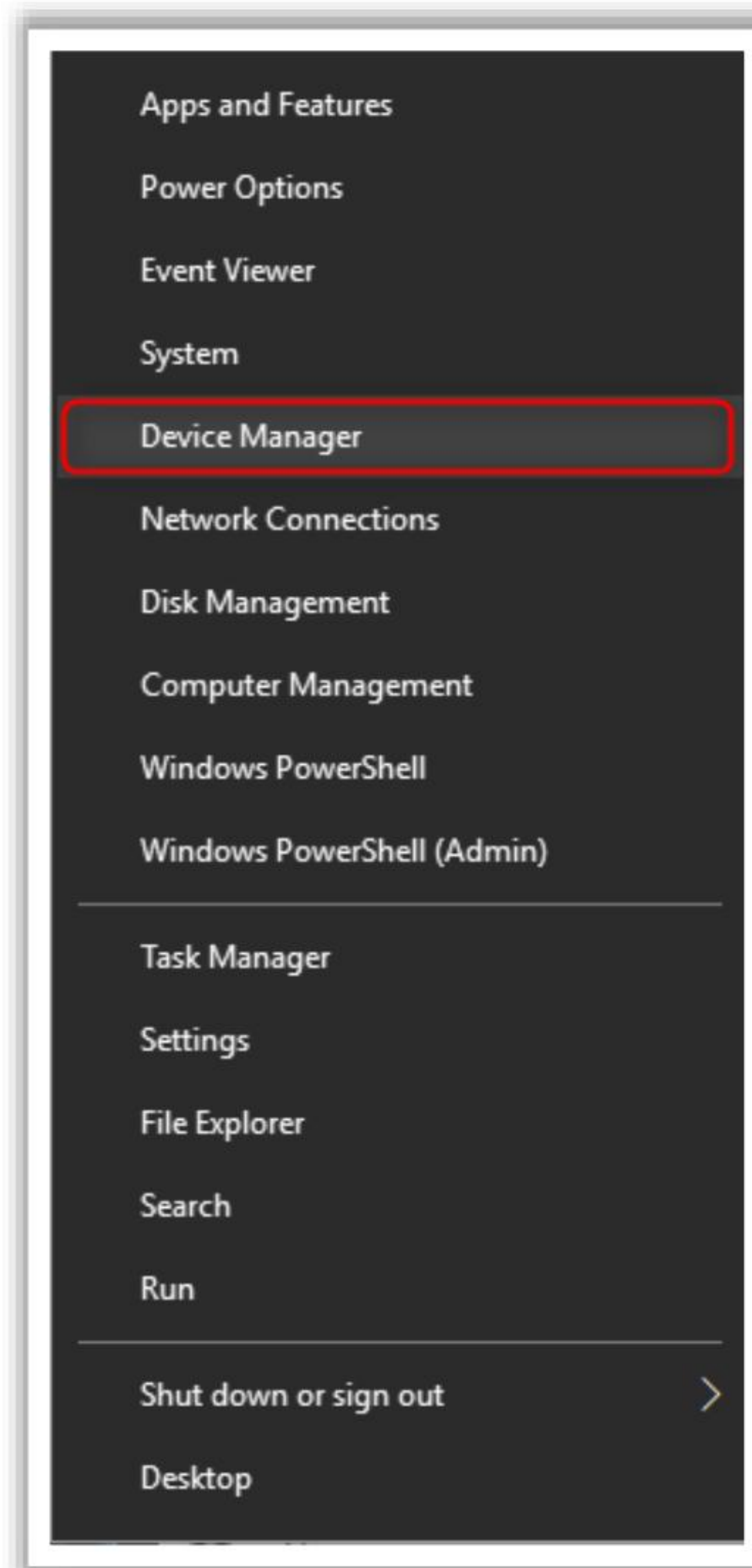


Figure 16.17: Selecting the “Device Manager”

- **Step 2:** In the Device Manager, navigate to **Network adapters** and click on it to expand the list of network adapters. Browse to the Ethernet Adapter in use (Intel(R) Ethernet Connection (7) I219-V). Right-click on the Ethernet Adapter and select **Properties**.

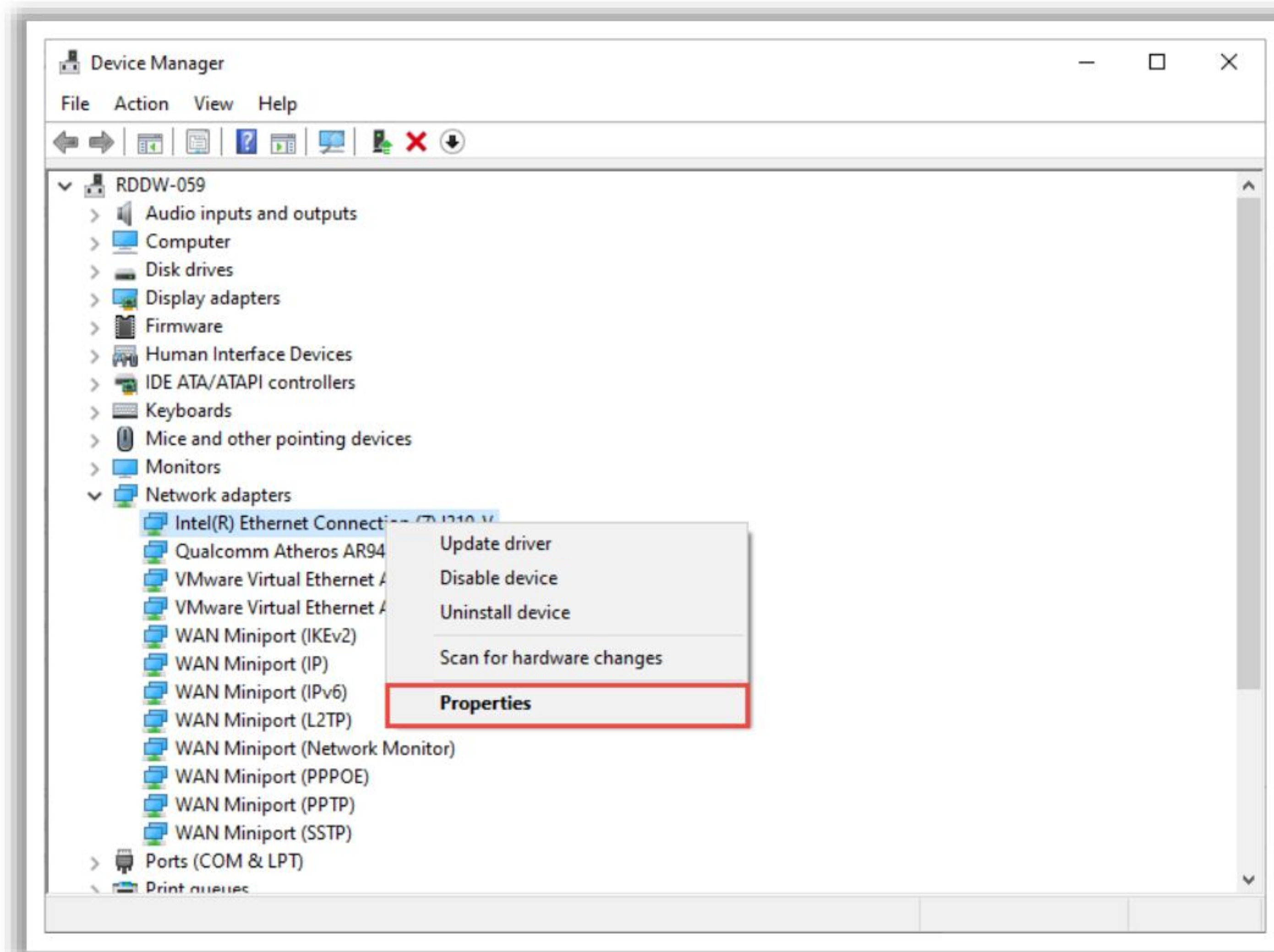


Figure 16.18 Selecting “Properties” for the Ethernet adapter

- **Step 3:** In the new window that opens, click on the **Advanced** button, and under **Properties**, navigate to and select **Speed & Duplex**.

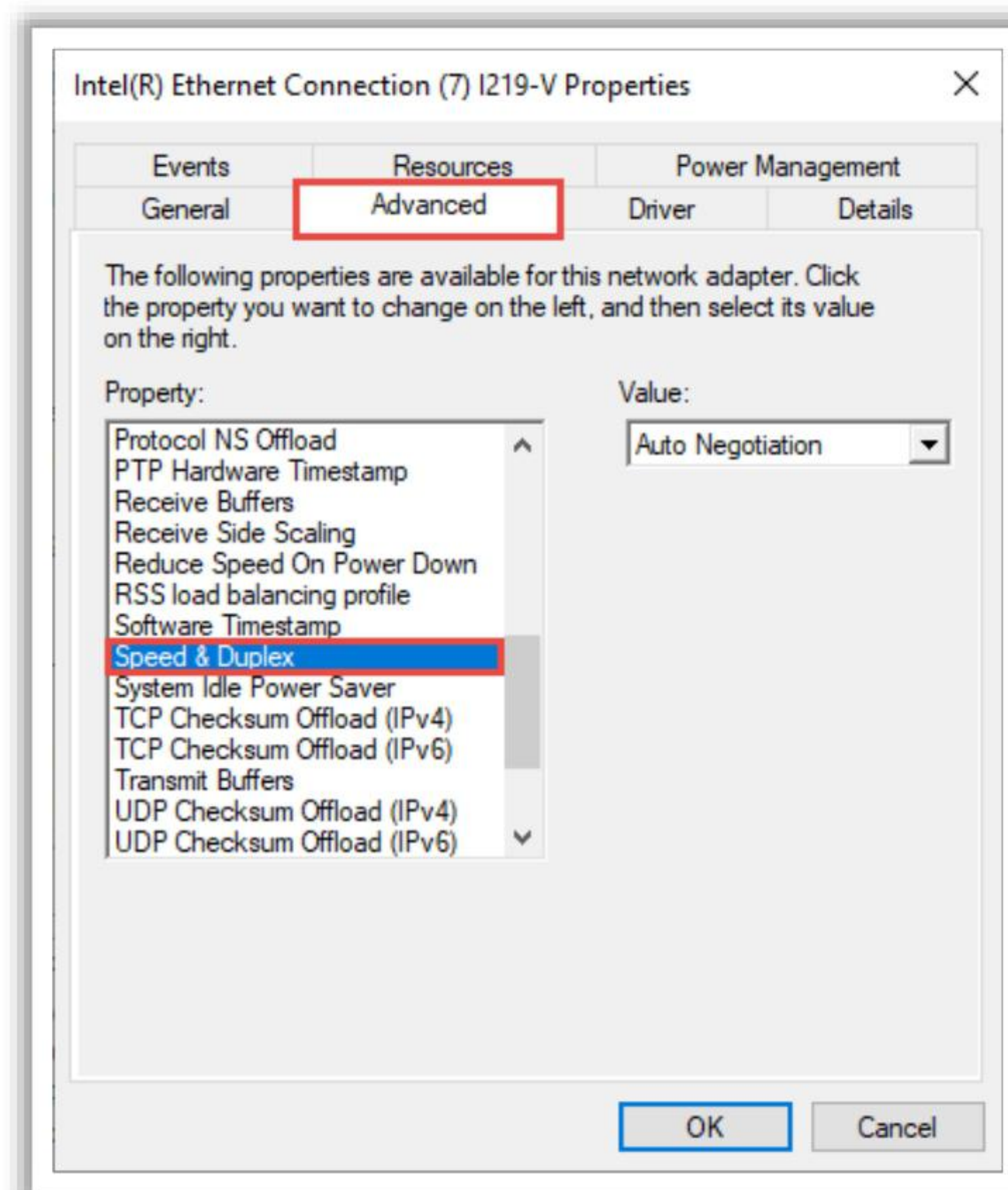


Figure 16.19: Selecting the “Speed & Duplex” property

- **Step 4:** Under the **Value** heading, change the value to **100 Mbps Half Duplex** and then click **OK**.

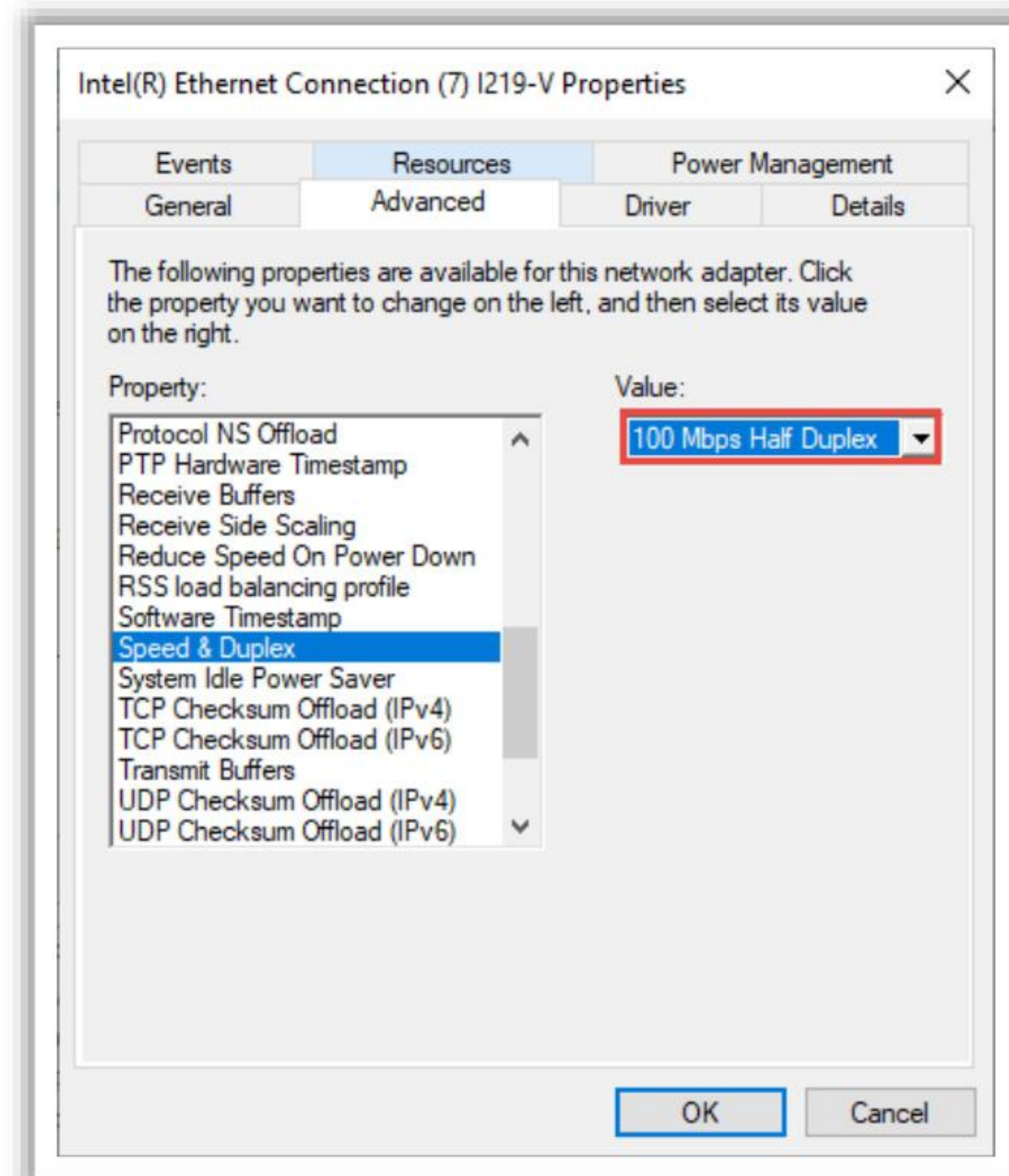


Figure 16.20: Changing the "Value"

- **Step 5:** Restart the computer.

Note: If this did not fix the problem, then change the value in Step 4 and check if the problem is resolved.

Disable the Ethernet Network Adapter

If the user is not using the network adapter, then it is advisable to disable it. This also applies to computers having built-in Ethernet adapters that run on Wi-Fi networks. To disable the Ethernet network adapter, perform the following steps:

- Open **Control Panel** and click on **Network and Internet**.

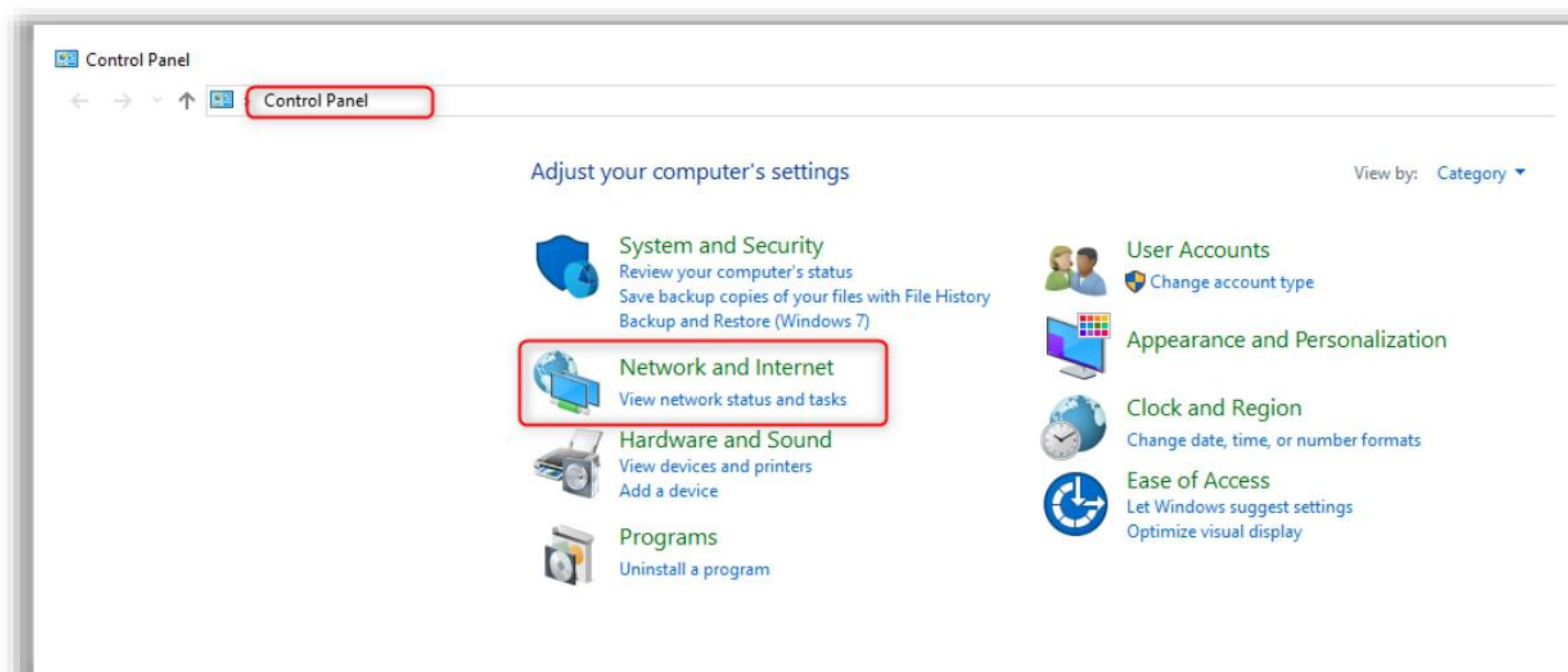


Figure 16.21: Control Panel

- Click on **Network and Sharing Center**.

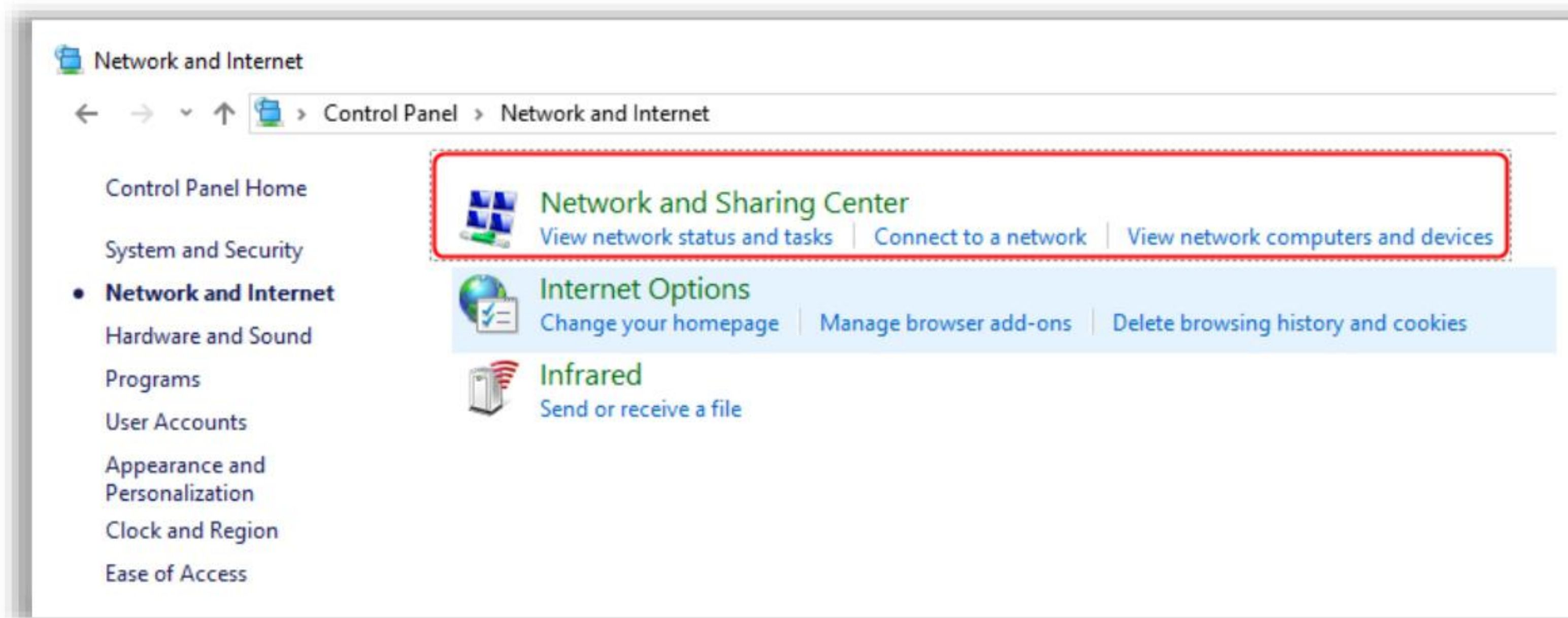


Figure 16.22: Click “Network and Sharing Center”

- Click on **Change adapter settings**.

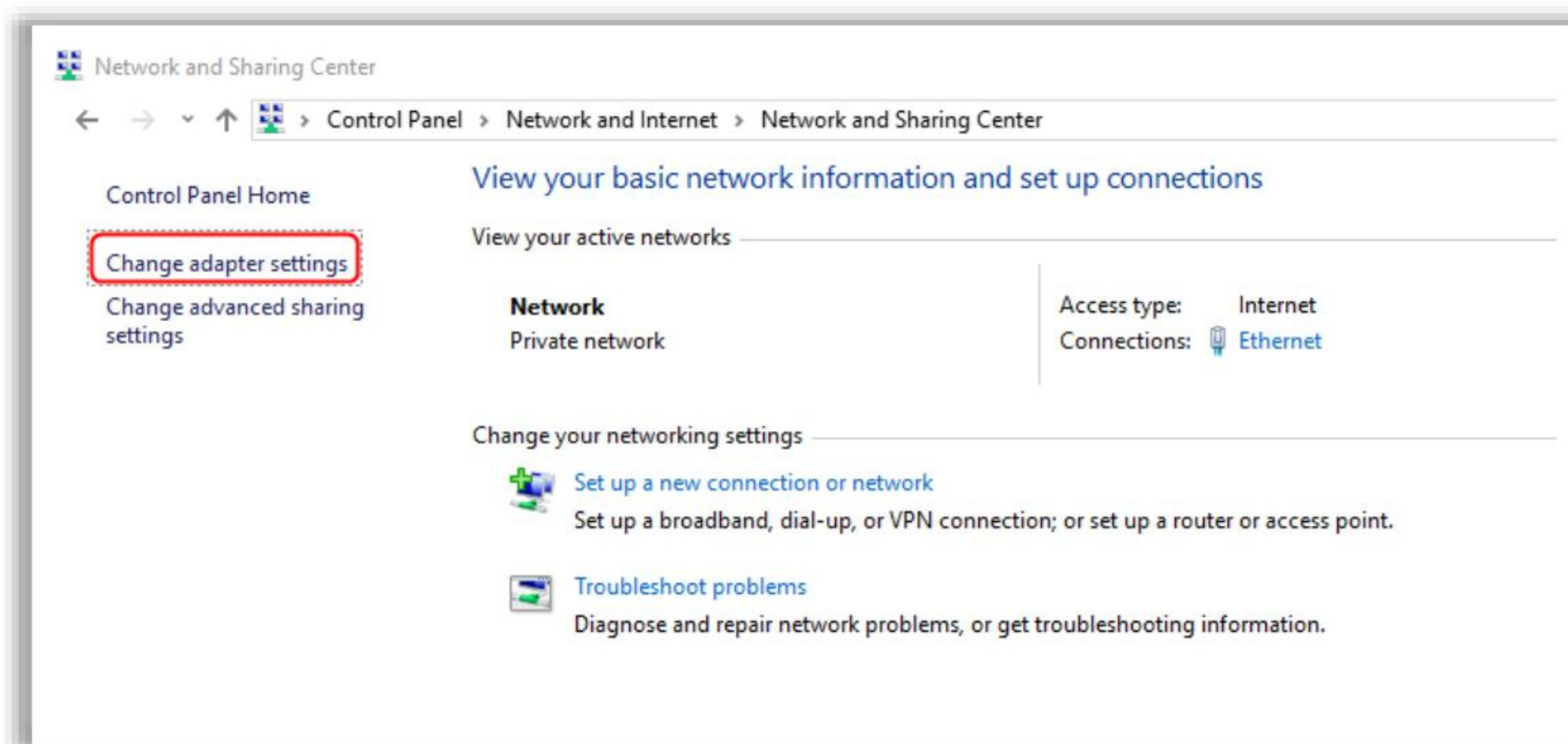


Figure 16.23: Select “Change adapter settings” Option

- A **Network Connections** window will open with the available connections. Depending on the connection the user wants to disable, right-click on the icon of the adapter and disable it.

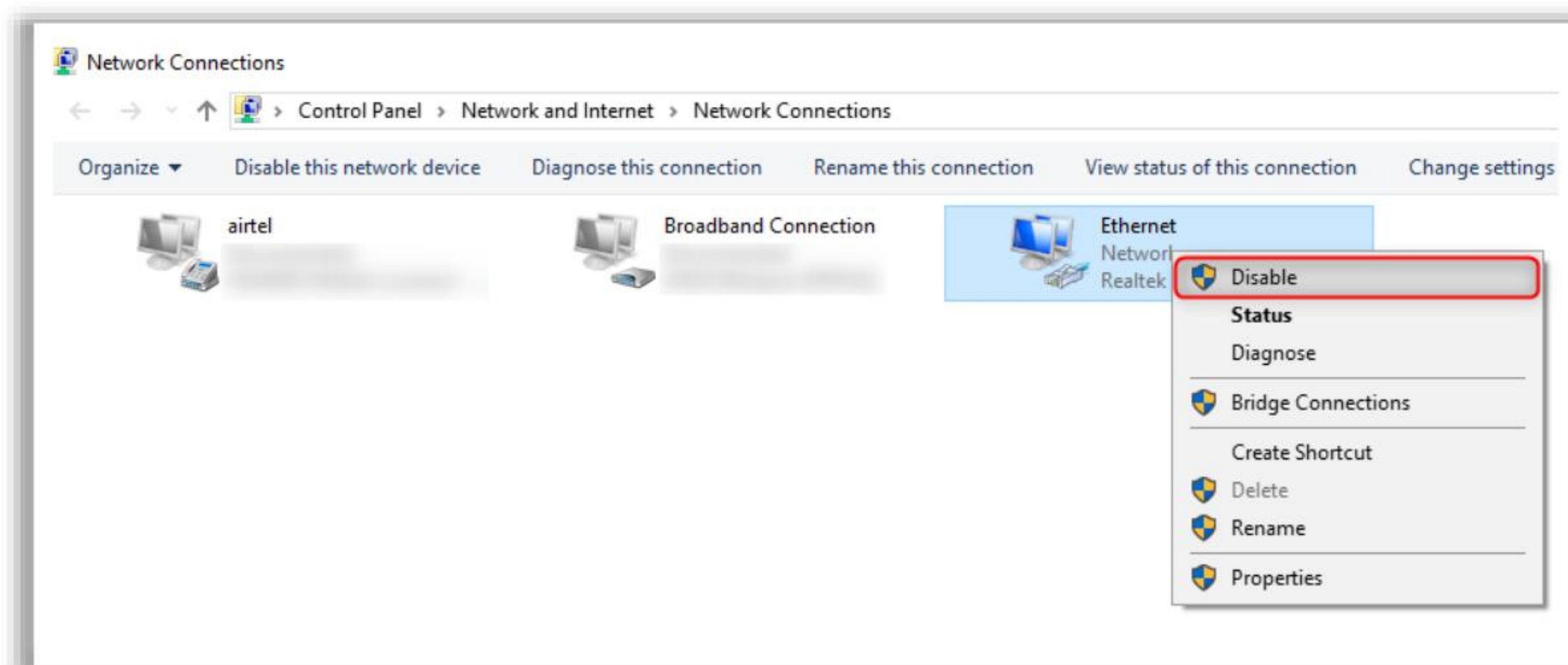


Figure 16.24: Disable an Ethernet Connection

Update the Network Adapter Driver

If the “network cable is unplugged” error occurred because of an outdated driver, then updating the network adapter driver will resolve the problem. To update the network adapter driver, follow the steps below.

- Right-click on the **Start Menu button** and select **Device Manager**.

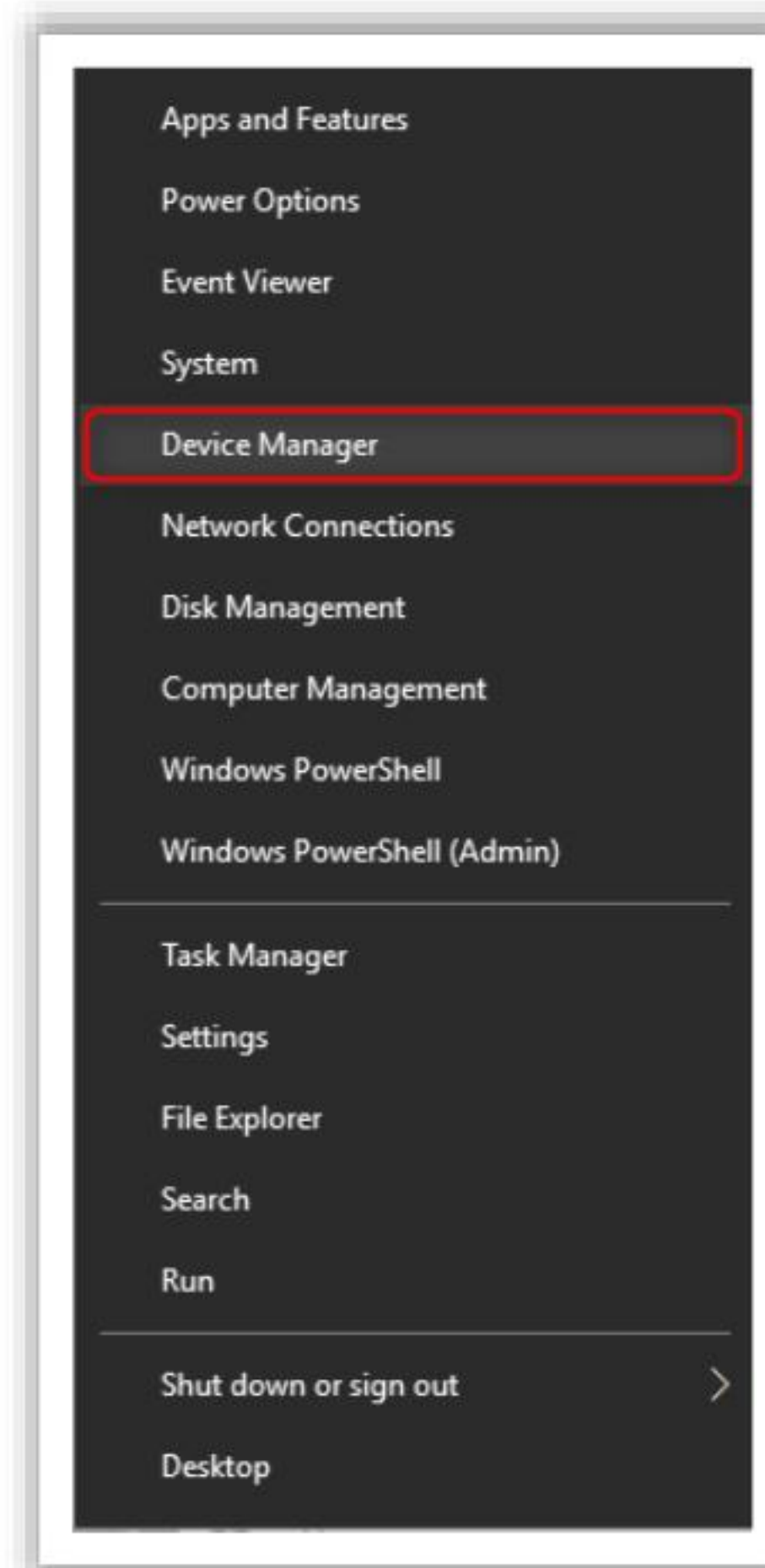


Figure 16.25: Go to “Device Manager”

- In the Device Manager, navigate to and click on **Network adapters** to expand the list of network adapters. Browse to the Ethernet adapter in use (Intel(R) Ethernet Connection (7) 1219-V). Right-click on the Ethernet adapter and select **Update driver**.

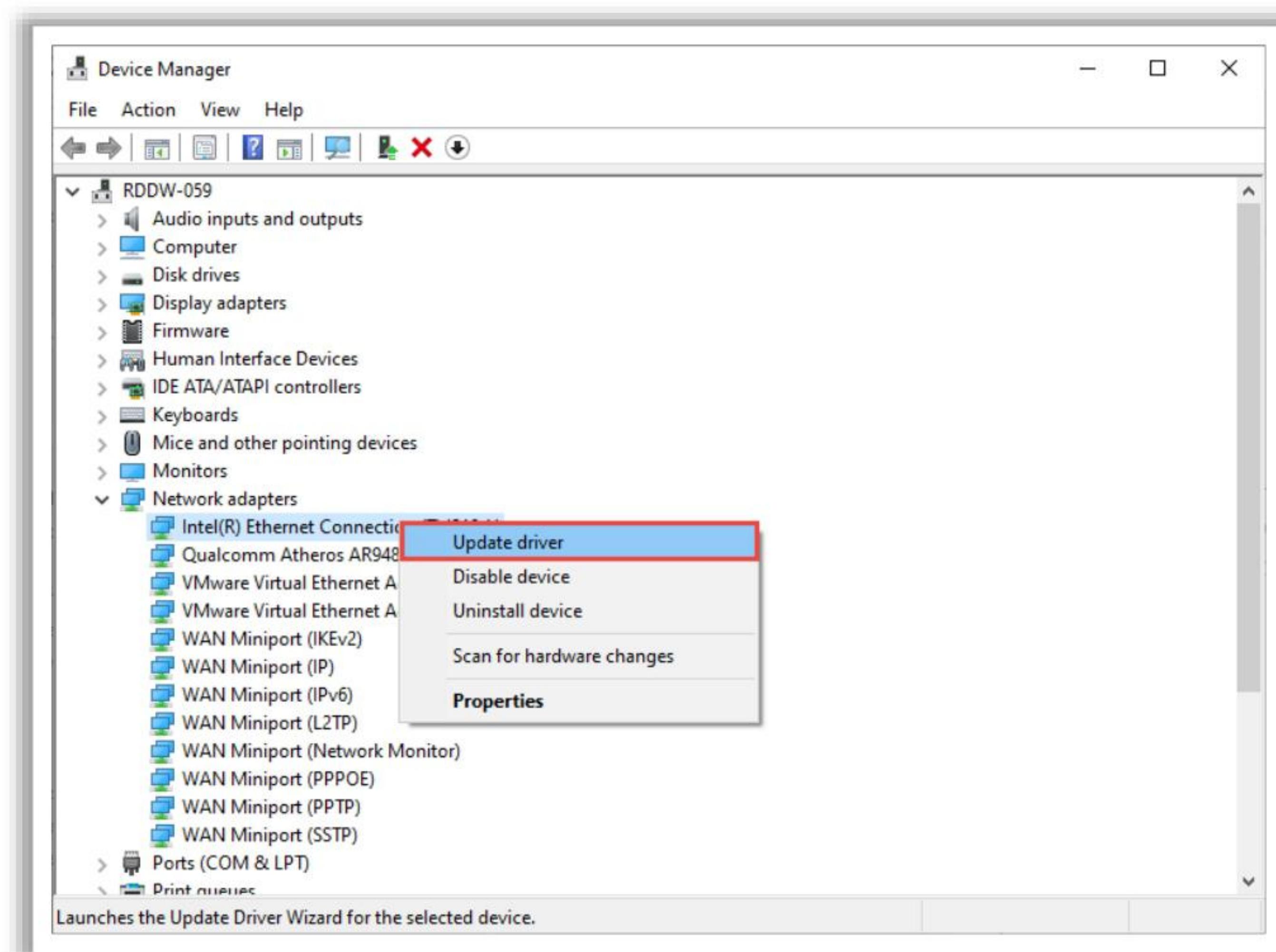


Figure 16.26: Selecting the “Update driver” Option for “Network adapters”

- Select **Search automatically for drivers**. Windows will update to the latest driver for the Ethernet adapter. After the completion of this process, restart the computer.

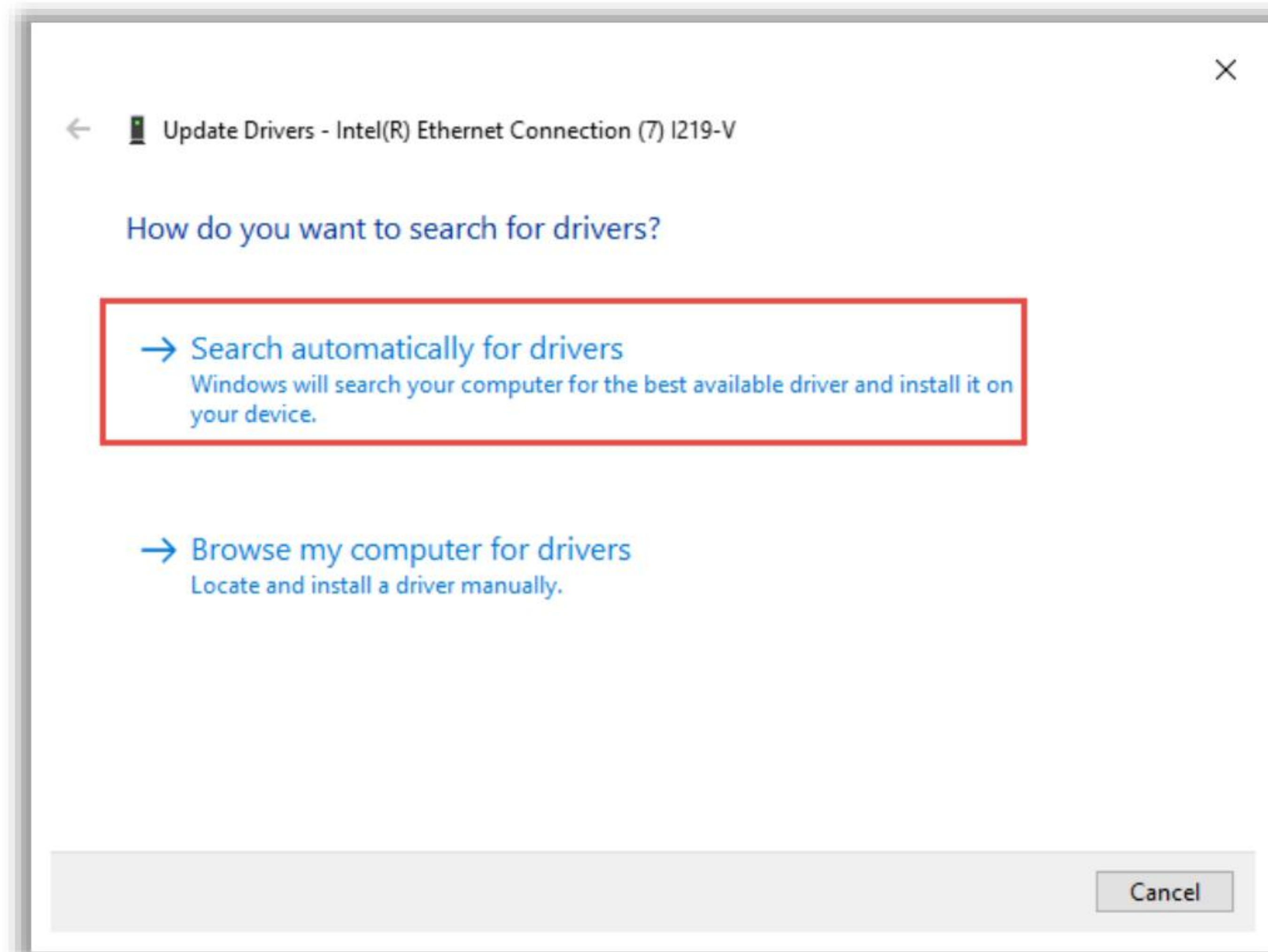


Figure 16.27 Selecting the option “Search automatically for drivers”

Troubleshooting Network Issues: Network Path Cannot Be Found

- ❑ The issue of “network path was not found” may arise for the following **reasons**:
 - The network connection does not exist
 - The network security settings **do not allow** sharing
 - The name of the computer with which the connection is to be established is too long
 - The connection of system is blocked by firewall or the antivirus software



Troubleshooting

- ✓ Verify the shared drive
- ✓ Ping the IP address of the target computer
- ✓ Modify the network security settings
- ✓ Reinstall the network adapter drive
- ✓ Enable NetBIOS over TCP/IP
- ✓ Rename the computer
- ✓ Disable the firewall

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: Network Path Cannot Be Found

Error code 0x80070035, “The network path was not found,” appears when hosting a system on a network to access files from another system without Ethernet.



Figure 16.28: Network error

A few different and unrelated technical problems generate this error. This issue often occurs for the following reasons:

- The network connection does not exist.
- Network security settings do not allow sharing.
- The name of the computer with which the connection is to be established is too long.
- The connection or system is blocked by firewall or antivirus software.

Methods to Troubleshoot “The Network Path Cannot Be Found” Error

- **Verify the shared drive:**
 - Right-click the drive that is to be shared and select **Properties**.

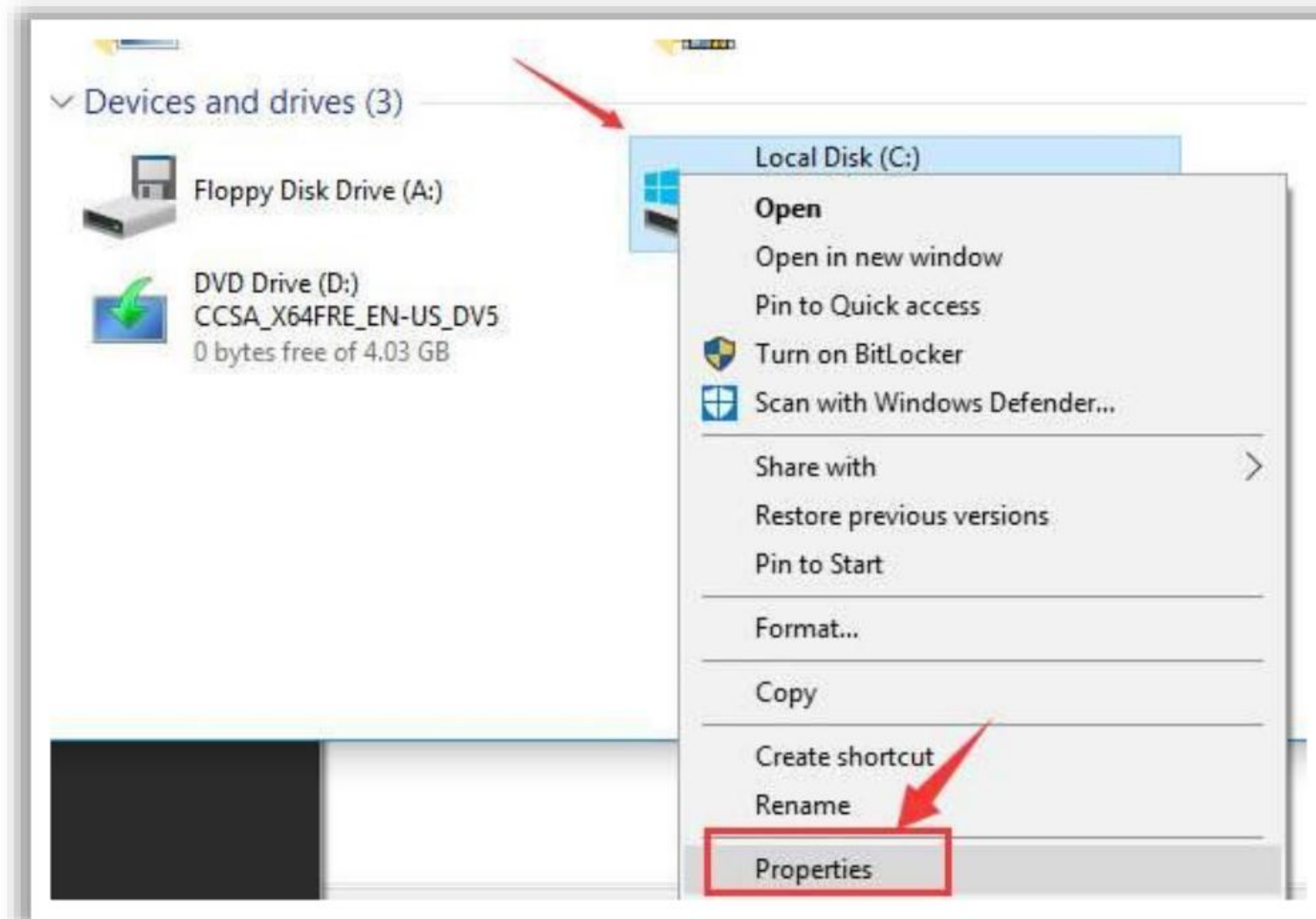


Figure 16.29: Select the Drive to be Shared

- Navigate to the **Sharing** tab. Click **Advanced Sharing...** if it says **Network Path: Not Shared** under the section **Network File and Folder Sharing**.

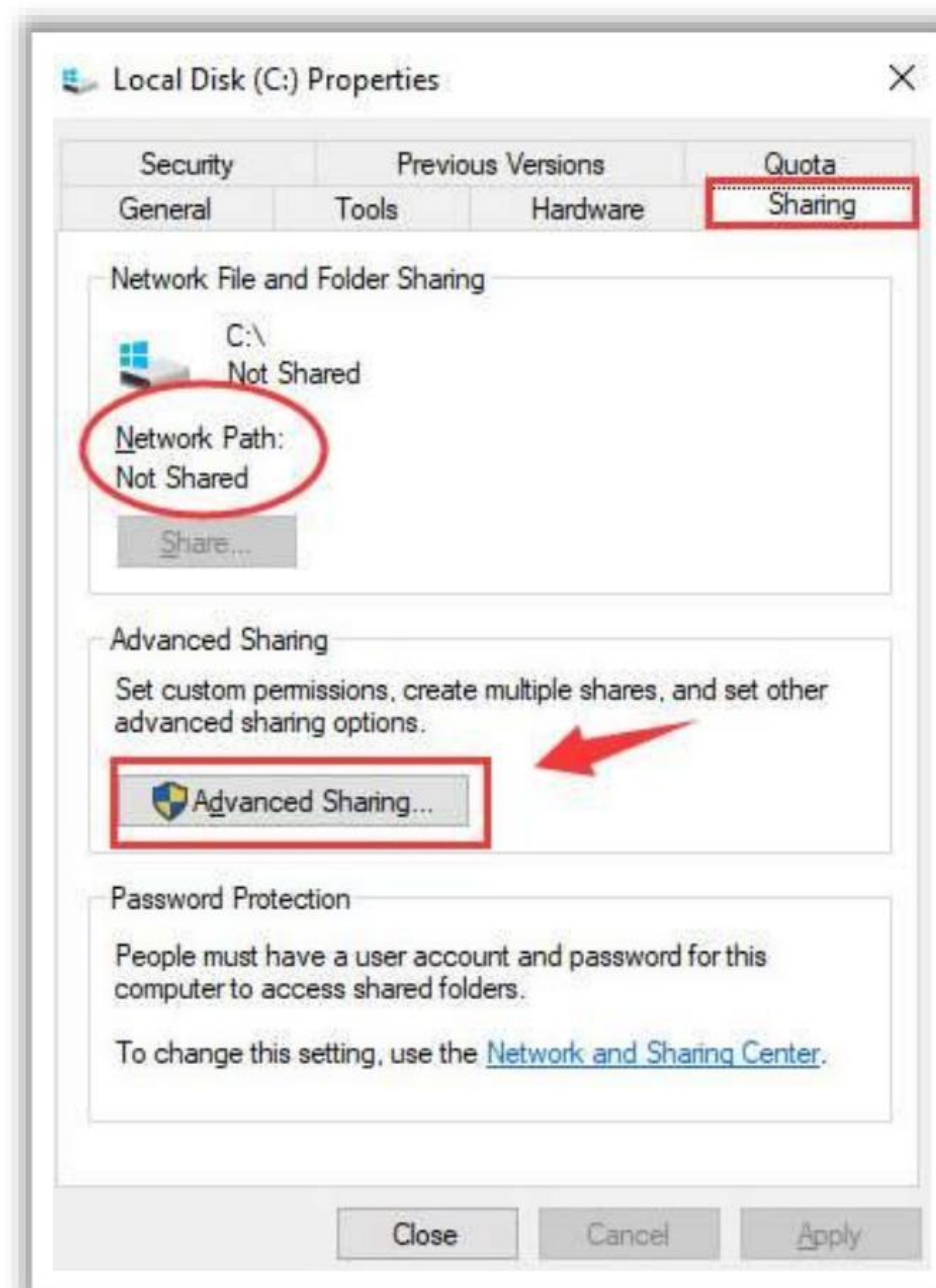


Figure 16.30: Share the Selected Drive

- Tick the box for **Share this folder**, ensure the **Share name** is correct, and click **Apply** and **OK**.
- To ensure the drive is shared, open the **Run** window in the system that needs to access the drive, type the name of the drive, and click **OK**.

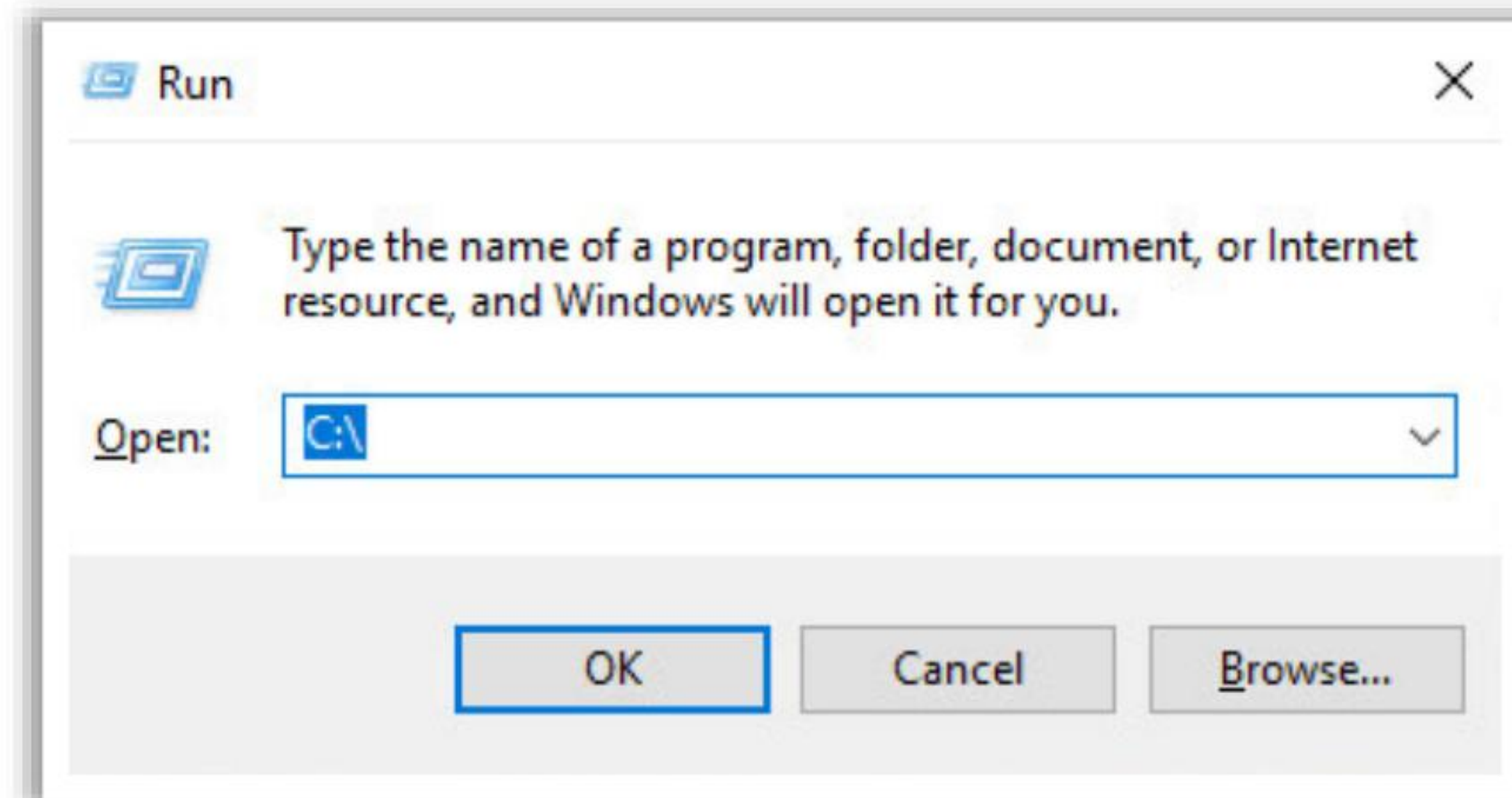


Figure 16.31: Ensure the Drive is Shared

▪ **Ping the IP address of the target system:**

- On the target system, type in the following command in the command prompt.

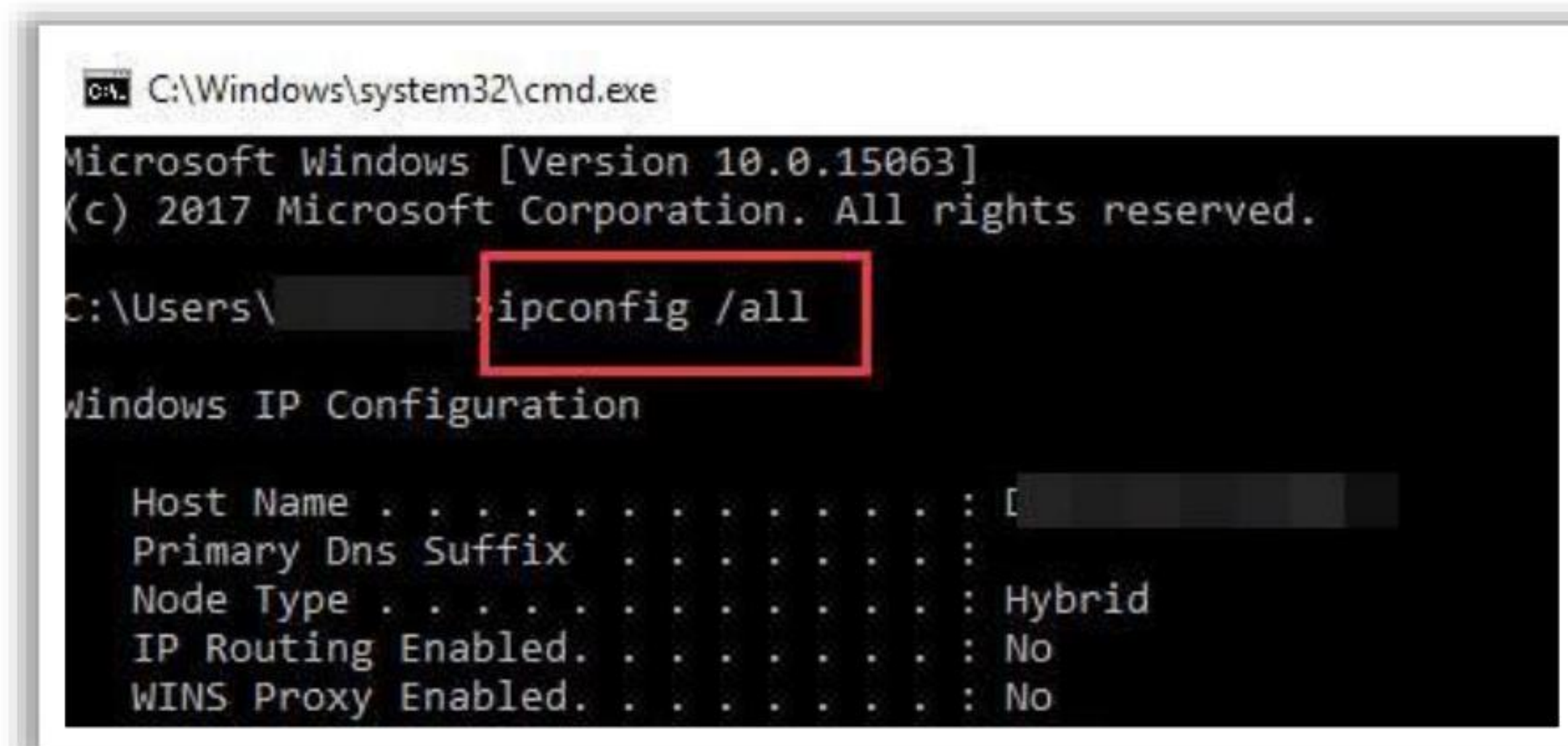


Figure 16.32: Execute "ipconfig" Command

- Locate the category **IPv4 Address** and note down the address (192.168.43.157) here.

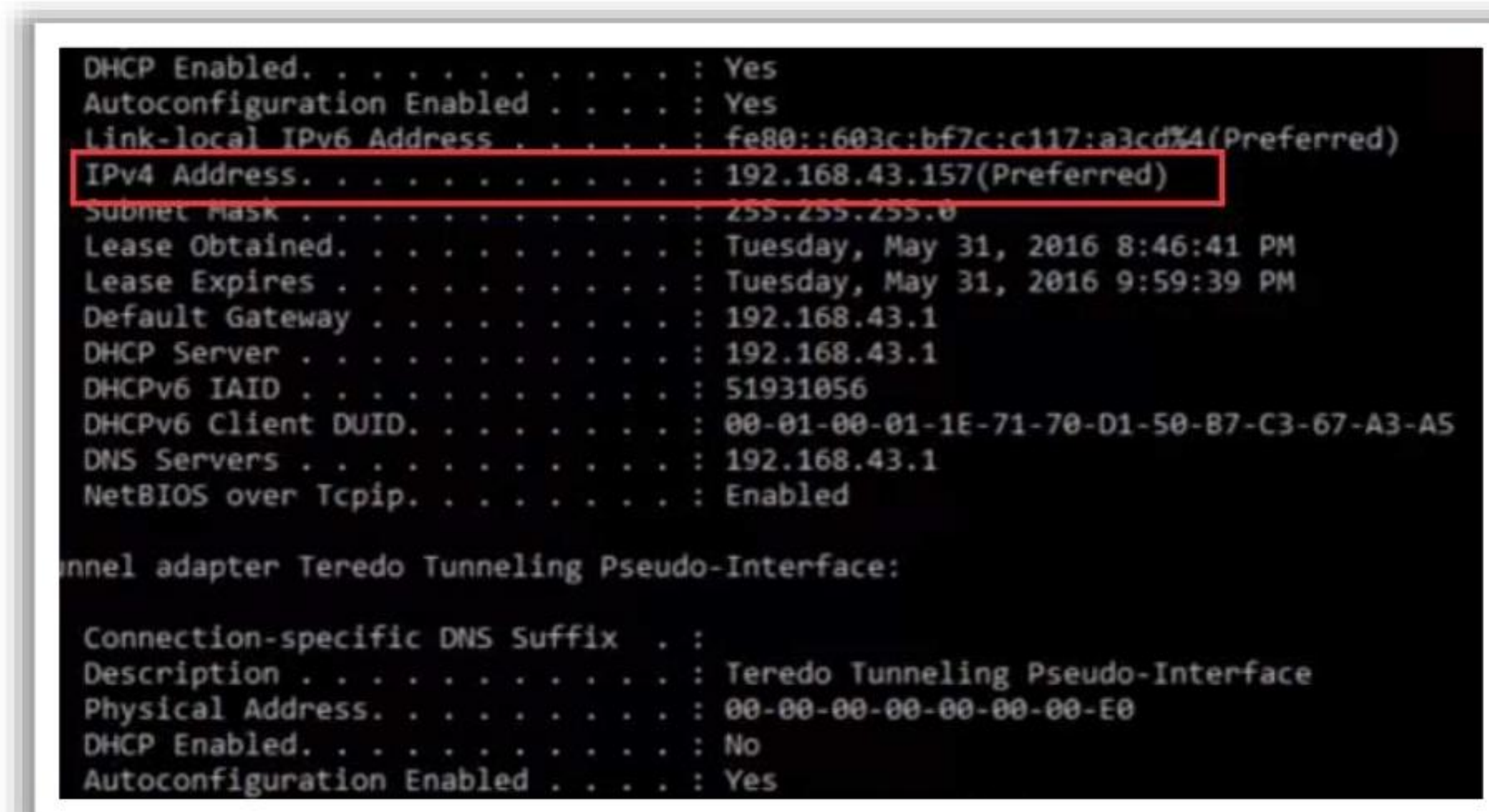


Figure 16.33: Locate IPv4

- Again, type in the **Run** window **\\IPv4 Address**(the drive that is to be accessed).

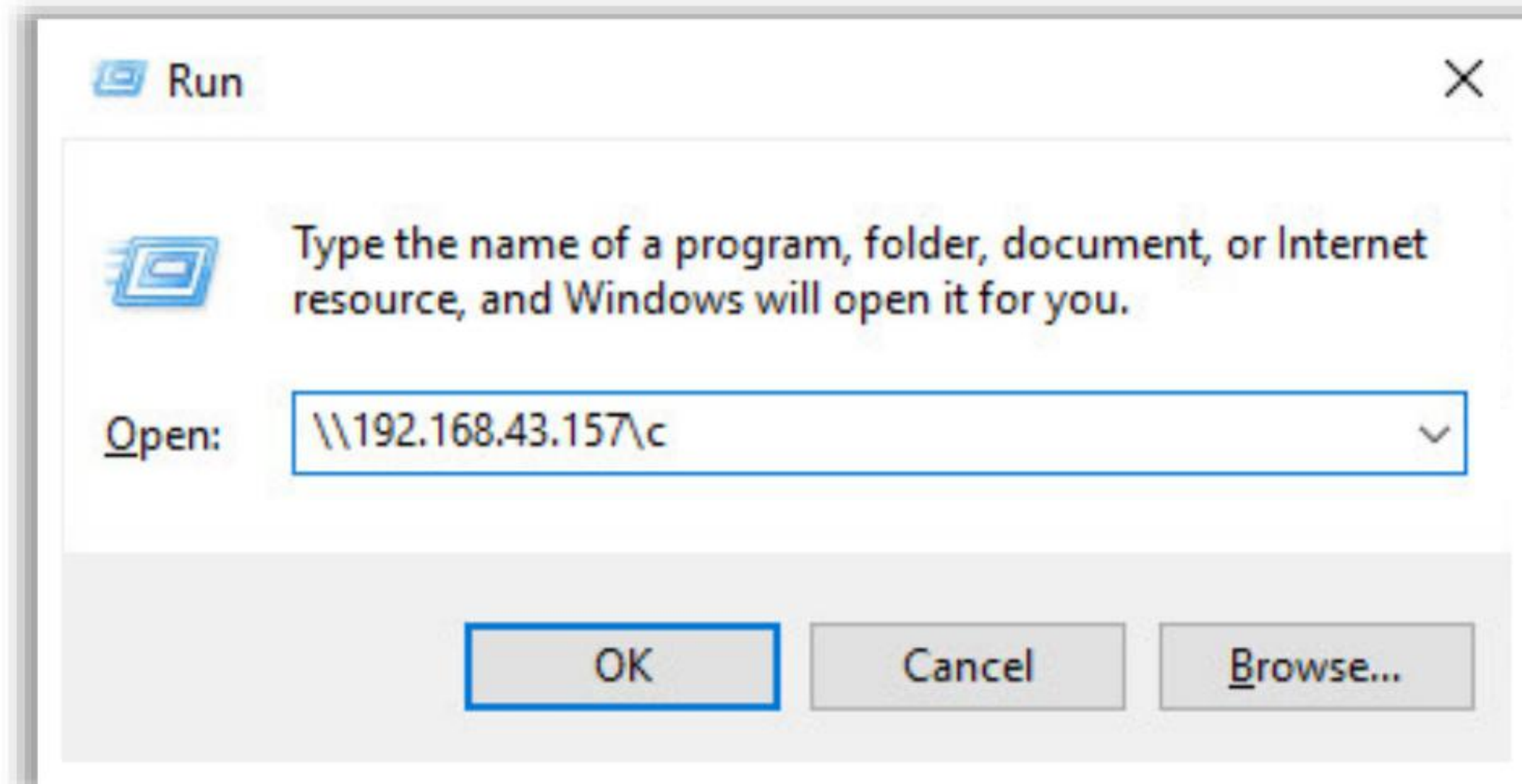


Figure 16.34: Ensure the Drive to be Accessed

- Ensure that the specified drive is the one that is shared (the C drive in this case).

- **Modify the network security settings:**

In case the above two methods did not work, check whether the problem is with the network connection.

- Double-click **Local Policies/Security Options/Network security: LAN Manager authentication level**.
- From the drop-down menu, select **Send LM & NTLM-use NTLMv2 session security if negotiated** and click **Apply** and **OK**.
- Try accessing the shared drive now.

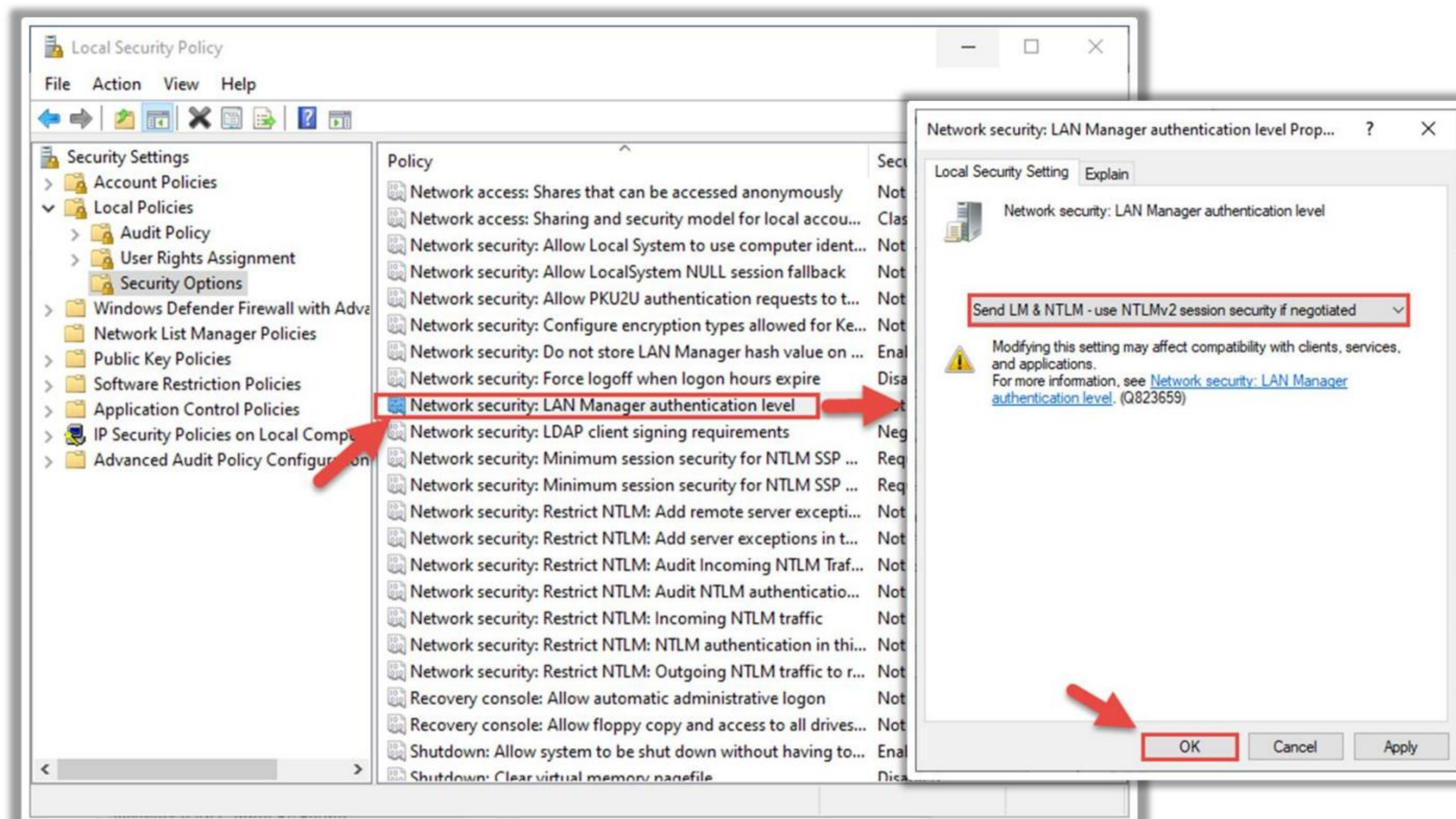


Figure 16.35: Modify Network Security Settings

- **Reinstall network adapters**

- Connect the computer to the Internet with a physical wire and run the command **devmgmt.msc**.
- Click the **View** tab in the **Device Manager** window and check **Show hidden devices**.

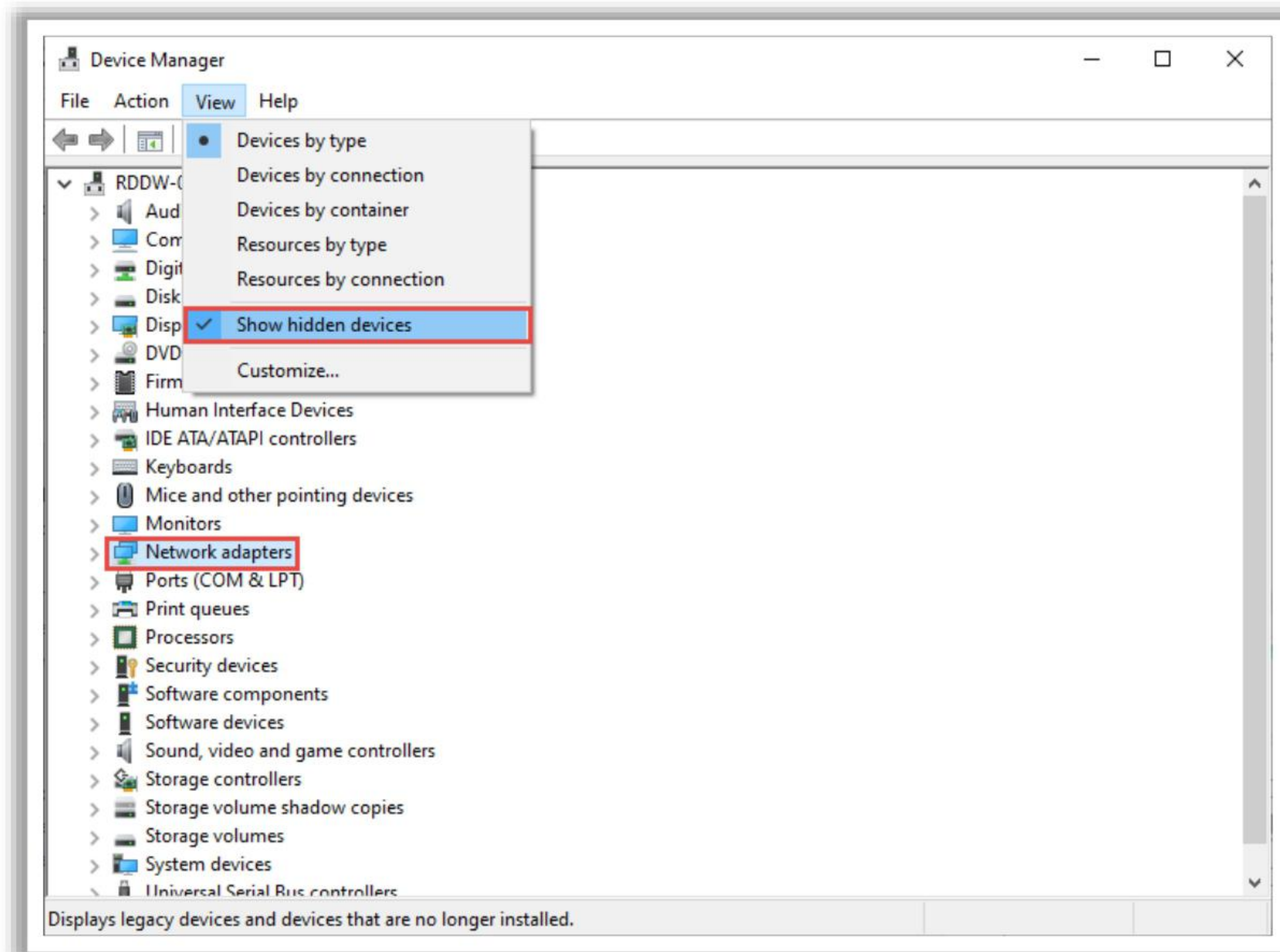


Figure 16.36: Check “Show hidden devices”

The hidden adapters are shown with light-colored translucent icons when the list of network adapters is expanded.

- Right-click all the drivers and uninstall them by selecting **Uninstall device**.

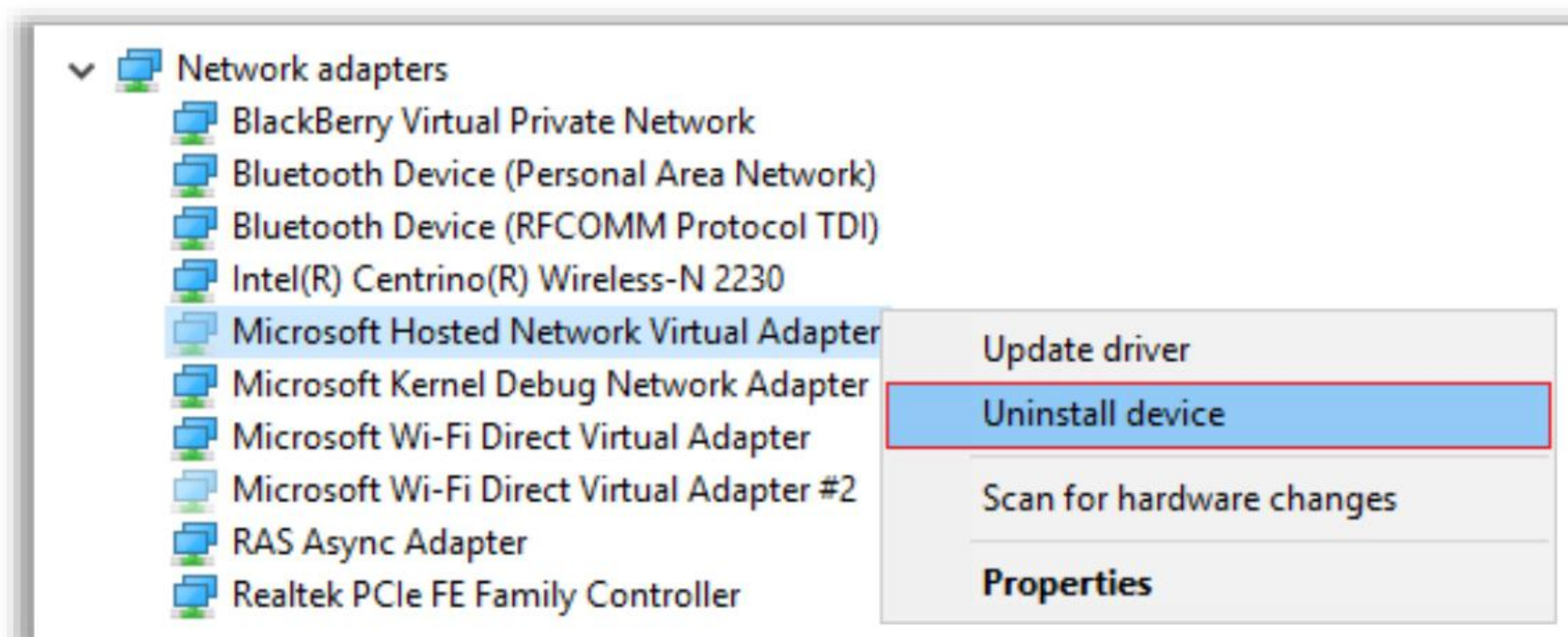


Figure 16.37: Uninstall Network Adapter

- Reboot the system and check if the issue is resolved.

- **Enable NetBIOS over TCP/IP**
 - Run the command **ncpa.cpl**.

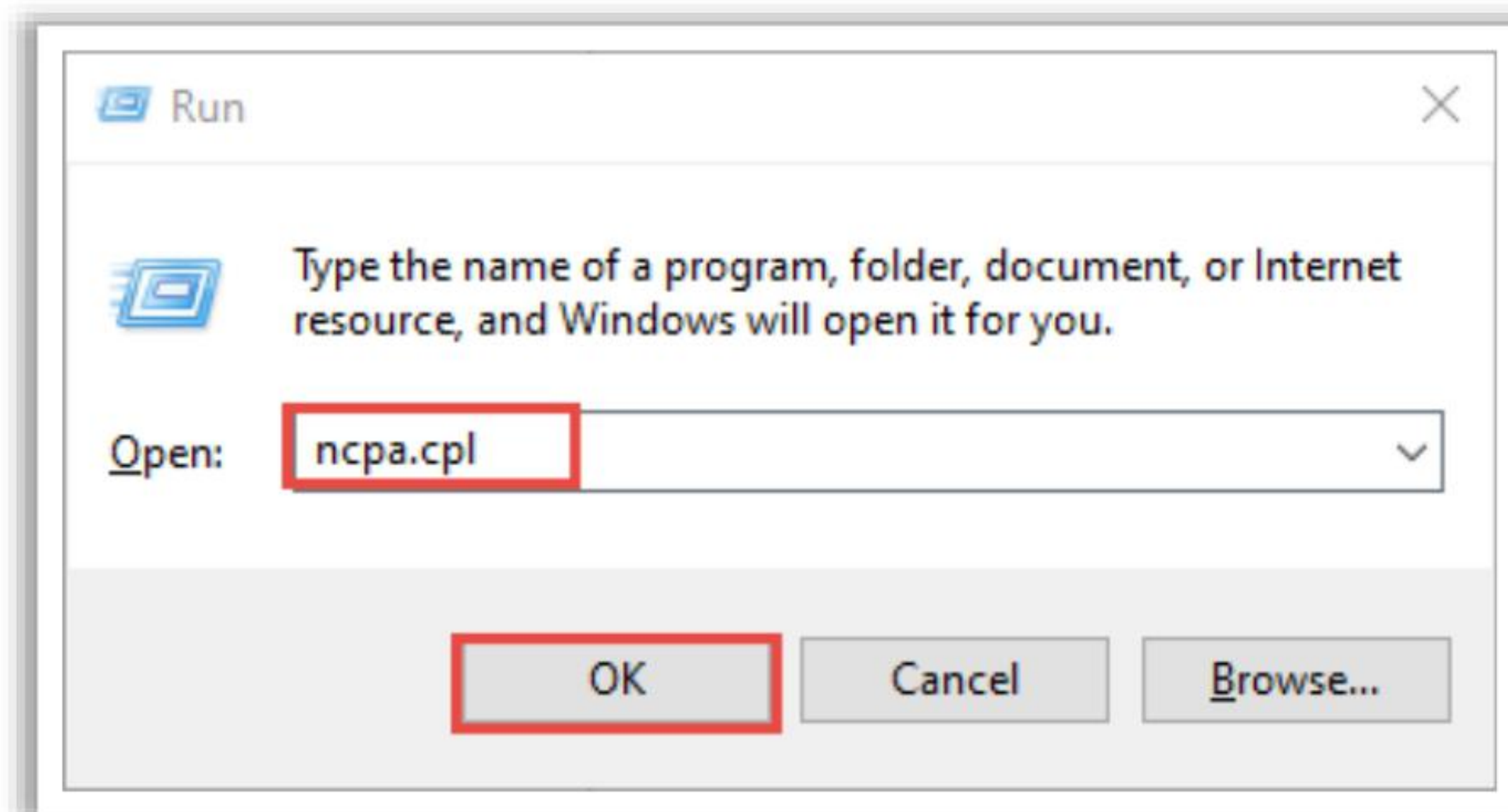


Figure 16.38: Run “ncpa.cpl”

- Right-click the active Wi-Fi or Ethernet connection and select **Properties**.
- Select **Internet Protocol Version 4 (TCP/IPv4)** and click **Properties**.

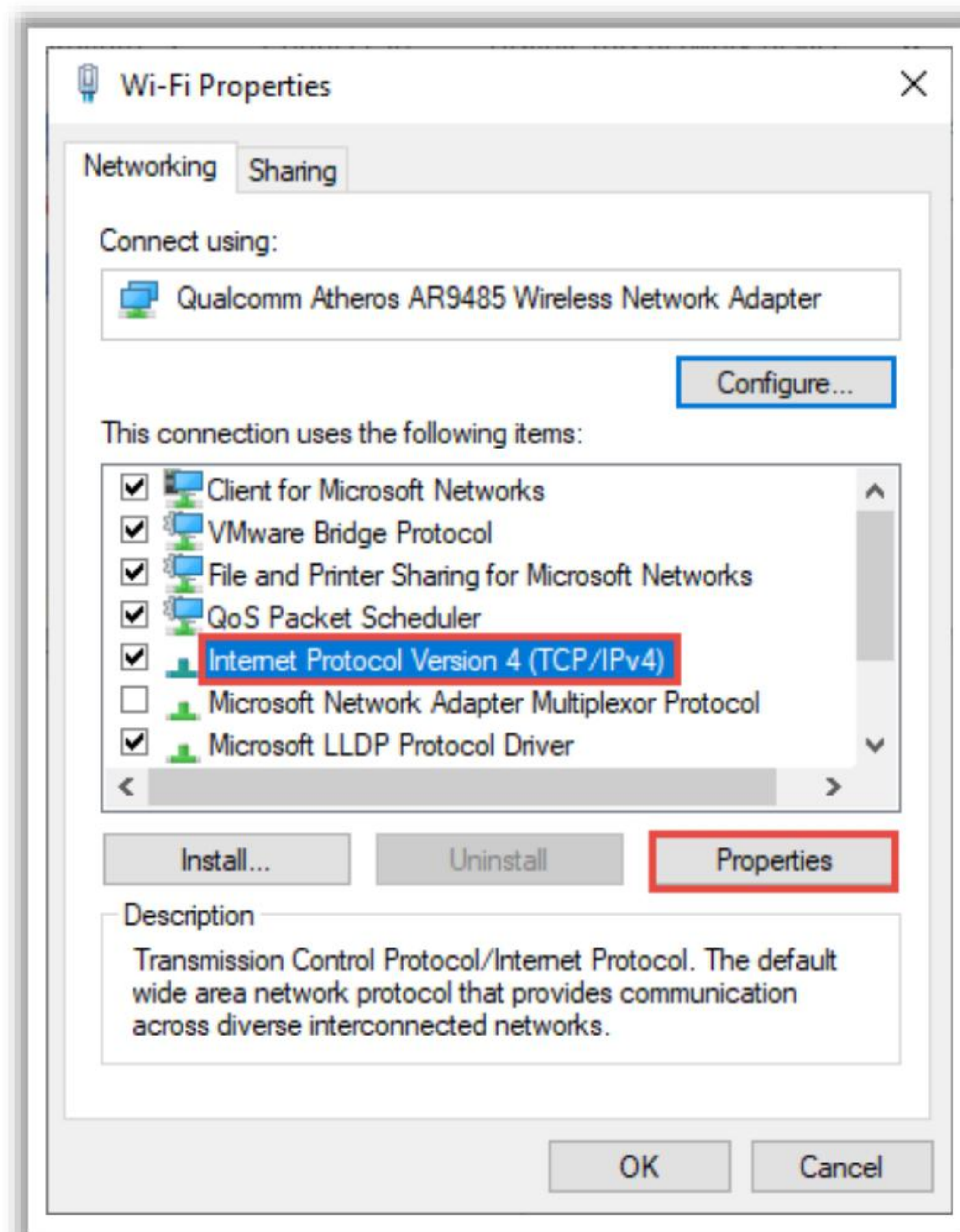


Figure 16.39: Select Properties of TCP/IPv4

- Click **Advanced** in the next window and switch to the **WINS** tab under **Advanced TCP/IP Settings**.

- Under **NetBIOS setting**, check **Enable NetBIOS over TCP/IP** and click **OK**.

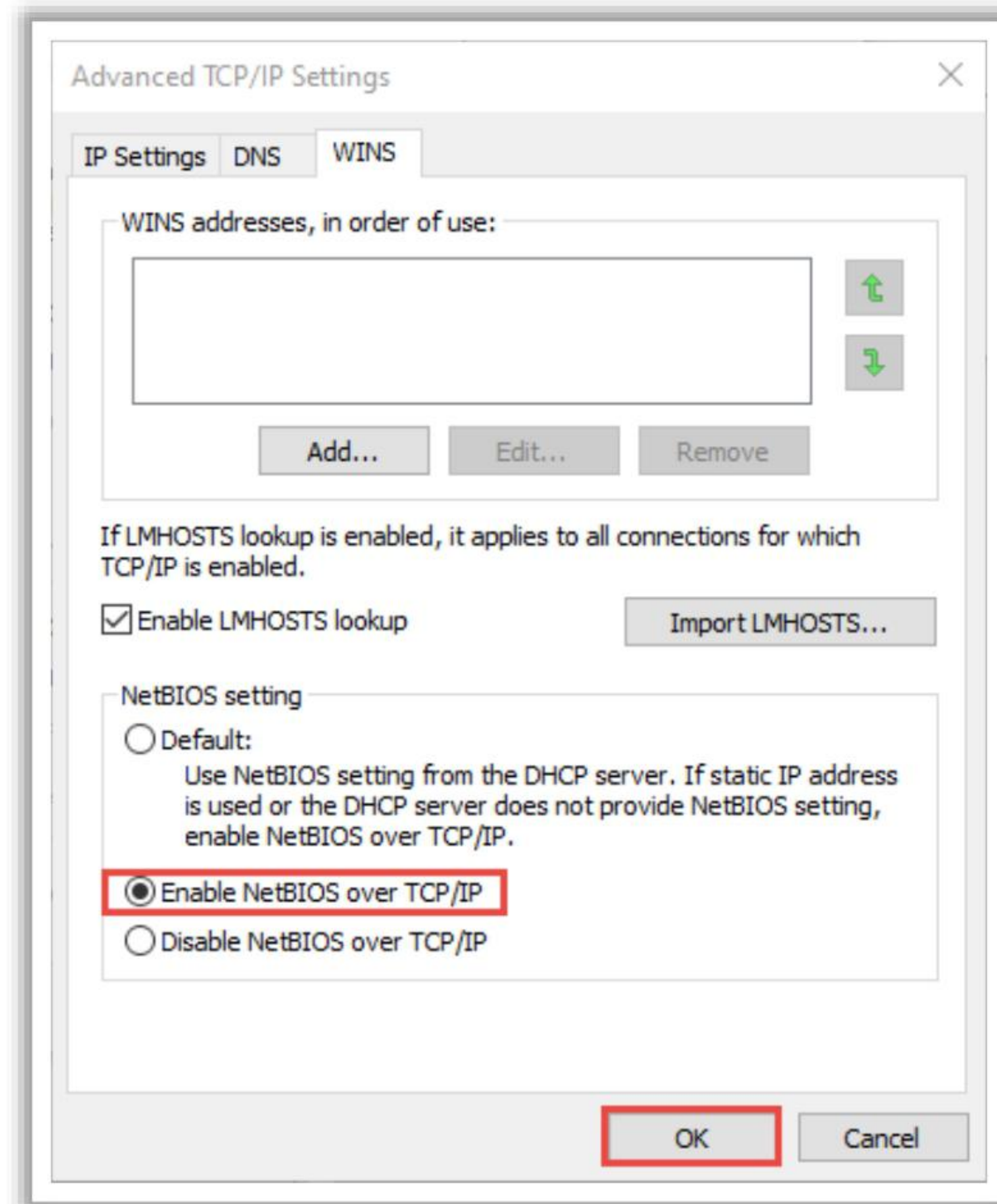


Figure 16.40: Enable NetBIOS over TCP/IP

- Reboot the PC to save the changes.
- **Rename the computer**
 - Go to **Control Panel/Credential Manager**.
 - Select **Windows Credentials** and click **Add a Windows credential**.

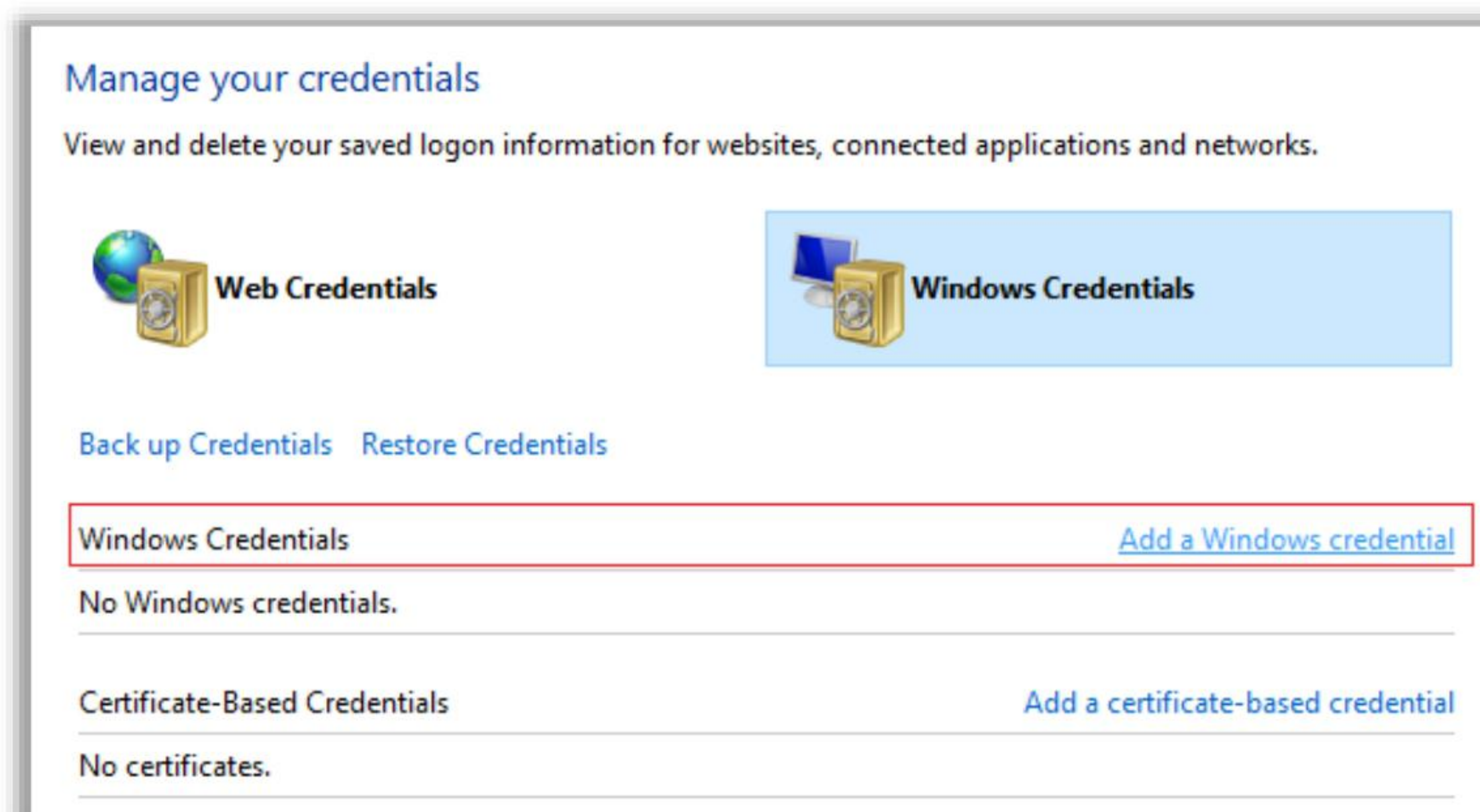


Figure 16.41: Add a Windows Credential

- Sequentially enter the username and password of each machine connected to the network.

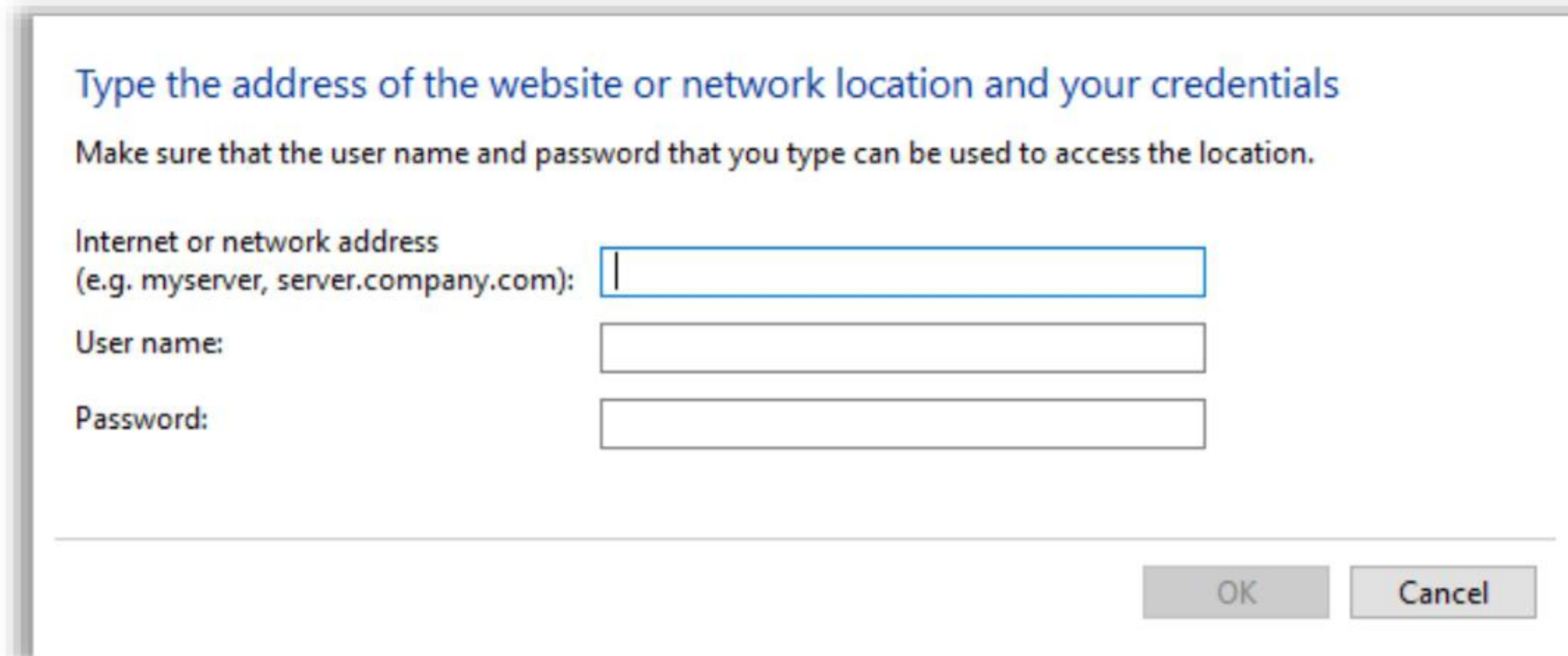


Figure 16.42: Enter Credentials for Machine

- Follow the above procedure for the connected PCs.
- **Disable antivirus and firewall**
 - Right-click the antivirus program icon from the system tray and select **Disable**.

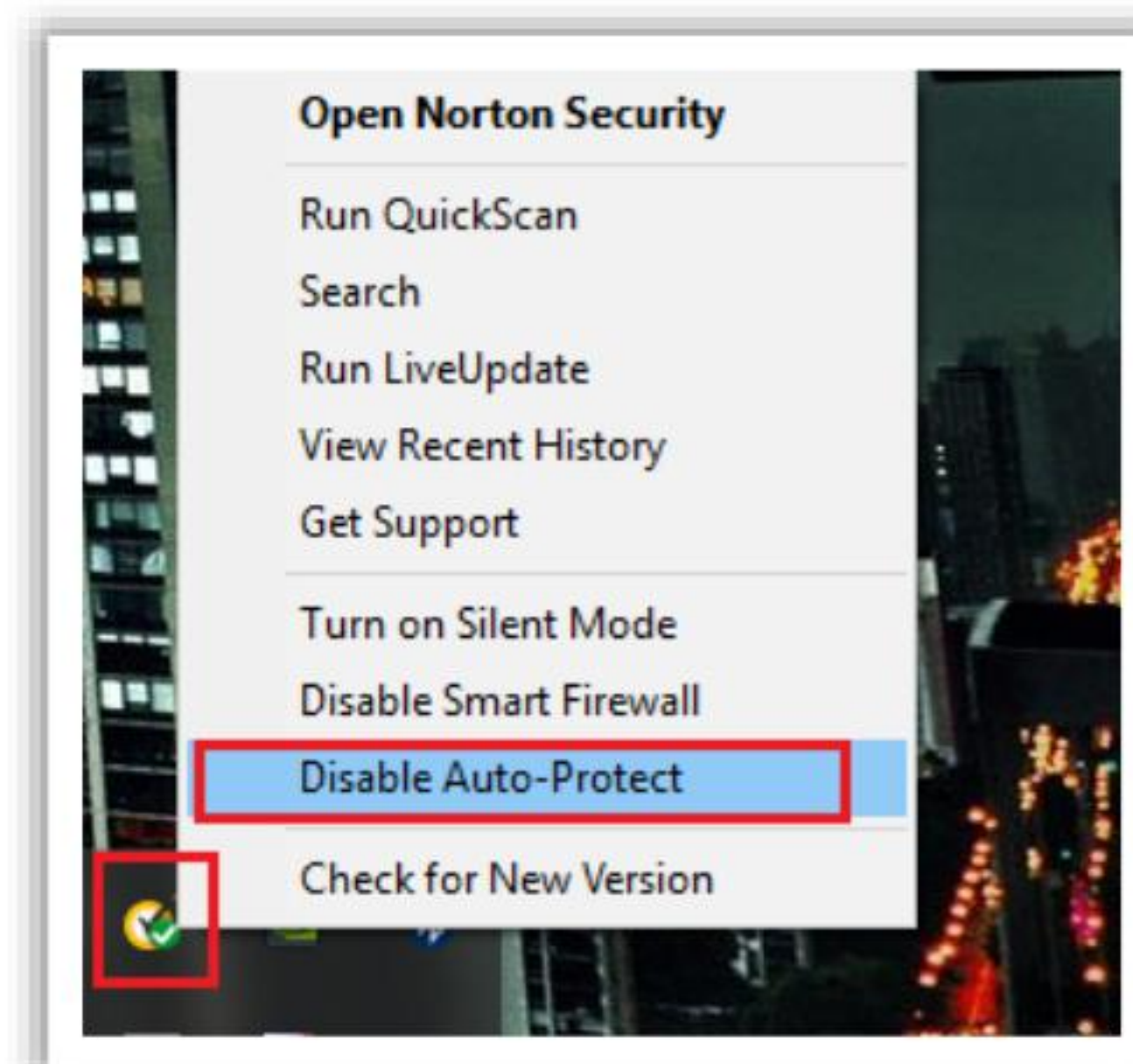


Figure 16.43: Disable Anti-virus

- Select the time frame for which the antivirus program will remain disabled.

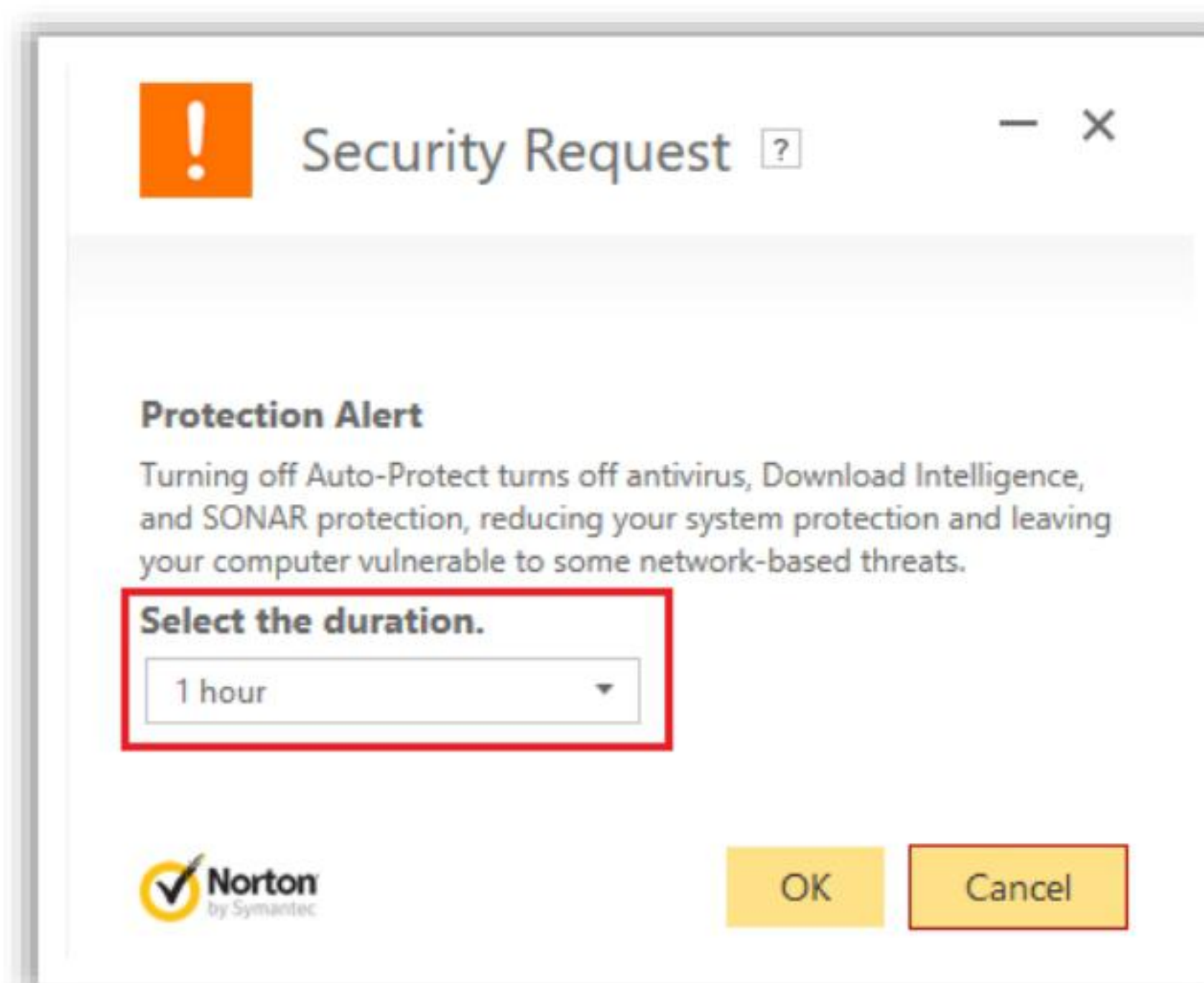


Figure 16.44: Duration Selection

- Once done, check whether the error is resolved.
- Go to **Control Panel/System and Security**.
- Click **Windows Firewall**.

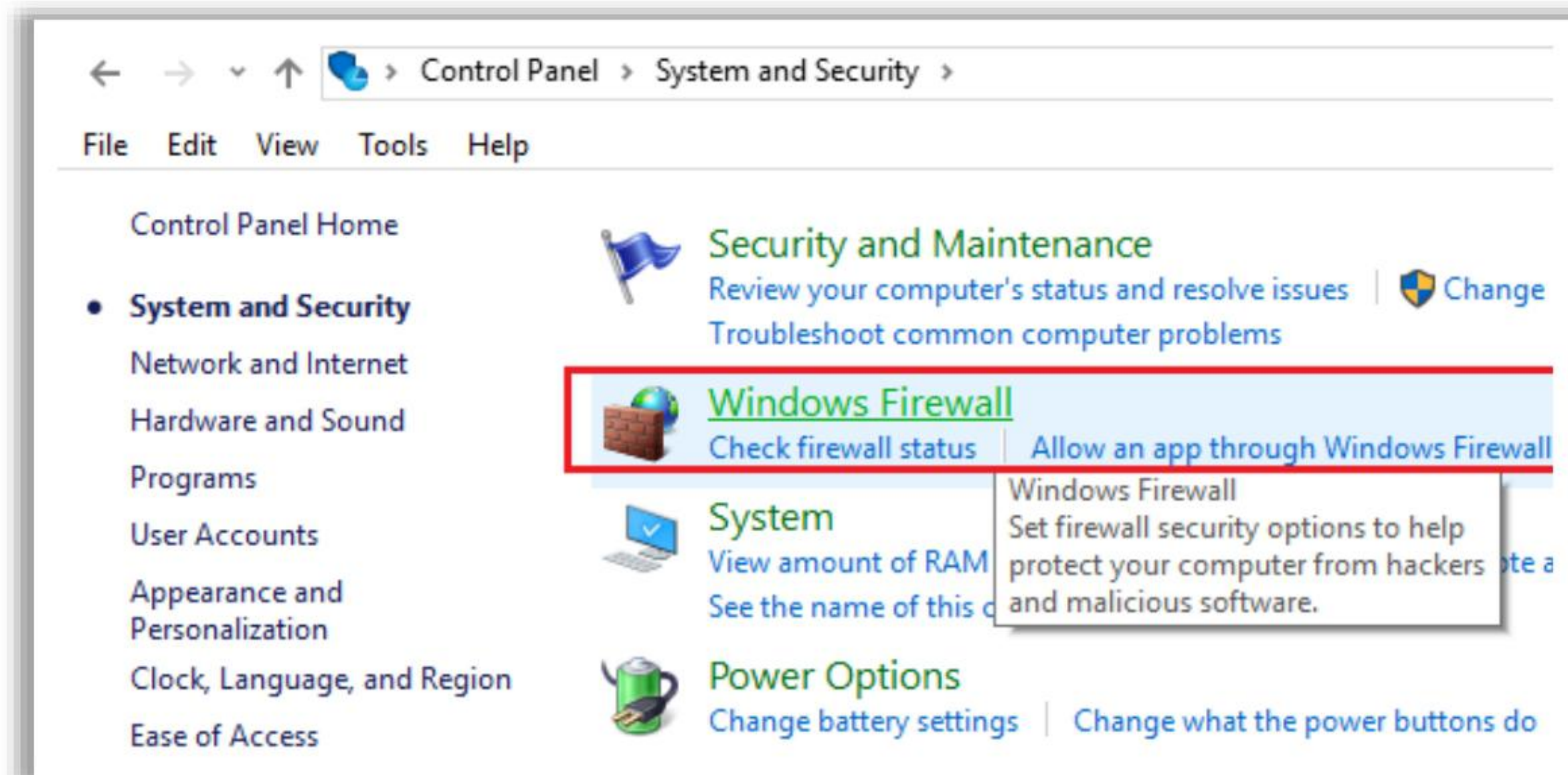


Figure 16.45: Click “Windows Firewall”

- Click **Turn Windows Firewall on or off** from the left pane.

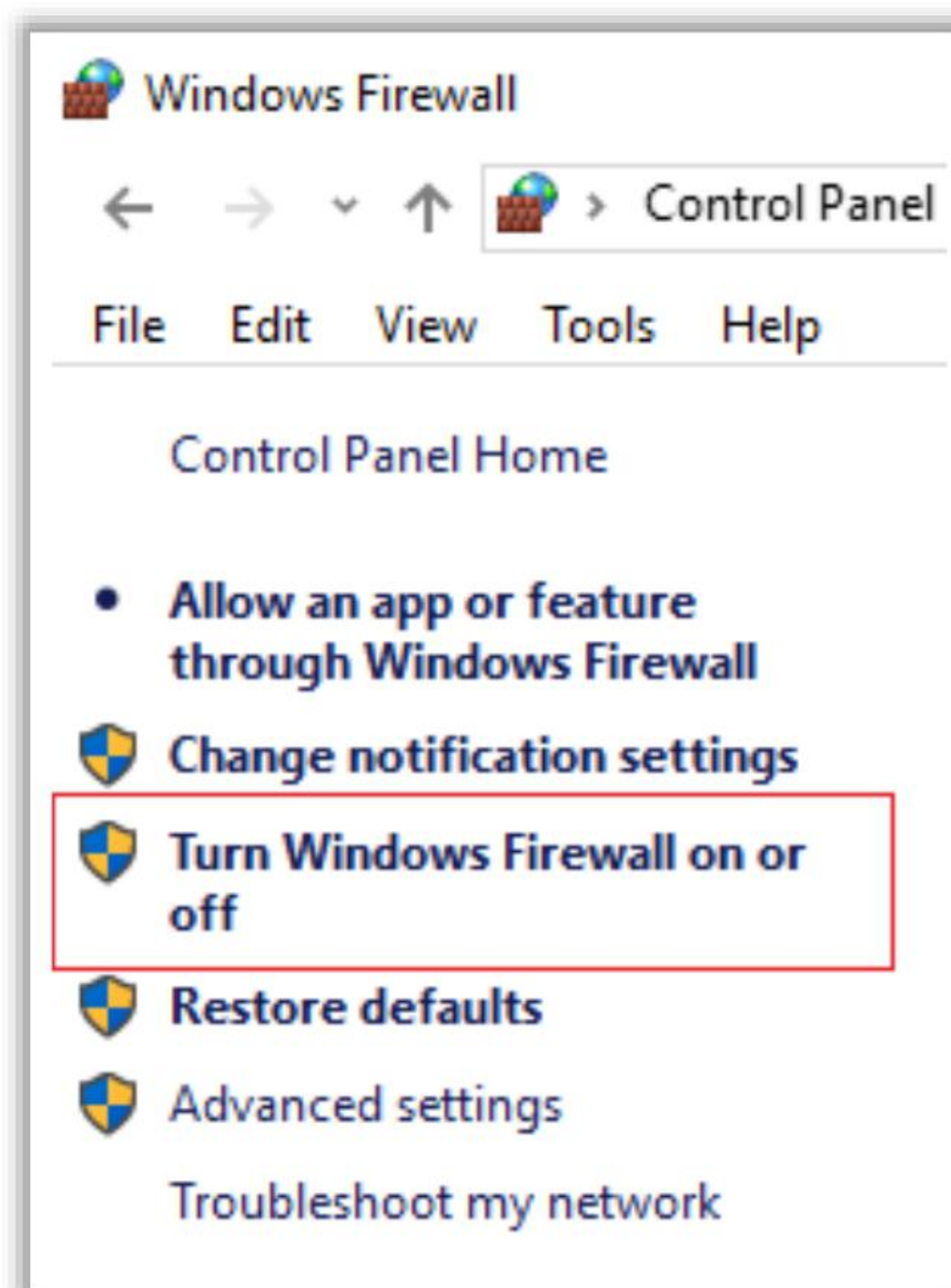


Figure 16.46: Select “Turn Windows Firewall on or off” Option

- Select **Turn off Windows Firewall** and restart the PC.



Troubleshooting Network Issues: Physical Connectivity Issues

Steps to Troubleshoot Physical Connectivity Issues

- Check for cable connectivity issues:**
 - Check that suitable cables are used for connections between devices
 - Loose connections are to be avoided
 - If there are no loose connections, check for old cables and replace them with new ones and then try to connect the device
 - If the problem persists, then it may be faulty port issue
- Check for faulty ports:**
 - Check the ports where the link is established and confirm that the indicator lights are on
- Check for traffic overload:**
 - Crosscheck the capacity of the devices in the network and the traffic flowing through it
 - Exceeding the specified limit could lead to the interruption of communication between the source and destination

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: Physical Connectivity Issues

Steps to Troubleshoot Physical Connectivity Issues

- **Check for cable connectivity issues:** The user should primarily check that the network cable is plugged in. If the cable is not plugged in, then plug it into the network card of the computer. If the cable is securely plugged in, then reboot the system and check for cable connectivity issues.

The cable used to connect computers may be physically damaged, shorted, or faulty. Check that suggested and suitable cables are used for connections between devices. For example, check that the connectivity between a router and computer is established using a crossover pair of cables. Loose connections are to be avoided. If there are no loose connections, check for old cables and replace them with new ones and attempt to connect the device. If the problem persists, then it can be a faulty port issue.

- **Check for faulty ports:** Check the ports where the link is established and confirm that the indicator lights are on. The user should verify the duplex mode and speed. If the port is working properly but the connectivity issue persists, then check the indicator lights on each device to determine the working status of the port. Check whether the port is physically radiating or the indicator lights are off. From the light status, the user can determine whether the port is physically malfunctioning. In this scenario, the user should configure the link on some other free port/interface.
- **Check for traffic overload:** Crosscheck the capacity of the devices in the network and the traffic flowing through it. Exceeding the specified limit could lead to the interruption of communication between the source and destination. If traffic overload occurs at the link or interface, then it would function abnormally at some point; therefore, the user should verify these criteria to avoid any connectivity issues.

Troubleshooting Network Issues: Local Connectivity Issues

Steps to Troubleshoot Local Connectivity Issues

- ❖ **Ping the destination** if the source and the destination are of the same subnet mask
- ❖ **Ping the gateway IP** of the router if the source and destination are not of the same subnet mask
- ❖ If the ping fails, check that the route followed by the **subnet mask is defined correctly** in the routing table
- ❖ If everything is fine, check if the **source is pinging** a hop/router in the network
- ❖ If the ping fails, it could be a **configuration issue** or an IP address conflict
- ❖ Resolve any **IP address conflict** by disconnecting the doubtful device and pinging it with other devices in the network
- ❖ If the device is pinging, then the disconnected device was using the same IP as the pinged device. Then, **modify the IP** according to the plan



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: Local Connectivity Issues







If the user encounters LAN connectivity issues, then to identify the root cause and resolve the issue, it is recommended to follow the steps below.

Steps to Troubleshoot Local Connectivity Issues

- The user should ping the destination IP if the source and destination are of the same subnet mask.
- The user should ping the gateway IP of the router if the source and destination are not of the same subnet mask.
- In case both pings fail, then the user must check whether the configuration settings of the subnet mask and the route to the destination are defined properly in the routing table.
- If the configuration settings are fine, then the user should check whether the source host can ping a different hop in the LAN network other than the destination host.
- If the user is unable to ping another device, then there might be various issues such as IP address conflict, configuration issues, and physical connectivity issues.
- To rectify any IP address conflict, the user should disconnect all doubtful devices from the LAN. At this point, the user needs to ping the device from some other device on the same subnet/LAN. A successful ping signifies that the disconnected device is using the same IP as the pinged device; then, modify the IP according to the plan. If the local connectivity issue persists, then there might be a problem with the physical connectivity or configuration.

Troubleshooting Network Issues: Routing Problems

Steps to Troubleshoot Physical Routing Issues

-  Use the **tracert** utility to locate the hop/router responsible for the problem
-  If the issue persists, investigate each **hop/router** to find where the problem has occurred
-  When the problematic **hop/router** is detected, login to it using Telnet and ping the destination and source
-  If the ping is unsuccessful, configure the routes between the source and destination with a **subnet mask** if they are not defined
-  Check for a **routing loop** by pinging again. If it exists, rectify it by tracing and reconfiguring the routing
-  Check the **routing protocol** if the problem persists, and change it according to the network



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: Routing Problems

When data packets are routed across a network, there are chances of fault occurrence. Based on the type of fault, the user should make a plan to resolve the fault. The type of faults that are observed between source and destination hosts when data packets are routed across a network are as follows:

- Failure to define the route in a router between the source and destination
- Usage of an incorrect routing protocol to find the route to the next hop/destination
- Fault in the router's software
- Firewall or filter hindering the movement of data packets to the destination node
- Configuration faults at the source router end

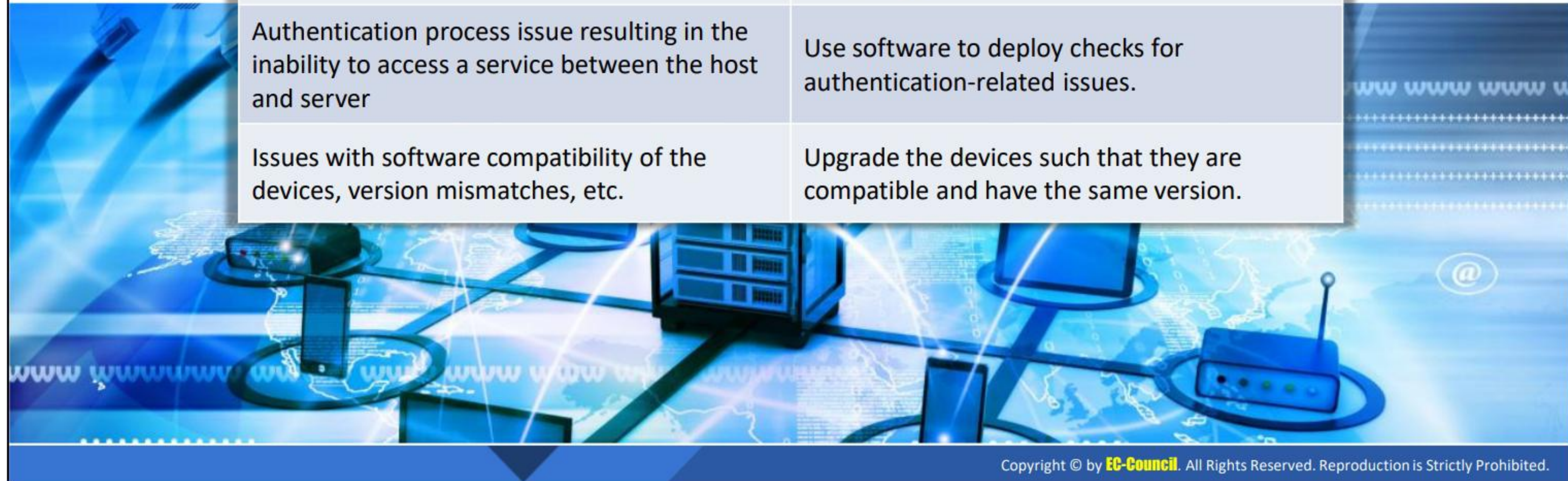
Steps to Troubleshoot Routing Issues

- The first step to troubleshoot routing issues is to locate the hop between the source and destination where the problem occurred.
- From the source to the destination, verify the IP connectivity and routing protocols at each hop.
- Use the traceroute utility to locate the hop/router responsible for the problem.
- After identifying the hop where the problem occurred, login to the router through Telnet and ping the source and destination hosts.
- If the ping is unsuccessful, then the routing table should be verified for the route between the source and destination.

- If there is a small percentage of successful ping responses, then there might be multiple defined paths to reach the destination, among which one results in failure owing to the occurrence of a routing loop in the path. To resolve this issue, the looping hop should be traced, and the configuration should be rectified.
- After rectification, if the problem persists, then check the used routing protocol and change it according to the network.

Troubleshooting Network Issues: Upper-layer Faults

Common problems	Rectification Steps
Firewall blocking the flow of incoming and outgoing traffic	Move the host in the network such that it bypasses the firewall blocking the traffic.
Server or service down	Replace the server that is down with a temporary server to continue the services.
Authentication process issue resulting in the inability to access a service between the host and server	Use software to deploy checks for authentication-related issues.
Issues with software compatibility of the devices, version mismatches, etc.	Upgrade the devices such that they are compatible and have the same version.



Troubleshooting Network Issues: Upper-layer Faults

If the user has already checked the physical connectivity, local connectivity, IP connectivity, and routing issues but was unable to find a fault, then there is a chance that the fault has occurred in the transport- and application-layer protocols.

The following are potential reasons for the occurrence of faults:

- The data connection is not working.
- The firewall/packet filter prevents the incoming and outgoing traffic.
- A specific service on the server is not working.
- There are authentication and access concerns between the client and server host.
- There are issues related to the incompatibility of software or mismatch of its version between the source and destination host.

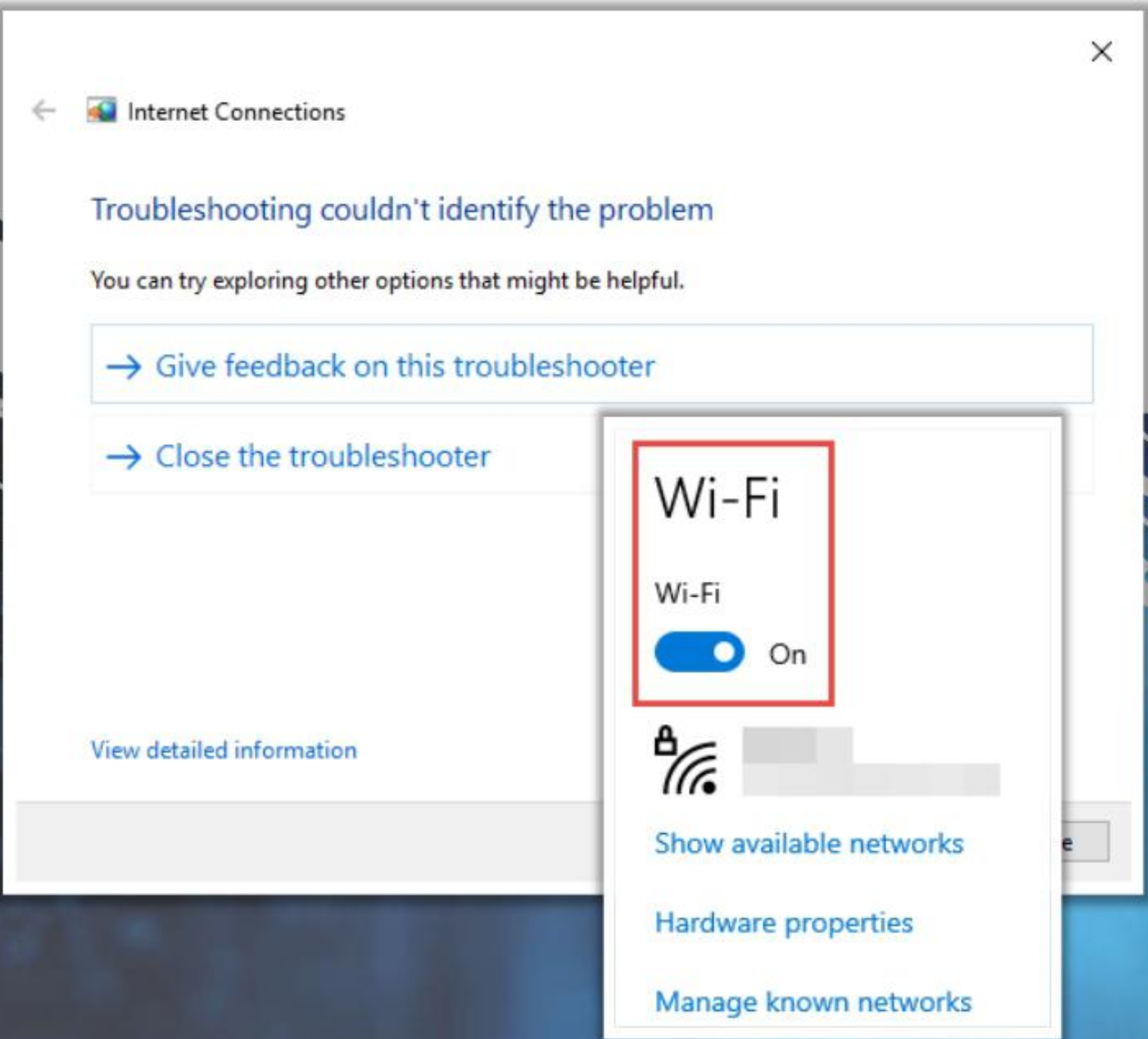
The following table outlines rectification steps based on the category of the fault.

Common Problems	Rectification Steps
Firewall blocking the flow of incoming and outgoing traffic	Move the host in the network such that it bypasses the firewall blocking the traffic.
Service/server down	Replace the server that is down with a temporary server to continue services.
Authentication process issue resulting in the inability to access services between the host and server	Use software to deploy checks for authentication-related issues.
Issues with the software compatibility of devices, version mismatches, etc.	Upgrade the devices such that they are compatible and have the same software version.

Table 16.2: Rectification Steps Based on the Category of the Fault

Troubleshooting Network Issues: Wireless Network Connection Issues

- ❑ Check whether Wi-Fi is enabled in the devices:
 - Go to **Settings** → **Network & Internet** → **Wi-Fi**
- ❑ If the problem persists,
 - Check and change the service set identifier (SSID) and access points to allocate the requesting device an IP address
- ❑ Use the **Windows Network Diagnostics** tool to troubleshoot the network-related issue:
 - Windows Network Diagnostics will troubleshoot and detect the problem by downloading and installing available patches
- ❑ Restore the router to its factory settings and restart it



Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Network Issues: Wireless Network Connection Issues

To troubleshoot a wireless network connection issue, the user primarily needs to check whether Wi-Fi is enabled on the devices by performing the following steps:

- Click on the **Start Menu** button and go to **Settings**. Next, click on **Network & Internet**.
- Enable **Wi-Fi**.

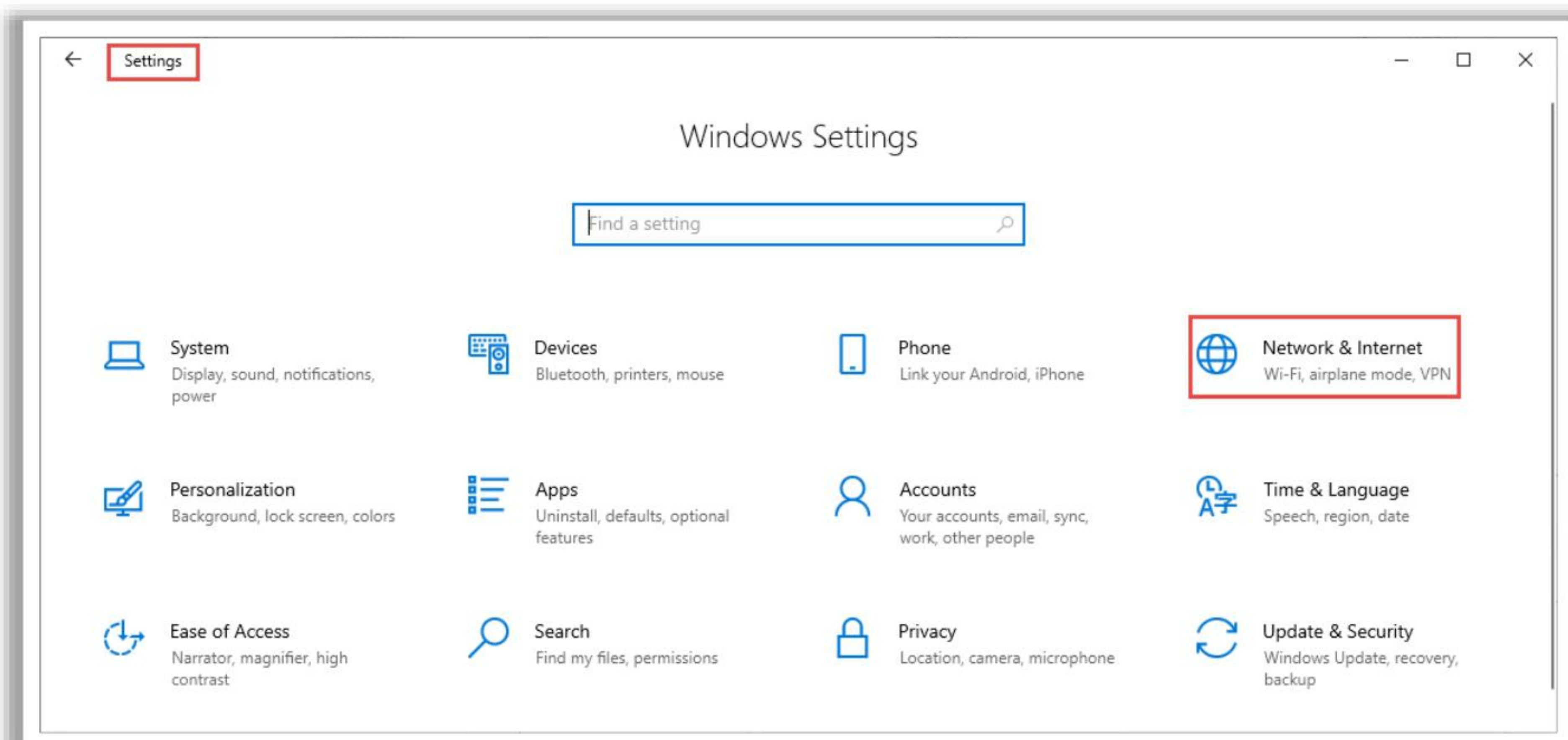


Figure 16.47: Go to “Network & Internet” Setting

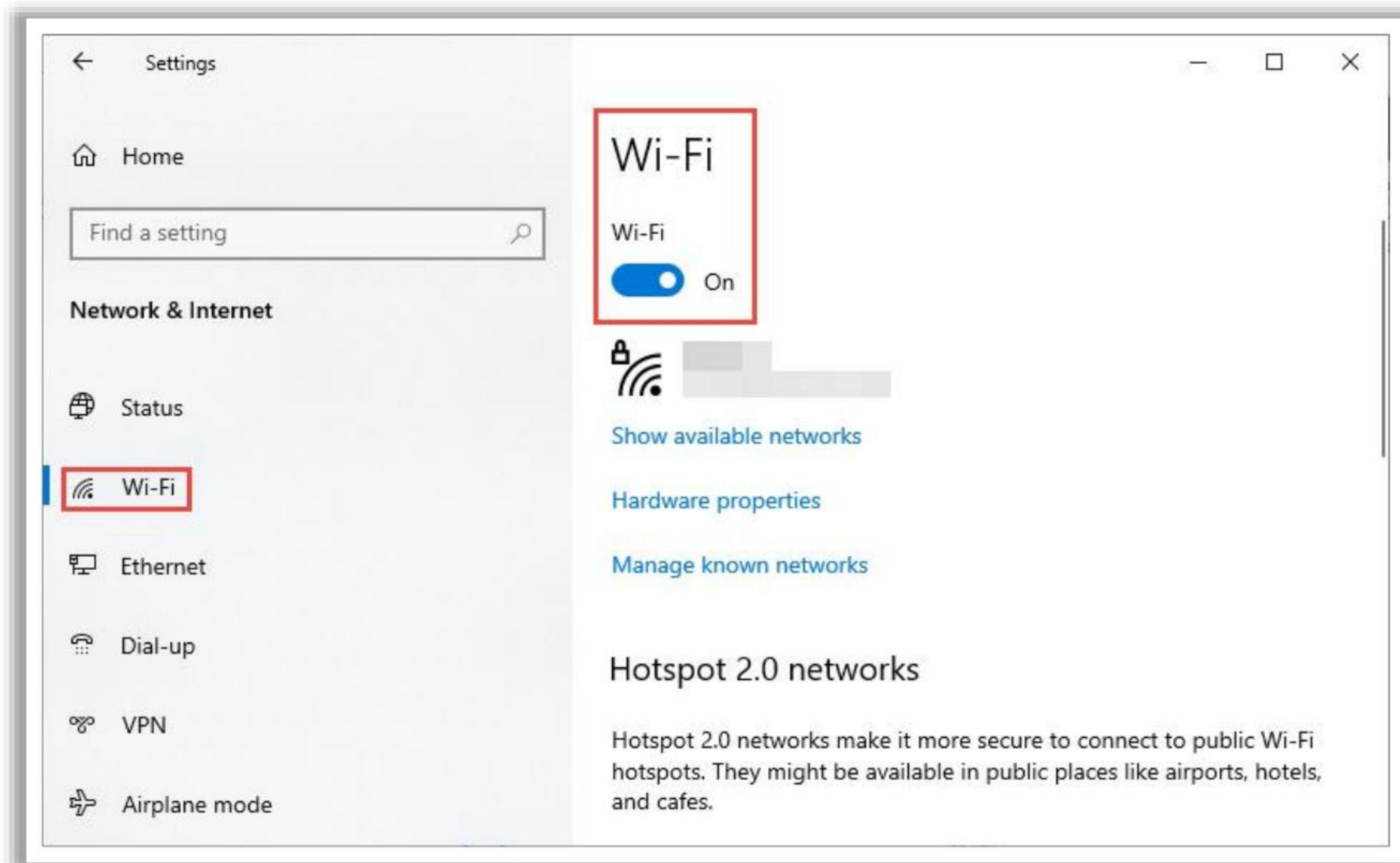


Figure 16.48: Enable Wi-Fi

If Wi-Fi is not listed in **Network & Internet**, then the user would not be able to connect to the wireless network. If there is no Wi-Fi on the computer, then the Internet access icon will also be missing from the taskbar.

Steps to Troubleshoot Wireless Network Connection Issues

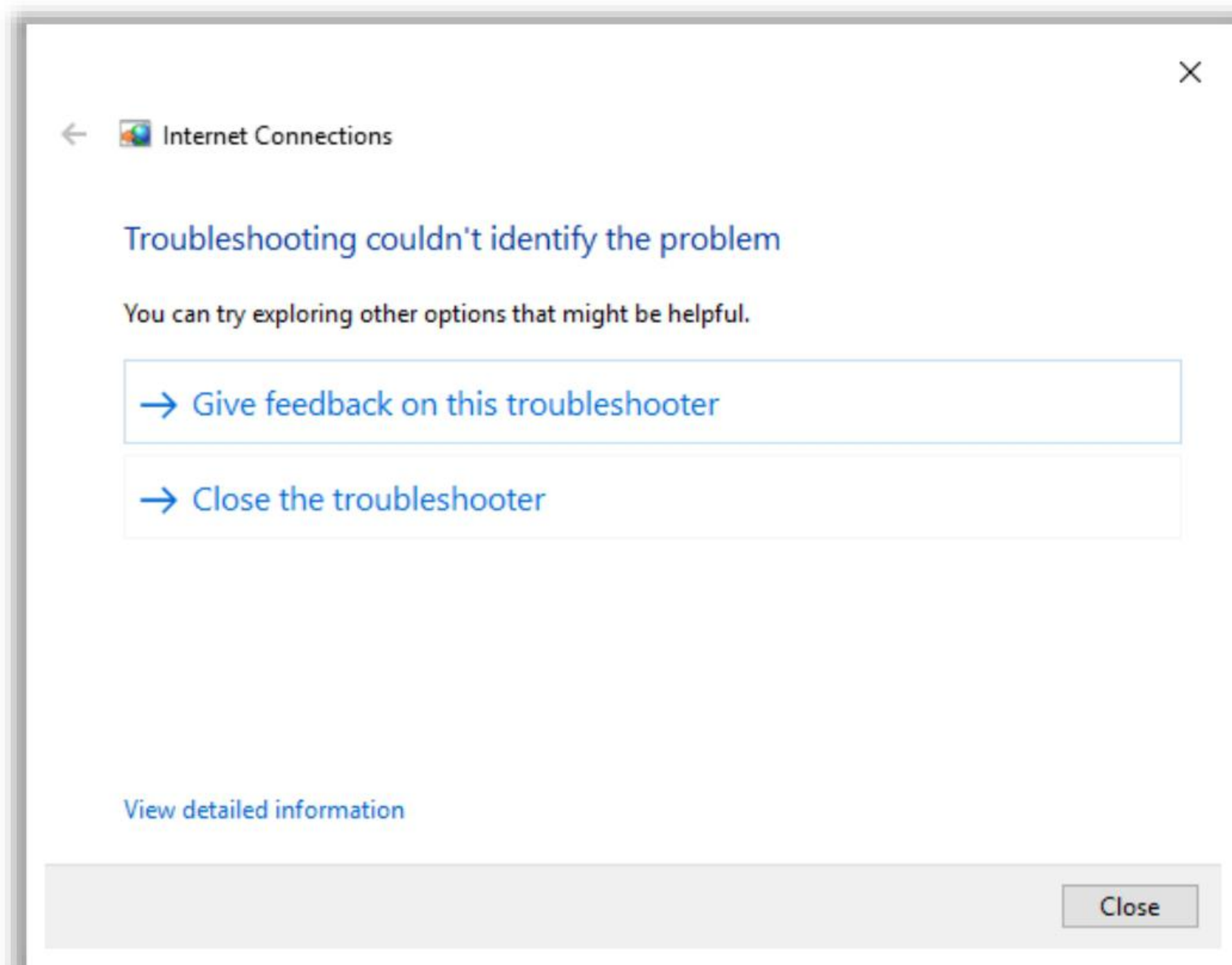
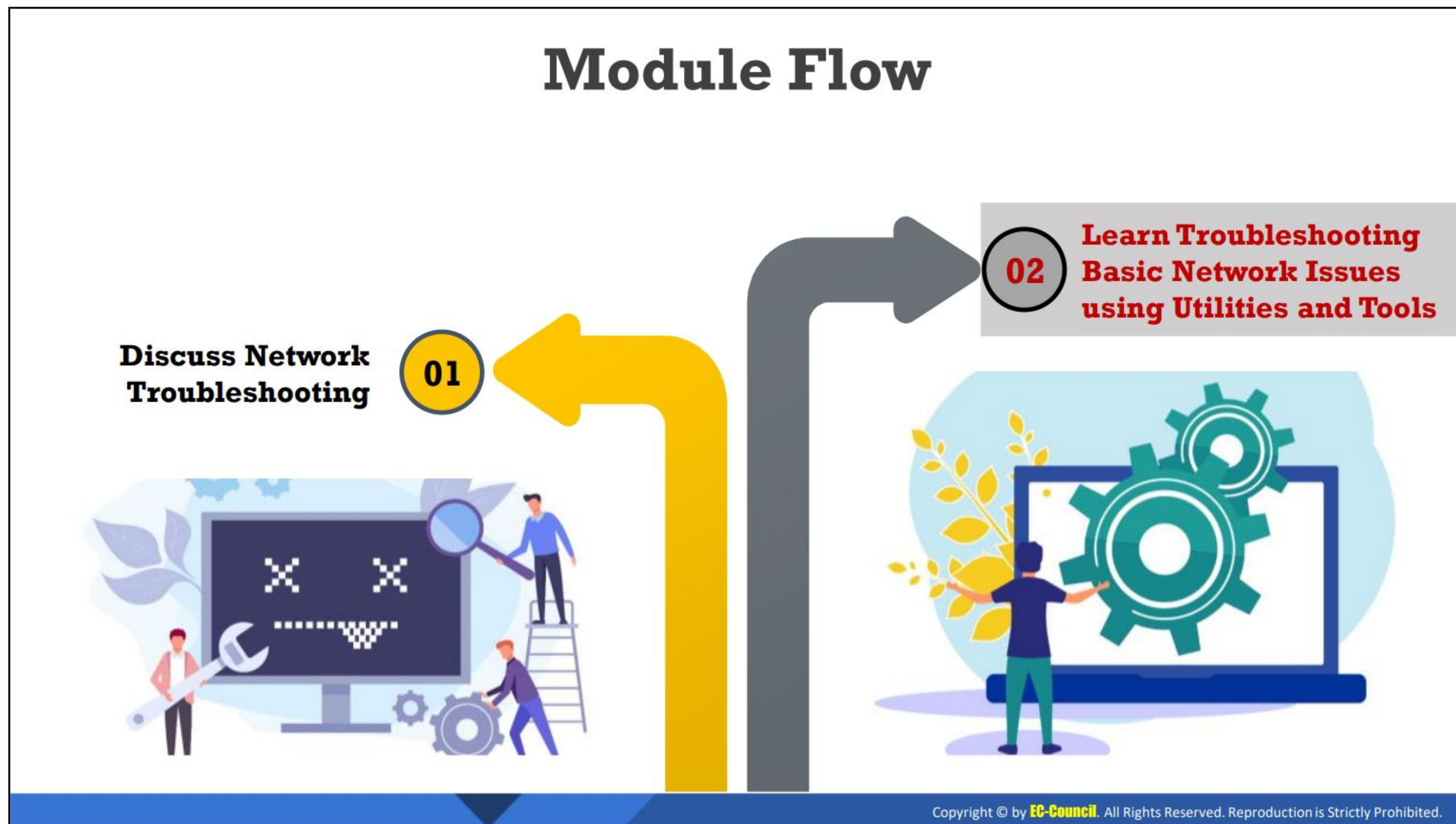


Figure 16.49: Troubleshoot Internet Connections issues





- If the user is unable to connect the device using Wi-Fi, then the user must first check the LAN and WAN cable connections. Ensure that the Ethernet cable connection is tight, and check the status of indicator lights. If the light is not green, then there might be a problem with the cable or port. Then, the user must change the faulty port or cable connection.
- If the problem persists, then the user should check the Wi-Fi network adaptor settings and enable network adaptor settings. The user should ensure that the airplane mode on the device is disabled. If the airplane mode is enabled, then the device would not allow wireless network connectivity.
- If the problem persists, check and change the service set identifier (SSID) and access points to allocate the requesting device an IP address.
- Use the Windows Network Diagnostics tool to troubleshoot network-related issues. Windows Network Diagnostics will troubleshoot the defect/problem by downloading and installing available patches.
- Restore the router to its factory settings and restart it.
- After performing all the steps, if the user is unable to resolve the wireless network connectivity issue, then there might be other reasons such as a firewall or packet filter preventing the user from connecting to the network or an issue with the authentication protocol. Hence, the user should reconfigure all the network settings and verify the IP routing by using ping.

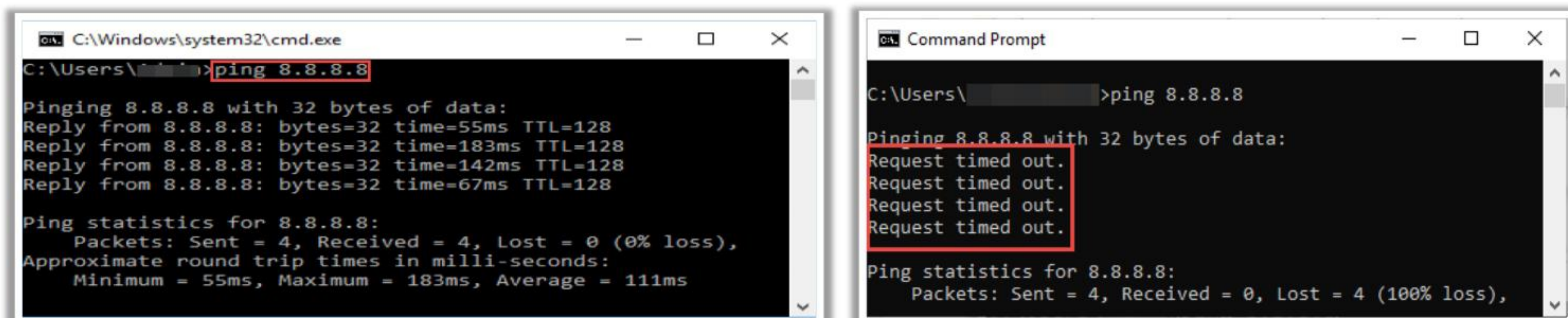


Learn Troubleshooting Basic Network Issues using Utilities and Tools

This section discusses how to troubleshoot various network issues using network troubleshooting tools and utilities.

Network Troubleshooting Utilities and Tools: Ping

-  The **ping** utility is used to test whether an IP address or a website is accessible by the host
-  A reply from the pinged IP address indicates that the packets are transferring between the system and **given IP**
-  Launch command prompt and execute **ping x.x.x.x** or **ping example.com** to check the availability of the host to the computer
-  The message "Request timed out" indicates that there is **no connection** between the system and host or that the system is unable to connect to the host



```
C:\Windows\system32\cmd.exe
C:\Users\>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=55ms TTL=128
Reply from 8.8.8.8: bytes=32 time=183ms TTL=128
Reply from 8.8.8.8: bytes=32 time=142ms TTL=128
Reply from 8.8.8.8: bytes=32 time=67ms TTL=128

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 55ms, Maximum = 183ms, Average = 111ms

Command Prompt
C:\Users\>ping 8.8.8.8

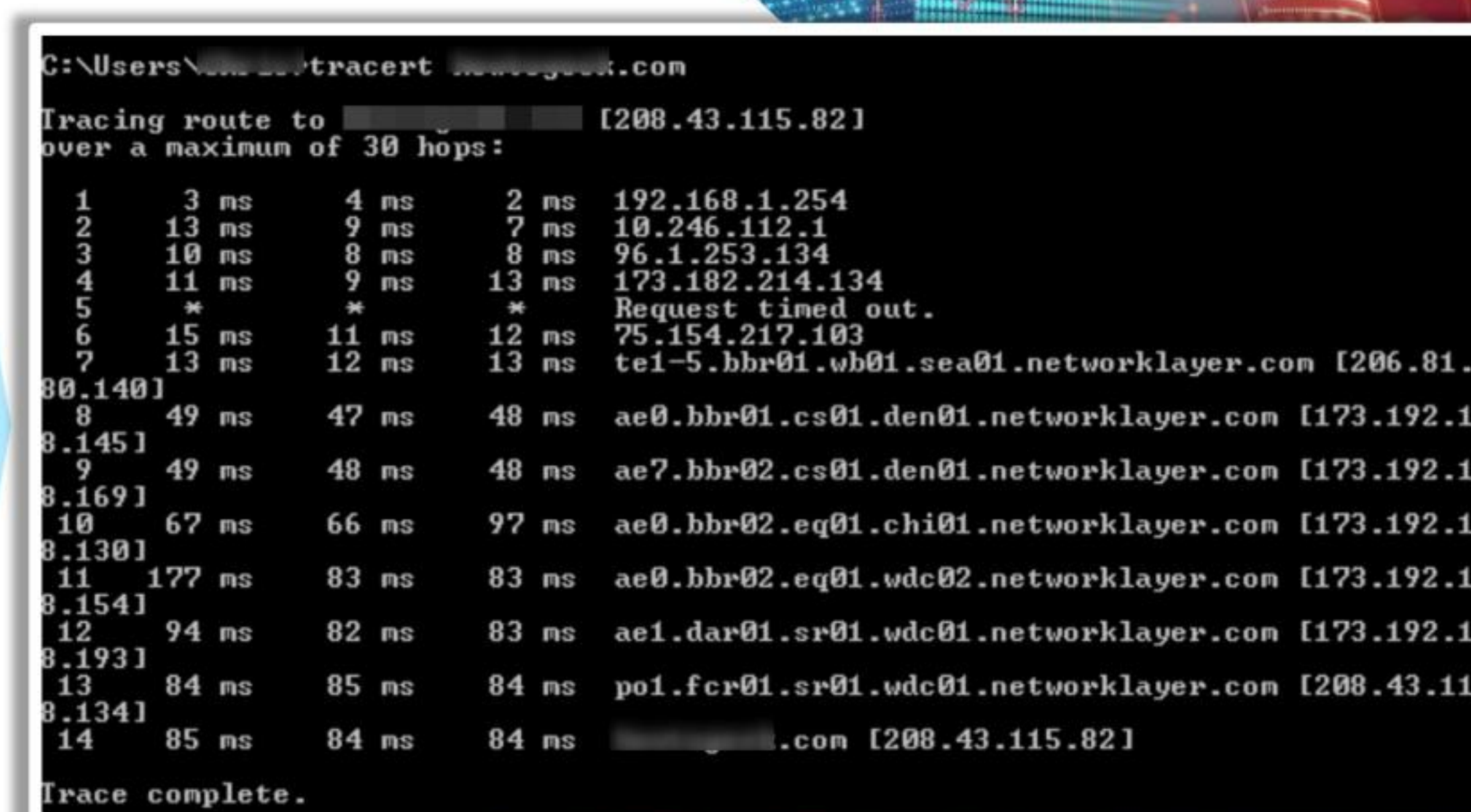
Pinging 8.8.8.8 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: traceroute/tracert

- The traceroute utility is used to **trace packets across a network** and to understand connections to a server
- It sends an **ICMP echo request** message to the specified destination
- If the destination is active, it sends **ICMP echo reply** messages as a response, which confirms that the connection is active
- If not, the destination may not be active, or it could be a connectivity issue of the source



```
C:\Users\>tracert google.com

Tracing route to google.com [208.43.115.82]
over a maximum of 30 hops:
 0  0 ns  0 ns  0 ns  192.168.1.254
 1  3 ns  4 ns  2 ns  10.246.112.1
 2  13 ns  9 ns  7 ns  96.1.253.134
 3  10 ns  8 ns  8 ns  173.182.214.134
 4  11 ns  9 ns  13 ns  *
 5  *  *  *  Request timed out.
 6  15 ns  11 ns  12 ns  75.154.217.103
 7  13 ns  12 ns  13 ns  tel-5.bbr01.wb01.sea01.networklayer.com [206.81.80.140]
 8  49 ns  47 ns  48 ns  ae0.bbr01.cs01.den01.networklayer.com [173.192.18.145]
 9  49 ns  48 ns  48 ns  ae7.bbr02.cs01.den01.networklayer.com [173.192.18.169]
10  67 ns  66 ns  97 ns  ae0.bbr02.eq01.chi01.networklayer.com [173.192.18.130]
11  177 ns  83 ns  83 ns  ae0.bbr02.eq01.wdc02.networklayer.com [173.192.18.154]
12  94 ns  82 ns  83 ns  ae1.dar01.sr01.wdc01.networklayer.com [173.192.18.193]
13  84 ns  85 ns  84 ns  po1.fcr01.sr01.wdc01.networklayer.com [208.43.115.134]
14  85 ns  84 ns  84 ns  google.com [208.43.115.82]

Trace complete.
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools : ipconfig/ifconfig



- ipconfig** is a command-line utility used to display all current TCP/IP network configuration values along with the IP address, subnet mask, and default gateway for all adapters
- ifconfig** is a similar utility but for Linux-based machines

```
cmd Select C:\Windows\system32\cmd.exe
C:\Users\>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::a51f:1c29:8fa1:2182%6
    IPv4 Address. . . . . : 10.10.10.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 10.10.10.2

Ethernet adapter Npcap Loopback Adapter:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::7c99:753f:b7c8:b27b%7
    Autoconfiguration IPv4 Address. . : 169.254.178.123
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . :
```

```
Parrot Terminal
[root@parrot ~]# ifconfig

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.10.13 netmask 255.255.255.0 broadcast 10.10.10.255
    inet6 fe80::8b64:ae6:facb:28a prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:eb:cd:af txqueuelen 1000 (Ethernet)
    RX packets 621 bytes 865044 (844.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 102 bytes 7793 (7.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 82 bytes 6826 (6.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 82 bytes 6826 (6.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools : nslookup

- The nslookup utility is used to **lookup a specific IP address** or multiple IP addresses associated with a domain name(s) at a time
- nslookup is used when a user can **access a resource** by specifying its IP address but not by specifying its domain name
- The nslookup utility is used to **resolve DNS address** resolution issues
- The nslookup command is executed in the command prompt to lookup the IP address for a domain name

```
cmd Select Command Prompt
C:\>nslookup www.google.com
Server: dns.google
Address: 8.8.8.8

Non-authoritative answer:
Name: www.google.com
Addresses: 2404:6800:4007:806::2004
          142.250.196.68
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: netstat

- ➔ **netstat** is a command-line utility used to display both the incoming and outgoing TCP/IP traffic
- ➔ The current state of the active hosts on the network can be determined using netstat
- ➔ netstat is also used to identify the services associated with user-defined ports
- ➔ Execute the **netstat** command without any parameters in the terminal to show the list of active connections
- ➔ Use the **netstat -e** command to show the statistics of various protocols

```
C:\> netstat -e
Interface Statistics

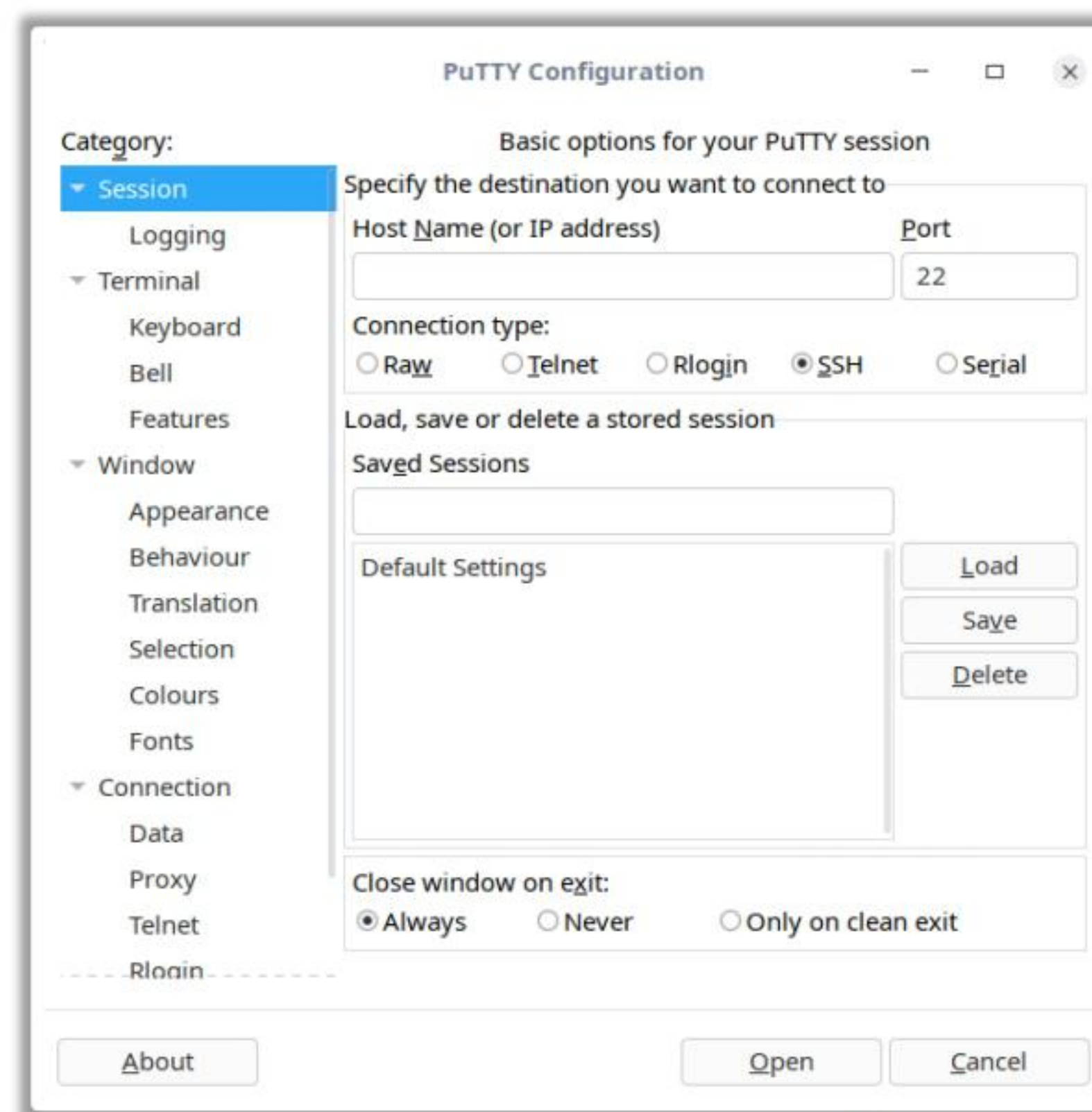
Received          Sent
Bytes            3850711362      493608888
Unicast packets  5477744         607283
Non-unicast packets
Discards         0               0
Errors           0               0
Unknown protocols
```

```
netstat -e
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       User
-----
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags   Type       State          I-Node      Path
-----
unix  18      [ ]     DGRAM     -              19719      /run/systemd/journal/dev-log
unix  3       [ ]     DGRAM     -              19258      /run/systemd/notify
unix  6       [ ]     DGRAM     -              19274      /run/systemd/journal/socket
unix  2       [ ]     DGRAM     -              40599      /run/user/0/systemd/notify
unix  2       [ ]     DGRAM     -              19678      /run/systemd/journal/syslog
unix  3       [ ]     STREAM   CONNECTED      42315      @/tmp/dbus-02JM0bhFo0
unix  2       [ ]     DGRAM     -              40772
unix  3       [ ]     STREAM   CONNECTED      44505      /run/user/0/bus
unix  3       [ ]     STREAM   CONNECTED      25563
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: PuTTY

PuTTY is an FTP or SSH FTP (SFTP) client for transferring files. It generates hashes for passwords



<https://www.putty.org>



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: IP Subnet Calculator



Subnet is used to find information about **IPv4** and **IPv6 subnets** and for the division of classes of subnets



The IP subnet calculator is used to **define possible IP addresses**, along with classes of IP



Broadcast ranges, network, and host ranges are calculated using the **IP calculator**

Subnet Calculator

IP Address and Netmask:

10.0.1.1/16

Calculate

Random IP

IP Address:	10.0.1.1
Netmask:	255.255.0.0
Wildcard Mask:	0.0.255.255
CIDR Notation:	/16
Network Address:	10.0.0.0
Usable Host Range:	10.0.0.1 - 10.0.255.254
Broadcast Address:	10.0.255.255
Binary Netmask:	11111111.11111111.00000000.00000000
Total number of hosts:	65,536
Number of usable hosts:	65,534
IP Class:	A (0.0.0.0 - 127.255.255.255)

Move to adjacent network

Backward

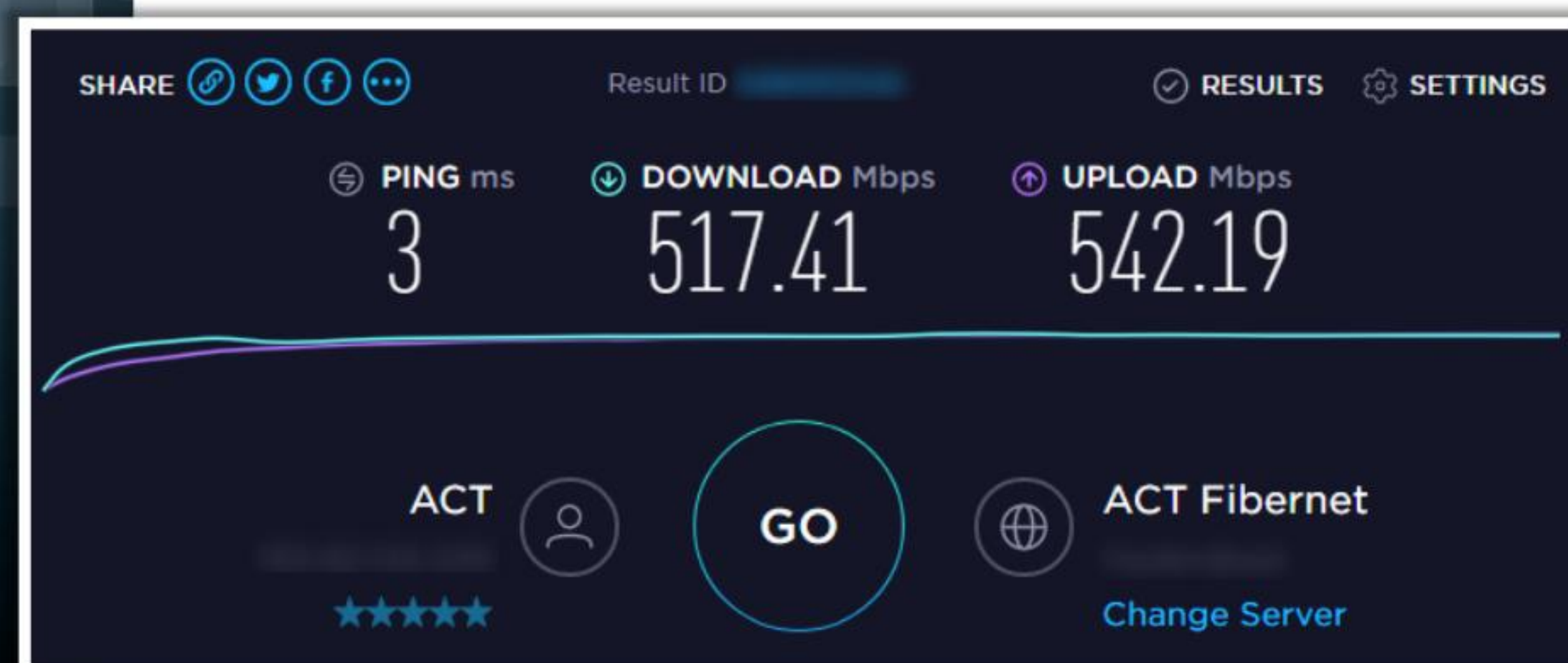
Forward

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: speedtest.net



- speedtest.net** is a website used to determine the available bandwidth for a host at a point of time
- The bandwidth values assigned by the service provider may differ from the actual bandwidth values
- The **time taken to upload and download a file** can also be determined using this website



Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: pathping



The **pathping** utility is used to provide detailed information about the **path characteristics** from a specific host to a specific destination in a single picture



In the first step, pathping traces the route to the destination. Next, it runs a 25-s test on this route and collects the rate at which data is lost at each router



Use the **pathping -n** command to show numeric IP numbers instead of DNS host names

```
Command Prompt - pathping 8.8.8.8
C:\>pathping 8.8.8.8
Tracing route to dns.google [8.8.8.8]
over a maximum of 30 hops:
 0  Windows10 [10.10.10.10]
 1  10.10.10.2
 2  192.168.1.1
 3  100.64.63.254
 4  192.168.34.201
 5  192.168.48.6
 6  192.168.48.2
 7  192.168.48.49
 8  nsg-corporate-229.104.185.122.airtel.in [122.185.104.229]
 9  182.79.198.6
10  72.14.208.234
11  108.170.234.3
12  142.251.55.227
13  dns.google [8.8.8.8]
Computing statistics for 325 seconds...
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: route

- ❑ The route utility is used to **show the ongoing status** of the routing table on the host
- ❑ Netmasks, network destinations, and gateways are displayed in the **active routes section** of the route utility
- ❑ `route [-p] command dest [mask subnet] gateway [-if interface]` is the command for adding, deleting, or changing a route entry



```
Select Command Prompt
C:\>route print
-----
Interface List
6...00 0c 29 df 79 4d .....Intel(R) 82574L Gigabit Network Connection
7...02 00 4c 4f 4f 50 .....Npcap Loopback Adapter
1.....Software Loopback Interface 1
14...00 00 00 00 00 00 e0 Microsoft ISATAP Adapter
12...00 00 00 00 00 00 e0 Teredo Tunneling Pseudo-Interface
11...00 00 00 00 00 00 e0 Microsoft ISATAP Adapter #2
-----

IPv4 Route Table
-----
Active Routes:
Network Destination        Netmask          Gateway          Interface        Metric
0.0.0.0                    0.0.0.0          10.10.10.2       10.10.10.10      266
10.10.10.0                  255.255.255.0    On-link          10.10.10.10      266
10.10.10.10                 255.255.255.255  On-link          10.10.10.10      266
10.10.10.255                255.255.255.255  On-link          10.10.10.10      266
127.0.0.0                   255.0.0.0        On-link          127.0.0.1        306
127.0.0.1                   255.255.255.255  On-link          127.0.0.1        306
127.255.255.255             255.255.255.255  On-link          127.0.0.1        306
169.254.178.123             255.255.255.255  On-link          169.254.178.123  266
169.254.255.255             255.255.255.255  On-link          169.254.178.123  266
224.0.0.0                   240.0.0.0        On-link          127.0.0.1        306
224.0.0.0                   224.0.0.0         240.0.0.0        On-link          169.254.178.123  266
224.0.0.0                   224.0.0.0         240.0.0.0        On-link          10.10.10.10      266
255.255.255.255             255.255.255.255  On-link          127.0.0.1        306
255.255.255.255             255.255.255.255  On-link          169.254.178.123  266
255.255.255.255             255.255.255.255  On-link          10.10.10.10      266
-----

Persistent Routes:
Network Address            Netmask          Gateway Address  Metric
0.0.0.0                    0.0.0.0          10.10.10.2       Default
-----

IPv6 Route Table
-----
Active Routes:
If Metric Network Destination      Gateway
12 306 ::70 On-link
12 306 2001::32 On-link
-----
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: Nmap

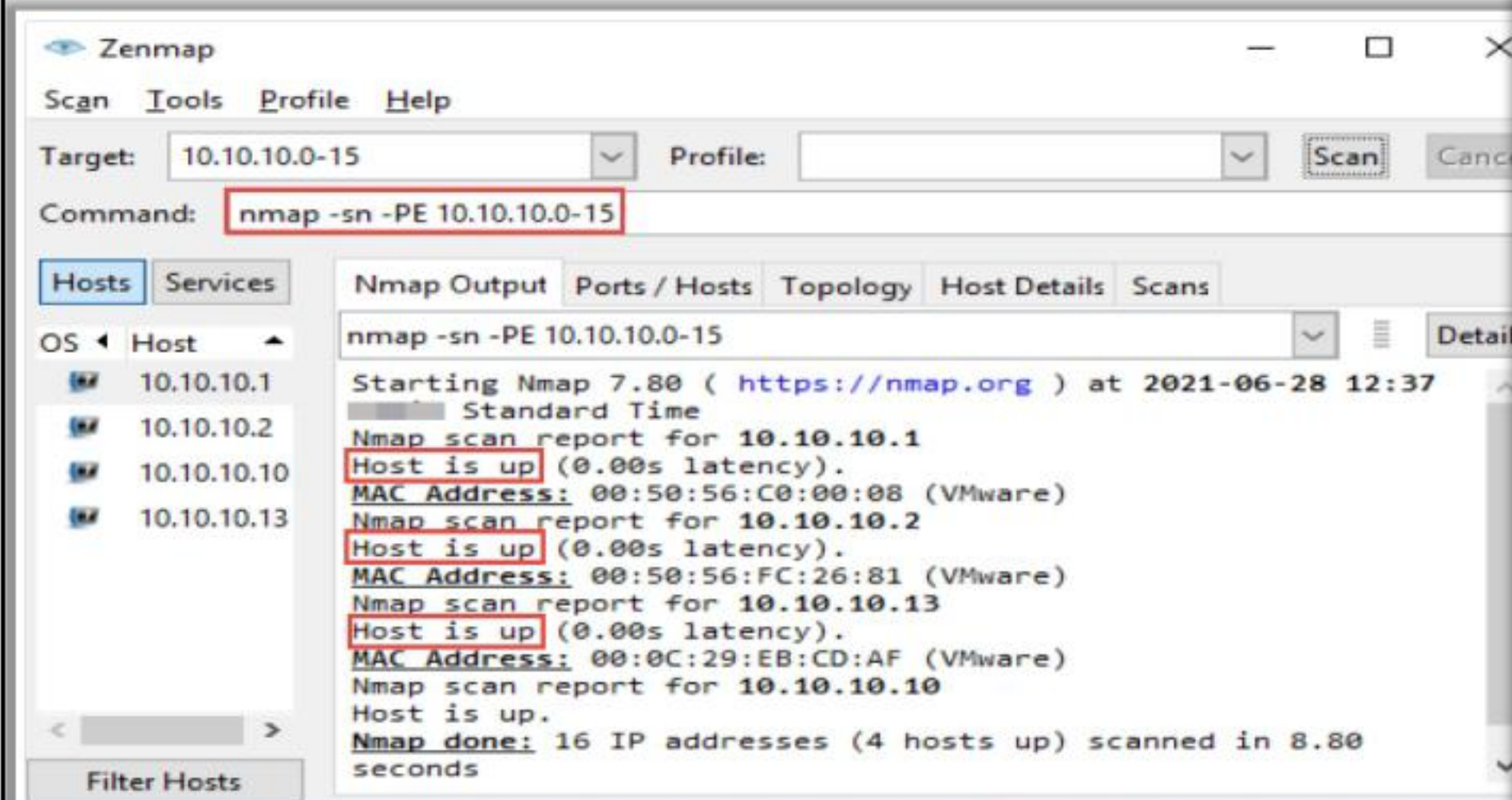


Nmap ("Network Mapper") is a security scanner for **network exploration**

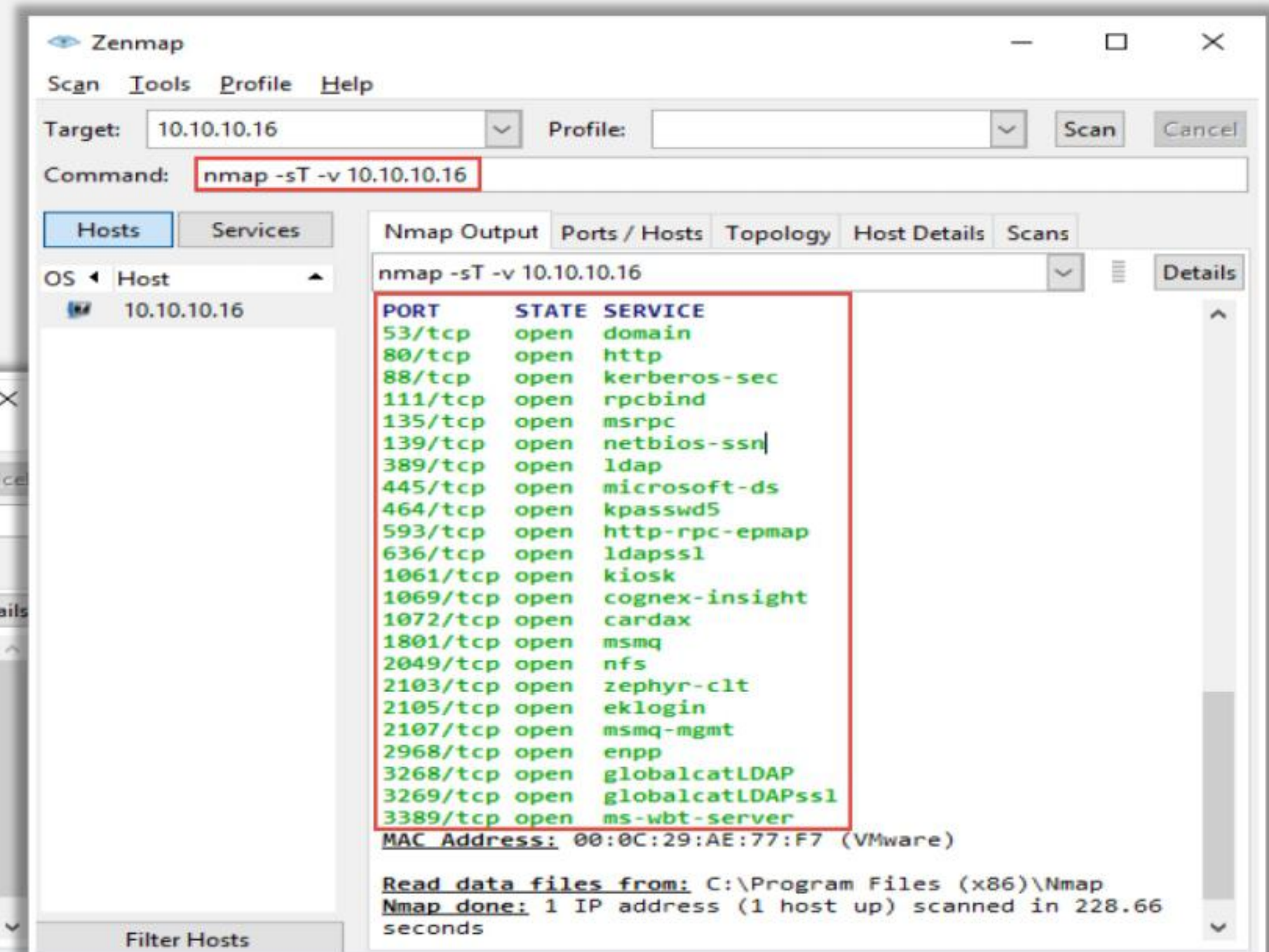


It allows the discovery of hosts, ports, and services on a computer network, thus creating a "map" of the network

Host Discovery



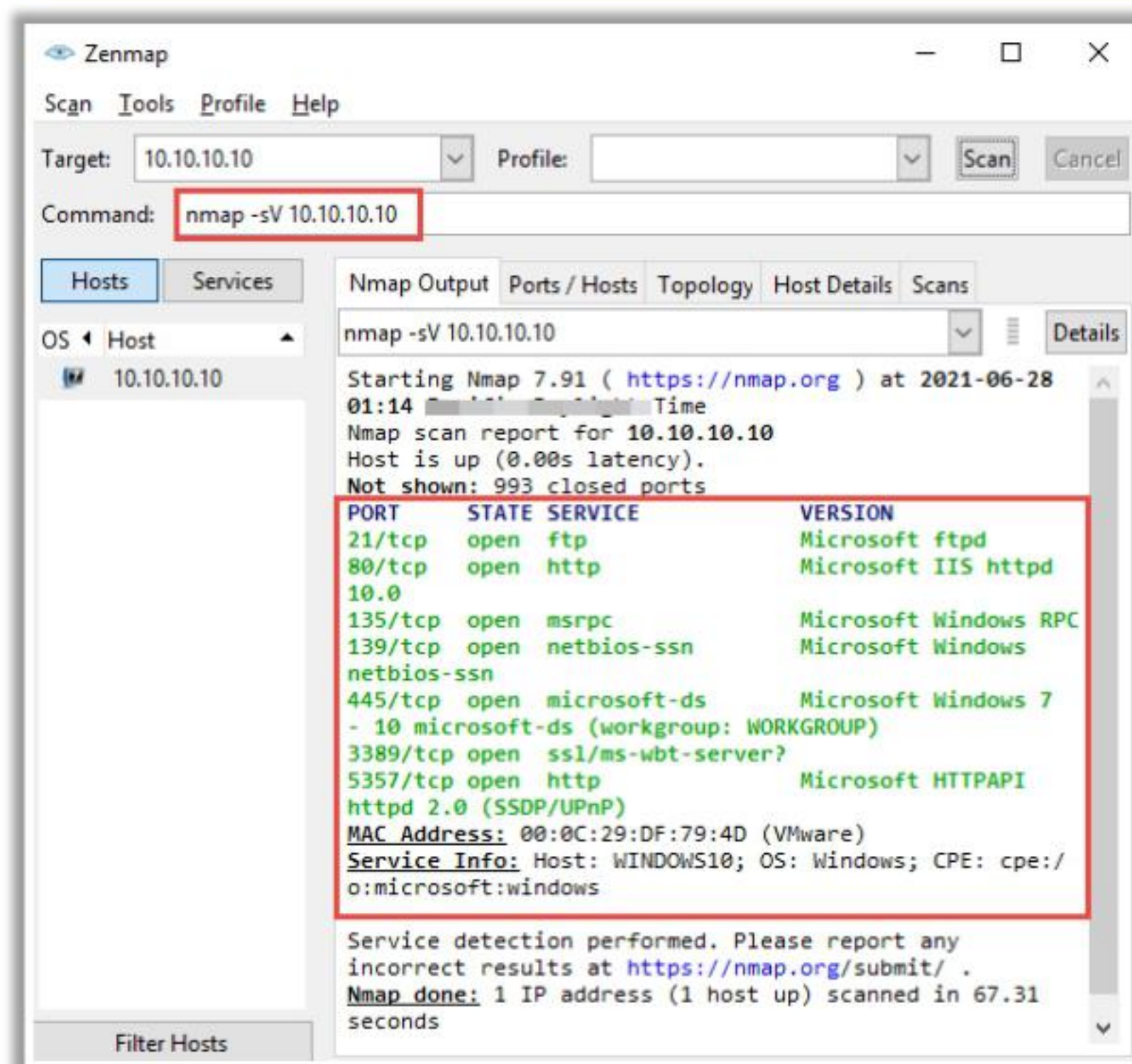
Service Discovery



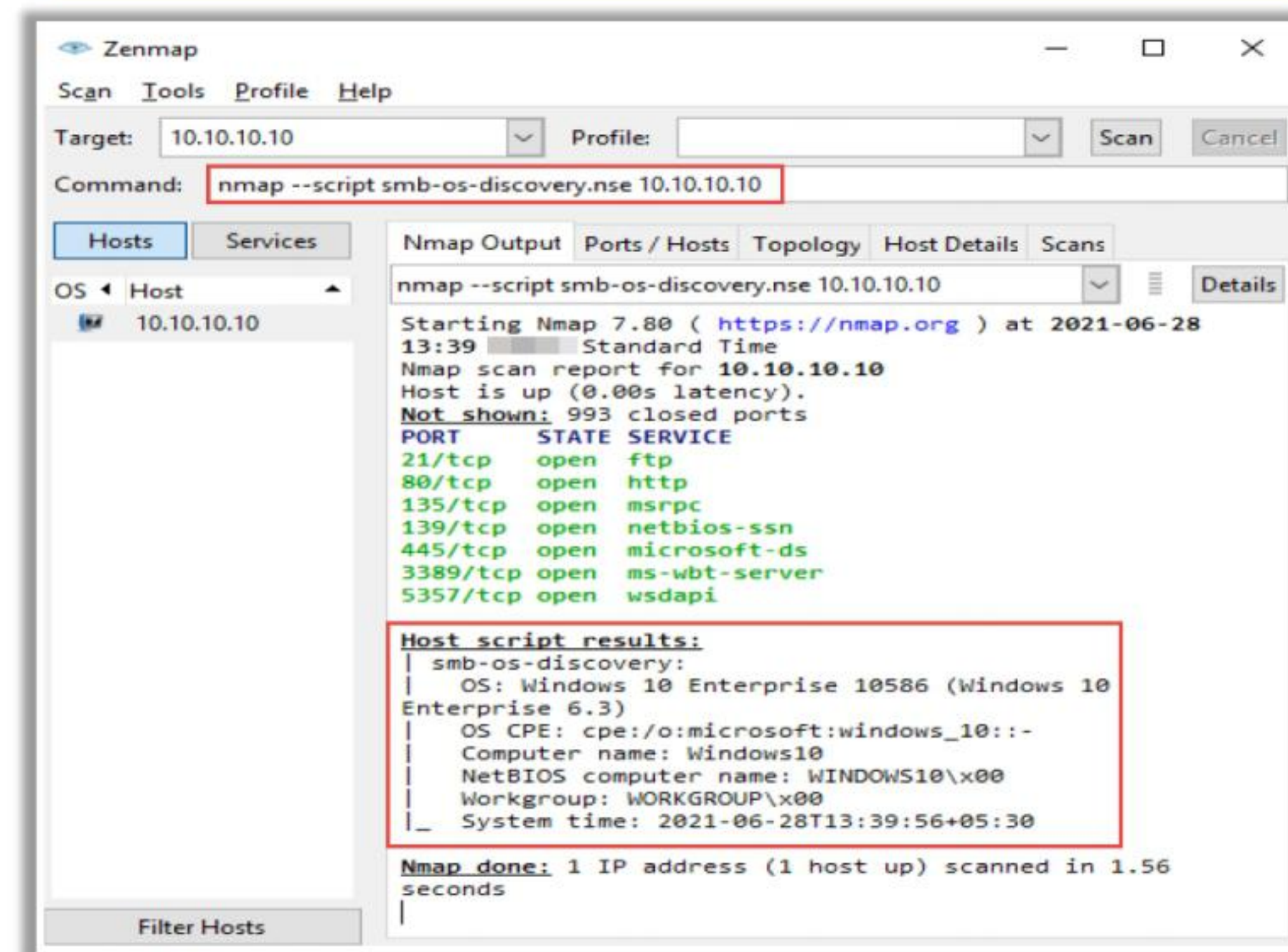
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: Nmap (Cont'd)

Service and Version Discovery



OS Discovery

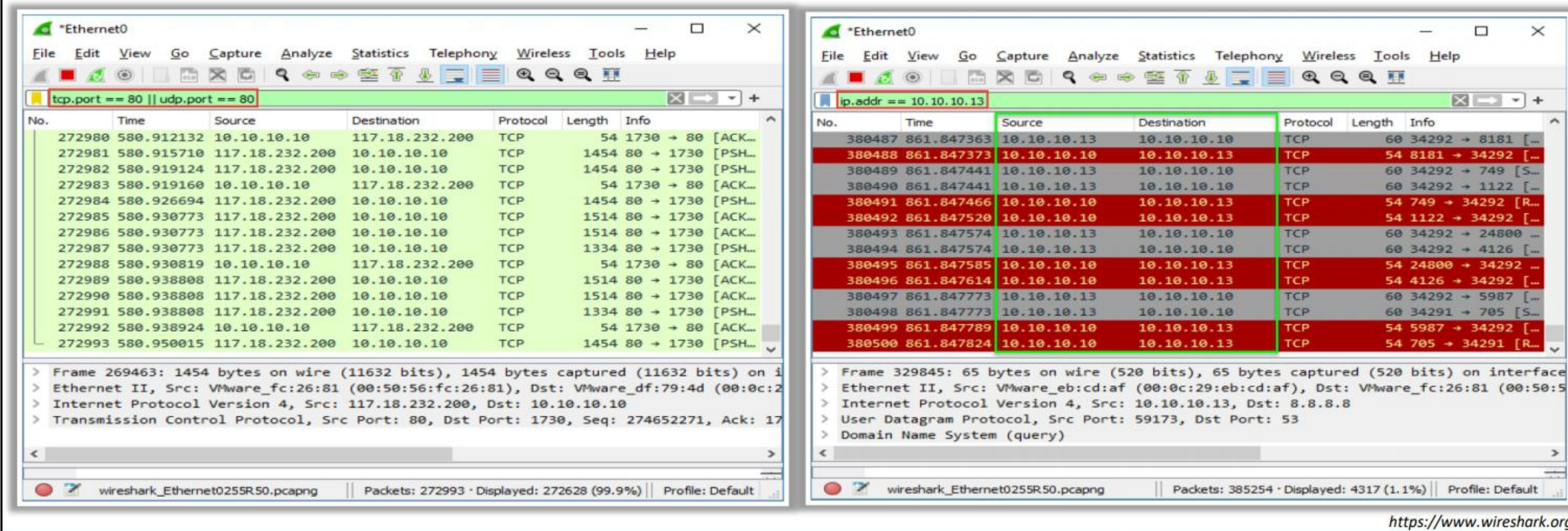


Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: Wireshark



- ❑ Wireshark enables **capturing** and **interactively browsing the traffic** in a computer network
- ❑ It captures **live network traffic** from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, and FDDI networks
- ❑ Wireshark features **display filters** that filter traffic on the target network by protocol type, IP address, port, etc.
- ❑ To set up a filter, enter the **protocol name**, such as arp, http, tcp, udp, dns, and ip, in the filter box of Wireshark



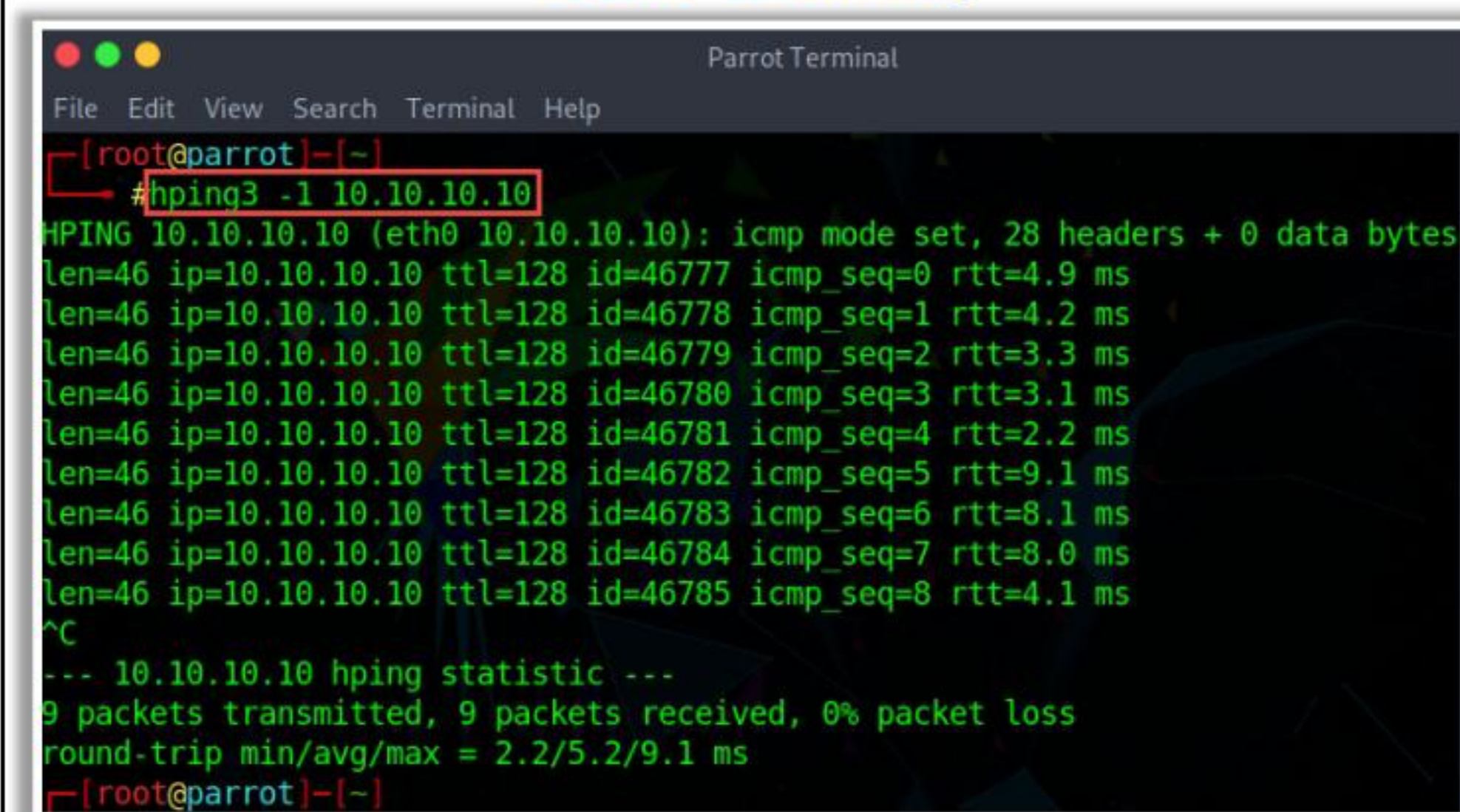
<https://www.wireshark.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

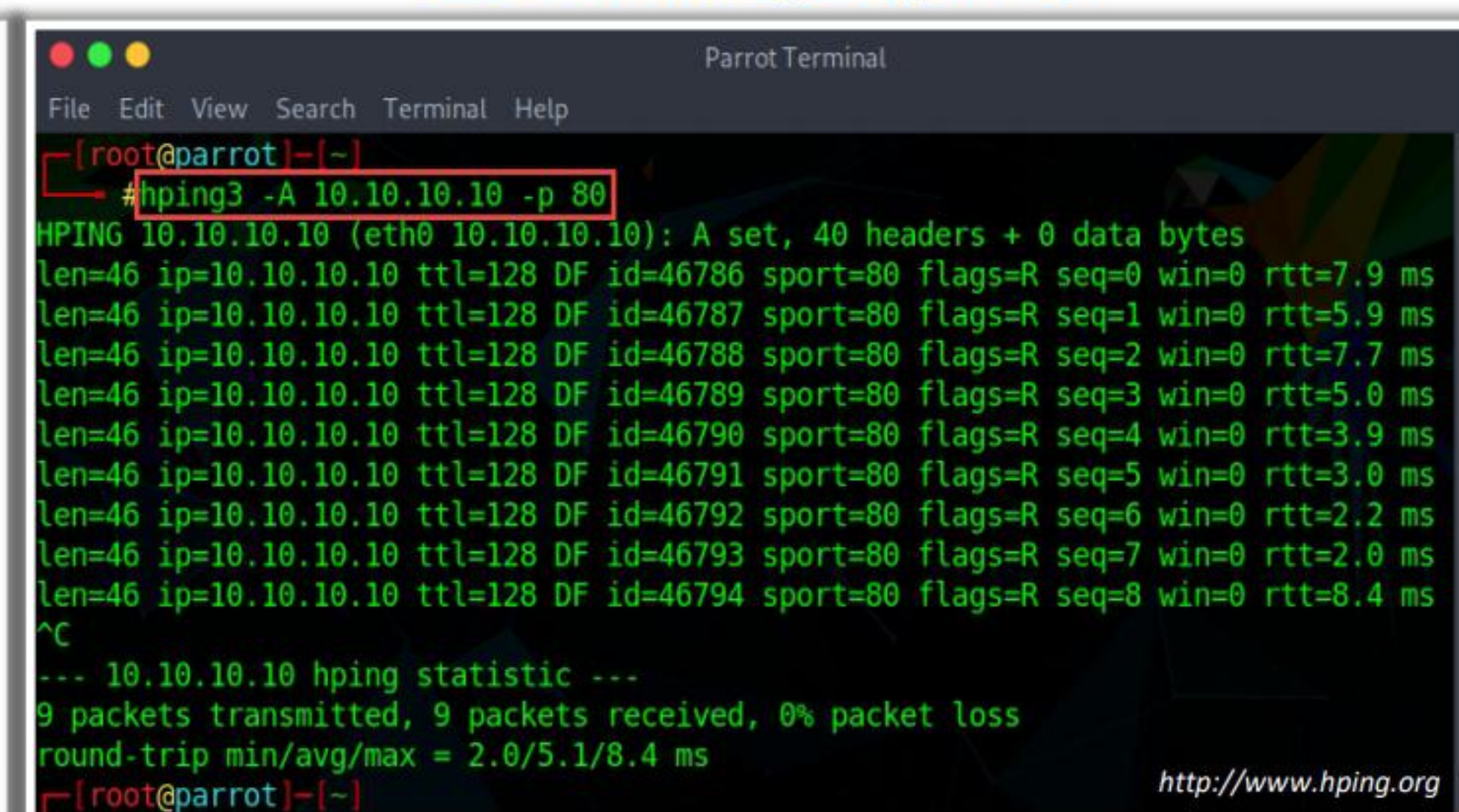
Network Troubleshooting Utilities and Tools: Hping2/Hping3

- 01** Hping2/Hping3 is command line **network scanning** and **packet crafting** tool for the TCP/IP protocol
- 02** It can be used for **network security auditing**, **firewall testing**, manual path MTU discovery, advanced traceroute, remote OS fingerprinting, remote uptime guessing, TCP/IP stacks auditing, etc.

ICMP Scanning



ACK Scanning on port 80



<http://www.hping.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: Hping2/Hping3 (Cont'd)



UDP scan on port 80

```
hping3 -2 10.0.0.25 -p 80
```



SYN scan on port 50-60

```
hping3 -8 50-60 -S  
10.0.0.25 -V
```



Collecting Initial Sequence Number

```
hping3 192.168.1.103 -Q  
-p 139 -s
```



FIN, PUSH and URG scan on port 80

```
hping3 -F -P -U 10.0.0.25  
-p 80
```



Firewalls and Timestamps

```
hping3 -S 72.14.207.99  
-p 80 --tcp-timestamp
```



Scan entire subnet for live host

```
hping3 -1 10.0.1.x --  
rand-dest -I eth0
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: netcat



- ❑ netcat is a network utility used for **testing network connections** on both Windows- and Linux-based systems
- ❑ It provides various commands to perform the thorough **monitoring of traffic flow** across a network
- ❑ It provides features such as **port scanning, OS fingerprinting, file transferring**, DNS checking, and source routing
- ❑ Test the connectivity with a remote host using the following command:

```
nc -w3 -4 -v <target host> <port number>
```
- ❑ Perform port scanning on a given IP address or host name using the following command:

```
nc -v -n <IP address> <port number/range>
```

```
ubuntu@ubuntu:~$ nc -v -n 10.10.10.10 21  
Connection to 10.10.10.10 21 port [tcp/*] succeeded  
220 Microsoft FTP Service
```

```
ubuntu@ubuntu:~$ nc -w3 -4 -v www.google.com 80  
Connection to www.google.com 80 port [tcp/http] succeeded!  
ubuntu@ubuntu:~$
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: dig

The dig command can be used on Linux-based systems to **query DNS servers** and **retrieve information** about target host addresses, name servers, and mail exchanges

Retrieve a specific DNS record

```
ubuntu@ubuntu:~$ dig ns certifiedhacker.com
; <<>> DiG 9.11.5-P1-1ubuntu2.6-Ubuntu <<>> ns certifiedhacker.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 11954
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;certifiedhacker.com.      IN      NS
;; ANSWER SECTION:
certifiedhacker.com.      5       IN      NS      ns1.bluehost.com.
certifiedhacker.com.      5       IN      NS      ns2.bluehost.com.
;; Query time: 129 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon Jun 28 05:17:34 PDT 2021
;; MSG SIZE rcvd: 93
ubuntu@ubuntu:~$
```

Test whether the target DNS allows zone transfers

```
ubuntu@ubuntu:~$ dig @ns1.bluehost.com certifiedhacker.com axfr
; <<>> DiG 9.11.5-P1-1ubuntu2.6-Ubuntu <<>> @ns1.bluehost.com certifiedhacker.com axfr
;; (1 server found)
;; global options: +cmd
;; Transfer failed.
ubuntu@ubuntu:~$
```



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: arp and tcpdump



A command-line tool used to **troubleshoot the ARP cache table** in network devices such as routers, switches, and other routing devices

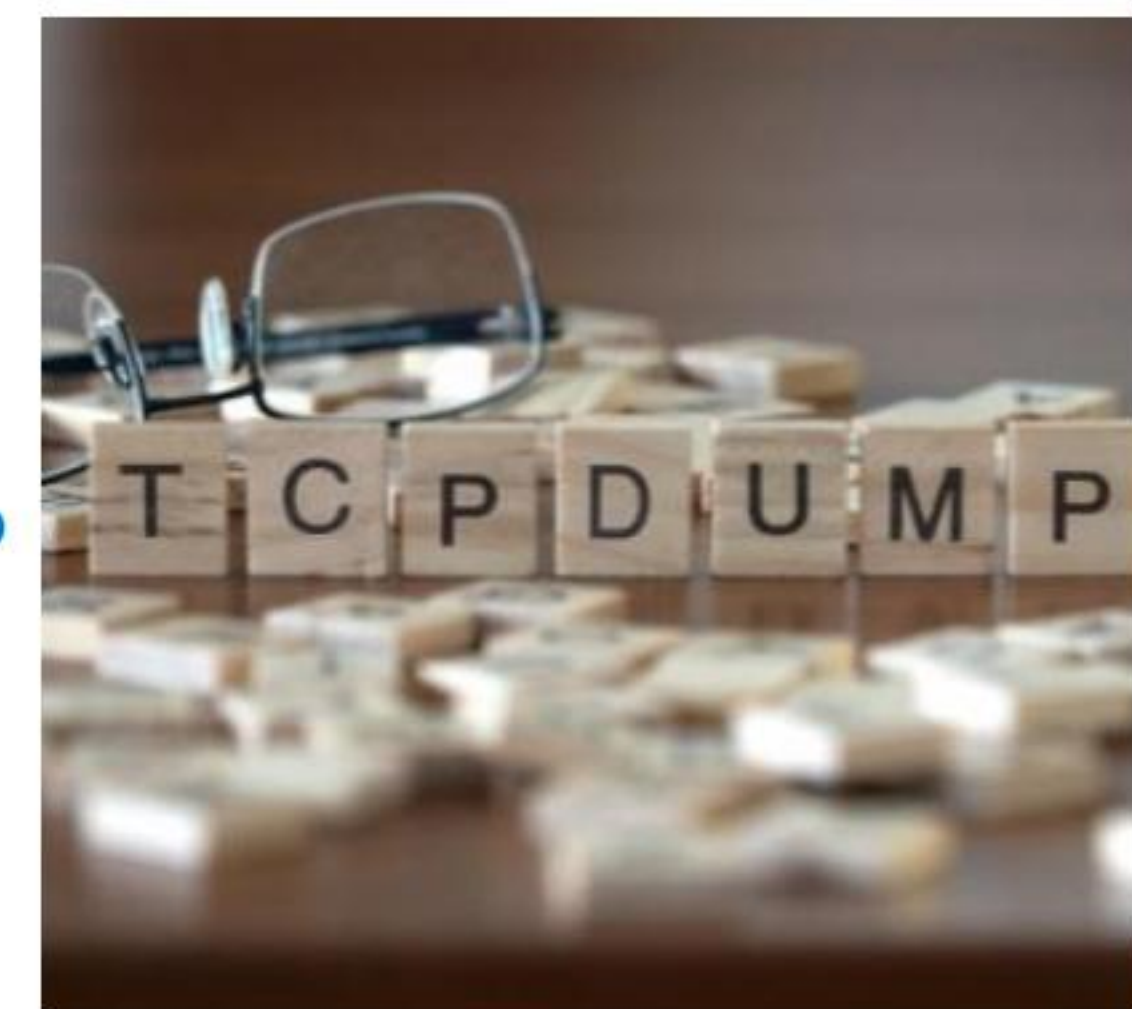


```
C:\Windows\System32\cmd.exe
C:\Windows\system32>arp -a

Interface: 10.10.10.10 --- 0x6
Internet Address      Physical Address      Type
10.10.10.2            00-50-56-fc-26-81    dynamic
10.10.10.255          ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.252           01-00-5e-00-00-fc    static
224.0.0.253           01-00-5e-00-00-fd    static
239.255.255.250       01-00-5e-7f-ff-fa    static
```

```
root@ubuntu: /home/ubuntu
root@ubuntu: /home/ubuntu# tcpdump -i any -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
05:49:36.801995 ARP, Request who-has _gateway (Broadcast) tell 10.10.10.10, length 46
05:49:36.809197 IP localhost.54838 > localhost.domain: 7288+ [1au] PTR? 2.10.10.10.in-addr.arpa. (52)
05:49:36.809413 IP ubuntu.60436 > _gateway.domain: 59922+ [1au] PTR? 2.10.10.10.in-addr.arpa. (52)
3 packets captured
29 packets received by filter
19 packets dropped by kernel
root@ubuntu: /home/ubuntu#
```

A network packet capturing tool used for **analyzing network traffic** and troubleshooting network issues



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools: tcpreplay and dnsenum

tcpreplay



A GPLv3 licensed utility that supports Unix-like operating systems for **modifying** and **replaying previously sniffed traffic** from tools such as Wireshark and tcpdump

```
Parrot Terminal
File Edit View Search Terminal Help
[*]-[root@parrot]~/
#tcpreplay -i eth0 mySample.pcapng
Warning in send_packets.c:send_packets() line 644:
Unable to send packet: Error with PF_PACKET send() [43]: Message
too long (errno = 90)
Actual: 42 packets (9244 bytes) sent in 12.40 seconds
Rated: 744.9 Bps, 0.005 Mbps, 3.38 pps
Statistics for network device: eth0
Successful packets: 42
Failed packets: 1
Truncated packets: 0
Retried packets (ENOBUFS): 0
Retried packets (EAGAIN): 0
```



dnsenum



A Perl script that **enumerates the DNS information** of a domain to discover noncontiguous IP blocks

```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~/
#dnsenum --enum google.com
Smartmatch is experimental at /usr/bin/dnsenum line 698.
Smartmatch is experimental at /usr/bin/dnsenum line 698.
dnsenum VERSION:1.2.4
Warning: can't load Net::Whois:IP module, whois queries disabled.
Warning: can't load WWW::Mechanize module, Google scraping disabled.
----- google.com -----
Host's addresses:
google.com. 202 IN A 142.250.193.174
Name Servers:
ns2.google.com. 21414 IN A 216.239.34.10
ns1.google.com. 18366 IN A 216.239.32.10
ns4.google.com. 21599 IN A 216.239.38.10
ns3.google.com. 21599 IN A 216.239.36.10
Mail (MX) Servers:
alt4.aspx.l.google.com. 292 IN A 64.233.171.27
alt3.aspx.l.google.com. 292 IN A 142.250.115.27
alt1.aspx.l.google.com. 292 IN A 173.194.202.27
aspx.l.google.com. 292 IN A 74.125.200.27
alt2.aspx.l.google.com. 292 IN A 142.250.141.27
```

<https://github.com>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Utilities and Tools

- Ping

The ping utility is used to test whether an IP address or a website is accessible by a host. Ping works by sending ICMP echo requests to the targeted host and waiting for ICMP echo replies. It measures the Internet speed and reports errors and losses of data packets. A reply from the pinged IP address indicates that the packets are transferring between the system and given IP. Launch the command prompt and execute **ping x.x.x.x** or **ping example.com** to check the availability of the host to the computer.

Network Troubleshooting Using Ping

The primary step is to ping the remote host. If the user has trouble connecting to a website, then the user must ping the URL. If the ping is returned, then the network is working properly, and the issues lie somewhere else.

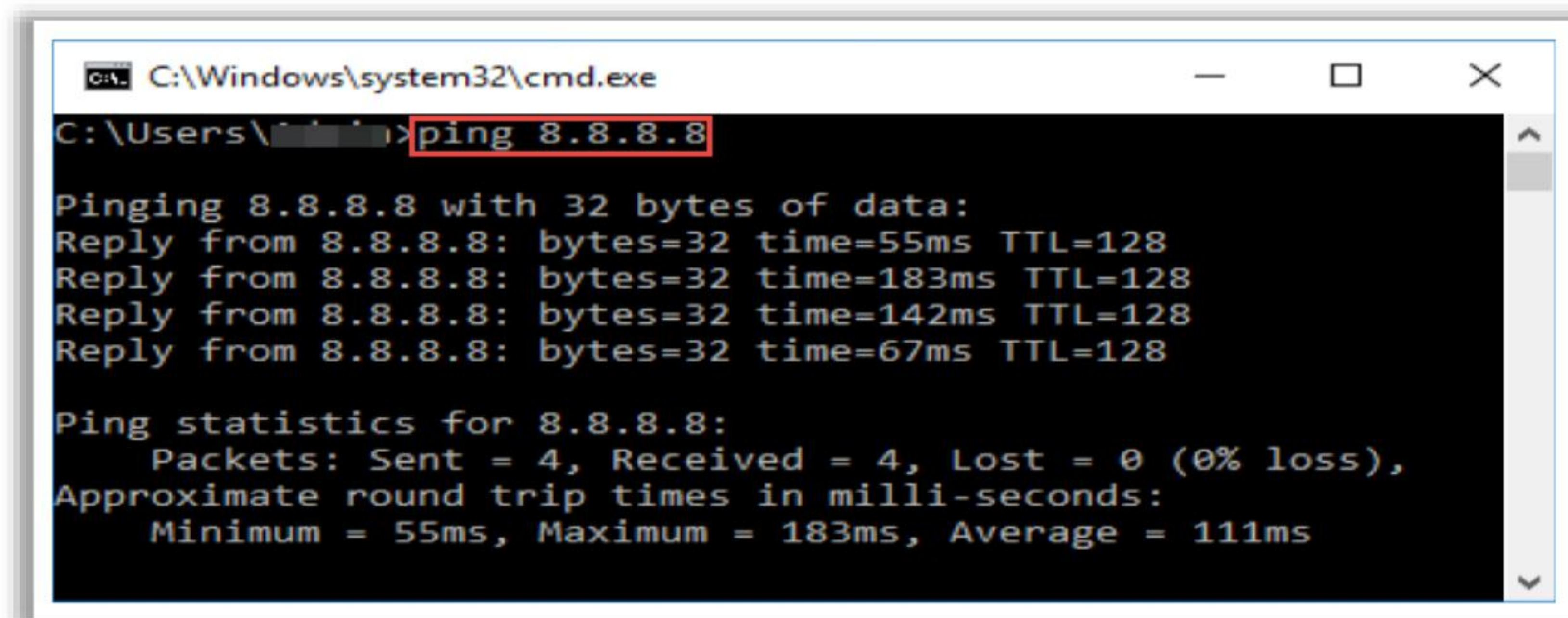
```
C:\Windows\system32\cmd.exe
C:\Users\>ping www.google.com
Pinging www.google.com [142.250.182.36] with 32 bytes of data:
Reply from : bytes=32 time=207ms TTL=128
Reply from : bytes=32 time=65ms TTL=128
Reply from : bytes=32 time=60ms TTL=128
Reply from : bytes=32 time=66ms TTL=128

Ping statistics for 142.250.182.36:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 60ms, Maximum = 207ms, Average = 99ms
```

Figure 16.50: Ping a Website

The four replies in the screenshot above indicate that the network connection is good and the server is reachable. In this scenario, the fault could lie with the web-server configuration. If the ping is missed, then there might be a problem with the network. If the replies have a wide variation in the maximum and minimum time of the ping, then there could be connection issues such as network congestion.

To test connectivity, some administrators ping Google's primary DNS server (ping 8.8.8.8) because it is easy to remember and is continuously running.



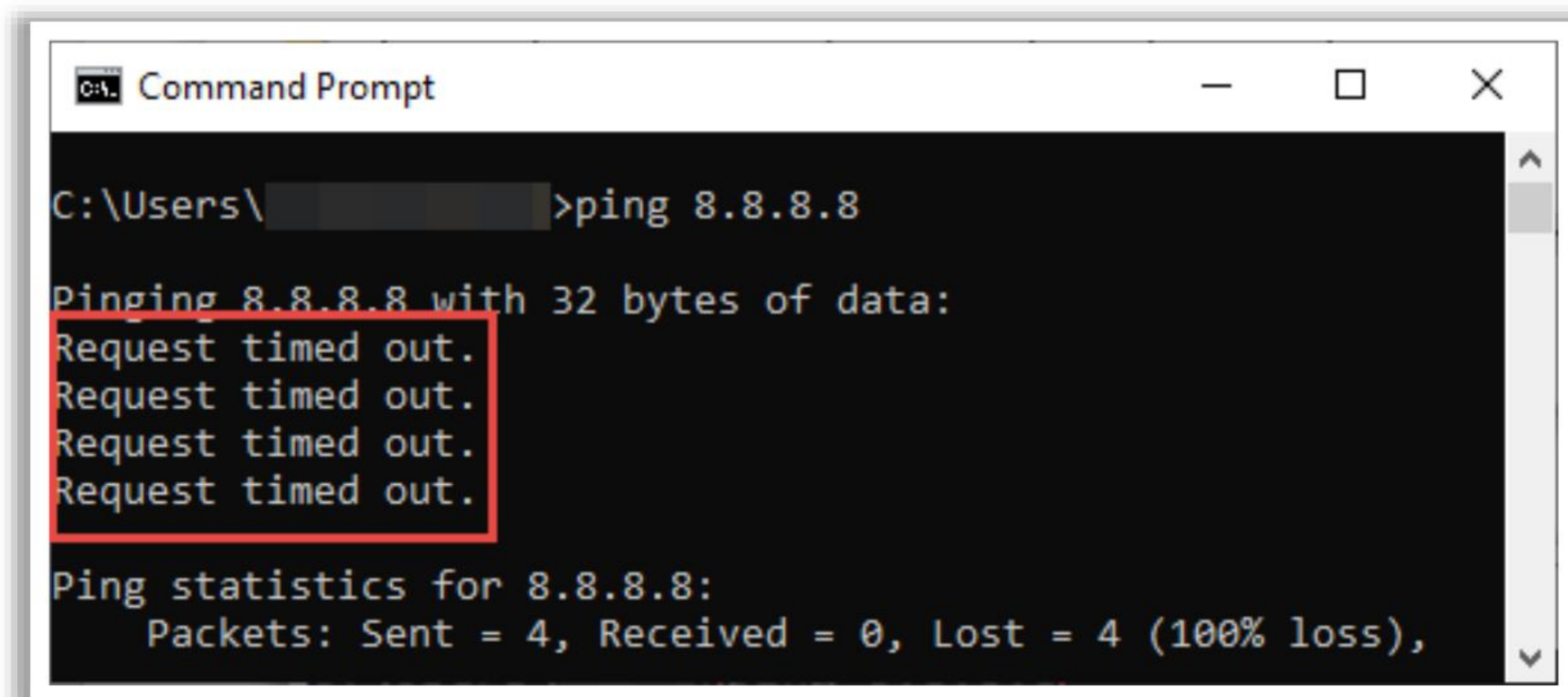
```
C:\Windows\system32\cmd.exe
C:\Users\>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=55ms TTL=128
Reply from 8.8.8.8: bytes=32 time=183ms TTL=128
Reply from 8.8.8.8: bytes=32 time=142ms TTL=128
Reply from 8.8.8.8: bytes=32 time=67ms TTL=128

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 55ms, Maximum = 183ms, Average = 111ms
```

Figure 16.51: Ping Google's Primary DNS Server

The message "Request timed out" in the screenshot above shows that there is no connection between the system and host, or the system is unable to connect to the host. It also indicates that the host might be down or unreachable, the host might be behind a firewall, or the user does not have network connectivity.



```
Command Prompt
C:\Users\>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Figure 16.52: No Connection between a System and a Host

- **tracert/tracert**

The multi-OS-compatible command-line tool trace route (tracert)/traceroute is used to trace packets across a network and to understand connections to a server. It allows the user to understand Internet connection problems, including packet loss and high latency.

tracert (for Windows) uses ICMP. It sends ICMP echo request messages to the specified destination. If the destination is active, it sends ICMP echo reply messages as a response, confirming that the connection is active. Otherwise, the destination may not be active, or it could be a connectivity issue of the source.

Steps to Use tracert (for Windows)

- Run the traceroute command with example.com (any website):

```
tracert example.com
```

```
C:\Users\New User>tracert google.com

Tracing route to google.com [208.43.115.82]
over a maximum of 30 hops:
  0  0 ms  0 ms  0 ms  192.168.1.254
  1  3 ms  4 ms  2 ms  192.168.1.254
  2  13 ms  9 ms  7 ms  10.246.112.1
  3  10 ms  8 ms  8 ms  96.1.253.134
  4  11 ms  9 ms  13 ms  173.182.214.134
  5  * * * Request timed out.
  6  15 ms  11 ms  12 ms  75.154.217.103
  7  13 ms  12 ms  13 ms  te1-5.bbr01.wb01.sea01.networklayer.com [206.81.
80.140]
  8  49 ms  47 ms  48 ms  ae0.bbr01.cs01.den01.networklayer.com [173.192.1
8.145]
  9  49 ms  48 ms  48 ms  ae7.bbr02.cs01.den01.networklayer.com [173.192.1
8.169]
 10  67 ms  66 ms  97 ms  ae0.bbr02.eq01.chi01.networklayer.com [173.192.1
8.130]
 11  177 ms  83 ms  83 ms  ae0.bbr02.eq01.wdc02.networklayer.com [173.192.1
8.154]
 12  94 ms  82 ms  83 ms  ae1.dar01.sr01.wdc01.networklayer.com [173.192.1
8.193]
 13  84 ms  85 ms  84 ms  po1.fcr01.sr01.wdc01.networklayer.com [208.43.11
8.134]
 14  85 ms  84 ms  84 ms  google.com [208.43.115.82]

Trace complete.
```

Figure 16.53: Run the Command “tracert”

Observe the route take form as the system receives responses from the routers along the way.

- Run the tracert command for another website hosted in different regions of the world. Observe how the paths differ.

```
C:\Users\New User>tracert google.com

Tracing route to google.com [123.125.114.144]
over a maximum of 30 hops:
  0  0 ms  0 ms  0 ms  192.168.1.254
  1  2 ms  1 ms  1 ms  192.168.1.254
  2  12 ms  13 ms  9 ms  10.246.112.1
  3  9 ms  8 ms  12 ms  96.1.253.134
  4  11 ms  8 ms  10 ms  173.182.214.134
  5  * * * Request timed out.
  6  42 ms  35 ms  46 ms  154.11.10.165
  7  36 ms  36 ms  36 ms  219.158.33.249
  8  186 ms  182 ms  178 ms  219.158.30.253
  9  180 ms  180 ms  177 ms  219.158.19.193
 10  190 ms  192 ms  196 ms  219.158.23.17
 11  216 ms  215 ms  216 ms  219.158.101.121
 12  227 ms  232 ms  229 ms  123.126.0.70
 13  212 ms  209 ms  213 ms  ht-227-018.hta.net.cn [202.106.227.18]
 14  232 ms  231 ms  227 ms  202.106.43.66
 15  * * * Request timed out.
 16  * * * Request timed out.
 17  229 ms  230 ms  229 ms  123.125.114.144

Trace complete.
```

Figure 16.54: Run the Command “tracert”

In the above screenshot, the first line represents the home router, assuming the user is behind a router. The remaining lines represent the ISP. The format of each line is as follows:

Hop RTT1 RTT2 RTT3 Domain Name [IP Address]

Hop: When a packet passes by a router, it is said to have performed a “hop.”

RTT1, RTT2, or RTT3: Also referred to as “latency,” it is the round-trip time that a packet takes to perform a hop and return to the system (in milliseconds).

*****: This output is produced when no response is received, indicating packet loss.

Domain name (IP address): The domain name allows the user to determine the location of a router. If it is not available, only the IP address of the router is displayed.

Steps to traceroute for *nix Systems

Traceroute uses UDP on typical *nix systems and sends traffic to port 33434 by default:

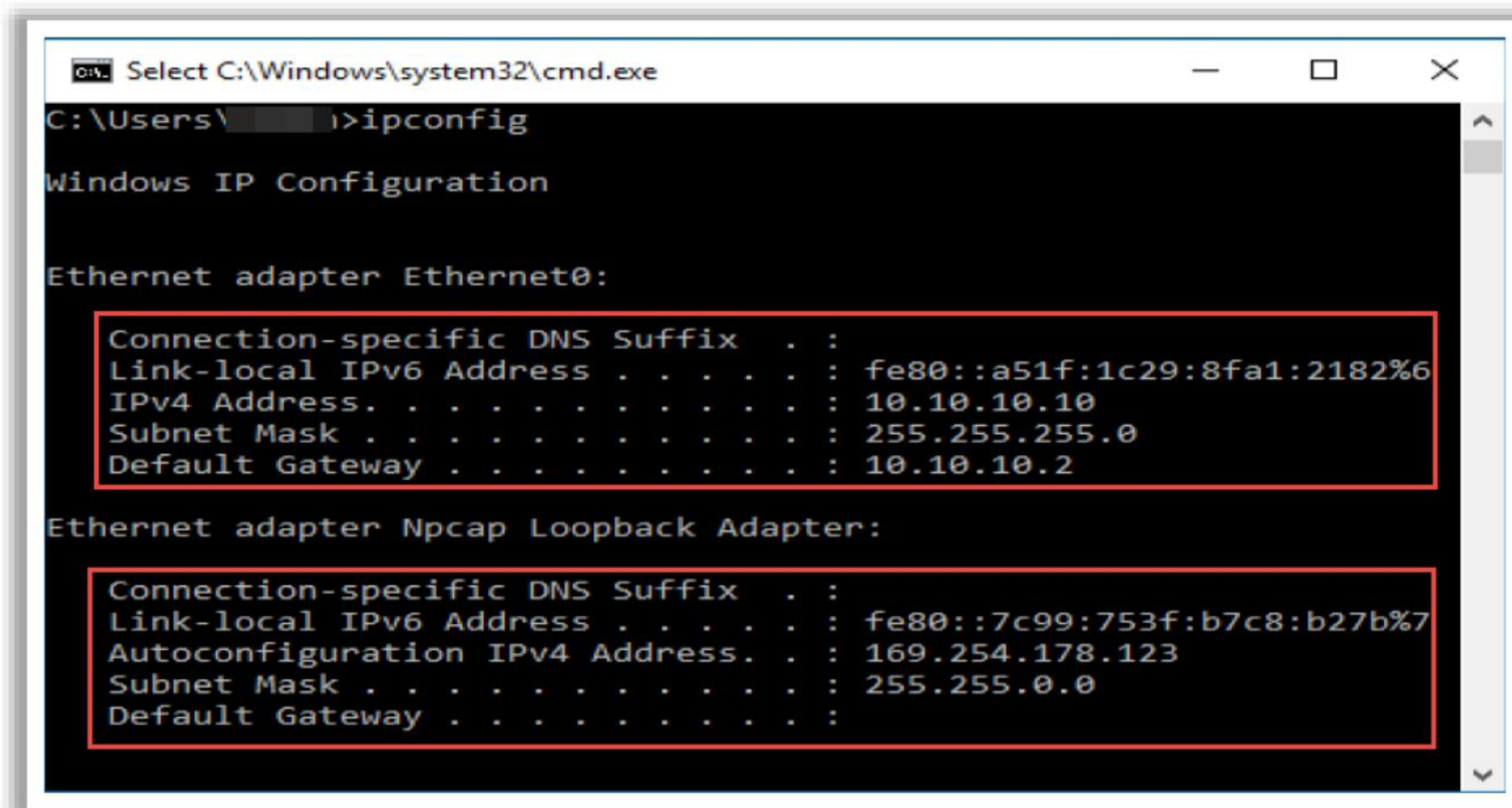
traceroute example.com

```
$ traceroute google.com
traceroute to google.com (172.217.10.46), 64 hops max, 52 byte packets
 1  192.168.1.1 (192.168.1.1)  1747.782 ms  1.812 ms  4.232 ms
 2  10.170.2.1 (10.170.2.1)  10.838 ms  12.883 ms  8.510 ms
 3  xx.xx.xx.xx (xx.xx.xx.xx)  10.588 ms  10.141 ms  10.652 ms
 4  xx.xx.xx.xx (xx.xx.xx.xx)  14.965 ms  16.702 ms  18.275 ms
 5  xx.xx.xx.xx (xx.xx.xx.xx)  15.092 ms  16.910 ms  17.127 ms
 6  108.170.248.97 (108.170.248.97)  13.711 ms  14.363 ms  11.698 ms
 7  216.239.62.171 (216.239.62.171)  12.802 ms
    216.239.62.169 (216.239.62.169)  12.647 ms  12.963 ms
 8  lga34s13-in-f14.1e100.net (172.217.10.46)  11.901 ms  13.666 ms
11.813 ms
```

Figure 16.55: Use “traceroute” for *nix Systems

- **ipconfig/ifconfig**

ipconfig: ipconfig is a command-line utility used to display all current TCP/IP network configuration values along with the IP address, subnet mask, and default gateway for all adapters. To display the basic configuration of the system, use ipconfig in the command prompt. For detailed information on the system configuration, execute **ipconfig /all** in the command prompt. ifconfig is a similar utility but for Linux-based machines.



```
Select C:\Windows\system32\cmd.exe
C:\Users\>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix . : 
    Link-local IPv6 Address . . . . . : fe80::a51f:1c29:8fa1:2182%6
    IPv4 Address. . . . . : 10.10.10.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 10.10.10.2

Ethernet adapter Npcap Loopback Adapter:

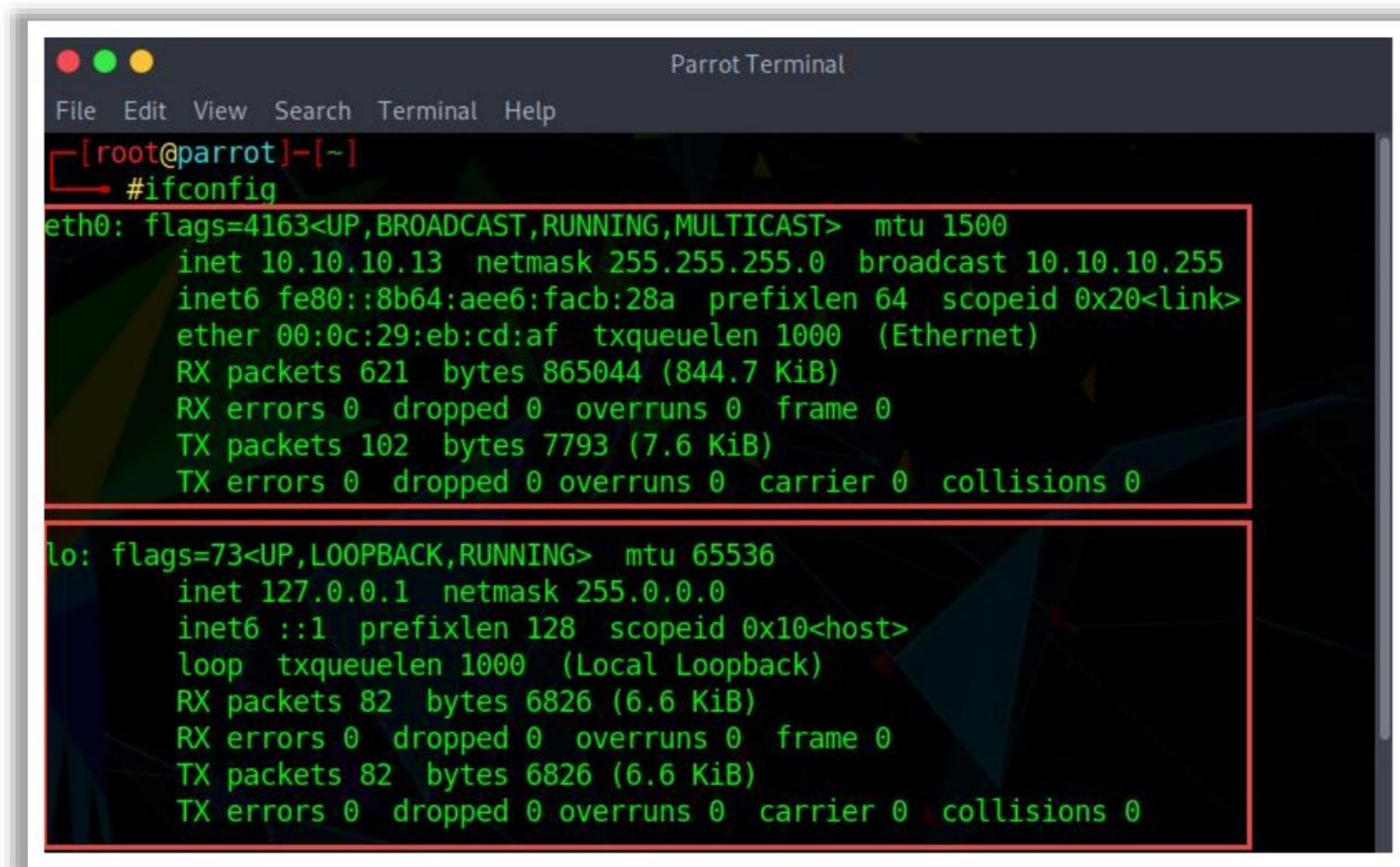
    Connection-specific DNS Suffix . : 
    Link-local IPv6 Address . . . . . : fe80::7c99:753f:b7c8:b27b%7
    Autoconfiguration IPv4 Address. . : 169.254.178.123
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . :
```

Figure 16.56: Using ipconfig in Windows System

ifconfig: When troubleshooting a networking issue, the Linux- or Unix-based OS utility interface configuration (ifconfig) helps display the specific IP address of the affected hosts, netmask of a network interface, and maximum transmission unit (MTU). This utility provides commands to configure and enable/disable a network interface.

Steps to Use ifconfig

- To display the network settings of all the active network interfaces on the system, use ifconfig without any options.

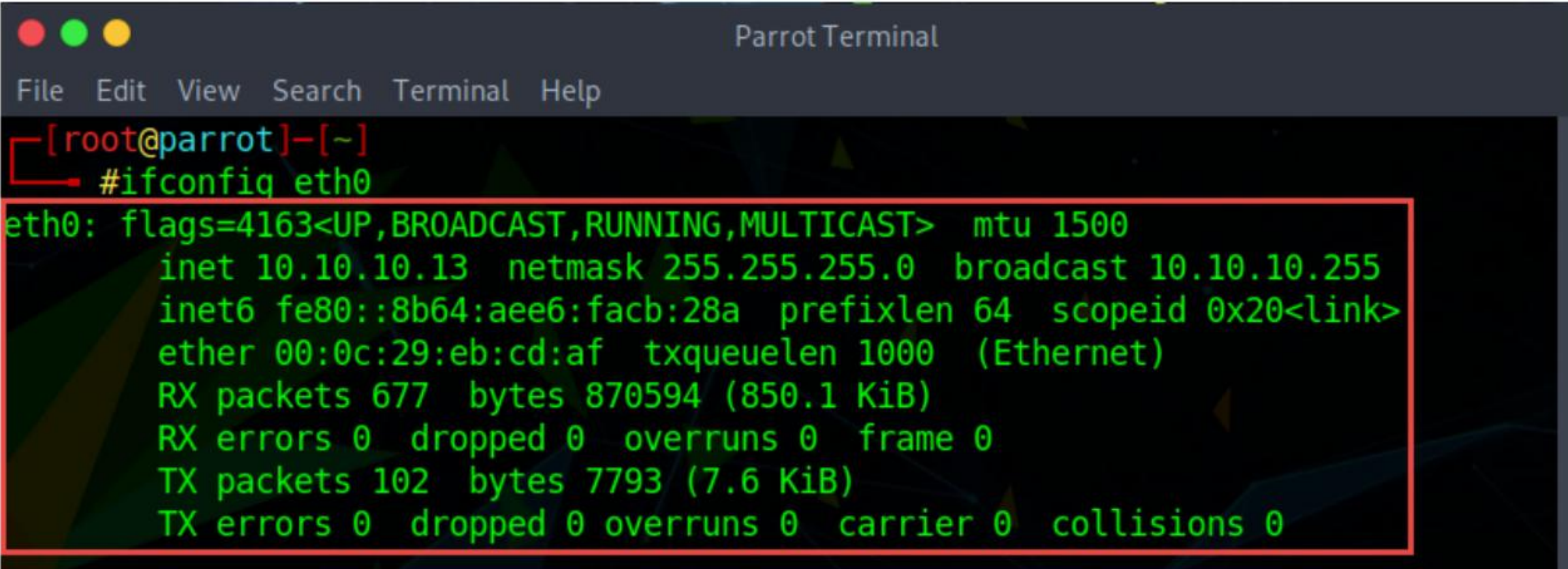


```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]-[~]
#ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.10.13 netmask 255.255.255.0 broadcast 10.10.10.255
    inet6 fe80::8b64:aee6:facb:28a prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:eb:cd:af txqueuelen 1000 (Ethernet)
    RX packets 621 bytes 865044 (844.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 102 bytes 7793 (7.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 82 bytes 6826 (6.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 82 bytes 6826 (6.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 16.57: Display Network Settings of a Network Interface

- To display the details of all network interfaces, use the command as shown in the screenshot below.
- Use the interface (eth0) command to display only specific interface details such as the IP address and MAC address.



```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]-[~]
#ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.10.13 netmask 255.255.255.0 broadcast 10.10.10.255
    inet6 fe80::8b64:aee6:facb:28a prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:eb:cd:af txqueuelen 1000 (Ethernet)
    RX packets 677 bytes 870594 (850.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 102 bytes 7793 (7.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 16.58: Display any Specific Interface Details

- To assign an IP address and gateway to interface on the fly, use the following command:
ifconfig eth0 192.168.50.5 netmask 255.255.255.0
The setting will be removed if system reboots.
- To enable a specific interface, use the following command:
ifup eth0
- To disable a specific Interface, use the following command:
ifdown eth0
- To set the required MTU, use the following command, where XXXX represents the size:
ifconfig eth0 mtu XXXX
- To set the interface in the promiscuous mode to capture all the packets and to analyze them later, use the following command:
ifconfig eth0 - promisc
- To configure an IP address, specify the interface to be configured, IP address, and subnet address.


```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~
#ifconfig eth0 5.5.5.5 netmask 255.255.255.0
[root@parrot]~
#ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 5.5.5.5 netmask 255.255.255.0 broadcast 5.5.5.255
    inet6 fe80::8b64:aee6:facb:28a prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:eb:cd:af txqueuelen 1000 (Ethernet)
    RX packets 717 bytes 873954 (853.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 102 bytes 7793 (7.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 82 bytes 6826 (6.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 82 bytes 6826 (6.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 16.59: Display an Interface, IP, and Subnet Address

- To enable/disable an interface, perform the following:
 - Use the “**up**” or “**ifup**” flag with the interface name (for example, eth0) that will activate a network interface.
 - Use the “**down**” or “**ifdown**” flag with the interface name that will deactivate the specified network interface.

```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~
#ifconfig eth0 down
[root@parrot]~
#ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 82 bytes 6826 (6.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 82 bytes 6826 (6.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@parrot]~
#ifconfig eth0 up
[root@parrot]~
#ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.10.13 netmask 255.255.255.0 broadcast 10.10.10.255
    inet6 fe80::8b64:aee6:facb:28a prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:eb:cd:af txqueuelen 1000 (Ethernet)
    RX packets 754 bytes 877470 (856.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 115 bytes 8822 (8.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 88 bytes 7060 (6.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 88 bytes 7060 (6.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 16.60: Enable/Disable an Interface

- **nslookup**

nslookup is a program that allows the administrator or system user to enter a host name and retrieve the corresponding IP address or DNS record. It is also used for reverse DNS lookup to find the host name for a given IP address. The nslookup utility is used to look up a specific IP address or multiple IP addresses associated with a domain name(s) at a time. nslookup is used when a user can access a resource by specifying its IP address but not by specifying its domain name. nslookup safeguards against phishing attacks and prevents cache poisoning.

The nslookup utility is used to resolve DNS address resolution issues. The nslookup command is executed in the command prompt to look up the IP address for a domain name. Subcommands can be used at the end of the nslookup command to perform queries or to set options. The optimal mail servers SMTP, Post Office Protocol (POP), and Internet Message Access Protocol (IMAP) for the desired domain can also be searched using nslookup.

Searching for the Domain Name Using nslookup

The user should enter the domain name into the command line to find the IP address or vice versa. nslookup gives the results shown in the screenshot below for google.com.

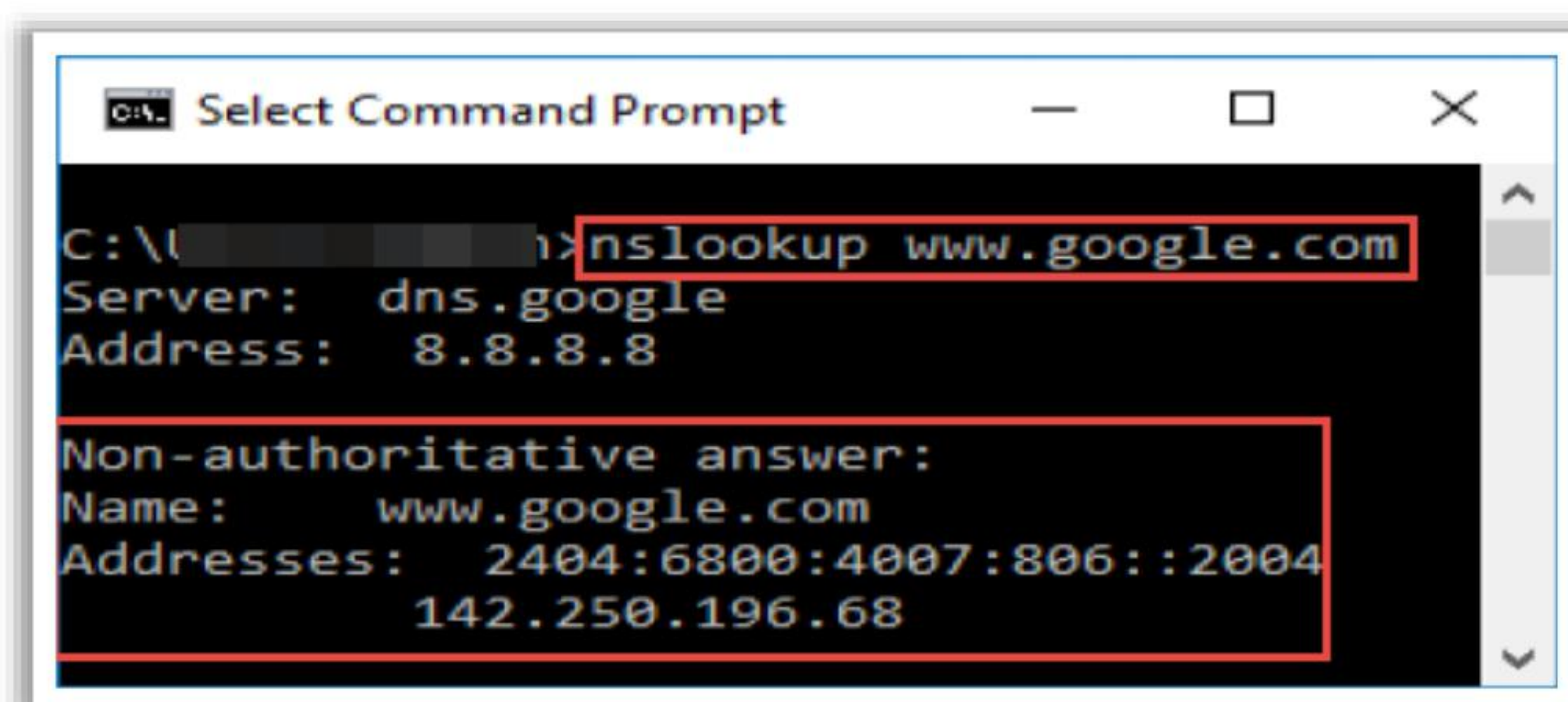


Figure 16.61: Search Domain Name using "nslookup" Command

The notification **Non-authoritative answer** indicates that the local DNS server failed to provide an answer to the query itself and contacted other name servers. The results of nslookup consist of IPv4 (four-figure) and IPv6 addresses (long and dived with colons) of the google domain.

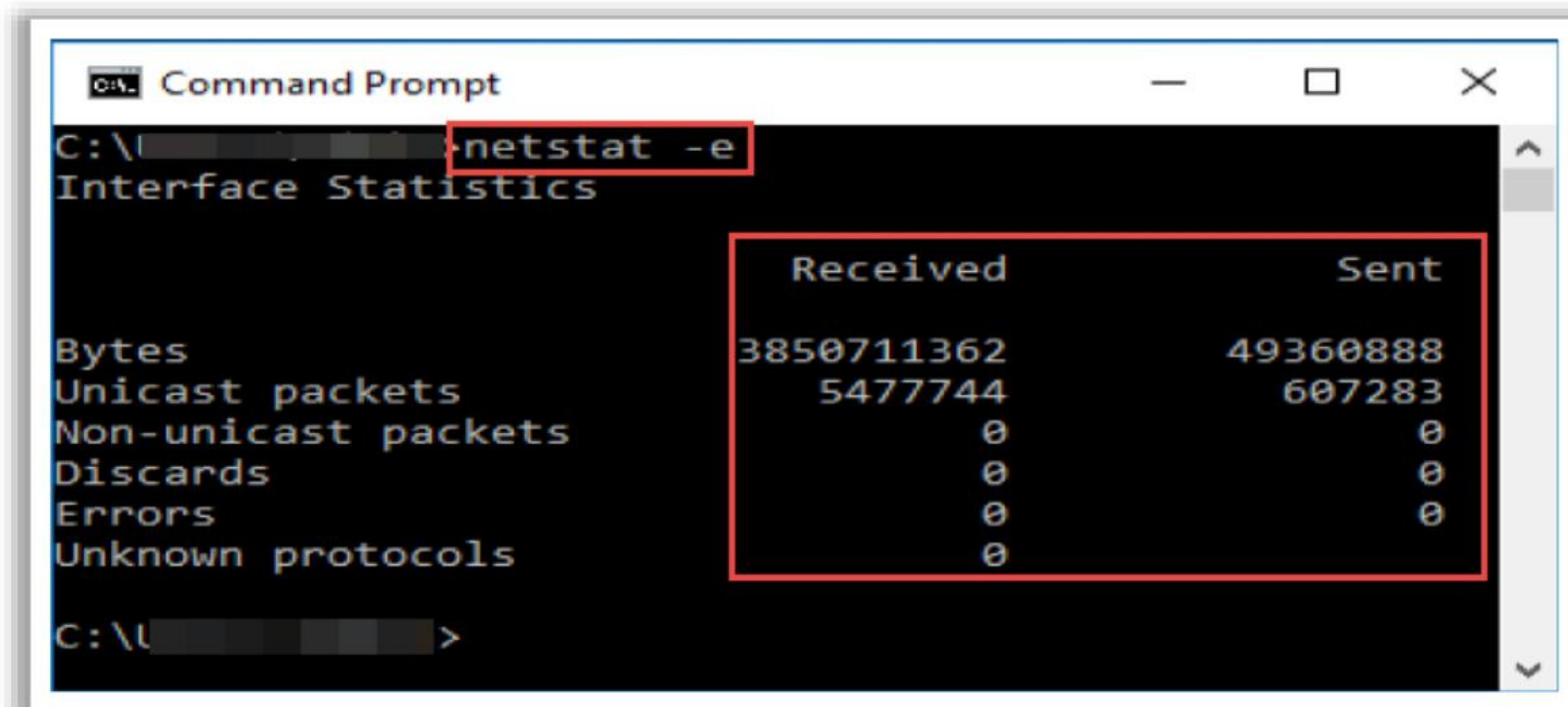
- **netstat**

The Linux/Windows utility Network Statistics (netstat) displays network connections (incoming and outgoing), network statistics, protocol statistics, and routing tables. It also displays connections that are not established properly and those that are being ended, and it helps solve slowdowns, bottlenecks, or outage problems in networks.

Steps to Use netstat

Follow the steps below to list various listening ports.

- Execute the netstat command without any parameters in the terminal to show the list of active connections.
- Use the `netstat -e` command to show the statistics of various protocols.

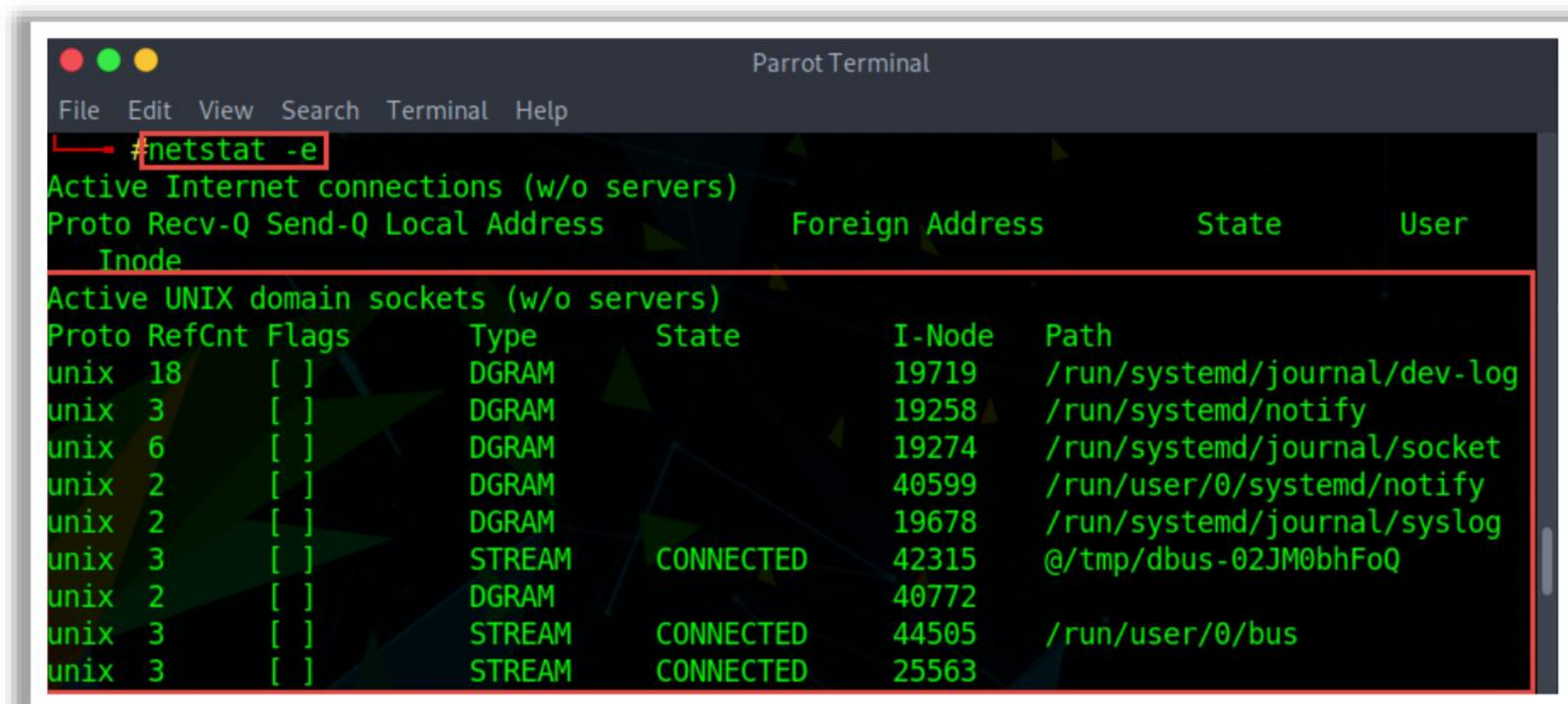


```
C:\> netstat -e
Interface Statistics

              Received              Sent
Bytes          3850711362          493608888
Unicast packets      5477744          607283
Non-unicast packets          0              0
Discards           0              0
Errors             0              0
Unknown protocols    0              0

C:\>
```

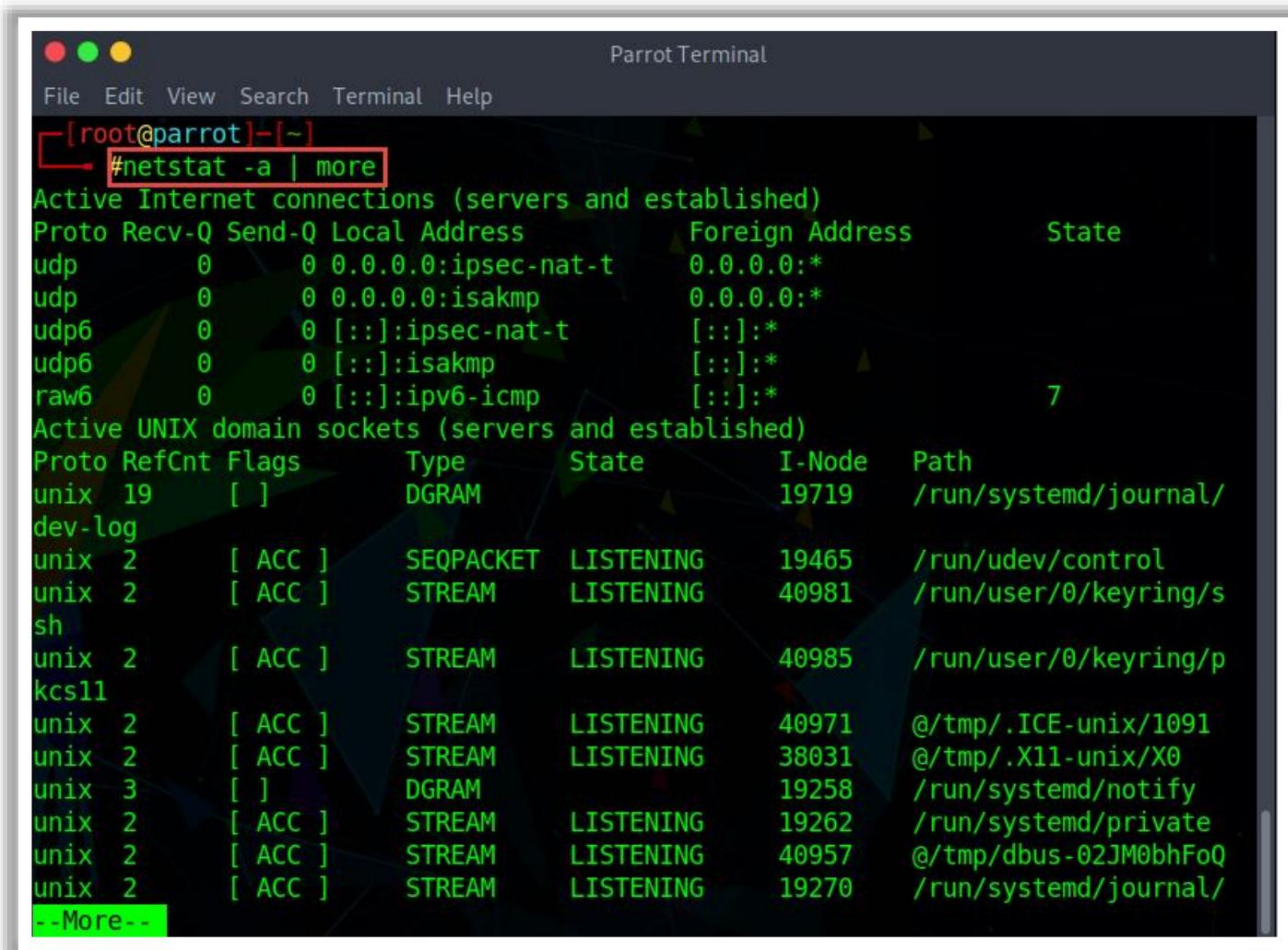
Figure 16.62: Using the netstat -e command in Windows



```
Parrot Terminal
File Edit View Search Terminal Help
# netstat -e
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       User
  Tnode
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags   Type       State       I-Node  Path
unix  18    [ ]     DGRAM     -           19719   /run/systemd/journal/dev-log
unix  3     [ ]     DGRAM     -           19258   /run/systemd/notify
unix  6     [ ]     DGRAM     -           19274   /run/systemd/journal/socket
unix  2     [ ]     DGRAM     -           40599   /run/user/0/systemd/notify
unix  2     [ ]     DGRAM     -           19678   /run/systemd/journal/syslog
unix  3     [ ]     STREAM   CONNECTED  42315   @/tmp/dbus-02JM0bhFoQ
unix  2     [ ]     DGRAM     -           40772
unix  3     [ ]     STREAM   CONNECTED  44505   /run/user/0/bus
unix  3     [ ]     STREAM   CONNECTED  25563
```

Figure 16.63: Using the netstat -e command in Linux

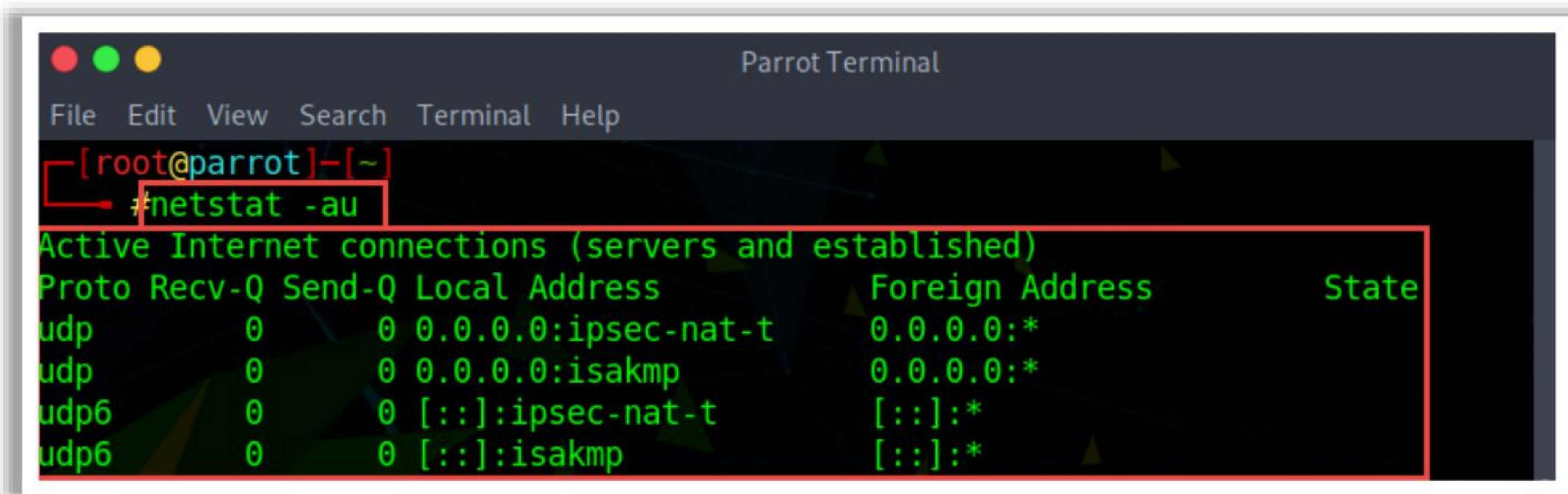
- The command “`netstat -a | more`” lists all the listening ports of TCP and UDP connections.



```
[root@parrot]-[~]
#netstat -a | more
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 0.0.0.0:ipsec-nat-t     0.0.0.0:*
udp        0      0 0.0.0.0:isakmp         0.0.0.0:*
udp6       0      0 [::]:ipsec-nat-t      [::]:*
udp6       0      0 [::]:isakmp           [::]:*
raw6       0      0 [::]:ipv6-icmp        [::]:*                7
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags               Type           State         I-Node  Path
unix  19      [ ]                 DGRAM          19719        /run/systemd/journal/
dev-log
unix  2      [ ACC ]              SEQPACKET     LISTENING     19465        /run/udev/control
unix  2      [ ACC ]              STREAM        LISTENING     40981        /run/user/0/keyring/s
sh
unix  2      [ ACC ]              STREAM        LISTENING     40985        /run/user/0/keyring/p
kcs11
unix  2      [ ACC ]              STREAM        LISTENING     40971        @/tmp/.ICE-unix/1091
unix  2      [ ACC ]              STREAM        LISTENING     38031        @/tmp/.X11-unix/X0
unix  3      [ ]                 DGRAM          19258        /run/systemd/notify
unix  2      [ ACC ]              STREAM        LISTENING     19262        /run/systemd/private
unix  2      [ ACC ]              STREAM        LISTENING     40957        @/tmp/dbus-02JM0bhFoQ
unix  2      [ ACC ]              STREAM        LISTENING     19270        /run/systemd/journal/
--More--
```

Figure 16.64: Listing the ports of TCP and UDP connections

- The command “`netstat -at`” lists TCP port connections.
- The command “`netstat -au`” lists UDP port connections.



```
[root@parrot]-[~]
#netstat -au
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 0.0.0.0:ipsec-nat-t     0.0.0.0:*
udp        0      0 0.0.0.0:isakmp         0.0.0.0:*
udp6       0      0 [::]:ipsec-nat-t      [::]:*
udp6       0      0 [::]:isakmp           [::]:*
```

Figure 16.65: Listing UDP port connections

Follow the steps below to list various listening connections.

- The command “`netstat -l`” lists all listening UDP connections.

```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]-[~]
#netstat -l
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State
udp 0 0 0.0.0.0:ipsec-nat-t 0.0.0.0:*
udp 0 0 0.0.0.0:isakmp 0.0.0.0:*
udp6 0 0 [::]:ipsec-nat-t [::]:*
udp6 0 0 [::]:isakmp [::]:*
raw6 0 0 [::]:ipv6-icmp [::]:* 7
Active UNIX domain sockets (only servers)
Proto RefCnt Flags Type State I-Node Path
unix 2 [ ACC ] SEQPACKET LISTENING 19465 /run/udev/control
unix 2 [ ACC ] STREAM LISTENING 40981 /run/user/0/keyring/ssh
unix 2 [ ACC ] STREAM LISTENING 40985 /run/user/0/keyring/pkcs11
unix 2 [ ACC ] STREAM LISTENING 40971 @/tmp/.ICE-unix/1091
unix 2 [ ACC ] STREAM LISTENING 38031 @/tmp/.X11-unix/X0
unix 2 [ ACC ] STREAM LISTENING 19262 /run/systemd/private
unix 2 [ ACC ] STREAM LISTENING 40957 @/tmp/dbus-02JM0bhFoQ
unix 2 [ ACC ] STREAM LISTENING 19270 /run/systemd/journal/stdout
unix 2 [ ACC ] STREAM LISTENING 34903 /var/run/charonctl
unix 2 [ ACC ] STREAM LISTENING 24153 /run/uidd/request
unix 2 [ ACC ] STREAM LISTENING 24156 /run/dbus/system_bus_socket
unix 2 [ ACC ] STREAM LISTENING 24160 /run/snapd.socket
unix 2 [ ACC ] STREAM LISTENING 24163 /run/snapd-snap.socket
unix 2 [ ACC ] STREAM LISTENING 24166 /run/pcscd/pcscd.comm
unix 2 [ ACC ] STREAM LISTENING 38032 /tmp/.X11-unix/X0
unix 2 [ ACC ] STREAM LISTENING 19587 /run/lvm/lvmpolld.socket
unix 2 [ ACC ] STREAM LISTENING 40603 /run/user/0/systemd/private
unix 2 [ ACC ] STREAM LISTENING 40610 /run/user/0/gnupg/S.gpg-agent
```

Figure 16.66: Listing all listening connections

- The command “`netstat -lt`” lists all TCP listening ports.
- The command “`netstat -lu`” lists all UDP listening ports.

```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]-[~]
#netstat -lu
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address Foreign Address State
udp 0 0 0.0.0.0:ipsec-nat-t 0.0.0.0:*
udp 0 0 0.0.0.0:isakmp 0.0.0.0:*
udp6 0 0 [::]:ipsec-nat-t [::]:*
udp6 0 0 [::]:isakmp [::]:*
```

Figure 16.67: Listing UDP listening ports

Follow the steps below to list statistics for different protocols.

- The command “`netstat -s`” lists the statistics for all protocols.



```
Parrot Terminal
File Edit View Search Terminal Help
#netstat -s
Ip:
  Forwarding: 2
  847 total packets received
  36 with invalid addresses
  0 forwarded
  0 incoming packets discarded
  811 incoming packets delivered
  141 requests sent out
Icmp:
  20 ICMP messages received
  0 input ICMP message failed
  ICMP input histogram:
    destination unreachable: 20
  20 ICMP messages sent
  0 ICMP messages failed
  ICMP output histogram:
    destination unreachable: 20
IcmpMsg:
  InType3: 20
  OutType3: 20
Tcp:
  4 active connection openings
  0 passive connection openings
  2 failed connection attempts
  0 connection resets received
  0 connections established
  85 segments received
  87 segments sent out
  0 segments retransmitted
  0 bad segments received
  4 resets sent
Udp:
  16 packets received
  20 packets to unknown port received
  0 packet receive errors
  36 packets sent
  0 receive buffer errors
  0 send buffer errors
  IgnoredMulti: 672
UdpLite:
TcpExt:
  57 packet headers predicted
  2 acknowledgments not containing data payload received
  6 predicted acknowledgments
  TCPRcvCoalesce: 11
  TCPOrigDataSent: 8
  TCPDelivered: 10
IpExt:
  InBcastPkts: 672
  InOctets: 895018
  OutOctets: 8512
  InBcastOctets: 54264
  InNoECTPkts: 1360
```

Figure 16.68: Listing statistics for all protocols

- The command “`netstat -st`” lists statistics for TCP.
- The command “`netstat -su`” lists statistics for UDP.

The command “`netstat -tp`” displays the service name with PID.

The command “`netstat -r`” displays the kernel IP routing table.

```
Parrot Terminal
File Edit View Search Terminal Help
#netstat -r
Kernel IP routing table
Destination Gateway Genmask Flags MSS Window irtt Iface
default 10.10.10.2 0.0.0.0 UG 0 0 0 eth0
10.10.10.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
[root@parrot]-[~]
#
```

Figure 16.69: Displaying the kernel IP routing table

The command “`netstat -i`” displays network interface packet transactions.

```
Parrot Terminal
File Edit View Search Terminal Help
#netstat -i
Kernel Interface table
Iface MTU RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0 1500 1706 0 0 0 120 0 0 0 BMRU
lo 65536 88 0 0 0 88 0 0 0 LRU
[root@parrot]-[~]
#
```

Figure 16.70: Displaying network interface packet transactions

The command “`netstat -ie`” displays the kernel interface table.

The command “`netstat -c`” prints netstat information continuously.

```
Parrot Terminal
File Edit View Search Terminal Help
#netstat -c
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address Foreign Address State
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags Type State I-Node Path
unix 18 [ ] DGRAM 19719 /run/systemd/journal/dev-log
unix 3 [ ] DGRAM 19258 /run/systemd/notify
unix 6 [ ] DGRAM 19274 /run/systemd/journal/socket
unix 2 [ ] DGRAM 40599 /run/user/0/systemd/notify
unix 2 [ ] DGRAM 19678 /run/systemd/journal/syslog
unix 3 [ ] STREAM CONNECTED 42315 @/tmp/dbus-02JM0bhFoQ
unix 2 [ ] DGRAM 40772
unix 3 [ ] STREAM CONNECTED 44505 /run/user/0/bus
unix 3 [ ] STREAM CONNECTED 25563
unix 3 [ ] STREAM CONNECTED 24687 /run/systemd/journal/stdout
unix 3 [ ] STREAM CONNECTED 41686 @/tmp/.X11-unix/X0
unix 3 [ ] STREAM CONNECTED 40746
unix 3 [ ] STREAM CONNECTED 41656 @/tmp/.ICE-unix/1091
unix 3 [ ] STREAM CONNECTED 40963
unix 3 [ ] STREAM CONNECTED 42325 /run/user/0/bus
unix 3 [ ] DGRAM 19261
```

Figure 16.71: Displaying netstat information continuously

The command “`netstat -ap | grep http`” lists listening programs.

The command “`netstat -statistics --raw`” displays raw network statistics.

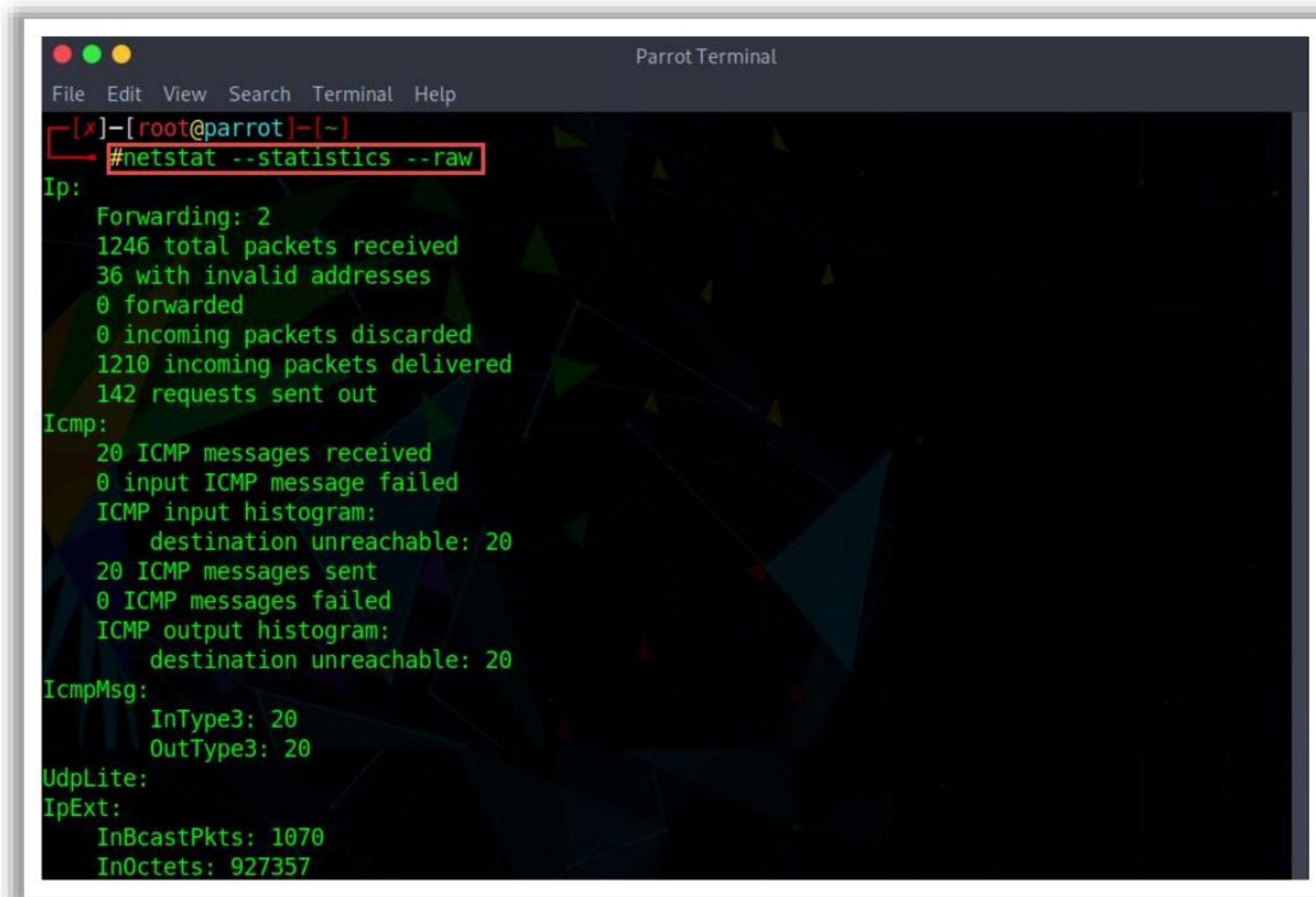


Figure 16.72: Displaying raw network statistics

■ PuTTY

The open-source graphical user interface (GUI) client PuTTY is a terminal emulator application that supports protocols such as SSH, Telnet, Rlogin, and serial for Windows and Unix-like operating systems (OSes). It helps in accessing and managing remote Linux servers. It is an FTP or SSH FTP (SFTP) client for transferring files. It generates hashes for passwords.

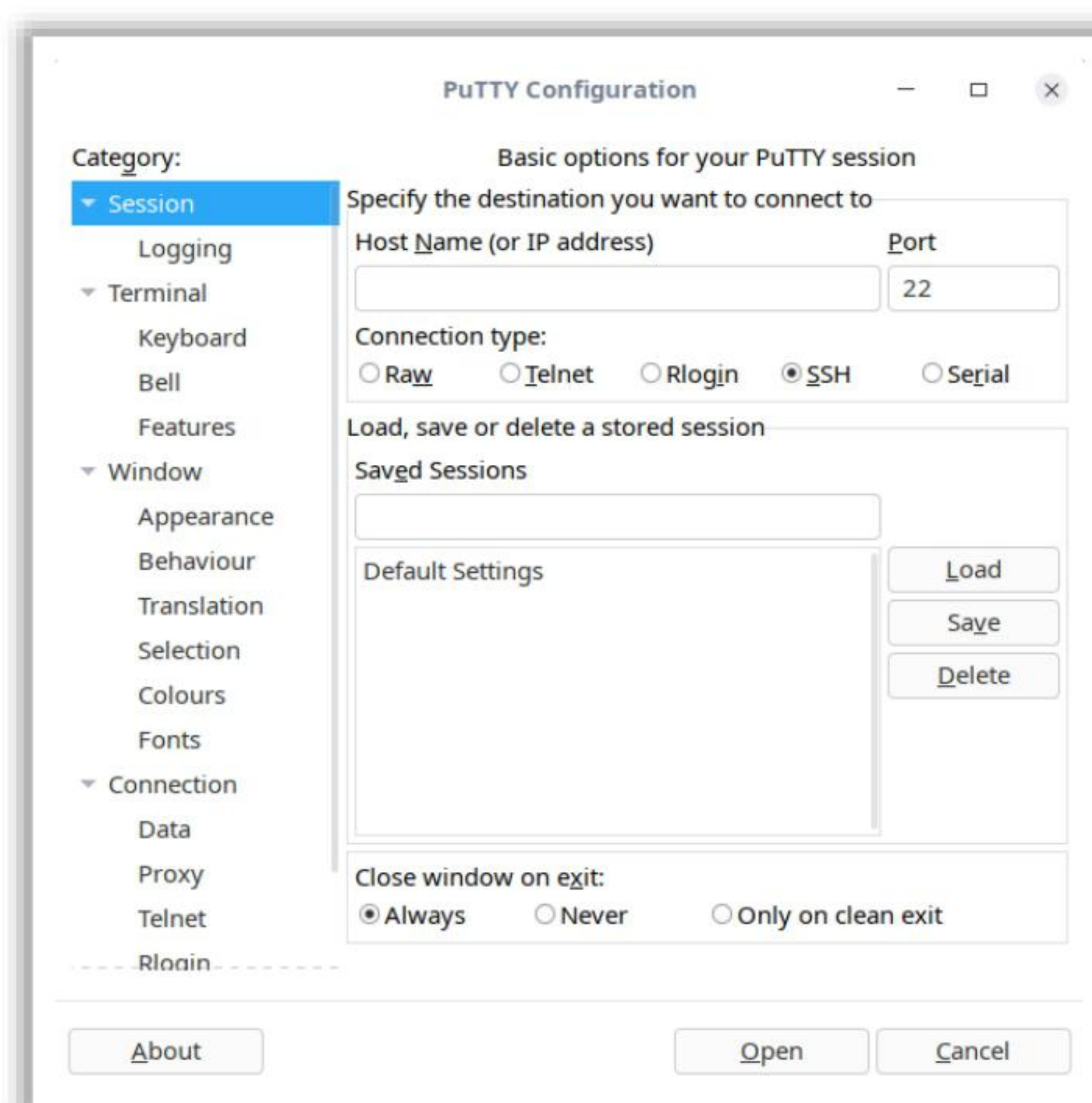


Figure 16.73: Putty Configuration

Features of PuTTY

- Control over the SSH encryption key and protocol version
- Command-line Secure Copy Protocol (SCP) and SSH File Transfer Protocol (SFTP) clients called “pscp” and “psftp,” respectively
- Control over port forwarding with SSH (local, remote, or dynamic port forwarding), including built-in handling of X11 forwarding
- Emulates most xterm, VT102 control sequences, as well as much of ECMA-48 terminal emulation
- IPv6 support
- Supports the Triple Data Encryption Standard (3DES), Advanced Encryption Standard (AES), Arcfour, Blowfish, and DES
- Public-key authentication support
- Support for local serial port connections

Steps to Use PuTTY to Access Remote Linux Servers

- Launch PuTTY and select the **Session** tab from the left pane.
- Enter the **Host Name (or IP address)** of the remote system to be connected and select **Connection type**. The selected connection type fills the default **Port** number automatically. Here, selecting **SSH** fills **Port** number 22. Click **Open**.

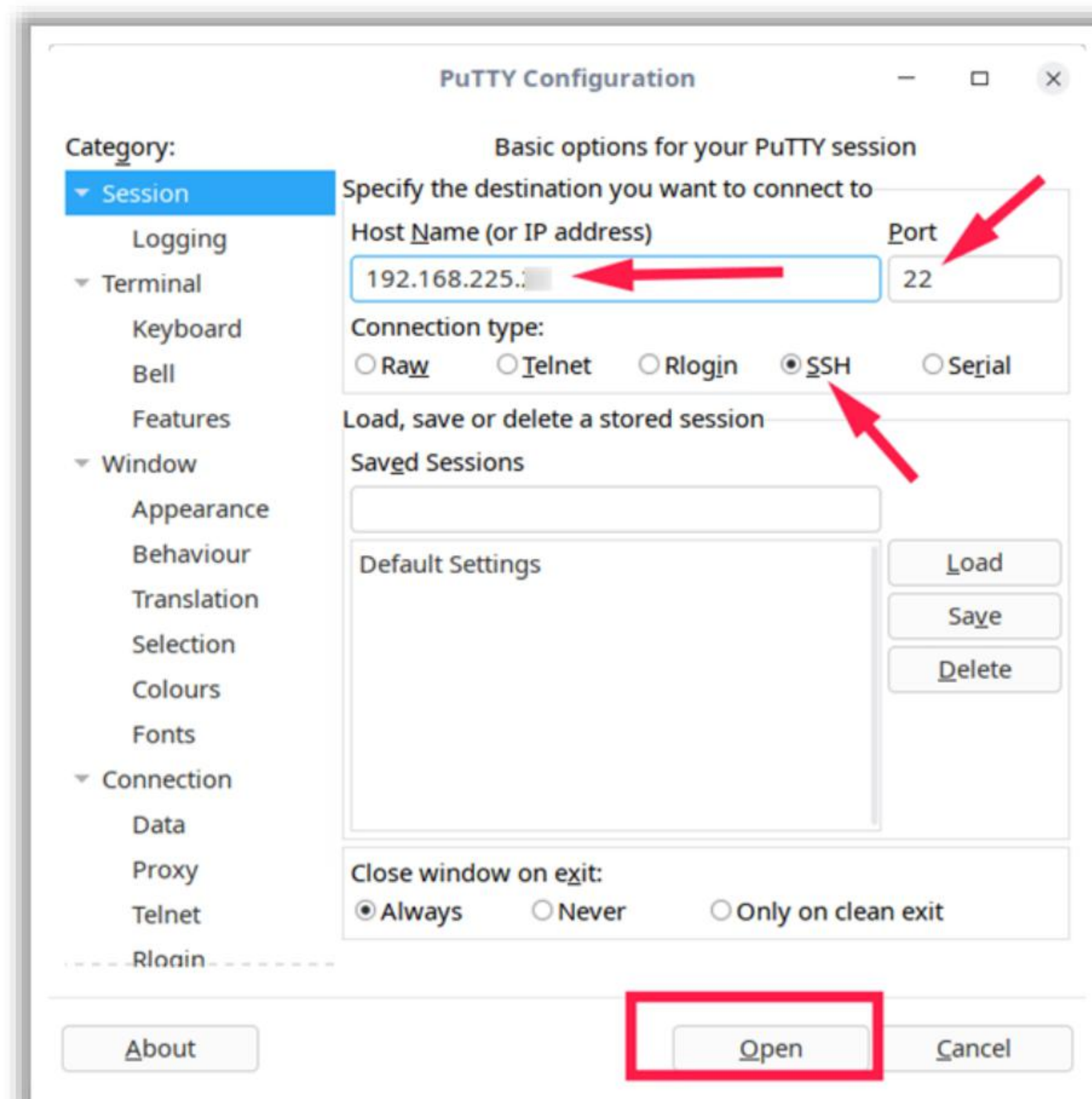


Figure 16.74: Use Putty to Access Remote Linux Servers

- If this is the first time the user is connecting to this remote system, PuTTY will display a security alert dialog box. Click **Accept** to add the remote system's host key to PuTTY's cache.

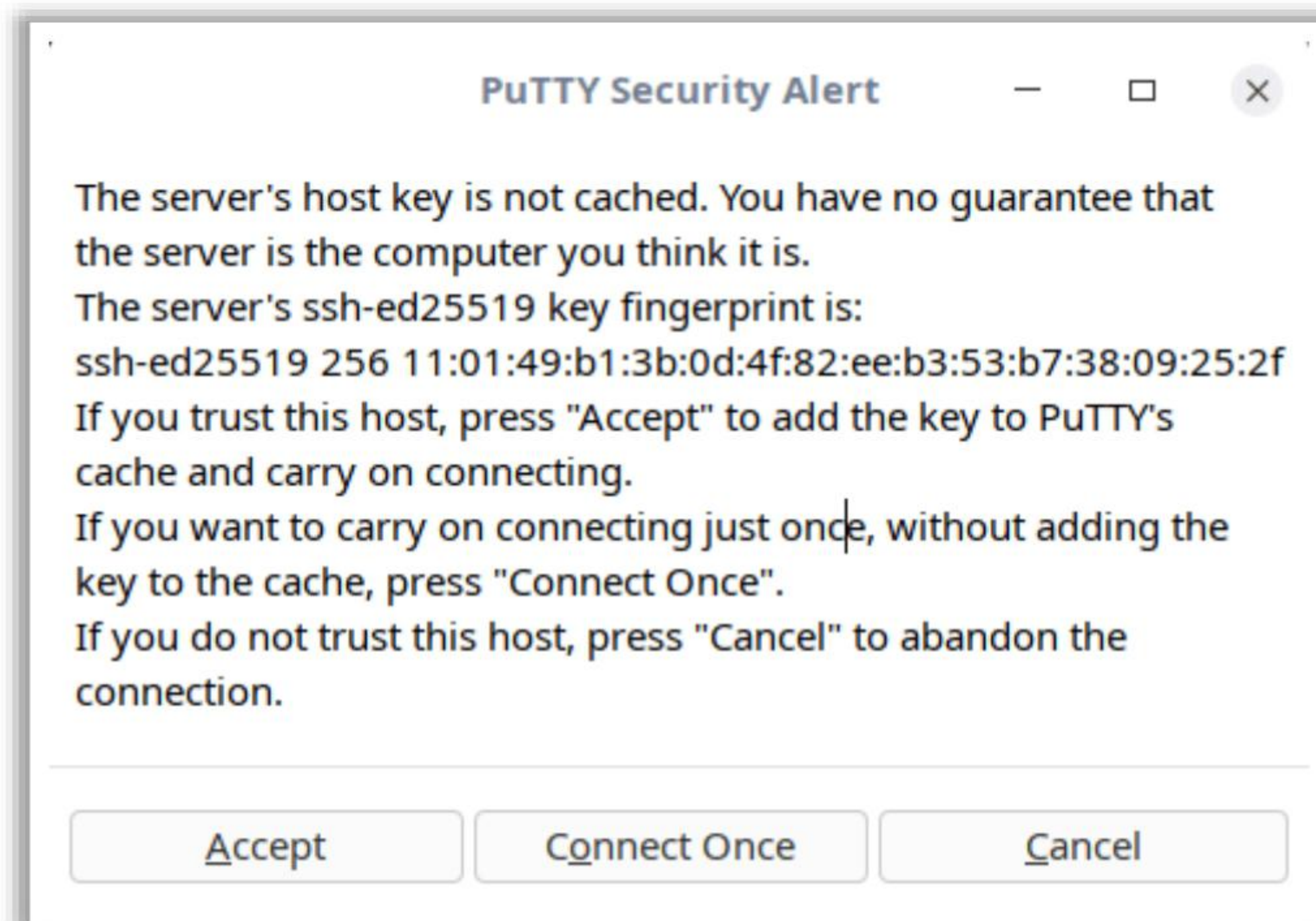


Figure 16.75: Putty Security Alert

- Enter the remote system's username and password. A connection is established to the remote system via SSH using PuTTY.

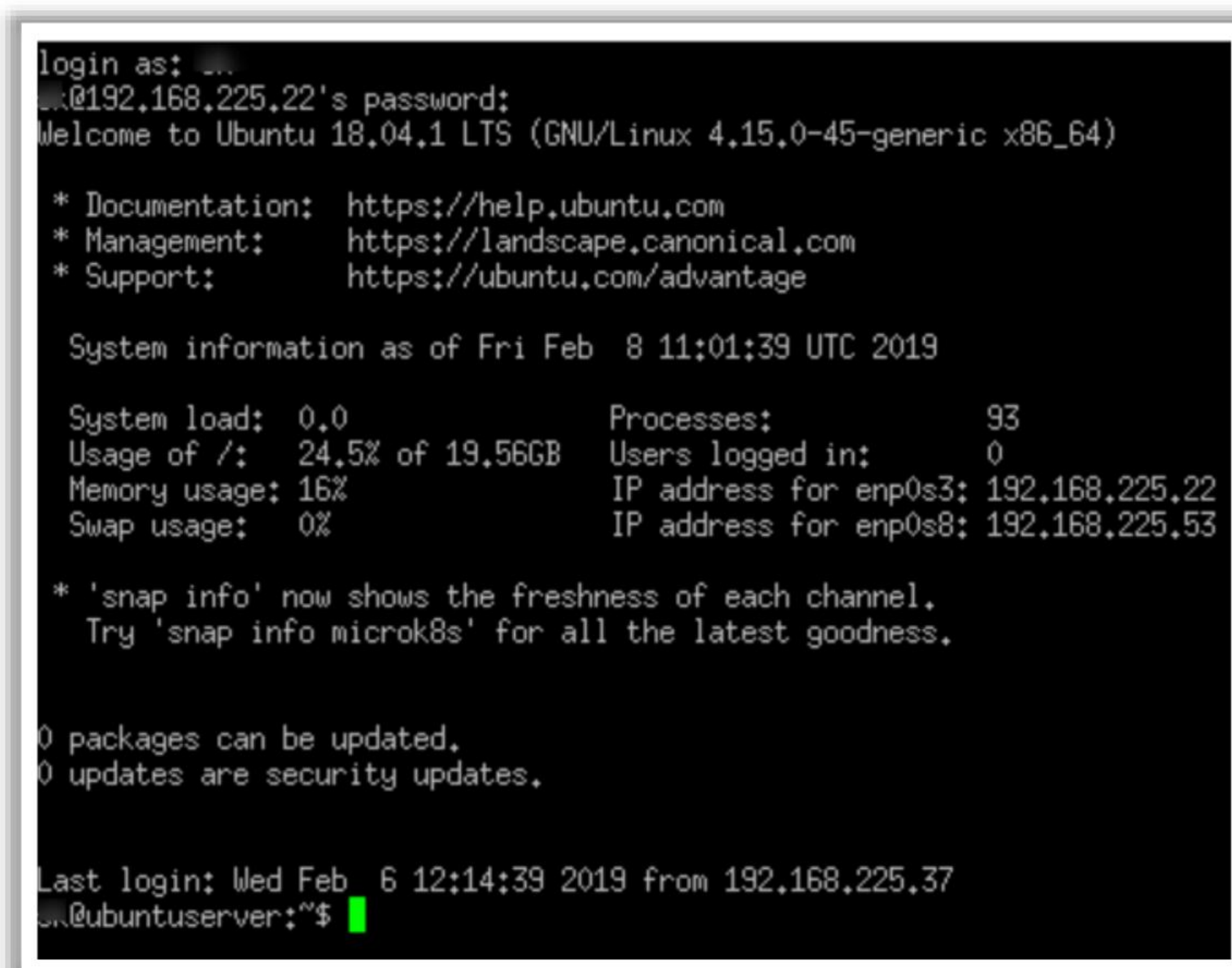


Figure 16.76: Establish Remote System Connection

Steps to Access Remote Systems Configured with Key-based Authentication

- Enter **Host Name (or IP address)** in the **Session** section.
- In the **Category** pane, expand **Connection**, expand **SSH**, and then choose **Auth**.
- Browse the location of the **.ppk** key file and click **Open**.

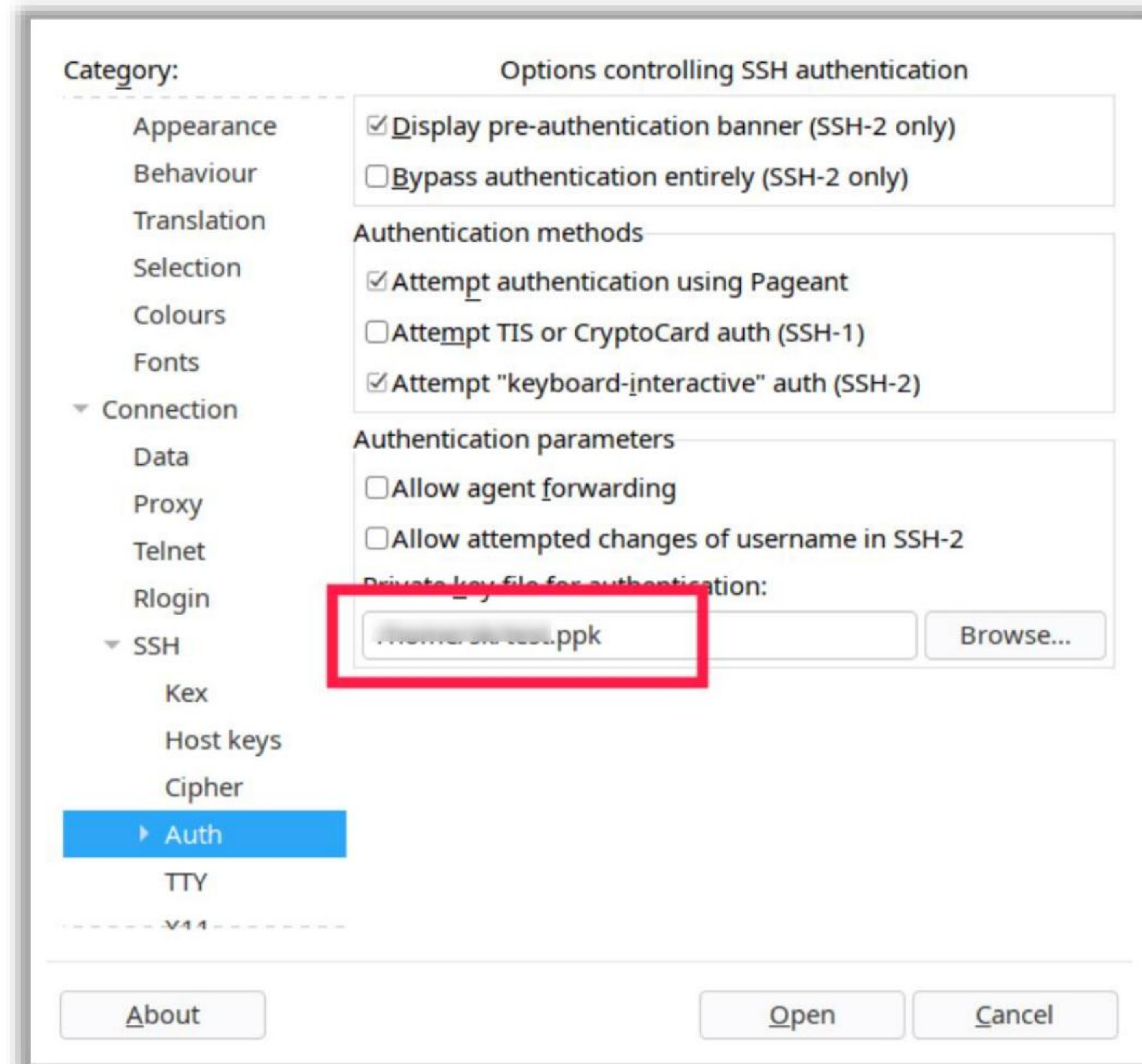


Figure 16.77: Browse for the Key File

- Click **Accept** to add the host key if it is the first time the user is connecting to the remote system. Enter the remote system's passphrase to connect.
- **IP Subnet Calculator**

A subnet is used to find information about IPv4 and IPv6 subnets as well as for the division of classes of subnets. For example, the SolarWinds Advanced Subnet Calculator helps ensure that IP addresses are not in conflict with one another and saves time in managing DHCP, DNS, and IP addresses. It allows the calculation of subnet masks and IP address management with the following:

 - Breaking down the IP address
 - Performing forward and reverse DNS resolution
 - Offering classful subnet calculations and CIDR subnet calculations
 - Providing a complete report of subnet addresses based on calculations

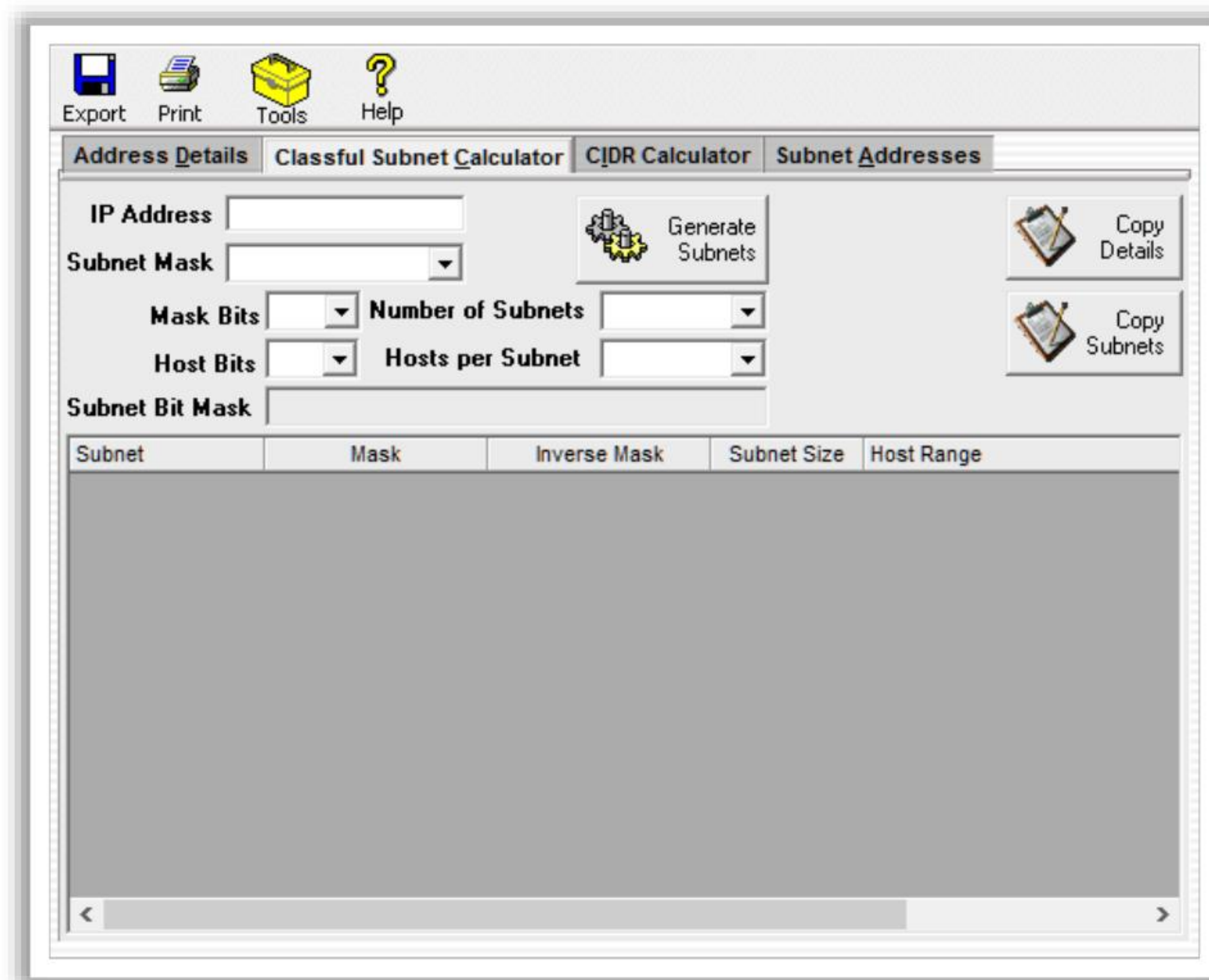


Figure 16.78: Solarwind's Advanced Subnet Calculator

Tunnels Up Subnet Calculator

The web-based Tunnels Up Subnet Calculator is an IPv4 and IPv6 calculator that allows for the input of a netmask, a Cisco wildcard mask/CIDR notation. It allows performing reverse subnetting if the number of hosts is known and the netmask is unknown.

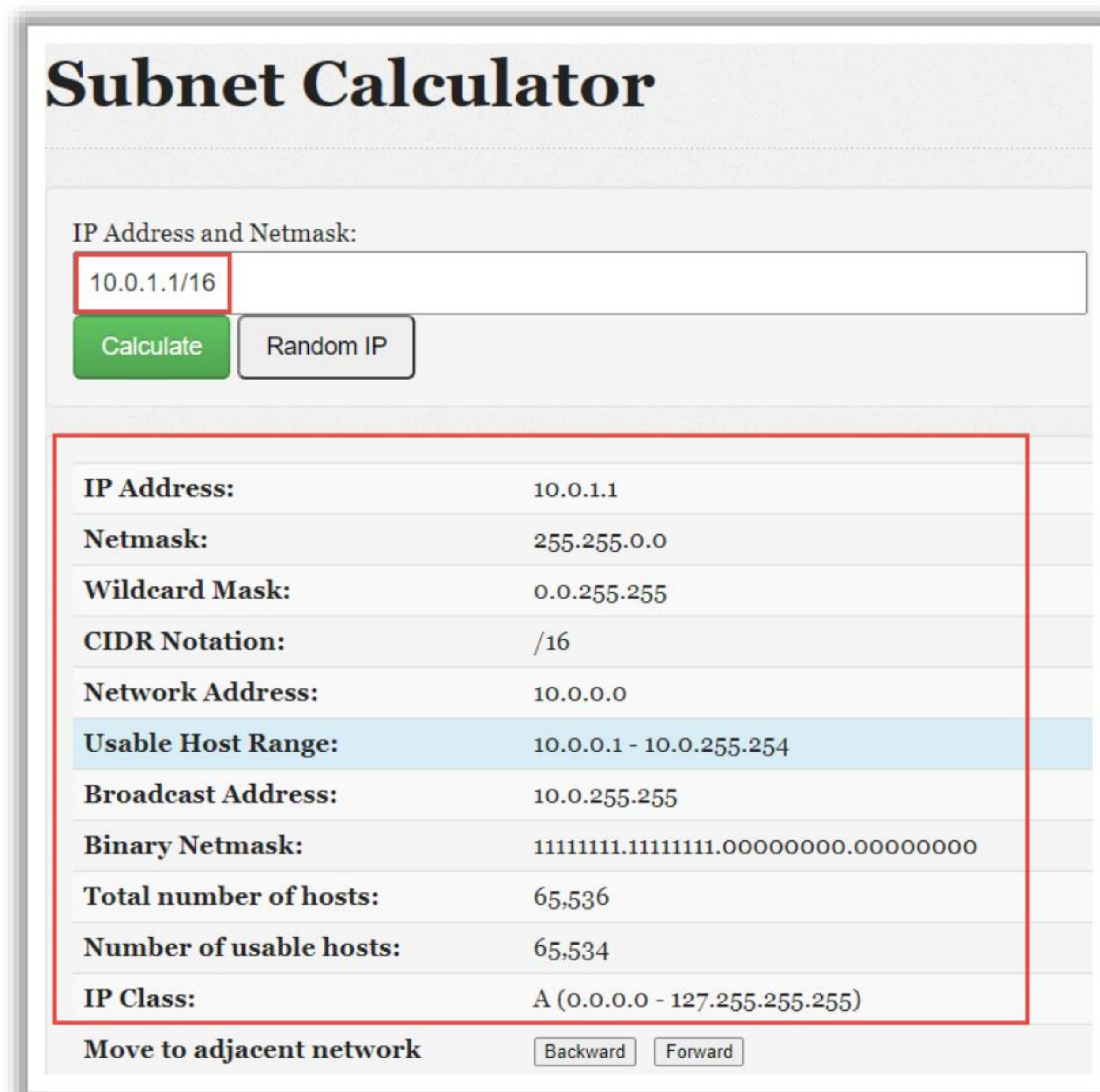


Figure 16.79: Screenshot of Subnet Calculator

- **speedtest.net**

The website speedtest.net is used to determine the available bandwidth for a host at a point of time. It provides free analysis of Internet access performance metrics. For example, it provides metrics such as the connection data rate (speed) and latency (connection delay). For enhanced test accuracy, it uses TCP sockets and a custom protocol for communication between servers and users.

The time taken to upload and download a file can also be determined using this website. All the tests performed by speedtest.net measure the data rate in the download direction (from the server to the user) and that in the upload direction (the user to the server).

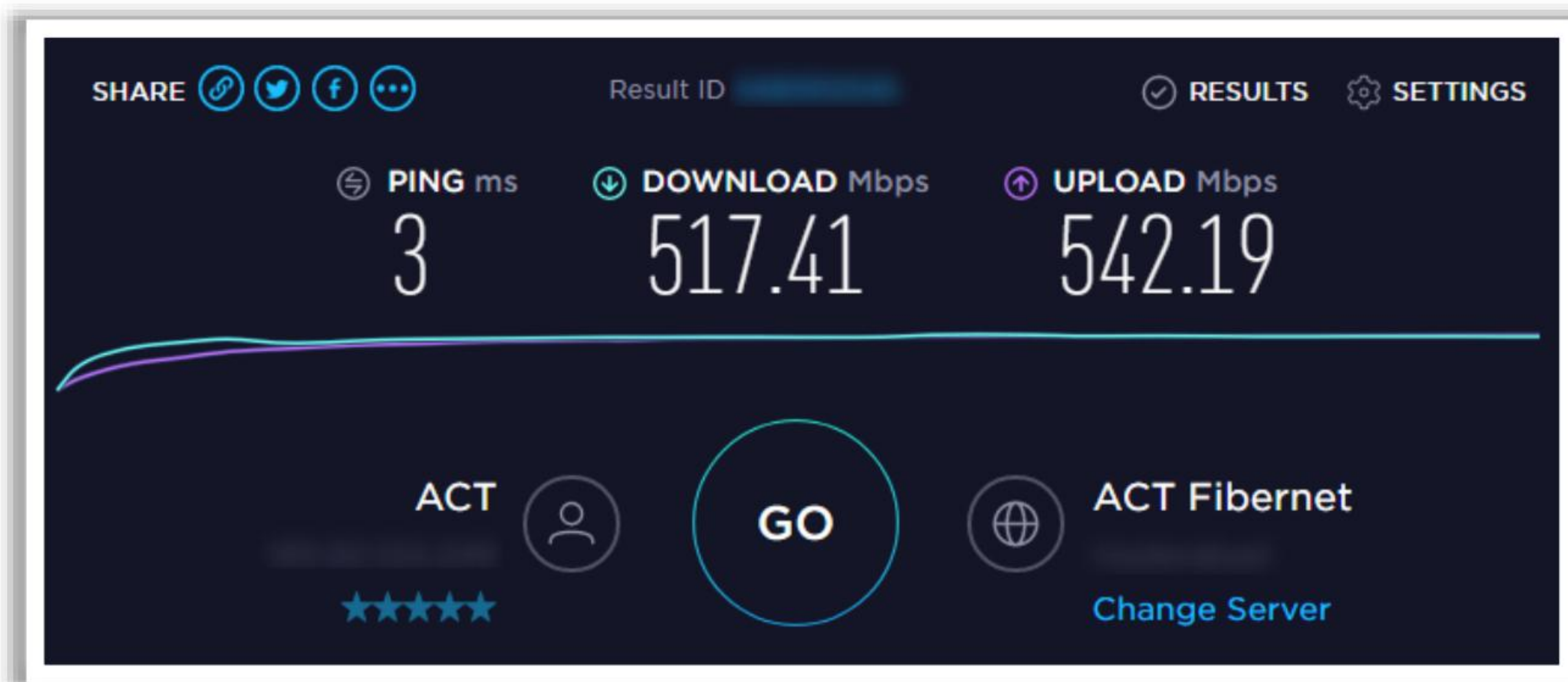


Figure 16.80: Working of “Speedtest.net”

- **pathping**

Source: <https://docs.microsoft.com>

The pathping utility provides detailed information about the path characteristics from a specific host to a specific destination in a single picture by taking advantage of the ping and tracert/traceroute commands. It helps diagnose packet loss and slow speed faults. Initially, running pathping traces the route to a destination address and launches a 25-s test for each hop to show the pathping statistics on the data loss to each hop.

```
pathping [/n] [/h <maximumhops>] [/g <hostlist>] [/p <Period>]
[/q <numqueries>] [/w <timeout>] [/i <IPaddress>] [/4 <IPv4>] [/6
<IPv6>] [<targetname>]
```

Options	Description
/n	Prevents the resolving of an IP address to a host name
/h <maximumhops>	Defines the maximum number of hops for reaching the target (30 by default)
/g <hostlist>	Defines a loose source route through a specified hostlist in the IP header (maximum allowed list is 9)

<code>/p <period></code>	Defines the wait time between pings (mostly in milliseconds)
<code>/q <numqueries></code>	Defines the number of ICMP queries for each hop (100 by default)
<code>/w <timeout></code>	Defines the wait time for each ICMP reply (3000 ms, i.e., 3 s, by default)
<code>/i <IPaddress></code>	Defines the source address
<code>/4 <IPv4></code>	Forces pathping to use only IPv4
<code>/6 <IPv6></code>	Forces pathping to use only IPv6
<code><targetname></code>	Defines the target or destination address that can be identified by the host name or IP address
<code>/?</code>	Displays all the available options on the command-line interface

Table 16.3: pathping command options

Steps to Use pathping for Networking Troubleshooting

- Run the command `pathping <IP address>` in Command Prompt. Interrupt pathping at any time by holding down ctrl + C.

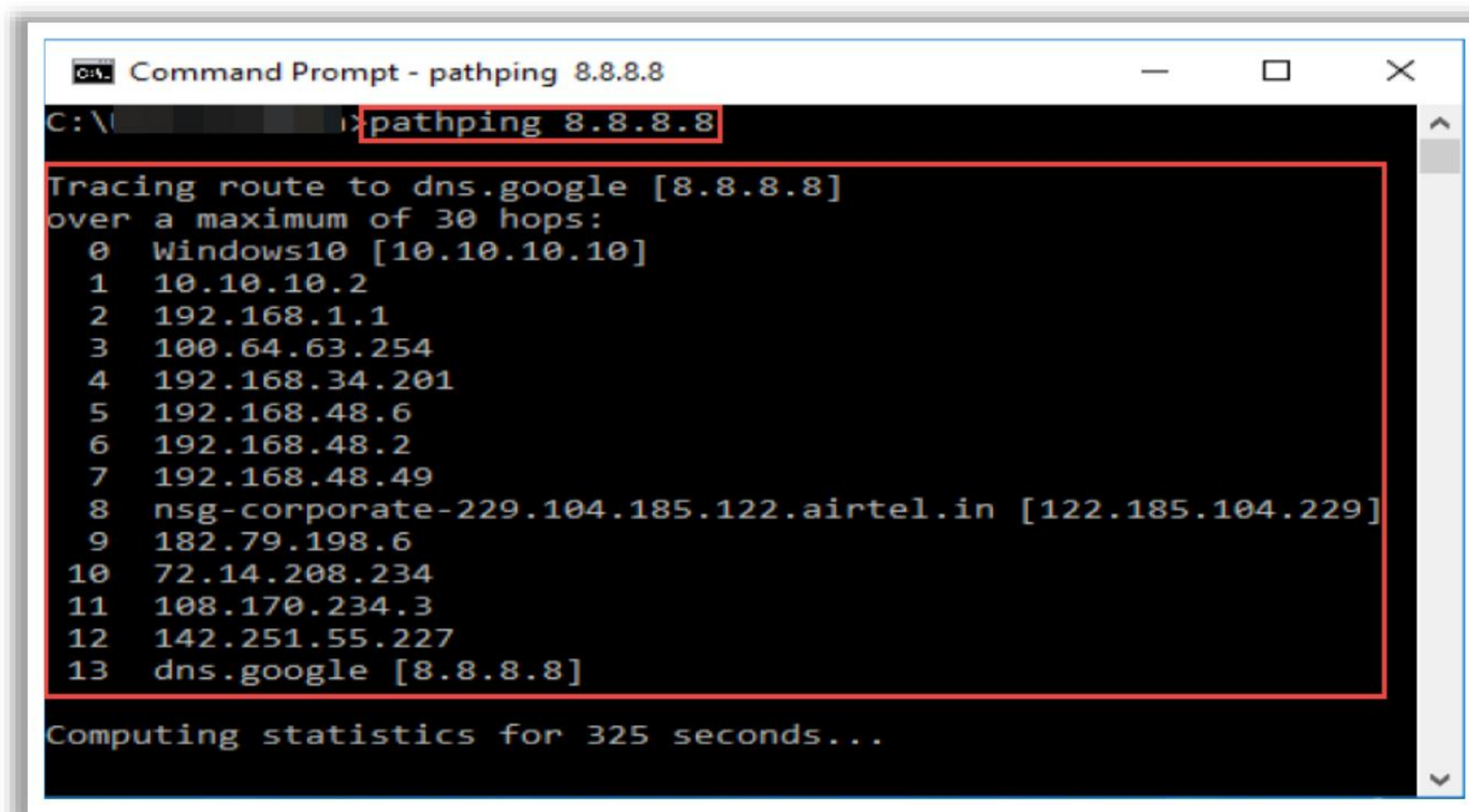


Figure 16.81: Running the command “pathping”

- Use the command `pathping -n` to show numeric IP numbers instead of DNS host names.

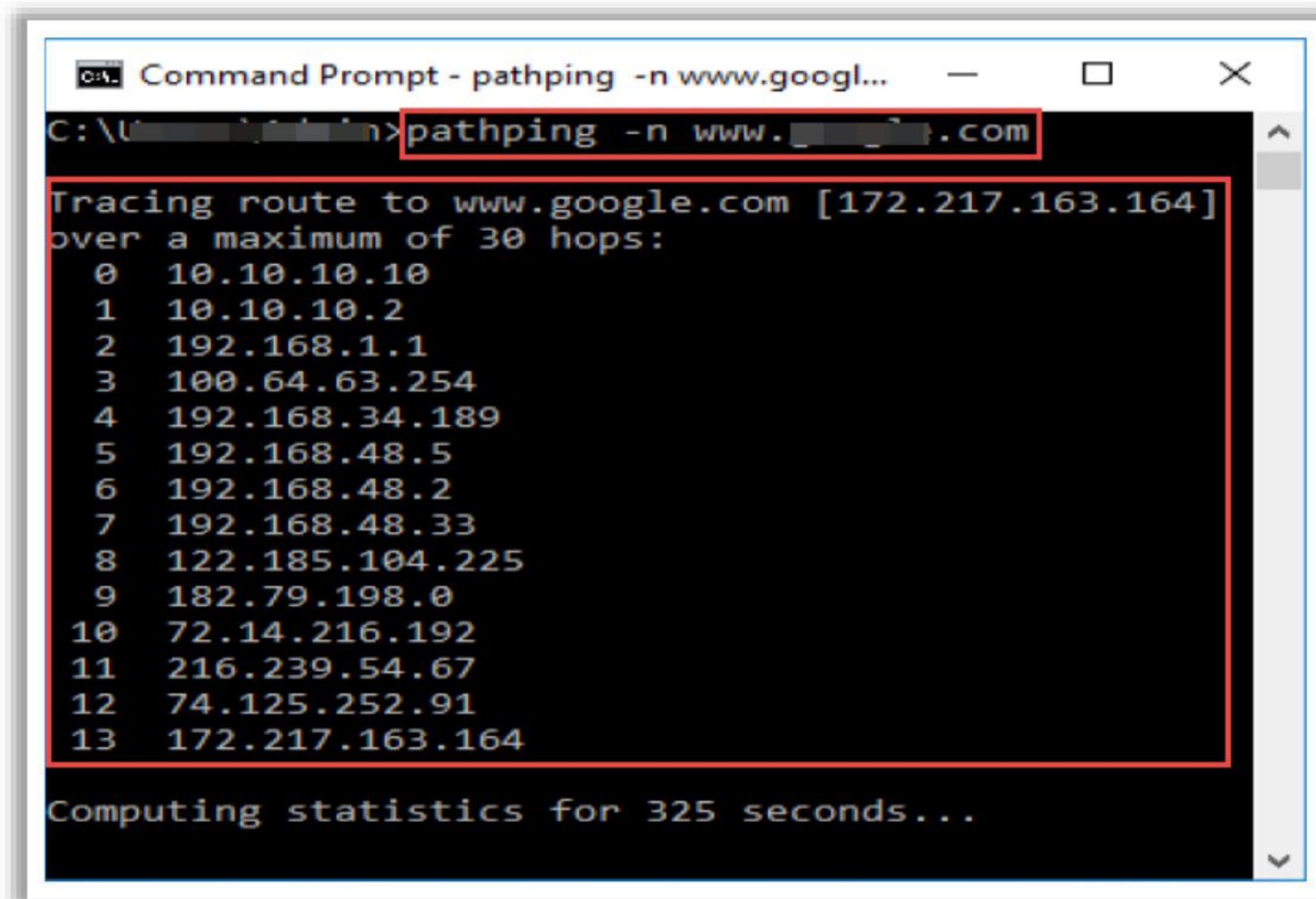


Figure 16.82: Displaying IP numbers instead of domain names

- **route**

Source: <https://docs.microsoft.com>

The route utility is used to show the ongoing status of and modifications to the routing table on the Windows host. It is useful when the host has multiple IPs and multiple hosts. Netmasks, network destinations, and gateways are displayed in the active routes section of the route utility. In Unix/Linux, the route command can be used without any command-line switches. The command shows similar outputs for both Windows and Unix/Linux.

```
route [/f] [/p] [<Command> [<Destination>] [mask <Netmask>]
[<Gateway>] [metric <Metric>]] [if <Interface>]]
```

Options	Description
/f	Wipes off all entries in the routing table
/p	Initializes the new routing table by adding a new route to the registry
<Command>	Defines the command to run (e.g., add, change, delete, or run)
<Destination>	Defines the destination or target of the route
mask <Netmask>	Defines the subnet masks associated with the target
<Gateway>	Defines the next hop address or transmission
metric <Metric>	Defines the integer cost metric for the route (between 0 and 9999)

if <Interface>	Defines the interface index for reaching the destination
/?	Displays all the available options

Table 16.4: route command options

Steps to Use the route Command for Network Troubleshooting

- In Windows, use the command **route print** to view the routing table (IPv4 and IPv6).

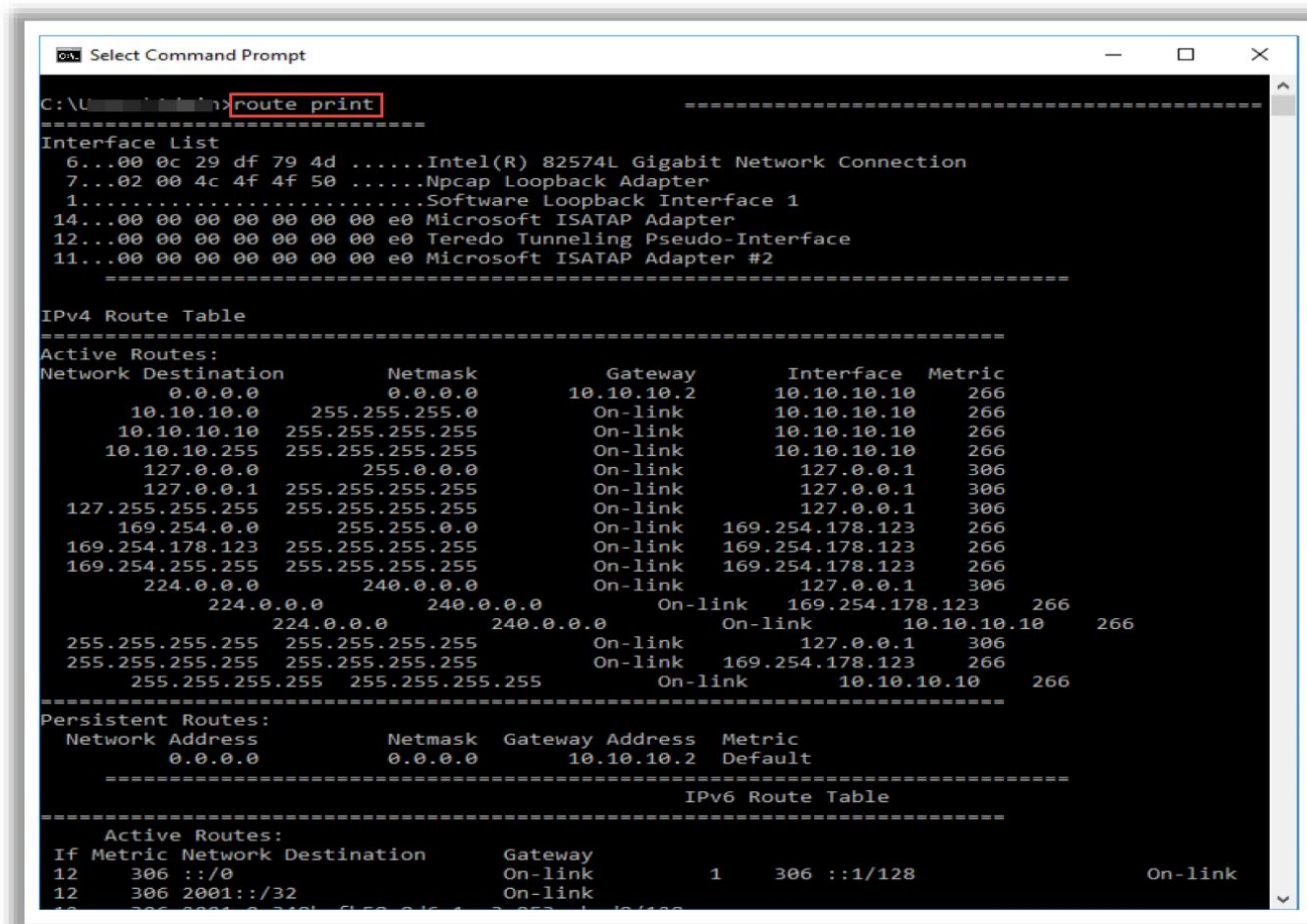


Figure 16.83: Viewing the routing table

- To add, delete, or change a route entry, use the following command:
route [/p] command dest [mask subnet] gateway [if interface]

Example:

```

C:\>route /p add 192.168.5.0 mask 255.255.255.0 192.168.1.500
C:\>route change 192.168.5.0 mask 255.255.255.0 192.168.1.542
C:\>route delete 192.168.5.0

```

▪ Nmap

Source: <https://nmap.org>

Nmap (“Network Mapper”) is a security scanner for network exploration and hacking. It allows the discovery of hosts, ports, and services on a computer network, thus creating a “map” of the network. It sends specially crafted packets to the target host and then analyzes the responses to accomplish its goal. It scans vast networks of hundreds of

thousands of machines. Nmap includes many mechanisms for port scanning (TCP and UDP), OS detection, version detection, ping sweeps, and so on.

Either a network administrator or an attacker can use this tool for their specific needs. Network administrators can use Nmap for maintaining network inventory, managing service upgrade schedules, and monitoring host or service uptime. Nmap can also be used to extract information such as live hosts on the network, open ports, services (application name and version), type of packet filters/firewalls, MAC details, and OSes, along with their versions.

Syntax: # `nmap <options> <Target IP address>`

Host Discovery

- Perform an ARP ping scan for discovering live hosts in the network. Use the `-PR` option to perform an ARP ping scan.

`nmap -sn -PR <IP address>`

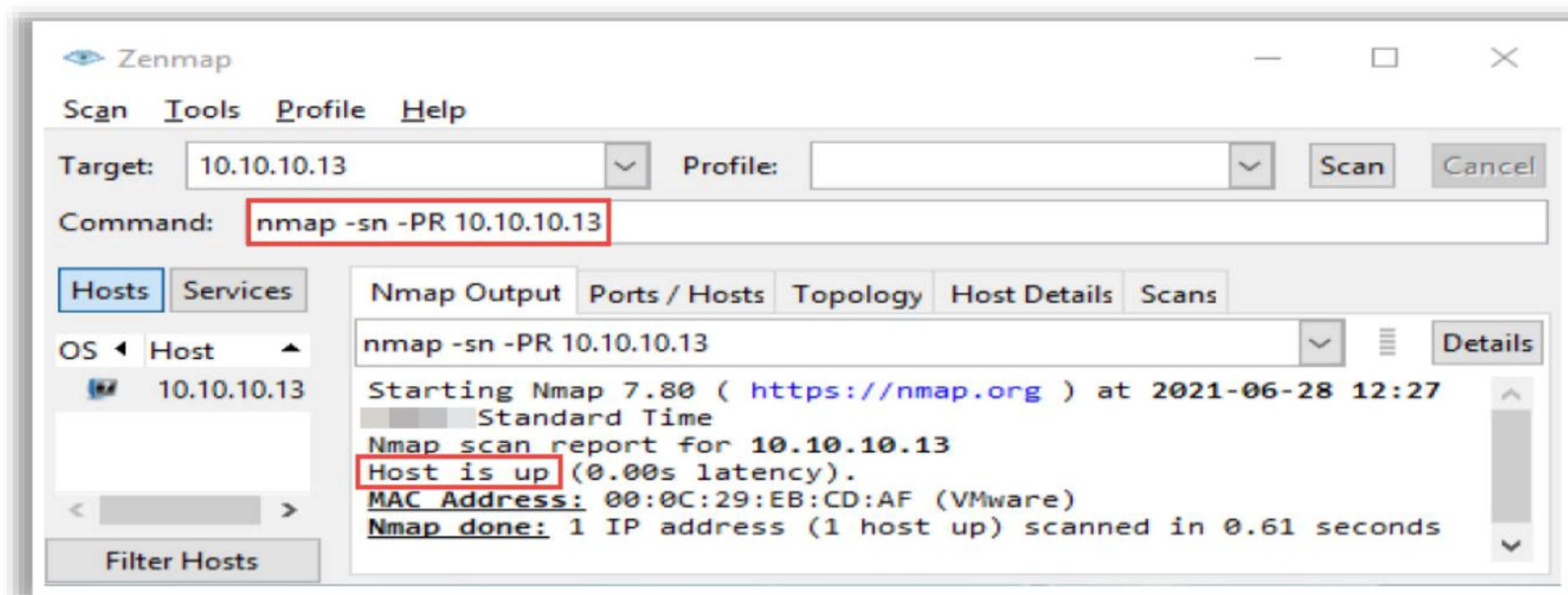


Figure 16.84: Screenshot of Nmap host discovery using an ARP ping scan

- ○ Use the `-PE` option to perform the ICMP ECHO ping scan. Active hosts are displayed as “Host is up,” as shown in screenshot.

`nmap -sn -PE <IP address>`

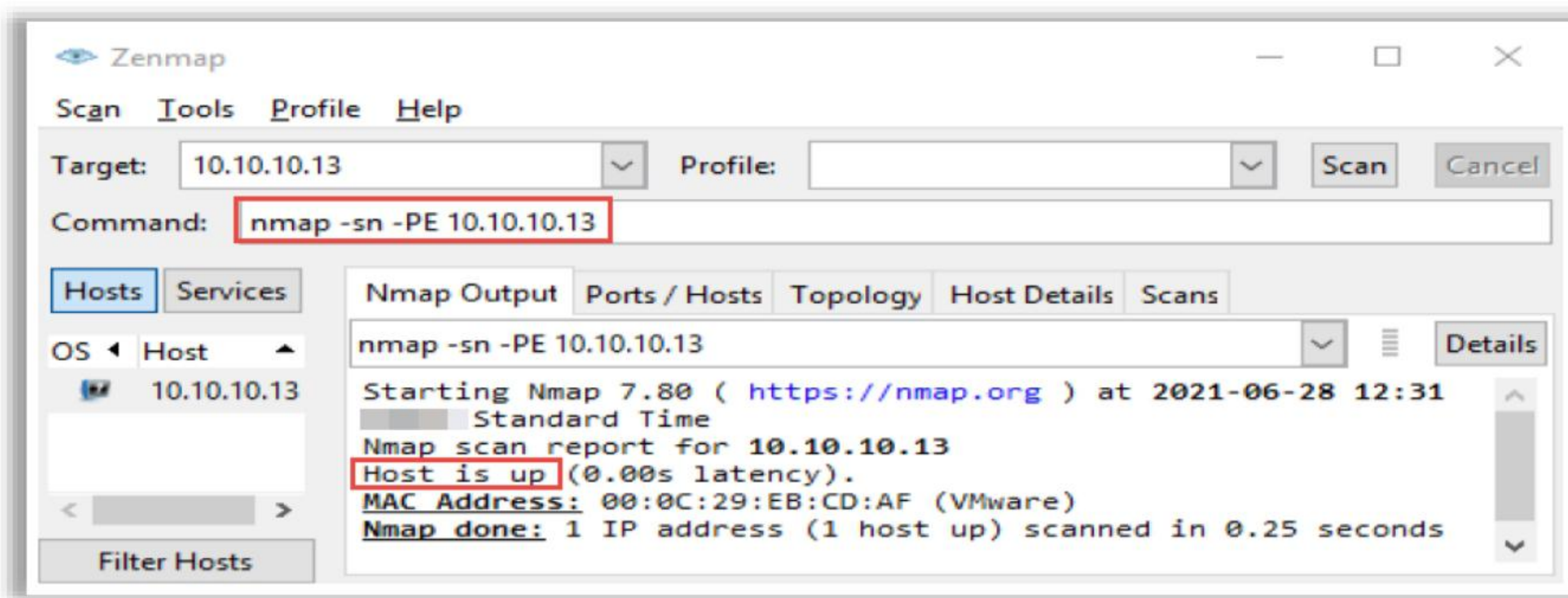


Figure 16.85: Screenshot of Nmap host discovery using an ICMP ECHO ping scan

- Use the -PE option with a list of IP addresses to perform an ICMP ECHO ping sweep.
nmap -sn -PR <IP address range>

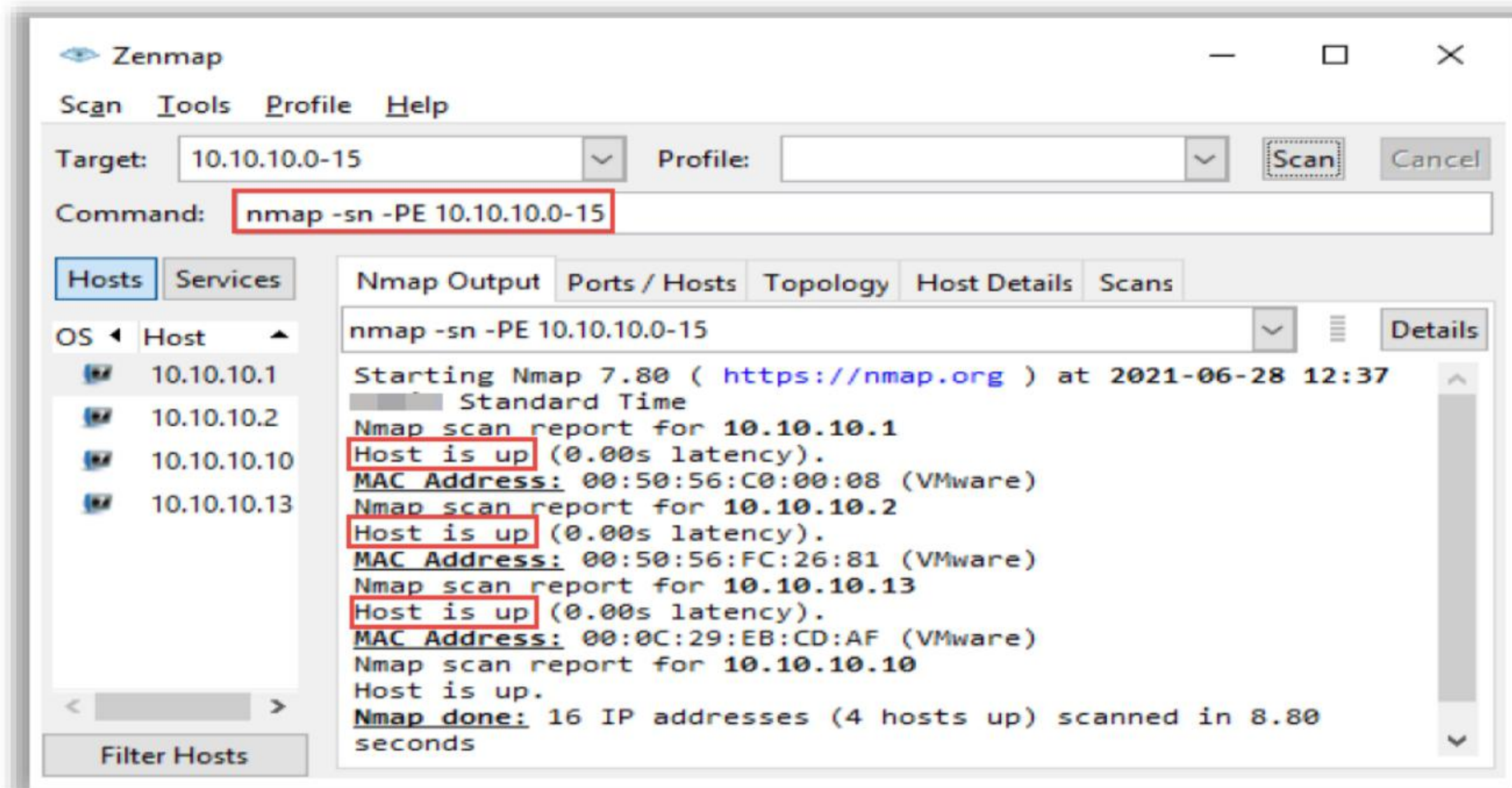


Figure 16.86: Screenshot of Nmap host discovery using an ICMP ECHO ping sweep

- To scan the given subnet, use the following command without any switches. This command works similarly to the ping command in that it sends TCP ACK packets for the ports 80 and 443 to check whether the target host is alive. It also performs an ARP scan and neighbor discovery scan; if it finds that any host is alive, it starts performing port scanning to detect running services.

nmap <IP subnet range>

Service and Version Discovery

- Use the -p option to scan for specified ports numbers or a port range.
nmap -p <Port Number or Range> <IP address>

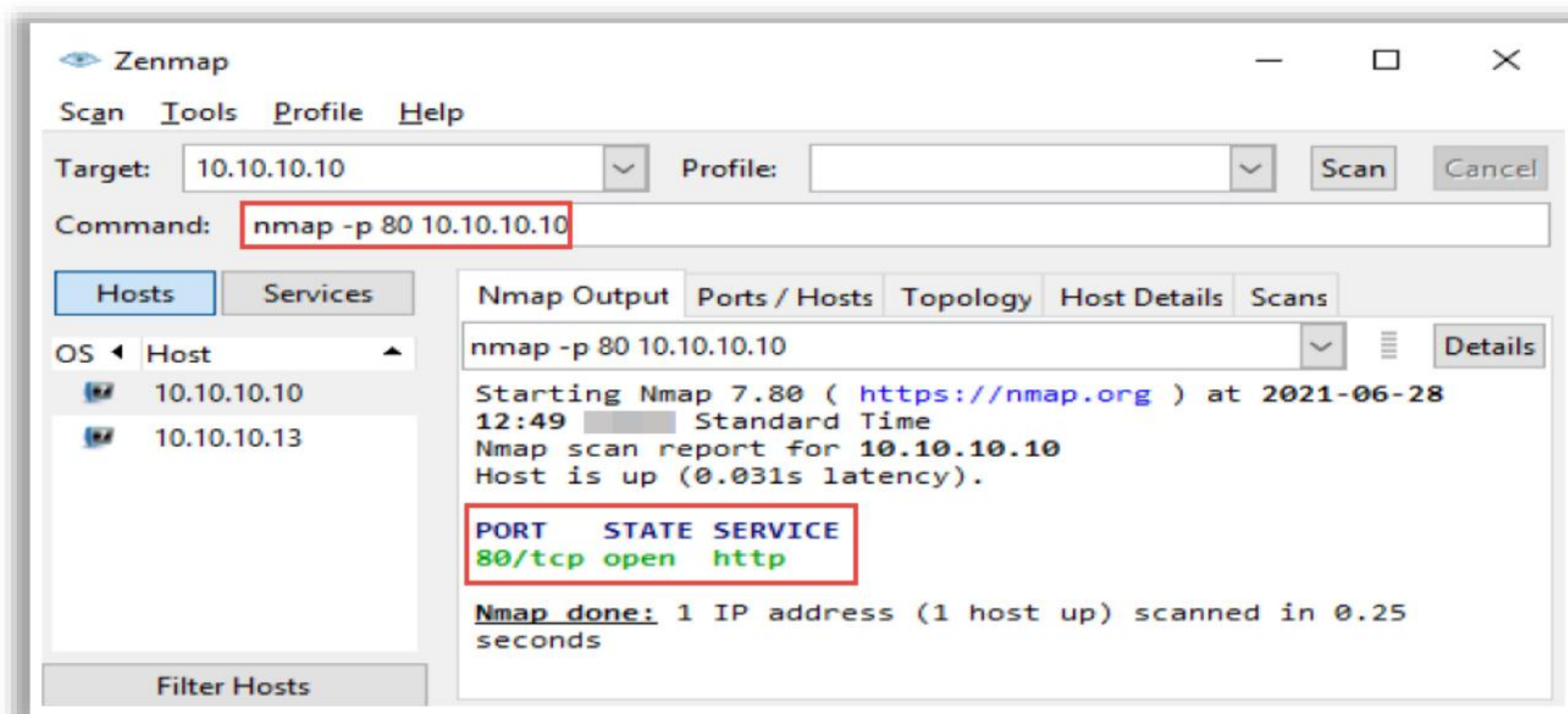


Figure 16.87: Screenshot of Nmap service discovery using a scan for a specified port number

- Use the `-sT` option to scan for only TCP ports.
`nmap -sT <IP address>`

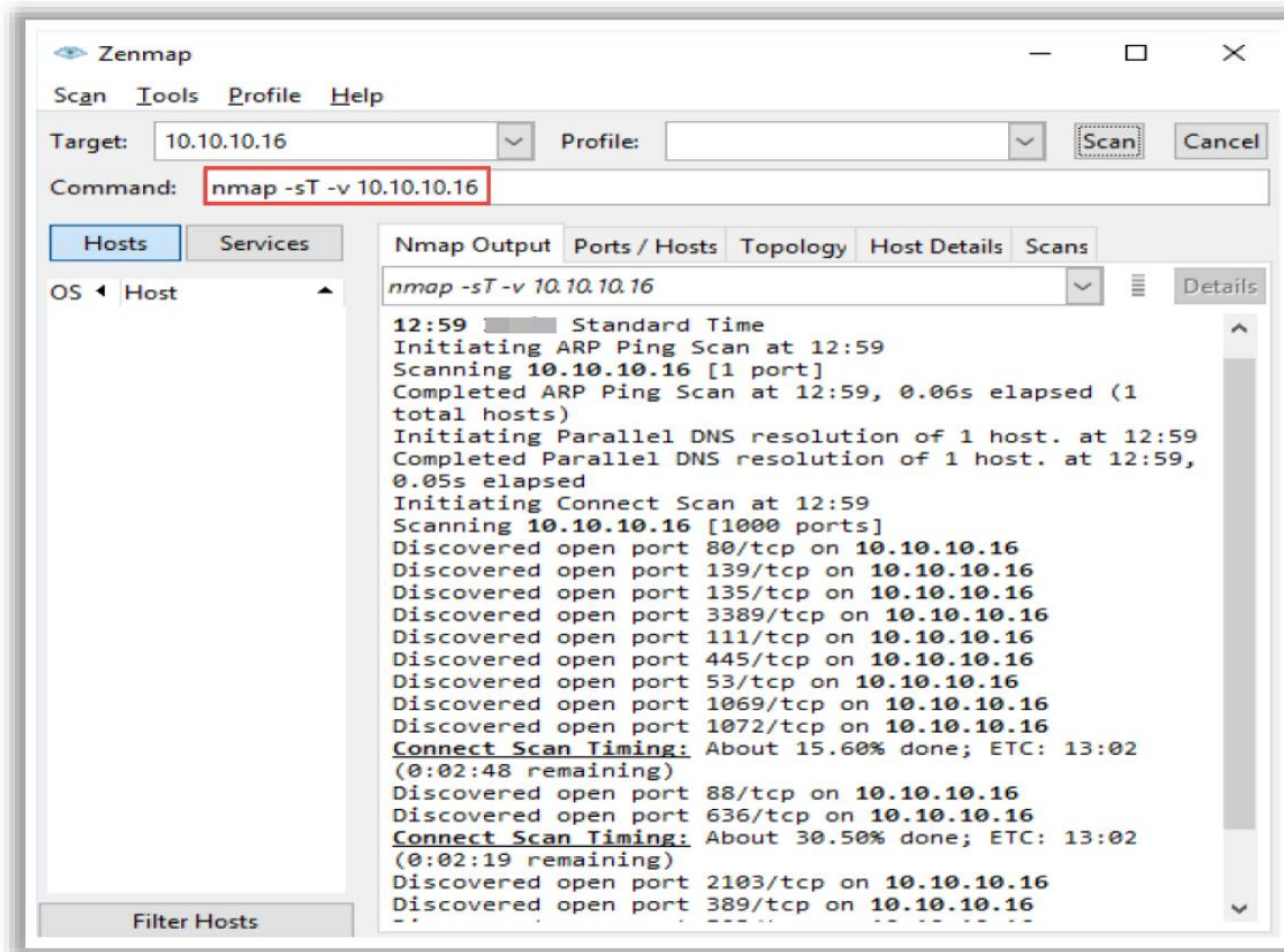


Figure 16.88: Screenshot of Nmap service discovery using a TCP scan

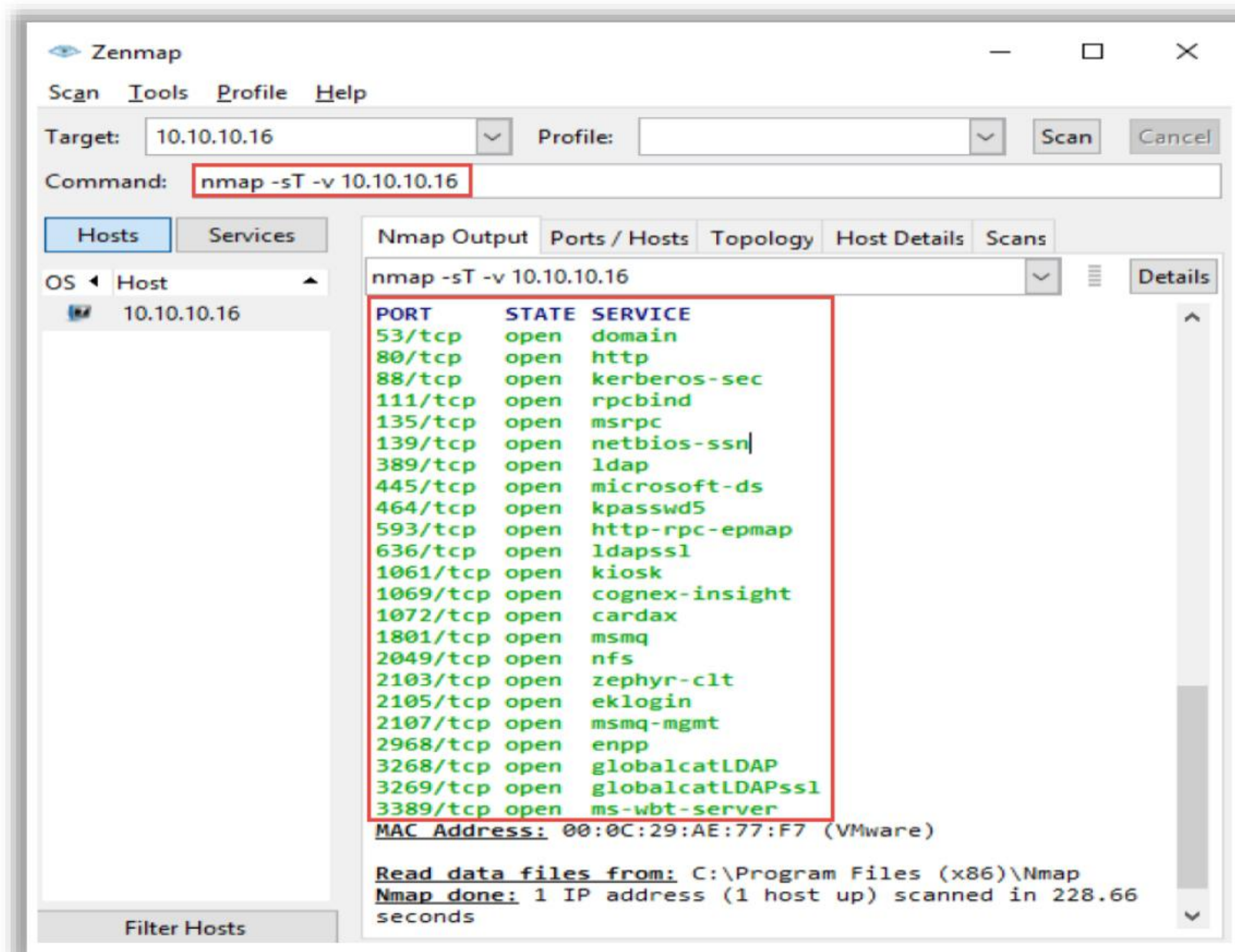


Figure 16.89: Screenshot of Nmap service discovery using a TCP scan

- Use the `-sU` option to scan for only UDP ports.

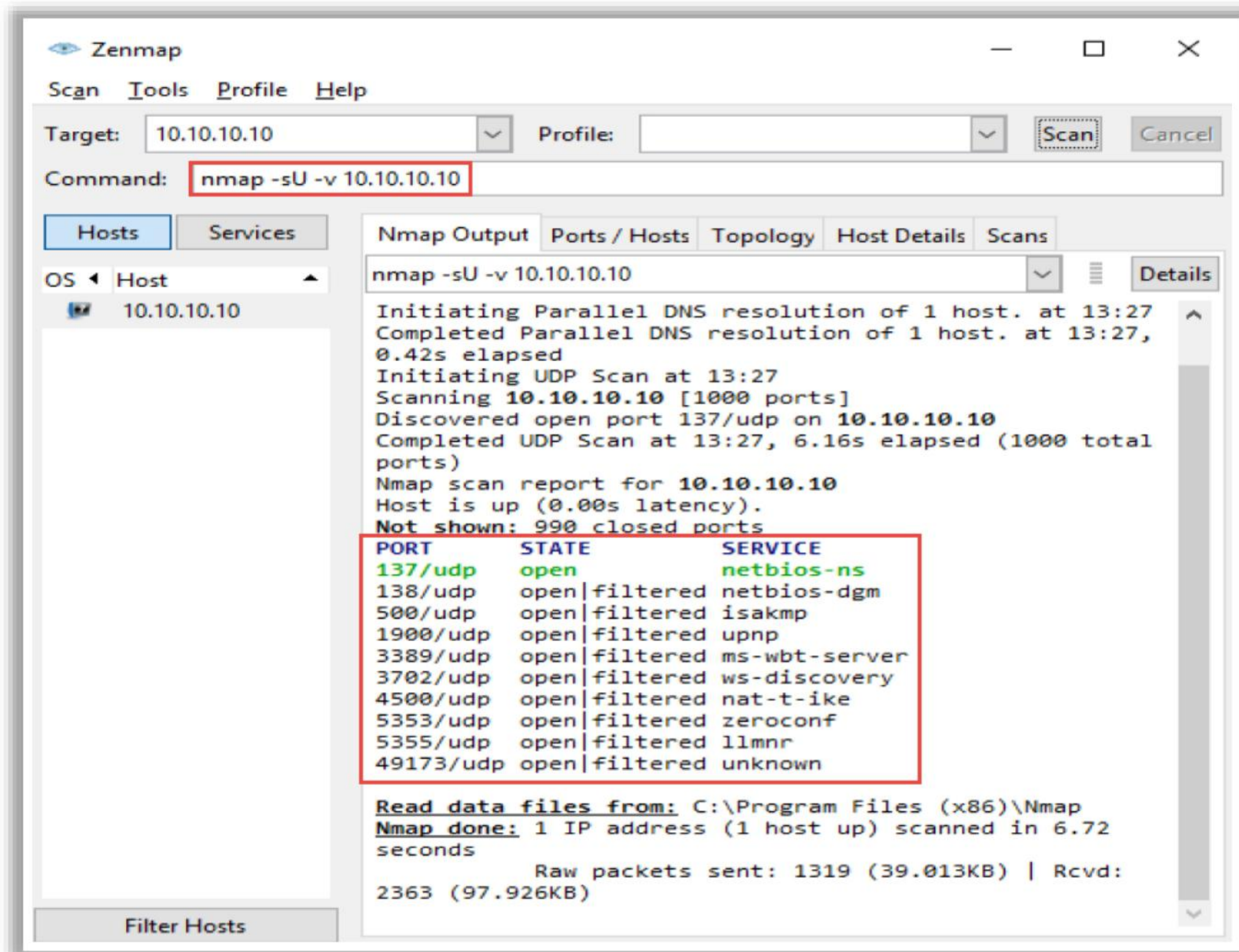


Figure 16.90: Screenshot of Nmap service discovery using a UDP scan

- Use the `-sS` option to perform a stealth scan/TCP half-open scan.
`nmap -sS <IP address>`

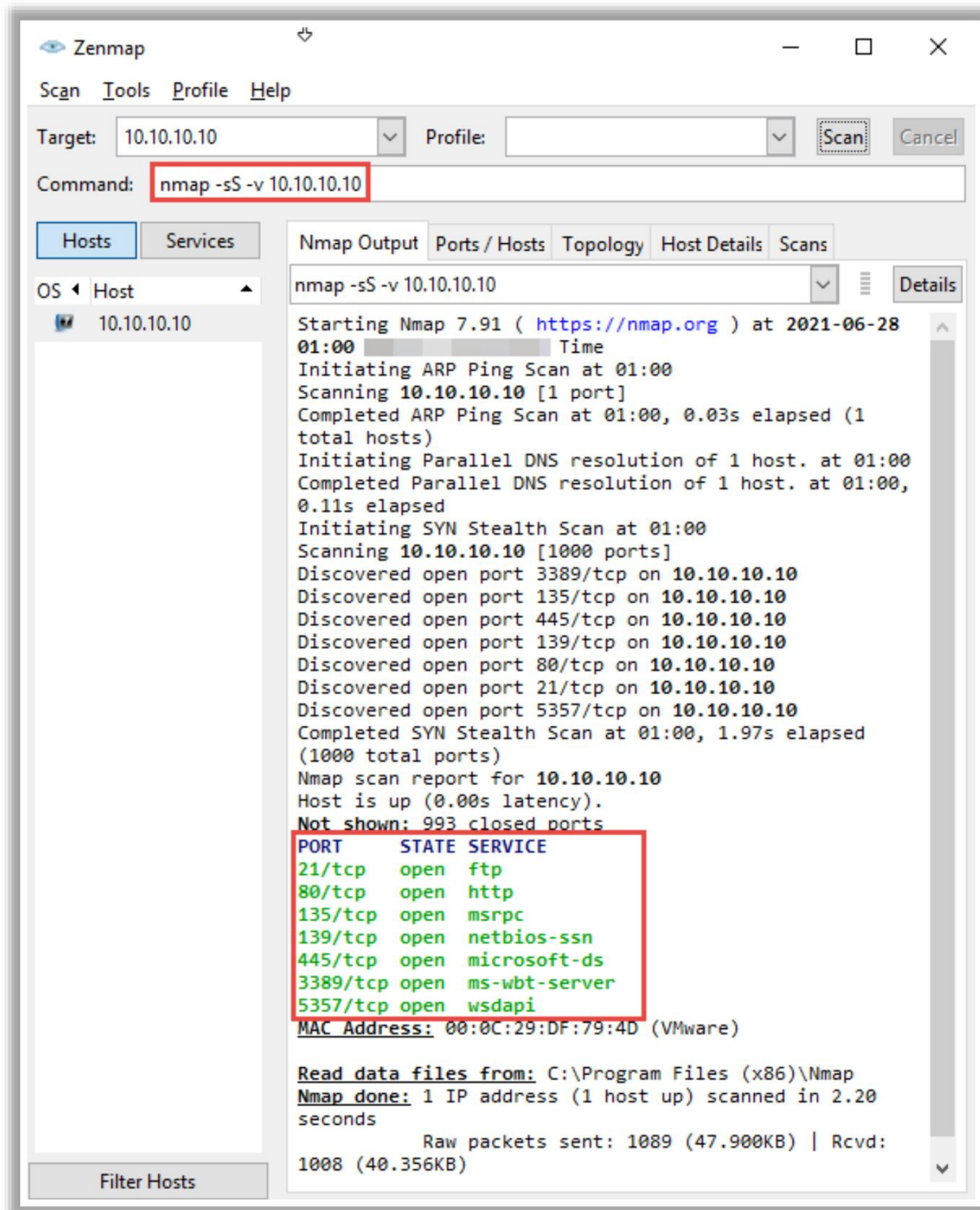


Figure 16.91: Screenshot of Nmap service discovery using a TCP half-open scan

- Use the `-sV` option to detect service versions.

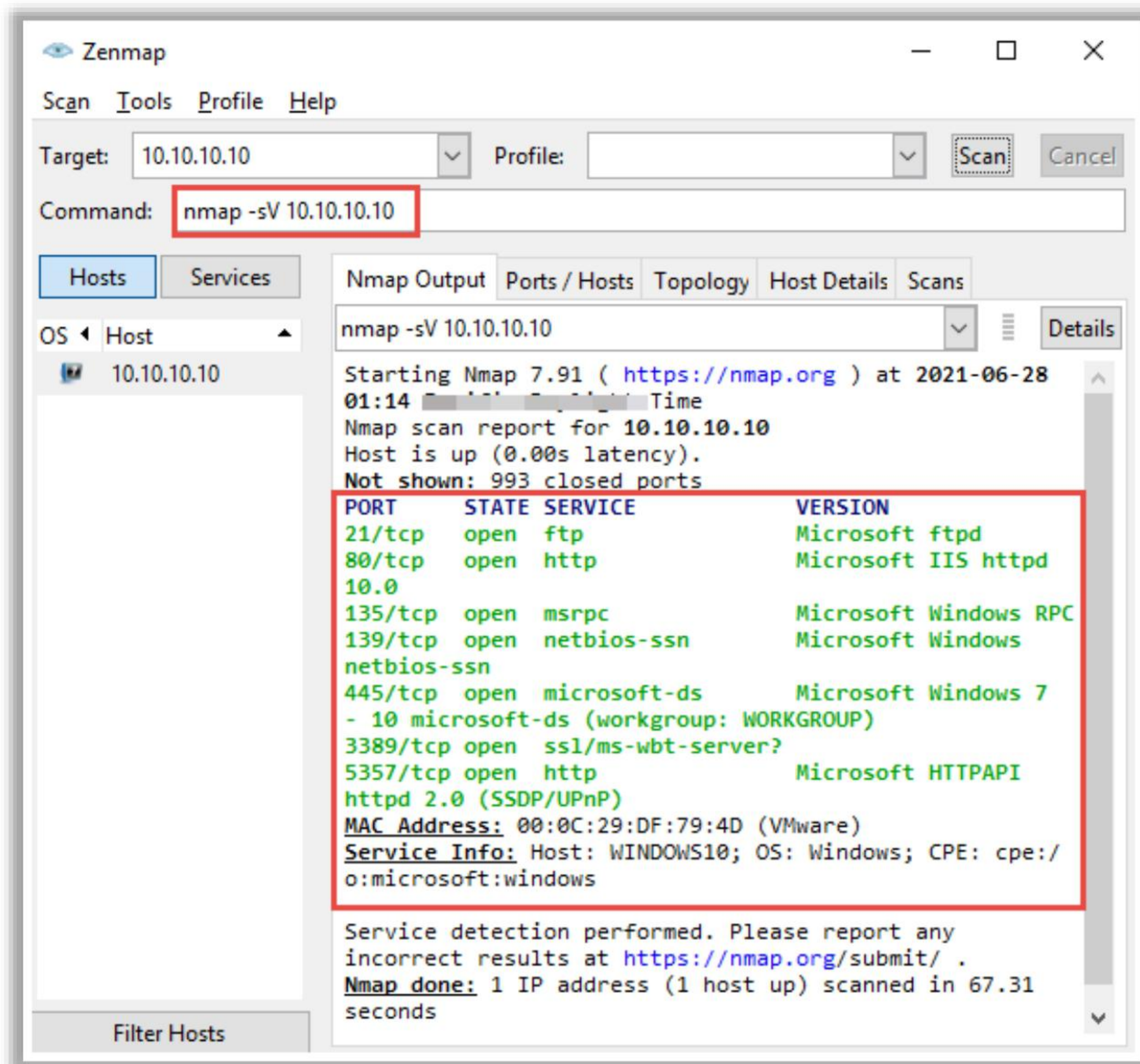


Figure 16.92: Screenshot of Nmap service discovery

- Use the following command to check 2000 common ports that manage UDP/TCP services:
`# nmap -sS -sU -PN <IP address>`
- Use the following command to scan all the ports:
`# nmap -sS -sU -PN -p 1-65535 <IP address>`

OS Discovery

- Use the `-O` option to perform OS discovery and obtain the OS details of the target machine.

`nmap -O <IP address>`

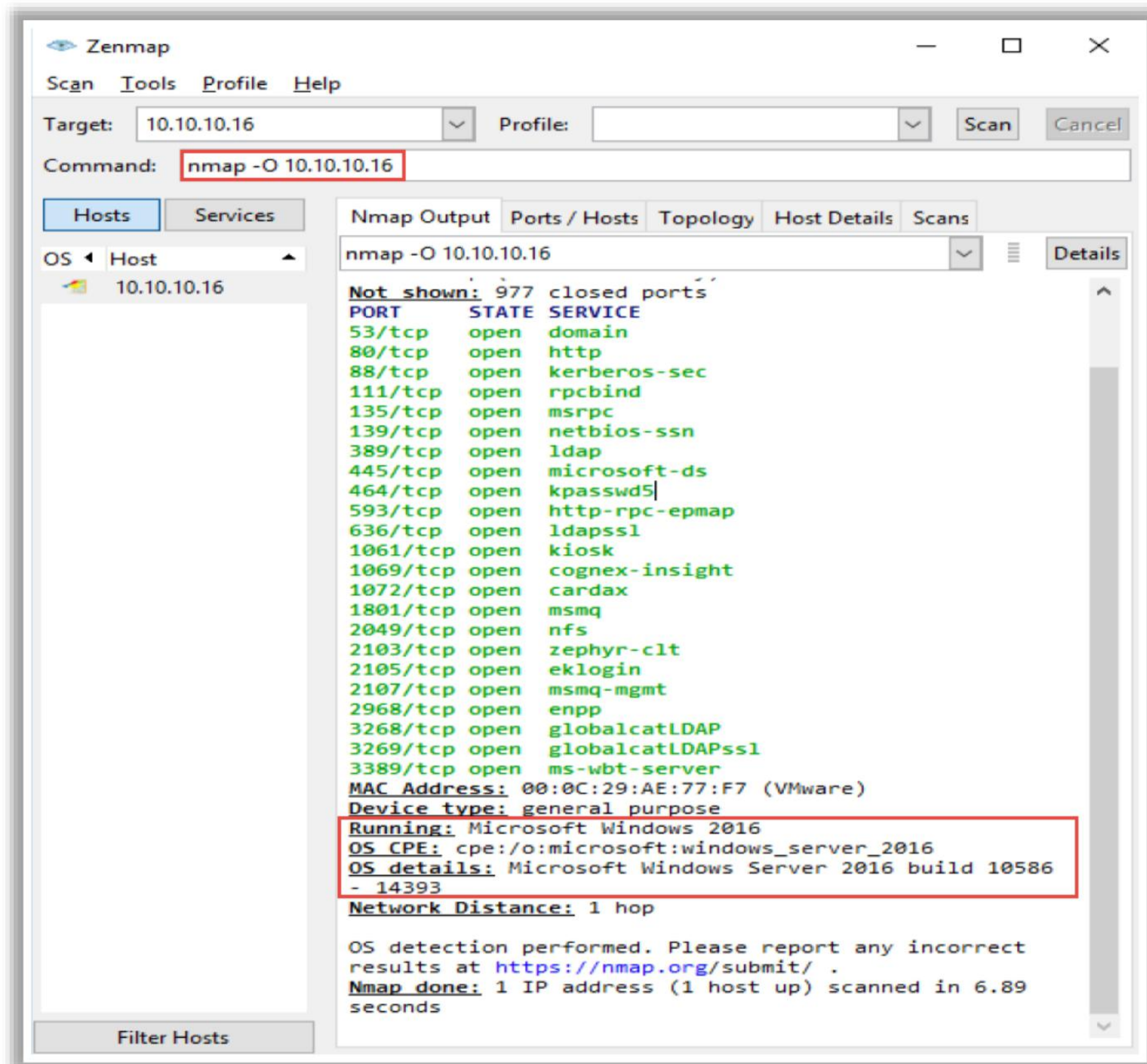


Figure 16.93: Screenshot of Nmap OS discovery scan

- In Nmap, use the **smb-os-discovery** NSE script for collecting OS information on the target machine through the SMB protocol.

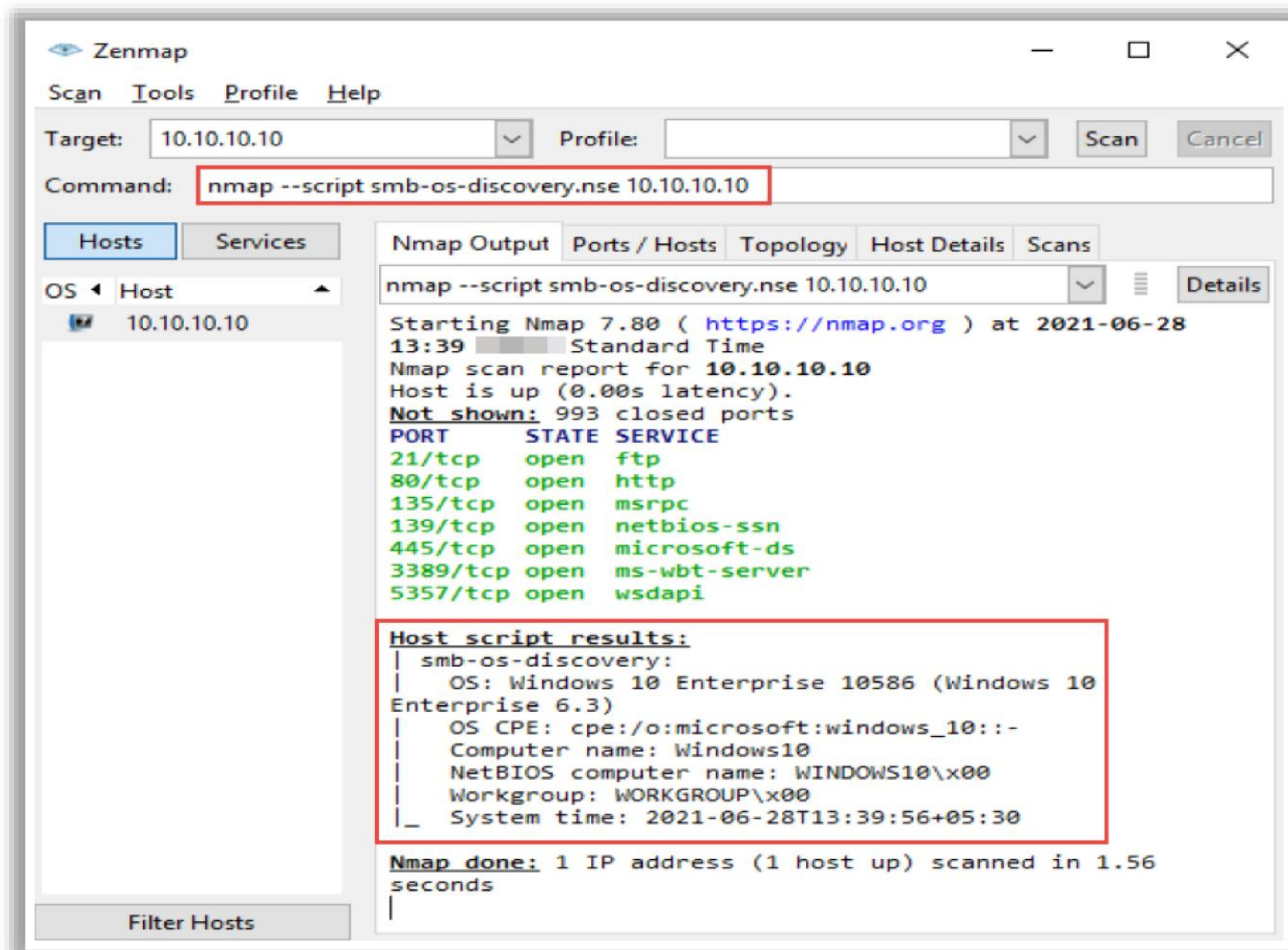


Figure 16.94: Screenshot of Nmap OS discovery using the Nmap script engine (NSE)

■ Wireshark

Source: <https://www.wireshark.org>

Wireshark allows capturing and interactively browsing the traffic in a computer network. This tool uses WinPcap to capture packets on its own supported networks. It captures live network traffic from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, and FDDI networks. The captured files can be programmatically edited via the command-line interface. A set of filters for customized data display can be refined using a display filter.

Wireshark assists administrators in troubleshooting network problems and performing real-time traffic analysis for diagnosing network-related issues. Wireshark can be used to troubleshoot some common issues such as packet drops, delay problems, and unnecessary activities over the network.

As shown in the screenshot, Wireshark can be used to sniff and analyze the packet flow in the target network and extract critical information.

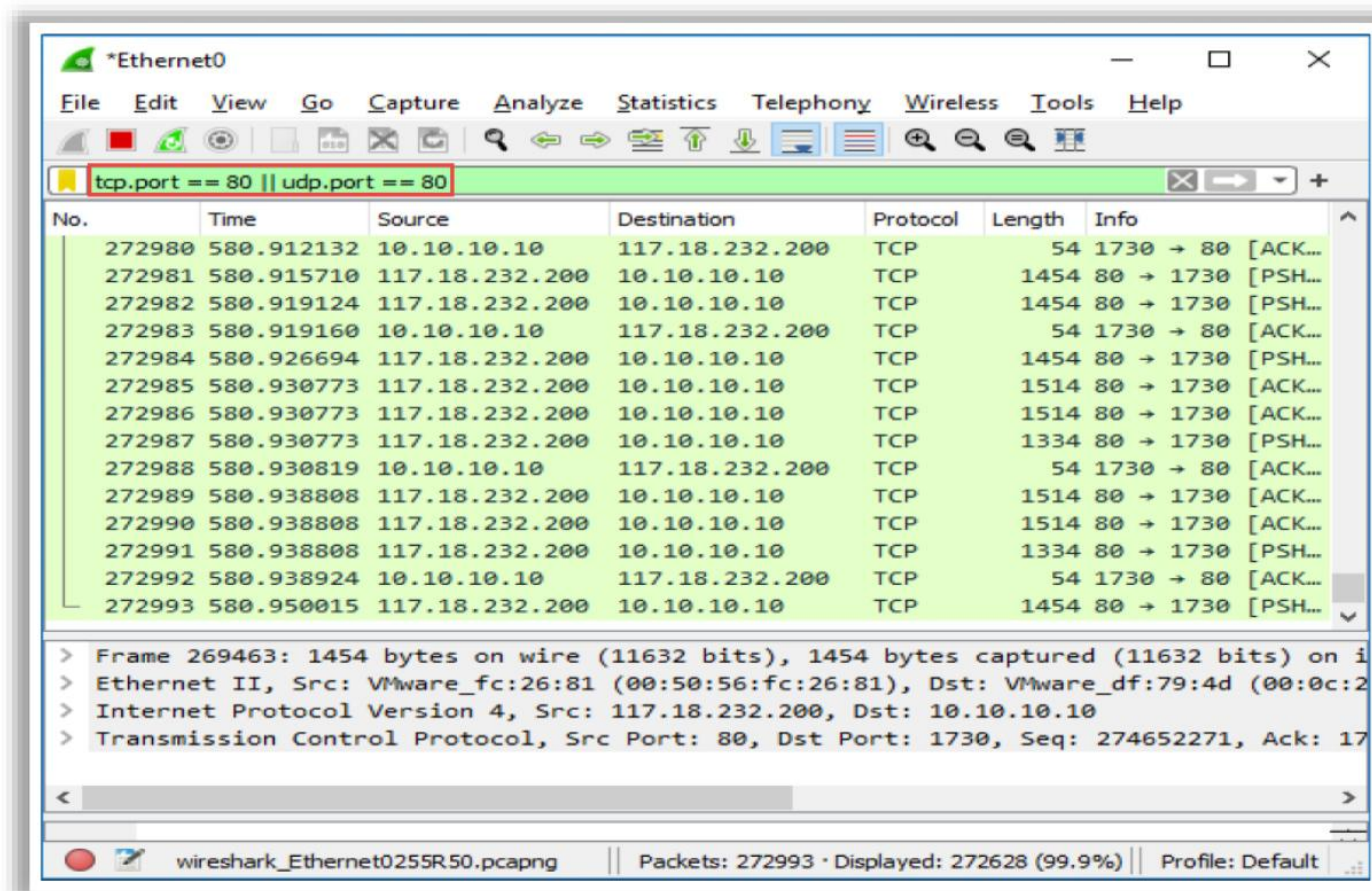


Figure 16.95: Screenshot of Wireshark

Wireshark features display filters that filter traffic on the target network by protocol type, IP address, port, etc. Display filters are used to change the view of packets in the captured files. To set up a filter, type the protocol name, such as arp, http, tcp, udp, dns, and ip, in the filter box of Wireshark. Wireshark can use multiple filters at a time.

Listed below are display filters in Wireshark that are commonly used for network troubleshooting.

- To investigate HTTP traffic, enter “http” as the filter option in the Wireshark window.

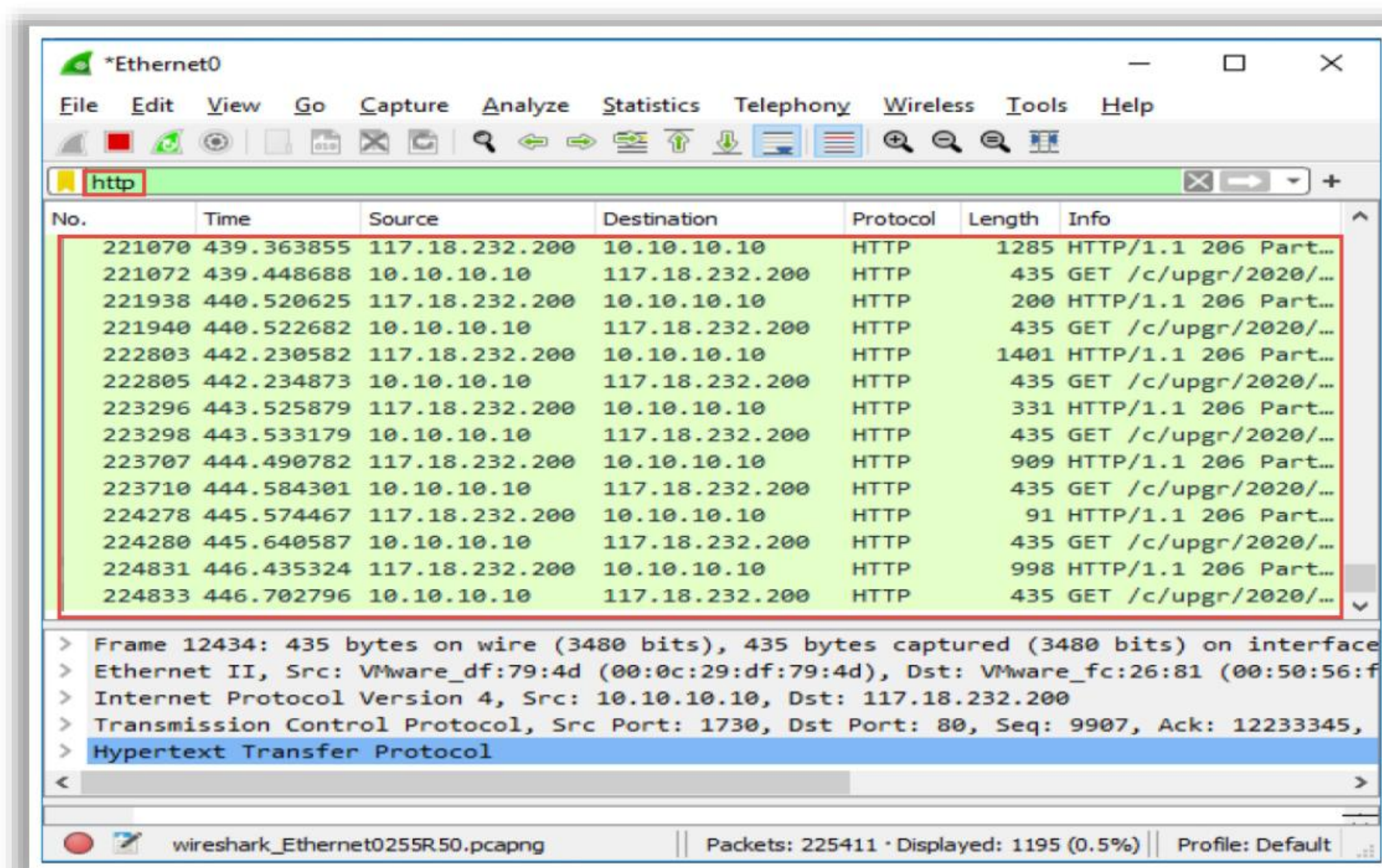


Figure 16.96: Screenshot of Wireshark showing an http filter

- Use the filter `ip.addr == <IP address>` to investigate HTTP traffic initiated from an IP address.

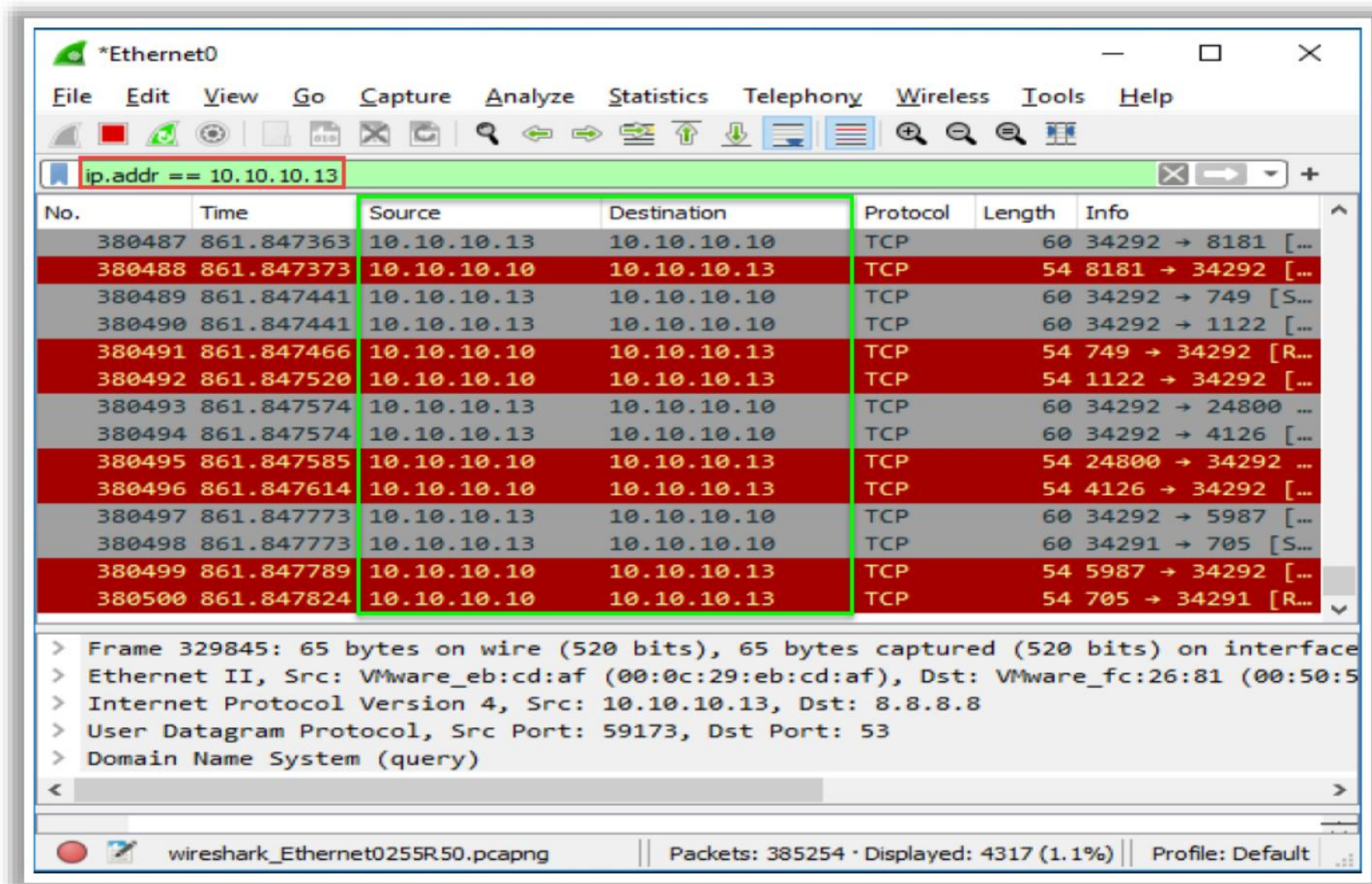


Figure 16.97: Screenshot of Wireshark showing ip.addr filter

- Use the filter `ip.dst==<IP address>&&http` to investigate HTTP traffic towards an IP address.
- Use the filter `!(ip.addr == <IP address>)` to discard packets destined to an IP address.
- Use the filter `ip.src==<IP address>/24 and ip.dst==<IP address>/24` to trace the local network traffic.
- To track the TCP data content, right-click on the selected packet and select “Follow TCP Stream.” A window will be displayed with TCP data content. The content includes headers and cleartext data forwarded while processing.

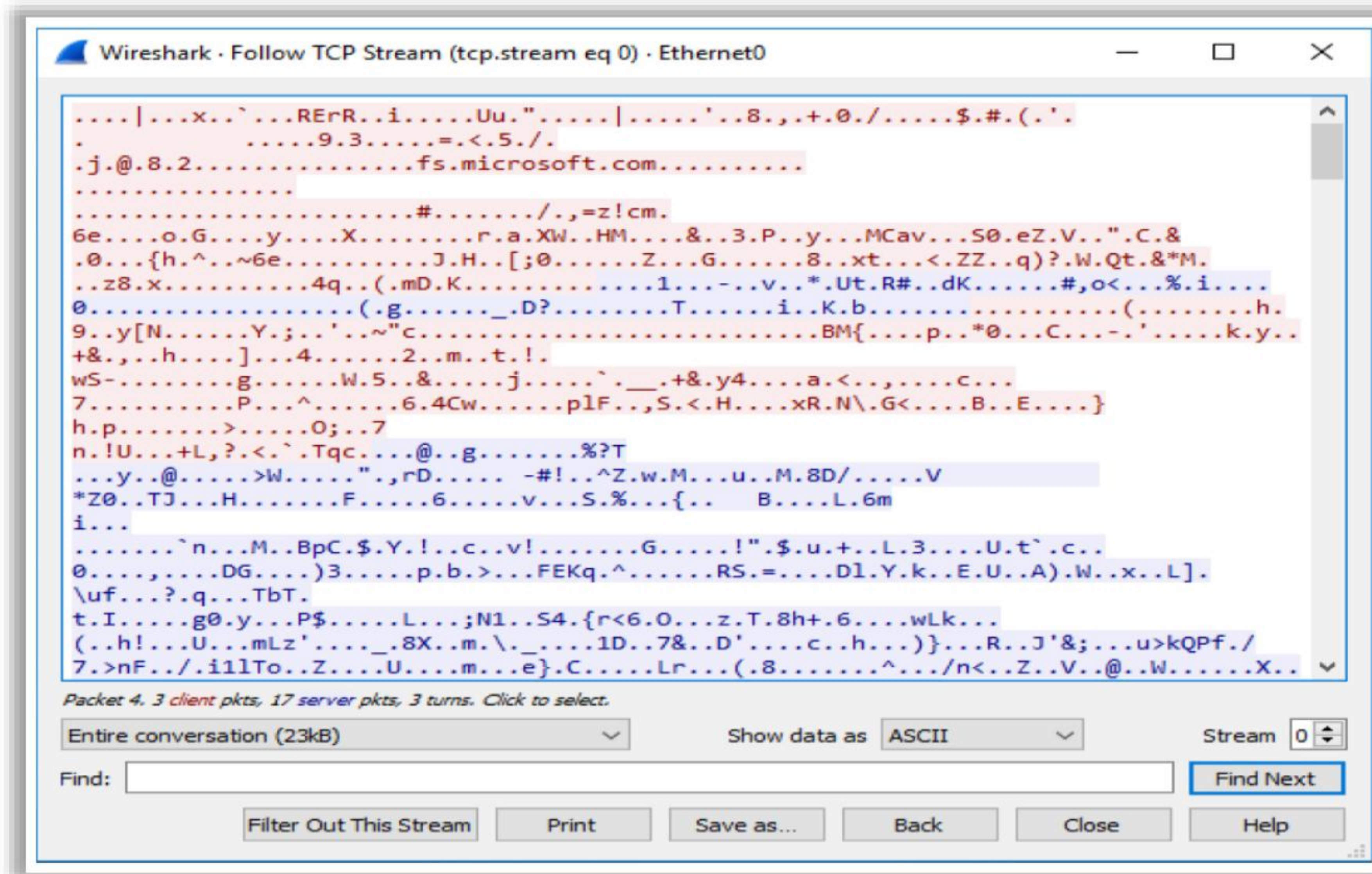


Figure 16.98: Screenshot of Wireshark showing the window “Follow TCP Stream”

- In the captured window, the traffic or any errors are displayed in different colors, which can be used for troubleshooting the network.

▪ **Hping2/Hping3**

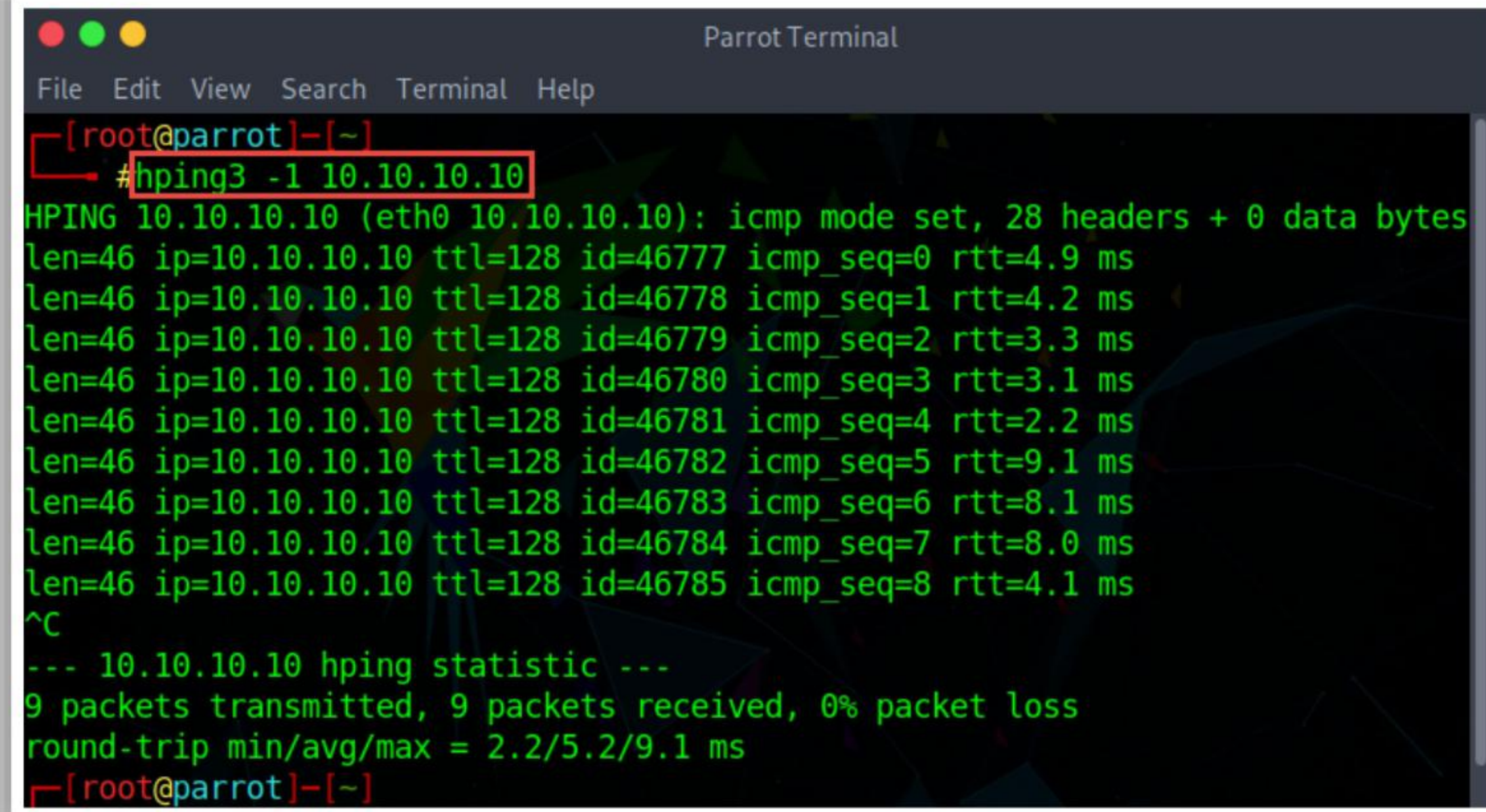
Hping2/Hping3 is a command-line-oriented network scanning and packet crafting tool for the TCP/IP protocol that sends ICMP echo requests and supports TCP, UDP, ICMP, and raw-IP protocols. It performs network security auditing, firewall testing, manual path MTU discovery, advanced traceroute, remote OS fingerprinting, remote uptime guessing, TCP/IP stacks auditing, and other functions. It can send custom TCP/IP packets and display target replies similarly to a ping program with ICMP replies. It handles fragmentation as well as arbitrary packet body and size, and it can be used to transfer encapsulated files under the supported protocols. It also supports idle host scanning. IP spoofing and network/host scanning can be used to perform an anonymous probe for services. It also determines whether the host is up even when the host blocks ICMP packets. Its firewall-like usage allows the discovery of open ports behind firewalls. It performs manual path MTU discovery and enables you to perform remote OS fingerprinting.

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.

Syntax: # **hping** <options> <Target IP address>

ICMP Scanning

A ping sweep or Internet Control Message Protocol (ICMP) scanning is a process of sending an ICMP request or ping to all the hosts on the network to determine the ones that are up.



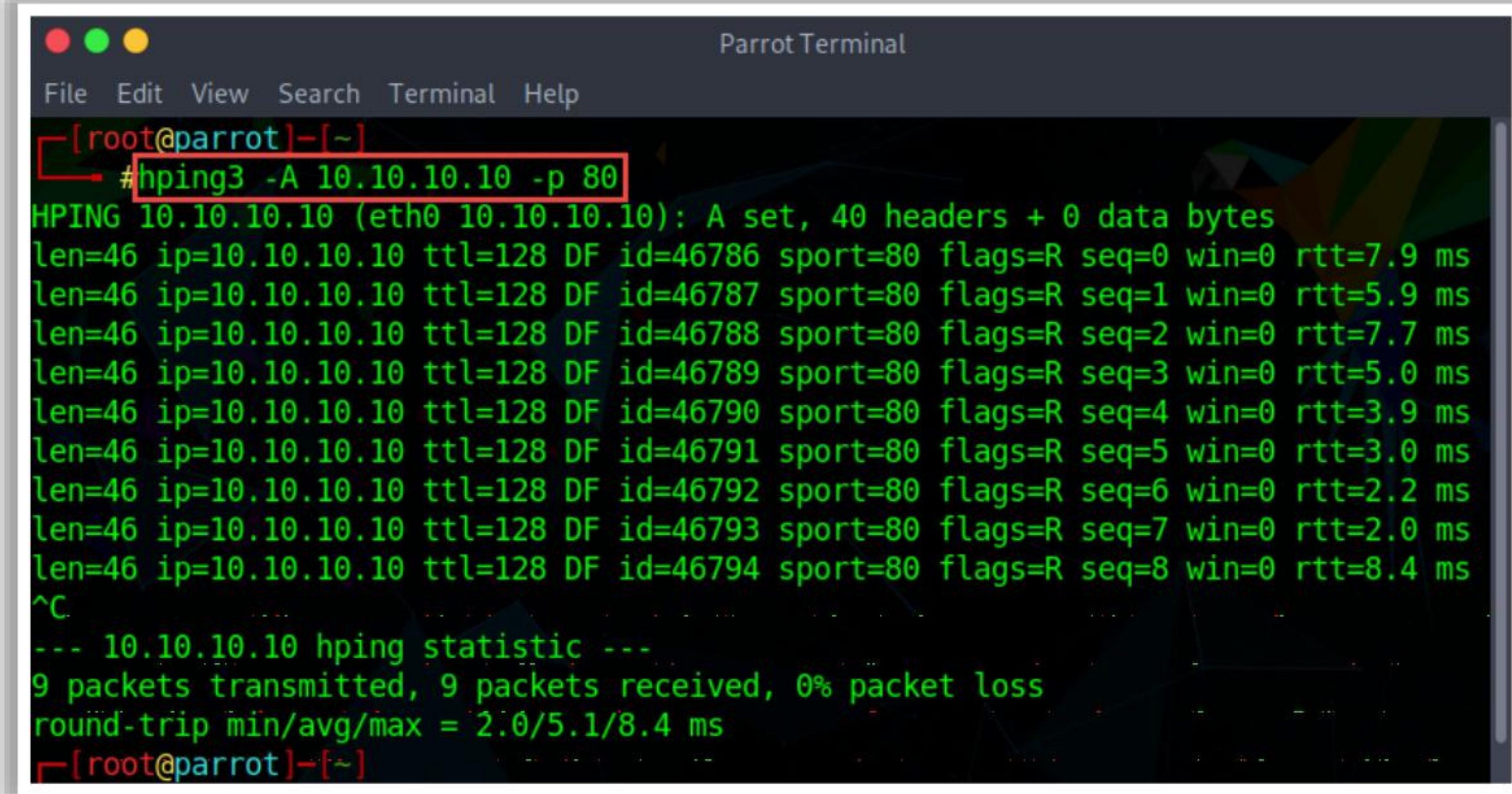
```
Parrot Terminal
File Edit View Search Terminal Help
[~]
#hping3 -1 10.10.10.10
HPING 10.10.10.10 (eth0 10.10.10.10): icmp mode set, 28 headers + 0 data bytes
len=46 ip=10.10.10.10 ttl=128 id=46777 icmp_seq=0 rtt=4.9 ms
len=46 ip=10.10.10.10 ttl=128 id=46778 icmp_seq=1 rtt=4.2 ms
len=46 ip=10.10.10.10 ttl=128 id=46779 icmp_seq=2 rtt=3.3 ms
len=46 ip=10.10.10.10 ttl=128 id=46780 icmp_seq=3 rtt=3.1 ms
len=46 ip=10.10.10.10 ttl=128 id=46781 icmp_seq=4 rtt=2.2 ms
len=46 ip=10.10.10.10 ttl=128 id=46782 icmp_seq=5 rtt=9.1 ms
len=46 ip=10.10.10.10 ttl=128 id=46783 icmp_seq=6 rtt=8.1 ms
len=46 ip=10.10.10.10 ttl=128 id=46784 icmp_seq=7 rtt=8.0 ms
len=46 ip=10.10.10.10 ttl=128 id=46785 icmp_seq=8 rtt=4.1 ms
^C
--- 10.10.10.10 hping statistic ---
9 packets transmitted, 9 packets received, 0% packet loss
round-trip min/avg/max = 2.2/5.2/9.1 ms
[~]
```

Figure 16.99: ICMP scanning using hping

The OS, router, switch, and IP-based devices use this protocol via the ping command for echo request and echo response as a connectivity tester between different hosts.

ACK Scanning on Port 80

This scanning technique can be used to probe the existence of a firewall and its rule sets. Simple packet filtering allows the establishment of a connection (packets with the ACK bit set), whereas a sophisticated stateful firewall does not allow the establishment of a connection.



```
Parrot Terminal
File Edit View Search Terminal Help
[~]
#hping3 -A 10.10.10.10 -p 80
HPING 10.10.10.10 (eth0 10.10.10.10): A set, 40 headers + 0 data bytes
len=46 ip=10.10.10.10 ttl=128 DF id=46786 sport=80 flags=R seq=0 win=0 rtt=7.9 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46787 sport=80 flags=R seq=1 win=0 rtt=5.9 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46788 sport=80 flags=R seq=2 win=0 rtt=7.7 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46789 sport=80 flags=R seq=3 win=0 rtt=5.0 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46790 sport=80 flags=R seq=4 win=0 rtt=3.9 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46791 sport=80 flags=R seq=5 win=0 rtt=3.0 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46792 sport=80 flags=R seq=6 win=0 rtt=2.2 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46793 sport=80 flags=R seq=7 win=0 rtt=2.0 ms
len=46 ip=10.10.10.10 ttl=128 DF id=46794 sport=80 flags=R seq=8 win=0 rtt=8.4 ms
^C
--- 10.10.10.10 hping statistic ---
9 packets transmitted, 9 packets received, 0% packet loss
round-trip min/avg/max = 2.0/5.1/8.4 ms
[~]
```

Figure 16.100: ACK scanning on port 80 using hping

Hping Commands

The various Hping commands are as follows:

- **ICMP ping**

Ex. `hping3 -1 10.0.0.25`

Hping performs an ICMP ping scan by specifying the argument -1 in the command line. You may use --ICMP or -1 as the argument in the command line. By issuing the above command, hping sends an ICMP echo request to 10.0.0.25 and receives an ICMP reply similarly to a ping utility.

- **ACK scan on port 80**

Ex. `hping3 -A 10.0.0.25 -p 80`

Hping can be configured to perform an ACK scan by specifying the argument -A in the command line. Here, you set the ACK flag in the probe packets and perform the scan. You perform this scan when a host does not respond to a ping request. By issuing this command, Hping checks if a host is alive on a network. If it finds a live host and an open port, it returns an RST response.

- **UDP scan on port 80**

Ex. `hping3 -2 10.0.0.25 -p 80`

Hping uses TCP as its default protocol. Using the argument -2 in the command line specifies that Hping operates in the UDP mode. You may use either --udp or -2 as the argument in the command line.

By issuing the above command, Hping sends UDP packets to port 80 on the host (10.0.0.25). It returns an ICMP port unreachable message if it finds the port closed and does not return a message if the port is open.

- **Collecting Initial Sequence Number**

Ex. `hping3 192.168.1.103 -Q -p 139 -s`

Using the argument -Q in the command line, Hping collects all the TCP sequence numbers generated by the target host (192.168.1.103).

- **Firewalls and Timestamps**

Ex. `hping3 -S 72.14.207.99 -p 80 --tcp-timestamp`

Many firewalls drop those TCP packets that do not have the TCP Timestamp option set. By adding the --tcp-timestamp argument in the command line, you can enable the TCP timestamp option in Hping and try to guess the timestamp update frequency and uptime of the target host (72.14.207.99).

- **SYN scan on port 50-60**

Ex. `hping3 -8 50-60 -S 10.0.0.25 -V`

Using the argument `-8` or `--scan` in the command line, you are operating Hping in the scan mode to scan a range of ports on the target host. Adding the argument `-S` allows you to perform a SYN scan.

Therefore, the above command performs a SYN scan on ports 50–60 on the target host.

- **FIN, PUSH and URG scan on port 80**

Ex. `hping3 -F -P -U 10.0.0.25 -p 80`

By adding the arguments `-F`, `-P`, and `-U` in the command line, you are setting FIN, PUSH, and URG packets in the probe packets. By issuing this command, you are performing FIN, PUSH, and URG scans on port 80 on the target host (10.0.0.25). If port 80 is open, you will not receive a response. If the port is closed, Hping will return an RST response.

- **Scan entire subnet for live host**

Ex. `hping3 -1 10.0.1.x --rand-dest -I eth0`

By issuing this command, Hping performs an ICMP ping scan on the entire subnet 10.0.1.x; in other words, it sends an ICMP echo request randomly (`--rand-dest`) to all the hosts from 10.0.1.0 to 10.0.1.255 that are connected to the interface eth0. The hosts whose ports are open will respond with an ICMP reply. In this case, you have not set a port; hence, Hping sends packets to port 0 on all IP addresses by default.

- **Intercept all traffic containing HTTP signature**

Ex. `hping3 -9 HTTP -I eth0`

The argument `-9` will set the Hping to the listen mode. Hence, by issuing the command `-9 HTTP`, Hping starts listening on port 0 (of all the devices connected in the network to interface eth0), intercepts all the packets containing the HTTP signature, and dumps from the signature end to the packet's end.

For example, on issuing the command `hping2 -9 HTTP`, if Hping reads a packet that contains data `234-09sdf1kjs45-HTTPhello_world`, it will display the result as `hello_world`.

- **SYN flooding a victim**

Ex. `hping3 -S 192.168.1.1 -a 192.168.1.254 -p 22 --flood`

TCP SYN flooding techniques using spoofed IP addresses can be adopted to perform a DoS attack for testing the target system.

- **netcat**

netcat is a network utility used for testing network connections on both Windows- and Linux-based systems. It provides various commands that allow the thorough monitoring of traffic flow across a network. netcat can be used to establish and analyze TCP/UDP connections. Moreover, it can be used as a backdoor to read and write raw data over

network connections. It provides features such as port scanning, OS fingerprinting, file transferring, DNS checking, and source routing.

netcat commands for network troubleshooting

- Test the connectivity with a remote host using the following command:

```
nc -w3 -4 -v <target host> <port number>
```

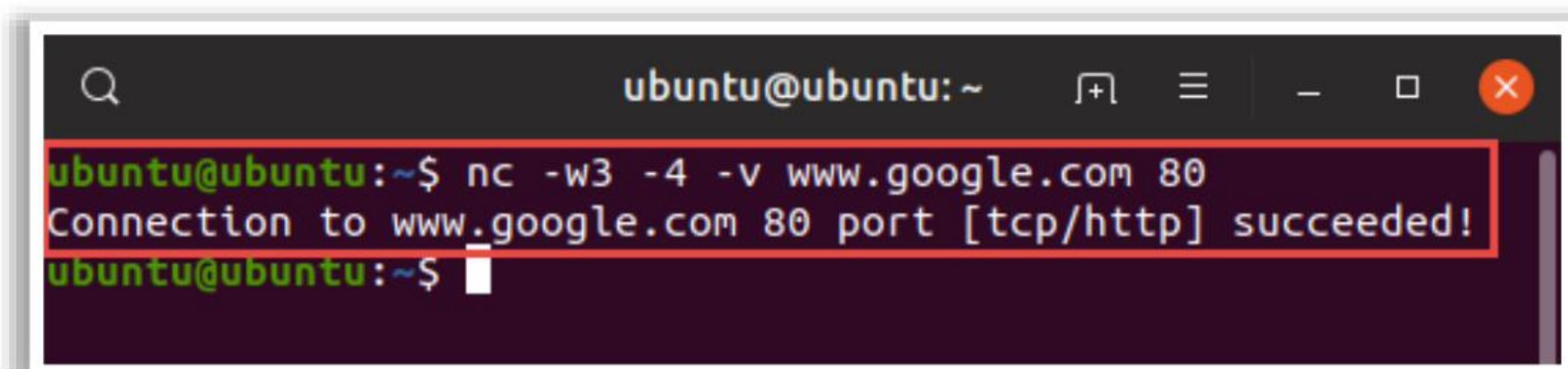


Figure 16.101: Screenshot of netcat

- Perform port scanning on a given IP address or host name using the following command:

```
nc -v -n <IP address> <port number/range>
```

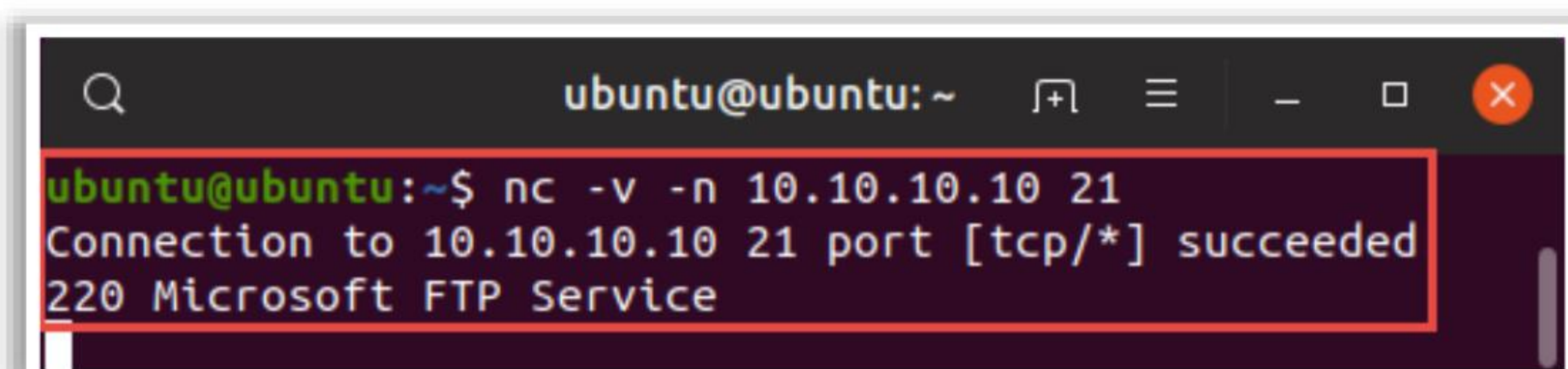


Figure 16.102: Screenshot of netcat performing a port scan

- Perform scan on a single port or given set of port numbers using the following command:

```
nc -zv <IP address> <port number(s)/port range>
```

- Create a listener on a specified port number using the following command:

```
nc -l -p <port number>
```

- Connect to a listening server using the following command:

```
nc <Server IP address> <port number>
```

- Create a proxy to redirect traffic destined to a host using the following command:

```
nc -l <port no> | nc <host name> < port number>
```

▪ dig

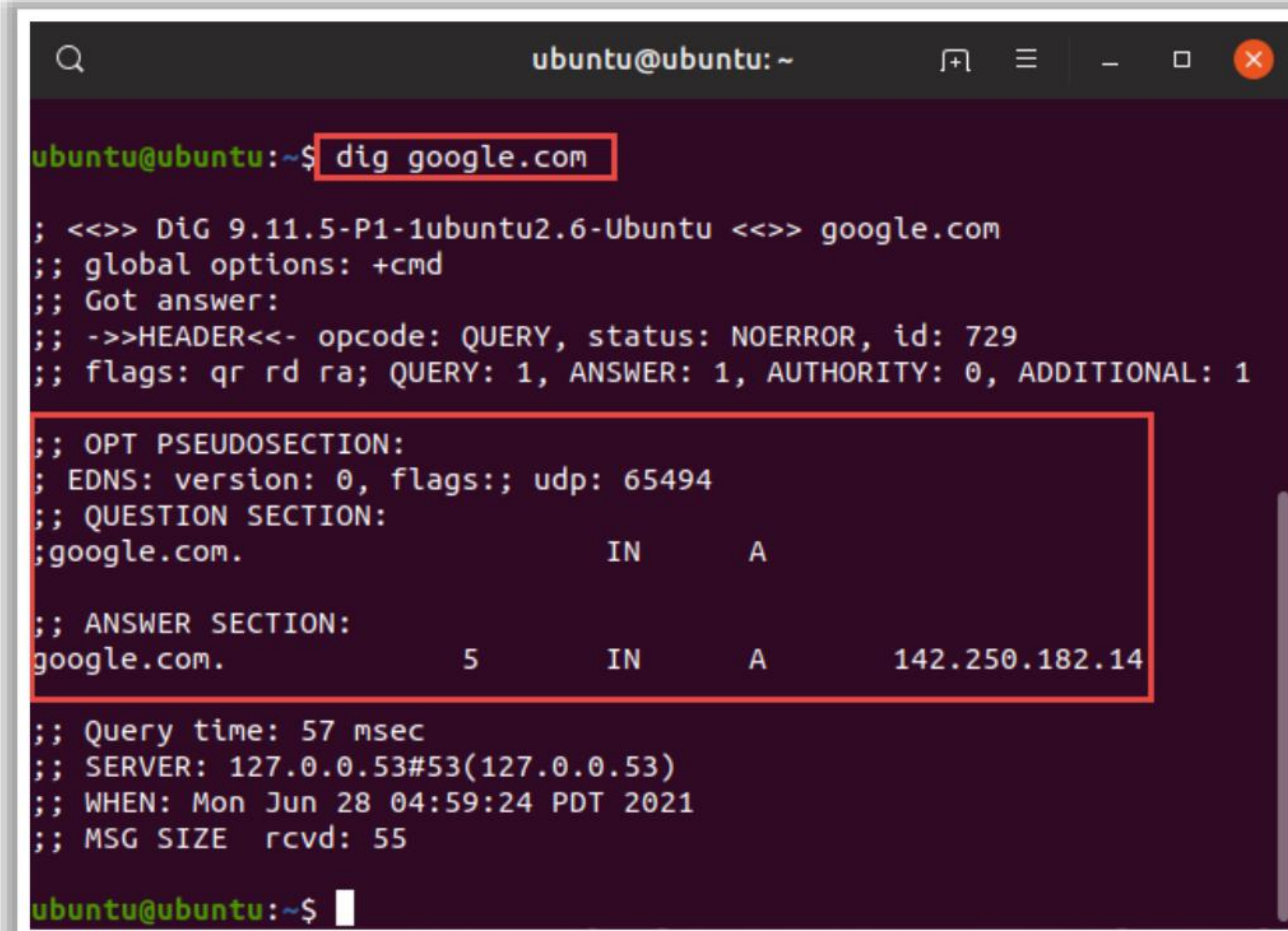
dig, which stands for “Domain Information Groper,” is used by the network administrators for troubleshooting networks and DNS name servers. It is a command-line tool used to query a DNS server directly to retrieve any type of DNS record information and to check whether the records are different when querying from different locations. It is a part of the BIND software suite. The dig command can be used

on Linux-based systems to query DNS name servers and retrieve information about the target host addresses, name servers, mail exchanges, etc.

dig commands

- Query any nameserver to retrieve DNS records using the following command:

dig <domain name>



```
ubuntu@ubuntu:~$ dig google.com

;<<>> DiG 9.11.5-P1-1ubuntu2.6-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 729
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;google.com.                IN      A
;; ANSWER SECTION:
google.com.                 5       IN      A      142.250.182.14

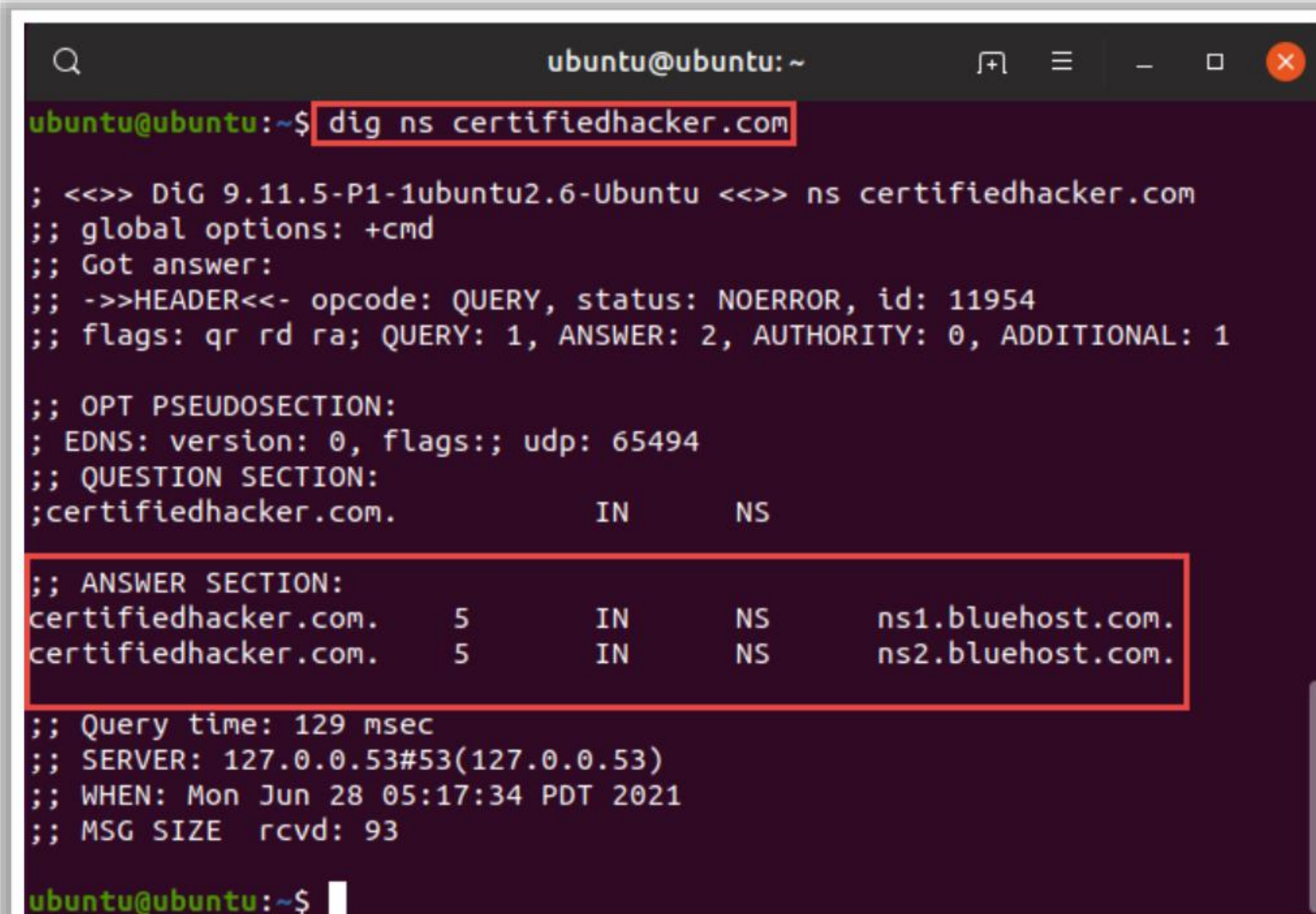
;; Query time: 57 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon Jun 28 04:59:24 PDT 2021
;; MSG SIZE rcvd: 55

ubuntu@ubuntu:~$
```

Figure 16.103: Screenshot of dig retrieving DNS records

- Retrieve a specific type of DNS record of a domain using the following command:

dig <record type> <domain name>



```
ubuntu@ubuntu:~$ dig ns certifiedhacker.com

;<<>> DiG 9.11.5-P1-1ubuntu2.6-Ubuntu <<>> ns certifiedhacker.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11954
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

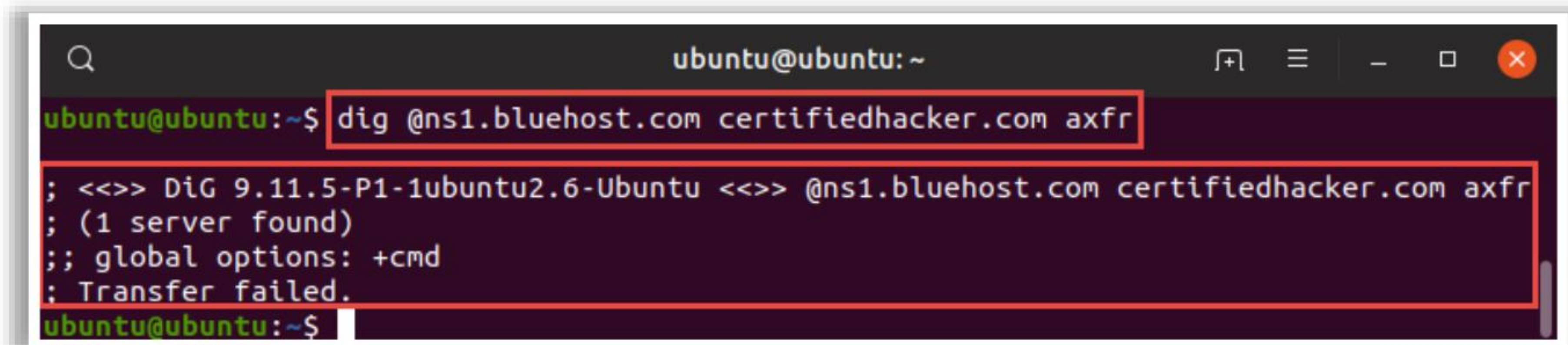
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;certifiedhacker.com.      IN      NS
;; ANSWER SECTION:
certifiedhacker.com.      5       IN      NS     ns1.bluehost.com.
certifiedhacker.com.      5       IN      NS     ns2.bluehost.com.

;; Query time: 129 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon Jun 28 05:17:34 PDT 2021
;; MSG SIZE rcvd: 93

ubuntu@ubuntu:~$
```

Figure 16.104: Screenshot of dig retrieving specific DNS record

- Display all the DNS records of a domain using the following command:
dig <domain name> ANY
- dig uses the local DNS configuration gateway setting to query a name server; it can be changed by specifying a hostname or IP address prefix with the “@” symbol:
dig <domain name> @ <IP address>
- Test whether the target DNS allows zone transfers using the following command:
dig @<domain of name server> <target domain> axfr



```
ubuntu@ubuntu:~$ dig @ns1.bluehost.com certifiedhacker.com axfr
; <<>> DiG 9.11.5-P1-1ubuntu2.6-Ubuntu <<>> @ns1.bluehost.com certifiedhacker.com axfr
; (1 server found)
;; global options: +cmd
; Transfer failed.
ubuntu@ubuntu:~$
```

Figure 16.105: Screenshot of dig testing DNS zone transfer

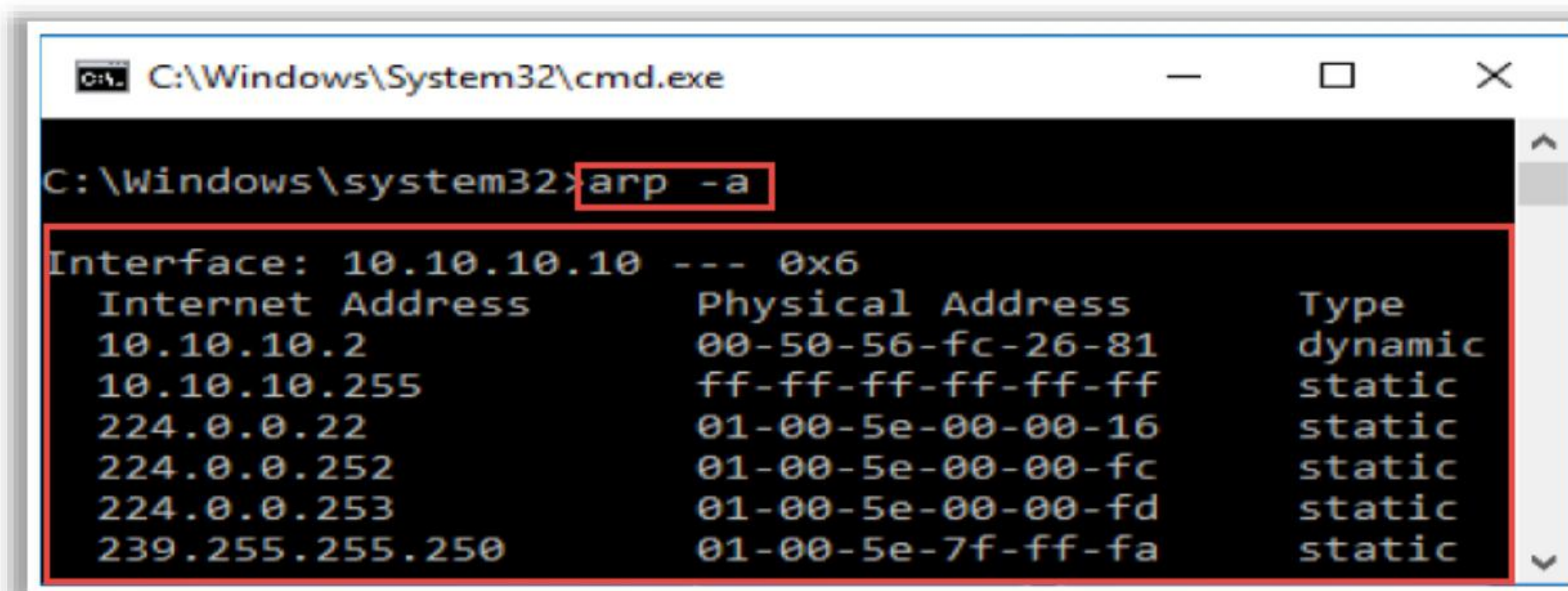
▪ **arp**

arp is a command-line tool used by network administrators to troubleshoot the ARP cache address table in network devices such as routers, switches, and other routing devices. It is used to check the ARP table to determine the presence of any conflict between the allocated logical addresses and a physical address. This tool allows making changes to the ARP table manually. Common conflicts in the ARP table such as duplicate IP addresses or duplicate MAC addresses can be easily identified using this command-line tool.

arp commands

- Display the current ARP cache table in a Windows system using the following command:

arp -a



```
C:\Windows\System32\cmd.exe
C:\Windows\system32>arp -a
Interface: 10.10.10.10 --- 0x6
Internet Address      Physical Address      Type
10.10.10.2            00-50-56-fc-26-81    dynamic
10.10.10.255          ff-ff-ff-ff-ff-ff    static
224.0.0.22            01-00-5e-00-00-16    static
224.0.0.252           01-00-5e-00-00-fc    static
224.0.0.253           01-00-5e-00-00-fd    static
239.255.255.250       01-00-5e-7f-ff-fa    static
```

Figure 16.106: Screenshot of the arp tool displaying the ARP cache table

- Display the MAC address for a specific IP address using the following command:

```
arp -a <IP address>
```

- Remove an entry from an ARP cache table using the following command:

```
arp -d <IP address>
```

- Remove all entries from an ARP cache table using the following command:

```
arp -d *
```

- Add a static entry to an existing ARP cache table using the following command:

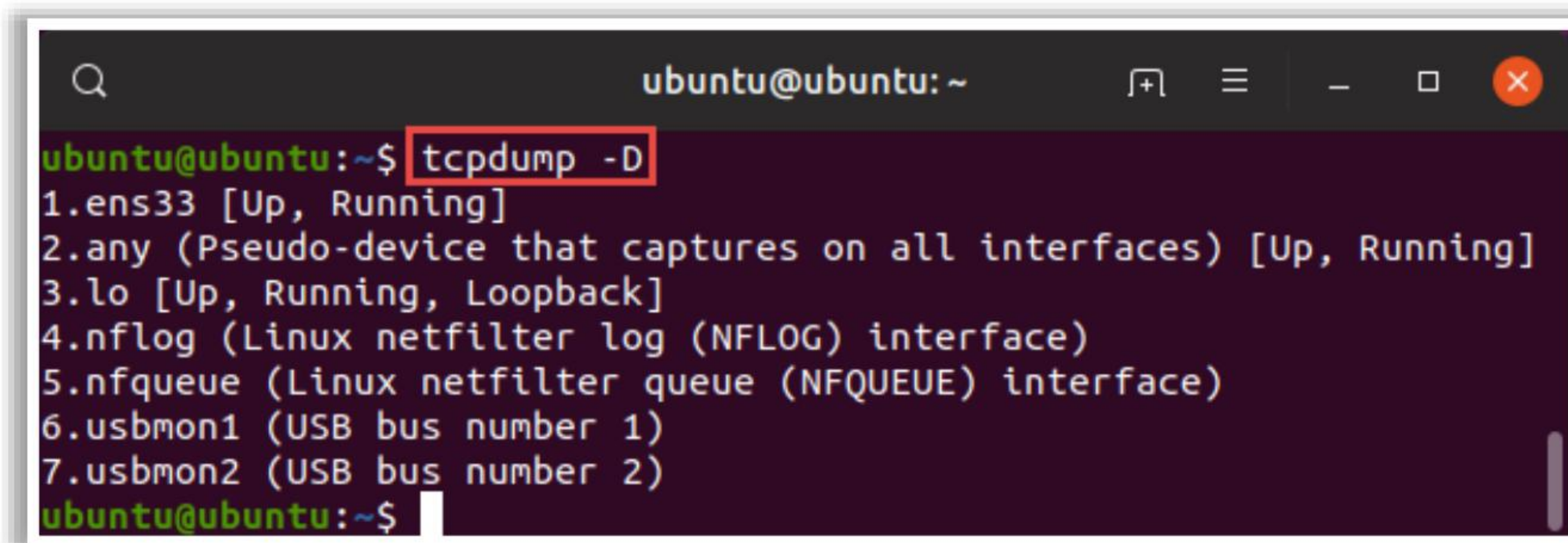
```
arp -s <IP address> <MAC address>
```

■ tcpdump

tcpdump is a network packet capturing tool used by network administrators for analyzing network traffic and troubleshooting network issues. It is a command-line tool available for both Linux and Windows OSes. It captures the traffic from any type of network interface and uses filters to regulate the captured traffic. It helps administrators easily identify problems in the network traffic.

- Display all the available network interfaces to capture traffic using the following command:

```
tcpdump -D
```



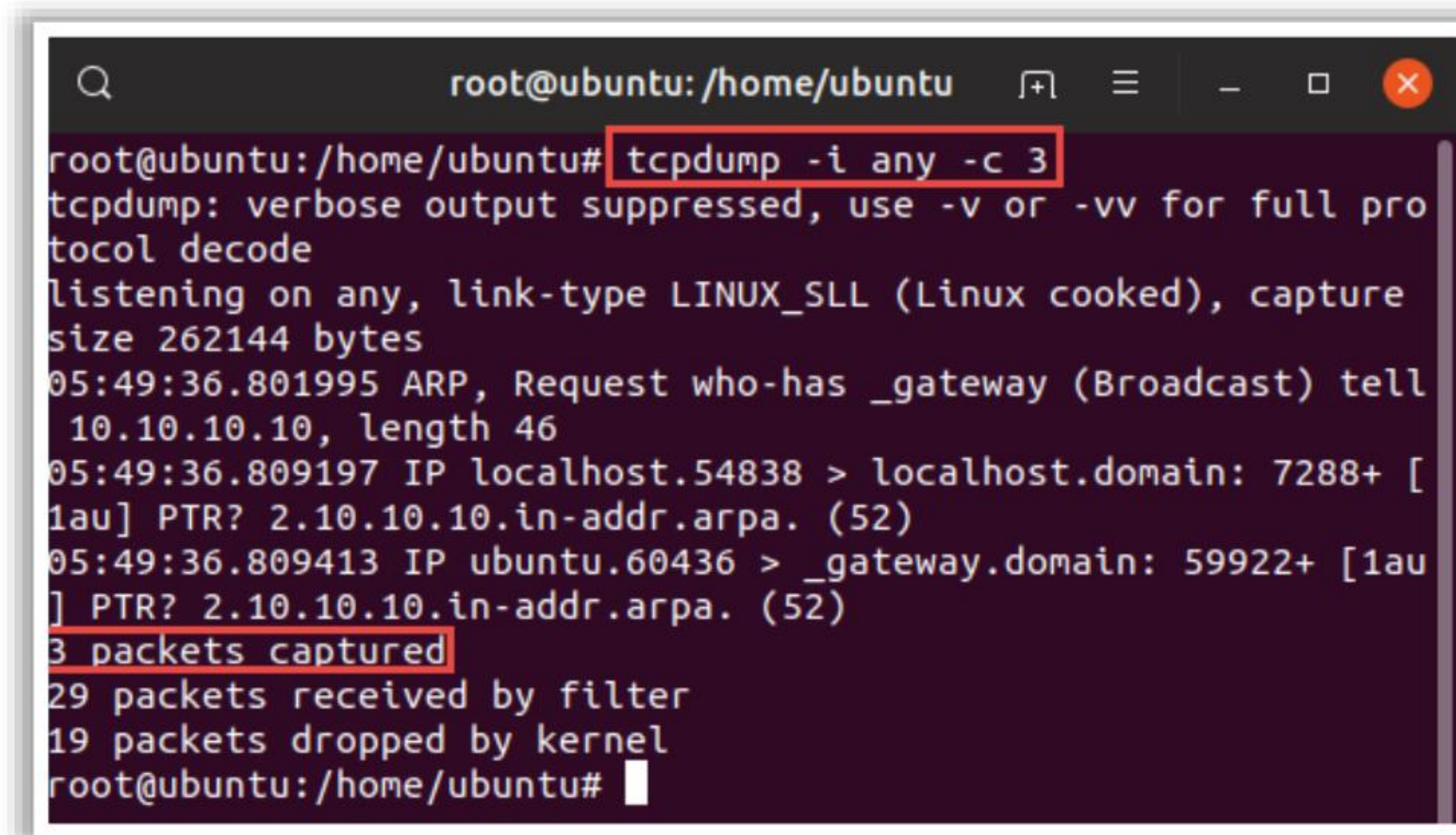
```
ubuntu@ubuntu:~$ tcpdump -D
1.ens33 [Up, Running]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.nflog (Linux netfilter log (NFLOG) interface)
5.nfqueue (Linux netfilter queue (NFQUEUE) interface)
6.usbmon1 (USB bus number 1)
7.usbmon2 (USB bus number 2)
ubuntu@ubuntu:~$
```

Figure 16.107: Screenshot of tcpdump displaying available network interfaces

- Start capturing the traffic from all interfaces or a specified interface using the following command:

```
tcpdump -i <interface name> -c 10
```

Here, `-c` specifies the number of packets to capture



```
root@ubuntu:/home/ubuntu# tcpdump -i any -c 3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
05:49:36.801995 ARP, Request who-has _gateway (Broadcast) tell 10.10.10.10, length 46
05:49:36.809197 IP localhost.54838 > localhost.domain: 7288+ [1au] PTR? 2.10.10.10.in-addr.arpa. (52)
05:49:36.809413 IP ubuntu.60436 > _gateway.domain: 59922+ [1au] PTR? 2.10.10.10.in-addr.arpa. (52)
3 packets captured
29 packets received by filter
19 packets dropped by kernel
root@ubuntu:/home/ubuntu#
```

Figure 16.108: Screenshot of tcpdump capturing network packets

- Disable the name and port numbers from the captured network packets using the following command:

```
tcpdump -i <interface name> -c 10 -nn
```

- **tcpreplay**

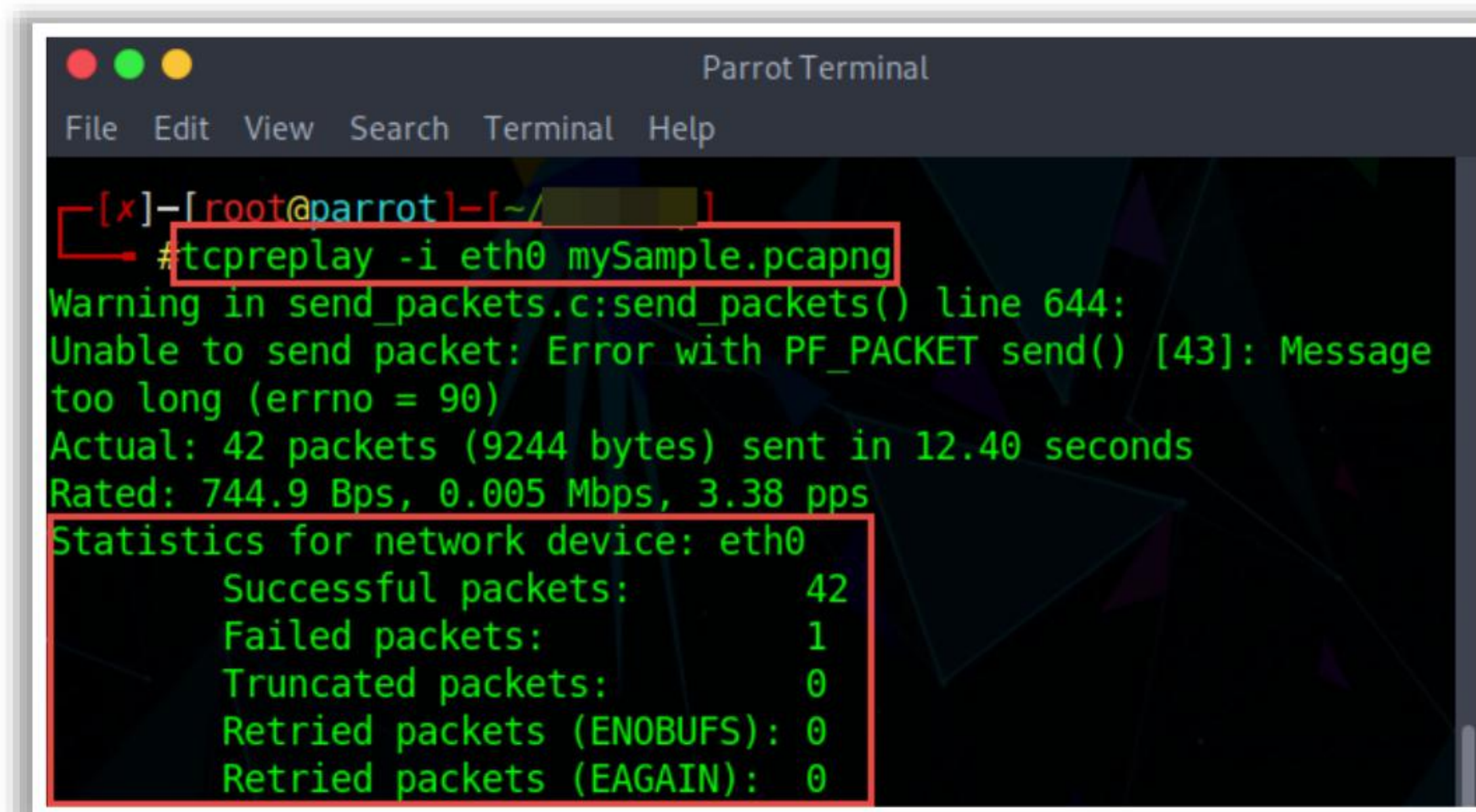
tcpreplay is a GPLv3 licensed utility that supports Unix-like OSes for modifying and replaying previously sniffed traffic from tools such as Wireshark and tcpdump. It replays the .pcaps files to a monitoring interface containing firewalls, NIDS, or IPSes to identify any suspicious network traffic patterns and malicious activities in the pcap. It also allows performing modifications in the pcap file to make it suitable for analysis.

tcpreplay commands

- Replay a pcap file to an interface using the following command:

```
tcpreplay -i eth0 sample.pcap
```

Here, -i specifies the interface to replay followed by the .pcap file name.



```
Parrot Terminal
File Edit View Search Terminal Help
[x]-[root@parrot]-[~/]
#tcpreplay -i eth0 mySample.pcapng
Warning in send_packets.c:send_packets() line 644:
Unable to send packet: Error with PF_PACKET send() [43]: Message too long (errno = 90)
Actual: 42 packets (9244 bytes) sent in 12.40 seconds
Rated: 744.9 Bps, 0.005 Mbps, 3.38 pps
Statistics for network device: eth0
Successful packets: 42
Failed packets: 1
Truncated packets: 0
Retried packets (ENOBUFS): 0
Retried packets (EAGAIN): 0
```

Figure 16.109: Screenshot of tcpreplay replaying network packets

- Replay the same pcap file 10 times on the specified interface using the following command:

```
tcpreplay --loop=10 -i eth0 sample.pcap
```

Here, `--loop` specifies how many times the packet has to be replayed.

- Replay the pcap file continuously until it is interrupted by pressing Ctrl + c.

```
tcpreplay --loop=0 -i eth0 sample.pcap
```

Here, `--loop=0` indicates continuous replay.

- **dnsenum**

Source: <https://github.com>

dnsenum is a Perl script that enumerates the DNS information of a domain to discover noncontiguous IP blocks. This tool performs the following operations:

- Obtain the host's address (A record)
- Obtain the name servers (threaded)
- Obtain the MX record (threaded)
- Perform axfr queries on name servers and obtain BIND VERSION (threaded)
- Obtain extra names and subdomains via Google scraping (Google query = "allinurl: -www site:domain")
- Brute-force subdomains from a file and perform recursion on a subdomain that has NS records (all threaded)
- Calculate C class domain network ranges and perform Whois queries on them (threaded)
- Perform reverse lookups on net ranges (C class or/and Whois net ranges) (threaded)
- Write to the domain_ips.txt file ip-blocks

dnsenum commands

- Display DNS records including name servers, IP addresses, and email records using the following command:

```
dnsenum <domain name/URL>
```

- Display additional details about the site using the following command. It will also attempt to collect Whois information and employ Google to discover if any subdomains are available.

```
dnsenum --enum <domain name/URL>
```




```
Parrot Terminal
File Edit View Search Terminal Help

[root@parrot1-[-]]
#dnsenum --enum google.com
Smartmatch is experimental at /usr/bin/dnsenum line 698.
Smartmatch is experimental at /usr/bin/dnsenum line 698.
dnsenum VERSION:1.2.4
Warning: can't load Net::Whois::IP module, whois queries disabled.
Warning: can't load WWW::Mechanize module, Google scraping disabled.

----- google.com -----

Host's addresses:
-----
google.com.                282      IN      A       142.250.193.174

Name Servers:
-----
ns2.google.com.           21414    IN      A       216.239.34.10
ns1.google.com.           10366    IN      A       216.239.32.10
ns4.google.com.           21599    IN      A       216.239.38.10
ns3.google.com.           21599    IN      A       216.239.36.10

Mail (MX) Servers:
-----
alt4.aspmx.l.google.com.  292      IN      A       64.233.171.27
alt3.aspmx.l.google.com.  292      IN      A       142.250.115.27
alt1.aspmx.l.google.com.  292      IN      A       173.194.202.27
aspmx.l.google.com.       292      IN      A       74.125.200.27
alt2.aspmx.l.google.com.  292      IN      A       142.250.141.27

Trying Zone Transfers and getting Bind Versions:
-----

Trying Zone Transfer for google.com on ns2.google.com ...
AXFR record query failed: corrupt packet

Trying Zone Transfer for google.com on ns4.google.com ...
AXFR record query failed: corrupt packet

Trying Zone Transfer for google.com on ns1.google.com ...
AXFR record query failed: corrupt packet

Trying Zone Transfer for google.com on ns3.google.com ...
AXFR record query failed: corrupt packet

brute force file not specified, bay.
[root@parrot1-[-]]
#
```

Figure 16.110: Screenshot of dnsenum

- Conduct brute forcing along with a custom text file to enumerate all the subdomains.

dnsenum -f subdomain.txt -r <domain name/URL>



Module Summary

- This module discussed the importance of network troubleshooting
- It discussed basic network issues
- It also discussed how to troubleshoot network issues
- Finally, this module presented an overview of network troubleshooting using various tools and utilities
- In the next module, we will discuss network traffic monitoring in detail

Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Summary

This module discussed the importance of network troubleshooting. Furthermore, it discussed basic network issues. It also discussed how to troubleshoot network issues. Finally, this module presented an overview of troubleshooting network issues using various tools and utilities.

In the next module, we will discuss network traffic monitoring in detail.