



# Assignment 1

CS 331 - Computer Networks

---

Name 1: Mohit

Roll no: 23110207

Name 2: Tanishq Bhushan Chaudhari

Roll no: 23110329

Date of Submission: 15th September, 2025

[Github Repository](#)

# Table of Contents

|  |          |
|--|----------|
| <b>TASK 1 - DNS Resolver</b>                 | <b>2</b> |
| Night  | 2        |
| Morning                                      | 2        |
| Afternoon                                    | 3        |
| <b>TASK 2 - Traceroute Protocol Behavior</b> | <b>4</b> |
| Purpose                                      | 4        |
| Procedure                                    | 4        |
| - Windows                                    | 4        |
| - Linux                                      | 4        |
| Questions                                    | 5        |
| Q1   | 5        |
| Q2   | 7        |
| Q3   | 8        |
| Q4   | 10       |
| Q5   | 12       |
| References                                   | 12       |

## TASK 1 - DNS Resolver

We ran the client and server 3 different times- one in the morning, one in the afternoon and one at night.

### Night

| Custom header value(HHMMSSID) | Domain name       | Resolved IP address |
|-------------------------------|-------------------|---------------------|
| 01215400                      | www.linkedin.com  | 192.168.1.11        |
| 01215401                      | www.reddit.com    | 192.168.1.12        |
| 01215402                      | www.facebook.com  | 192.168.1.13        |
| 01215403                      | www.bing.com      | 192.168.1.14        |
| 01215404                      | www.example.com   | 192.168.1.15        |
| 01215405                      | www.wikipedia.org | 192.168.1.11        |
| 01215406                      | www.github.com    | 192.168.1.12        |

### Morning

| Custom header value(HHMMSSID) | Domain name       | Resolved IP address |
|-------------------------------|-------------------|---------------------|
| 04210100                      | www.linkedin.com  | 192.168.1.1         |
| 04210101                      | www.reddit.com    | 192.168.1.2         |
| 04210102                      | www.facebook.com  | 192.168.1.3         |
| 04210103                      | www.bing.com      | 192.168.1.4         |
| 04210104                      | www.example.com   | 192.168.1.5         |
| 04210105                      | www.wikipedia.org | 192.168.1.1         |
| 04210106                      | www.github.com    | 192.168.1.2         |

## Afternoon

| Custom header value(HHMMSSID) | Domain name       | Resolved IP address |
|-------------------------------|-------------------|---------------------|
| 14222600                      | www.linkedin.com  | 192.168.1.6         |
| 14222601                      | www.reddit.com    | 192.168.1.7         |
| 14222602                      | www.facebook.com  | 192.168.1.8         |
| 14222603                      | www.bing.com      | 192.168.1.9         |
| 14222604                      | www.example.com   | 192.168.1.10        |
| 14222605                      | www.wikipedia.org | 192.168.1.6         |
| 14222606                      | www.github.com    | 192.168.1.7         |

## TASK 2 - Traceroute Protocol Behavior

OSes Chosen: Windows and Linux

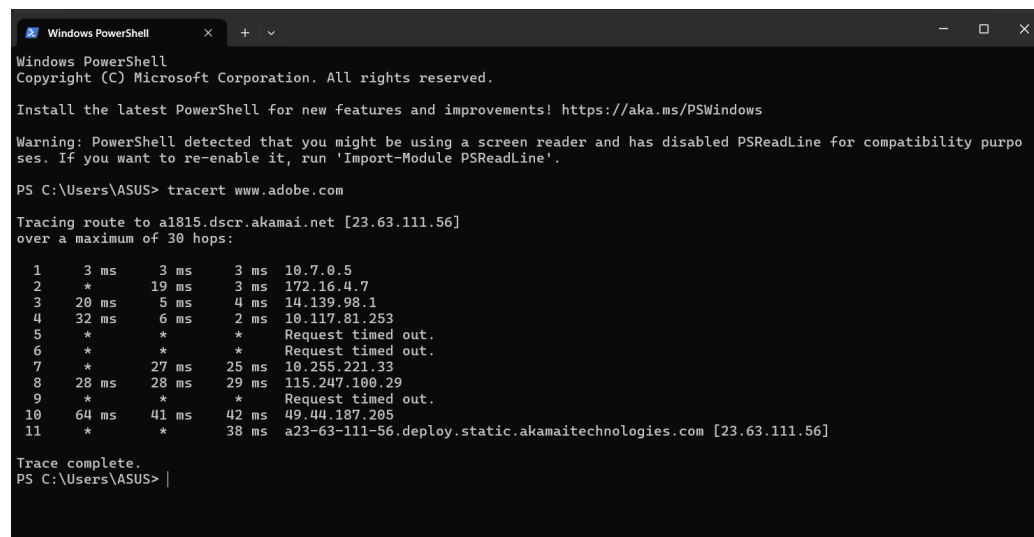
Website Traced: [www.adobe.com](http://www.adobe.com)

### Purpose

The purpose of this task is to understand how the traceroute utility works in different operating systems.

### Procedure

- Windows
  - Open Wireshark and select Wifi.
  - Open the terminal and run the command: `tracert www.adobe.com`
  - Upon completion of this tracing, stop the packet capturing and save the .pcapng file.



```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS C:\Users\ASUS> tracert www.adobe.com

Tracing route to a1815.dscr.akamai.net [23.63.111.56]
over a maximum of 30 hops:
  0  3 ms  3 ms  3 ms  10.7.0.5
  1  *  19 ms  3 ms  172.16.4.7
  2  20 ms  5 ms  4 ms  14.139.98.1
  3  32 ms  6 ms  2 ms  10.117.81.253
  4  *  *  *  Request timed out.
  5  *  *  *  Request timed out.
  6  *  *  *  Request timed out.
  7  *  27 ms  25 ms  10.255.221.33
  8  28 ms  28 ms  29 ms  115.247.100.29
  9  *  *  *  Request timed out.
 10  64 ms  41 ms  42 ms  49.44.187.205
 11  *  *  38 ms  a23-63-111-56.deploy.static.akamaitechnologies.com [23.63.111.56]

Trace complete.
PS C:\Users\ASUS>

```

- Linux
  - Open Wireshark and select wlan0.
  - Open the terminal and run the command: `traceroute www.adobe.com`
  - Upon completion of this tracing, stop the packet capturing and save the .pcapng file.

---

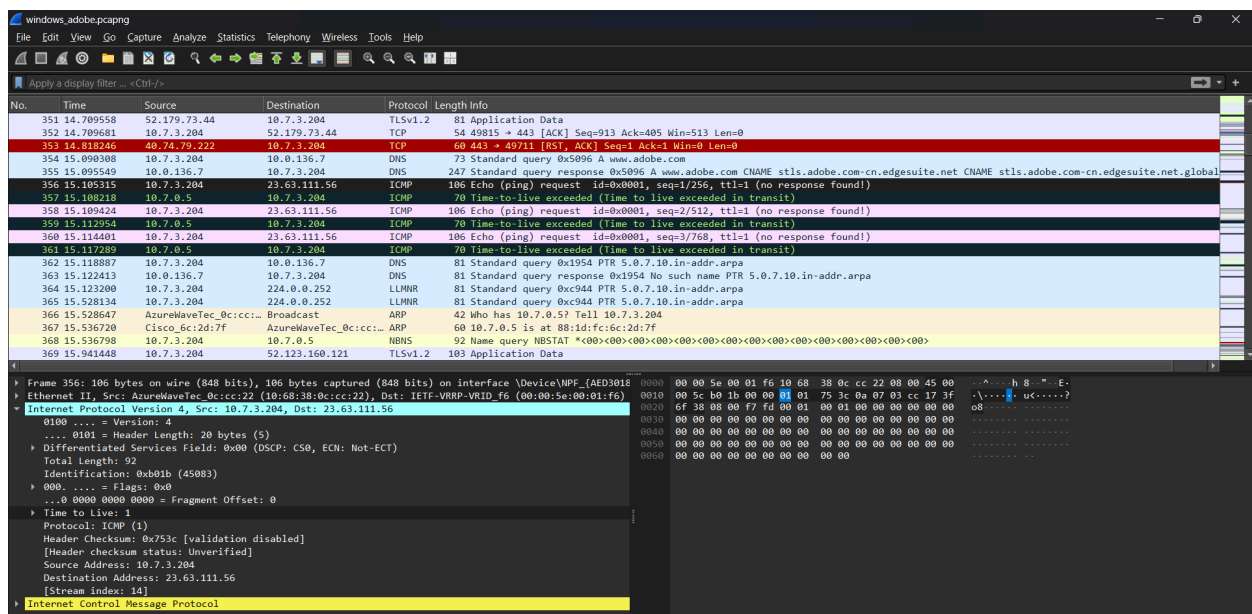
## Questions

01.

What protocol does Windows tracert use by default, and what protocol does Linux traceroute use by default?

Ans.

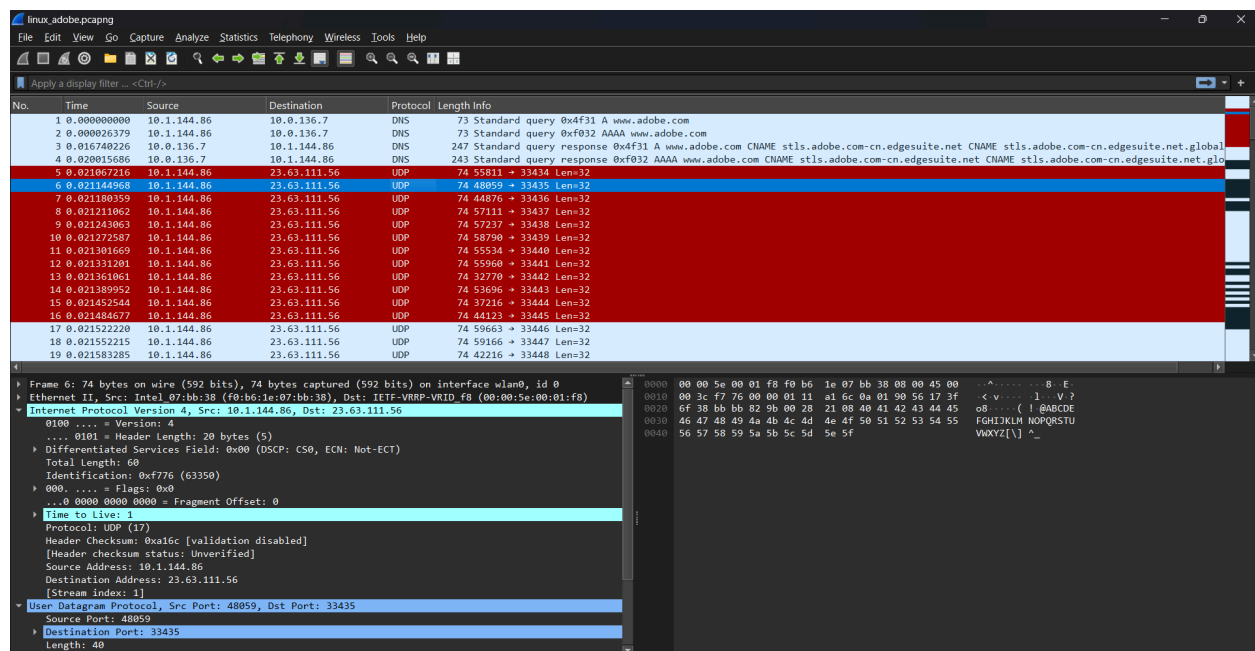
**Windows** tracer uses the ICMP protocol by default. I found this by looking at the packets in the pcap file and noticing that a sequence of ICMP Echo Request packets was being sent with TTL values increasing one by one, which is the typical behavior of tracer. This pattern helped me identify that these packets were generated by tracer and not just a normal ping. The following image shows the first such sequence with TTL=1 (the one packet selected in black colour and the 5 packets that follow it).



- I see ICMP Echo Request packets going out (under Internet Control Message Protocol → Echo (ping) request).

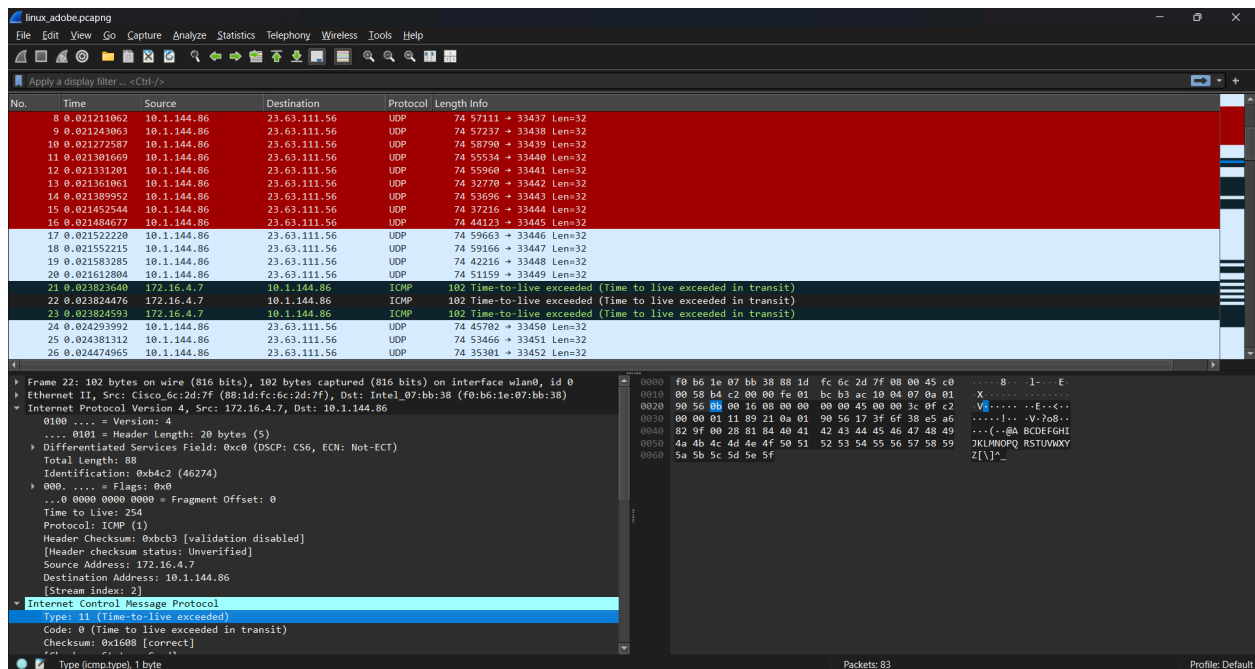
- Intermediate routers reply with ICMP (Time-to-live exceeded) messages.
- At the final hop, the destination sends an ICMP Echo Reply.

**Linux** traceroute uses the UDP protocol by default. I found this by looking at the packets in the pcap file and noticing that a sequence of UDP packets was being sent to destination ports starting from 33434 and increasing with each probe, along with TTL values increasing one by one. This repeating pattern is characteristic of traceroute and helped me identify these packets as coming from the traceroute command. The following image shows the second packet with TTL=1. (Its destination port is 33435.)



- I see UDP packets going out with destination ports beginning at 33434.
- Intermediate routers reply with ICMP (Time-to-live exceeded) messages.
- At the final hop, the destination replies with an ICMP Destination Unreachable message.

The following image shows the first sequence of ICMP replies.



Q2

Some hops in your traceroute output may show \*\*\*. Provide at least two reasons why a router might not reply.

Ans.

The terminal on both Linux and Windows showed \*\*\* for hops 5, 6 and 9 while tracing packets.

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS C:\Users\ASUS> tracert www.adobe.com

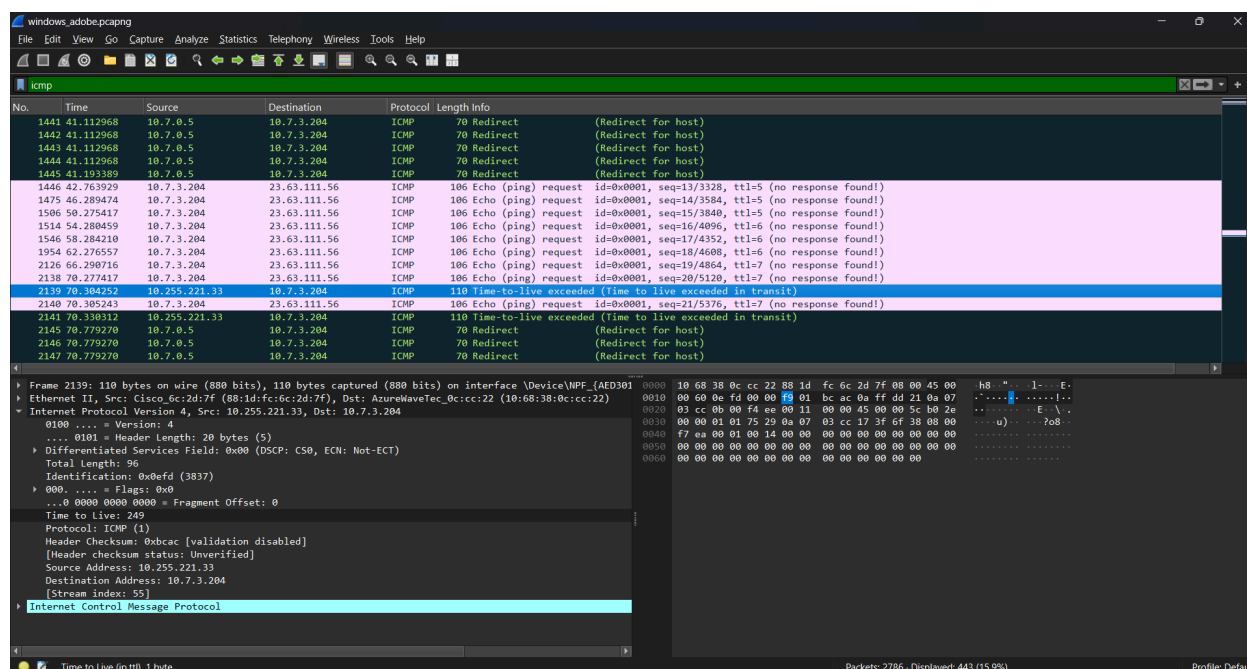
Tracing route to a1815.dscr.akamai.net [23.63.111.56]
over a maximum of 30 hops:

  0  3 ms    3 ms    3 ms  10.7.0.5
  1  *        19 ms   3 ms  172.16.4.7
  2  20 ms    5 ms    4 ms  14.139.98.1
  3  32 ms    6 ms    2 ms  10.117.81.253
  4  *        *        *
  5  *        *        *    Request timed out.
  6  *        *        *    Request timed out.
  7  *        27 ms   25 ms  10.255.221.33
  8  28 ms    28 ms   29 ms  115.247.100.29
  9  *        *        *    Request timed out.
 10  64 ms    41 ms   42 ms  49.44.187.205
 11  *        *        38 ms  a23-63-111-56.deploy.static.akamaitechnologies.com [23.63.111.56]

Trace complete.
PS C:\Users\ASUS>

```





Looking at the .pcapng, I see the probe packets going out, but no ICMP reply comes back for those hops. The routers were not faulty because later probes reached their destination, which was confirmed by their replies.

Possible reasons:

- Some routers/firewalls block or drop ICMP/UDP responses, so the probes never return and traceroute shows \*.
- Routers may rate-limit ICMP replies or be configured not to send TTL-expired responses, which also leads to \*.

Since hops 5, 6, and 9 dropped in both Windows and Linux, the most likely reason is suppression or blocking at those routers.

## Q3

In Linux traceroute, which field in the probe packets changes between successive probes sent to the destination?

Ans.

In Linux traceroute, I noticed mainly three fields that change between consecutive probes:

- Time to Live: It started from 1 and kept increasing in groups of 3 as traceroute sends 3 probes for each hop.
- Destination port: Starting from 33434, it kept increasing by 1 in each consecutive probe.
- Source port: It keeps changing in each hop, but I don't see any pattern in it.

## Hop1- probe1:

The screenshot shows a Wireshark capture of network traffic. The packet list on the left shows a series of packets, with packet 5 selected. The packet details pane on the right shows the structure of the selected packet, which is an Internet Protocol Version 4 (IPv4) packet. The packet is from source 10.1.144.86 to destination 23.63.111.56. The protocol is UDP, and the source port is 55811, destination port is 33434. The packet length is 40 bytes. The packet bytes pane on the right shows the raw data of the packet, which is a DNS query response.

| No. | Time        | Source      | Destination  | Protocol | Length | Info  |
|-----|-------------|-------------|--------------|----------|--------|---|
| 1   | 0.000000000 | 10.1.144.86 | 10.0.136.7   | DNS      | 73     | Standard query 0xf31 A www.adobe.com  |
| 2   | 0.000026379 | 10.1.144.86 | 10.0.136.7   | DNS      | 73     | Standard query 0xf32 AAAA www.adobe.com   |
| 3   | 0.016740226 | 10.0.136.7  | 10.1.144.86  | DNS      | 247    | Standard query response 0xf31 A www.adobe.com CNAME stls.adobe.com-cn.edgesuite.net CNAME stls.adobe.com-cn.edgesuite.net.global    |
| 4   | 0.02015686  | 10.0.136.7  | 10.1.144.86  | DNS      | 243    | Standard query response 0xf32 AAAA www.adobe.com CNAME stls.adobe.com-cn.edgesuite.net CNAME stls.adobe.com-cn.edgesuite.net.global |
| 5   | 0.021067216 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 55811 → 33434 Len=32  |
| 6   | 0.021144968 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 48059 → 33435 Len=32  |
| 7   | 0.021180359 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 44876 → 33436 Len=32  |
| 8   | 0.021218602 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 57111 → 33437 Len=32  |
| 9   | 0.021243063 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 57237 → 33438 Len=32  |
| 10  | 0.021272587 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 58790 → 33439 Len=32  |
| 11  | 0.021301669 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 55534 → 33440 Len=32  |
| 12  | 0.02131201  | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 55960 → 33441 Len=32  |
| 13  | 0.021361061 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 32770 → 33442 Len=32  |
| 14  | 0.021389952 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 53696 → 33443 Len=32  |
| 15  | 0.021452544 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 37216 → 33444 Len=32  |
| 16  | 0.021484677 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 44123 → 33445 Len=32  |
| 17  | 0.021522220 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 59663 → 33446 Len=32  |
| 18  | 0.021552215 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 59166 → 33447 Len=32  |
| 19  | 0.021583285 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 42216 → 33448 Len=32  |

Frame 5: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface wlan0, id 0  
 Ethernet II, Src: Intel 07:bb:38 (f0:b6:1e:07:bb:38), Dst: IETF-VRRP-VRID f8 (00:00:5e:00:01:f8)  
 Internet Protocol Version 4, Src: 10.1.144.86, Dst: 23.63.111.56  
 0100 .... = Version: 4  
 .... 0101 = Header Length: 20 bytes (5)  
 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)  
 Total Length: 60  
 Identification: 0x6c2c (27692)  
 0000 .... = Flags: 0x0  
 ...0 0000 0000 0000 = Fragment Offset: 0  
 Time to live: 1  
 Protocol: UDP (17)  
 Header Checksum: 0x2cb7 [validation disabled]  
 [Header checksum status: Unverified]  
 Source Address: 10.1.144.86  
 Destination Address: 23.63.111.56  
 [Stream index: 1]  
 User Datagram Protocol, Src Port: 55811, Dst Port: 33434  
 Source Port: 55811  
 Destination Port: 33434  
 Length: 40  
 User Datagram Protocol (udp), 4 bytes

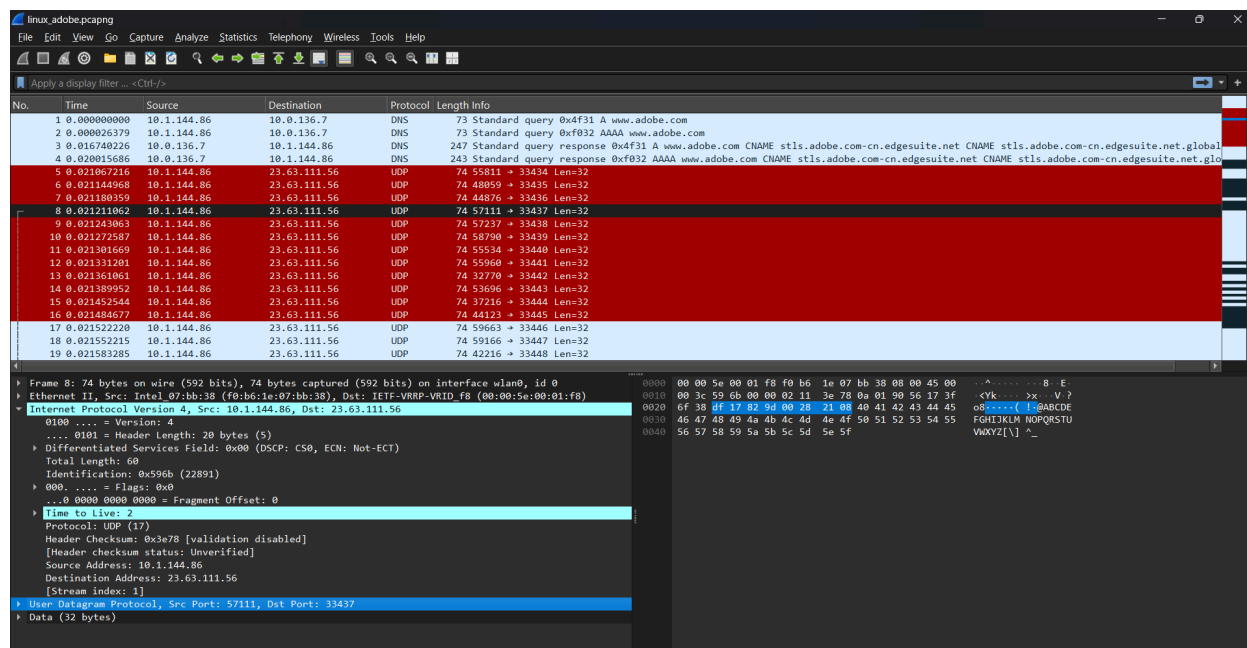
## Hop1 - probe2:

The screenshot shows a Wireshark capture of network traffic. The packet list on the left shows a series of packets, with packet 5 selected. The packet details pane on the right shows the structure of the selected packet, which is an Internet Protocol Version 4 (IPv4) packet. The packet is from source 10.1.144.86 to destination 23.63.111.56. The protocol is UDP, and the source port is 48059, destination port is 33435. The packet length is 32 bytes. The packet bytes pane on the right shows the raw data of the packet, which is a DNS query response.

| No. | Time        | Source      | Destination  | Protocol | Length | Info  |
|-----|-------------|-------------|--------------|----------|--------|---|
| 1   | 0.000000000 | 10.1.144.86 | 10.0.136.7   | DNS      | 73     | Standard query 0xf31 A www.adobe.com  |
| 2   | 0.000026379 | 10.1.144.86 | 10.0.136.7   | DNS      | 73     | Standard query 0xf32 AAAA www.adobe.com   |
| 3   | 0.016740226 | 10.0.136.7  | 10.1.144.86  | DNS      | 247    | Standard query response 0xf31 A www.adobe.com CNAME stls.adobe.com-cn.edgesuite.net CNAME stls.adobe.com-cn.edgesuite.net.global    |
| 4   | 0.02015686  | 10.0.136.7  | 10.1.144.86  | DNS      | 243    | Standard query response 0xf32 AAAA www.adobe.com CNAME stls.adobe.com-cn.edgesuite.net CNAME stls.adobe.com-cn.edgesuite.net.global |
| 5   | 0.021067216 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 55811 → 33434 Len=32  |
| 6   | 0.021144968 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 48059 → 33435 Len=32  |
| 7   | 0.021180359 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 44876 → 33436 Len=32  |
| 8   | 0.021218602 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 57111 → 33437 Len=32  |
| 9   | 0.021243063 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 57237 → 33438 Len=32  |
| 10  | 0.021272587 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 58790 → 33439 Len=32  |
| 11  | 0.021301669 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 55534 → 33440 Len=32  |
| 12  | 0.02131201  | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 55960 → 33441 Len=32  |
| 13  | 0.021361061 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 32770 → 33442 Len=32  |
| 14  | 0.021389952 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 53696 → 33443 Len=32  |
| 15  | 0.021452544 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 37216 → 33444 Len=32  |
| 16  | 0.021484677 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 44123 → 33445 Len=32  |
| 17  | 0.021522220 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 59663 → 33446 Len=32  |
| 18  | 0.021552215 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 59166 → 33447 Len=32  |
| 19  | 0.021583285 | 10.1.144.86 | 23.63.111.56 | UDP      | 74     | 42216 → 33448 Len=32  |

Frame 6: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface wlan0, id 0  
 Ethernet II, Src: Intel 07:bb:38 (f0:b6:1e:07:bb:38), Dst: IETF-VRRP-VRID f8 (00:00:5e:00:01:f8)  
 Internet Protocol Version 4, Src: 10.1.144.86, Dst: 23.63.111.56  
 0100 .... = Version: 4  
 .... 0101 = Header Length: 20 bytes (5)  
 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)  
 Total Length: 60  
 Identification: 0xf776 (63350)  
 0000 .... = Flags: 0x0  
 ...0 0000 0000 0000 = Fragment Offset: 0  
 Time to live: 1  
 Protocol: UDP (17)  
 Header Checksum: 0xa16c [validation disabled]  
 [Header checksum status: Unverified]  
 Source Address: 10.1.144.86  
 Destination Address: 23.63.111.56  
 [Stream index: 1]  
 User Datagram Protocol, Src Port: 48059, Dst Port: 33435  
 Data (32 bytes)  
 User Datagram Protocol (udp), 6 bytes

## Hop2 - probe1:



## Q4

At the final hop, how is the response different compared to the intermediate hop?

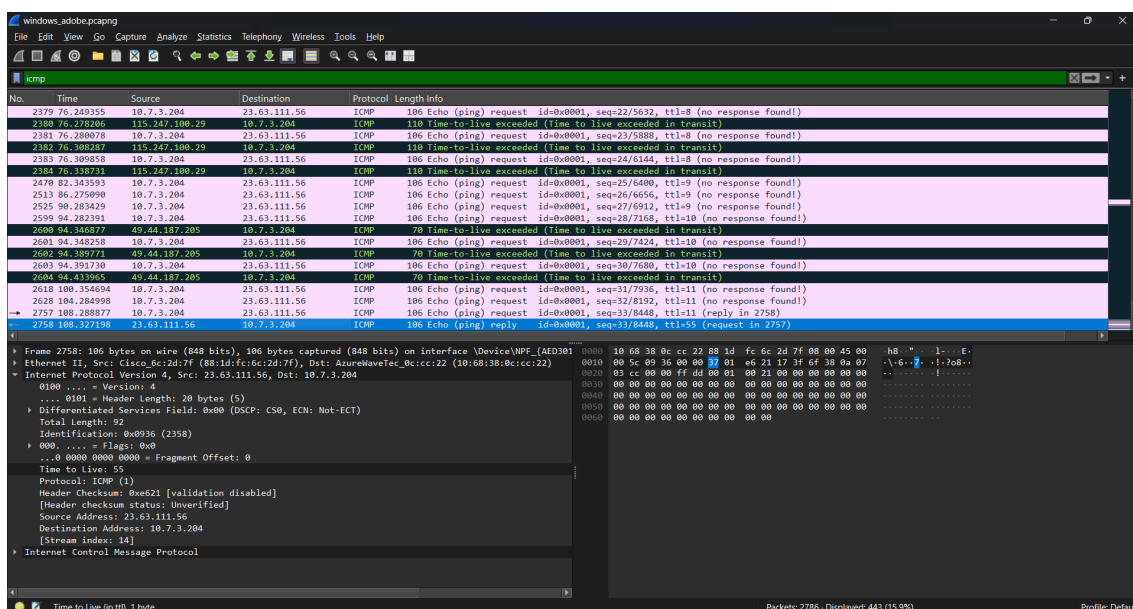
Ans.

## Windows:

- Intermediate hop: We get a Time-to-live exceeded ICMP message for each probe of the intermediate hop. (Type:11, Code:0)

|     |           |             |              |      |   |
|-----|-----------|-------------|--------------|------|---|
| 681 | 30.767394 | 10.7.3.204  | 23.63.111.56 | ICMP | 106 Echo (ping) request id=0x0001, seq=7/1792, ttl=3 (no response found!) |
| 682 | 30.787827 | 14.139.98.1 | 10.7.3.204   | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)               |
| 683 | 30.789207 | 10.7.3.204  | 23.63.111.56 | ICMP | 106 Echo (ping) request id=0x0001, seq=8/2048, ttl=3 (no response found!) |
| 684 | 30.794206 | 14.139.98.1 | 10.7.3.204   | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)               |
| 685 | 30.795256 | 10.7.3.204  | 23.63.111.56 | ICMP | 106 Echo (ping) request id=0x0001, seq=9/2304, ttl=3 (no response found!) |
| 686 | 30.799146 | 14.139.98.1 | 10.7.3.204   | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)               |

- Final hop: We get an ICMP - Echo (ping) reply message for each of the final hop probes. But in my case, only one probe led to a reply, as for the other two probes, my terminal printed \*. (Type:0, Code:0)

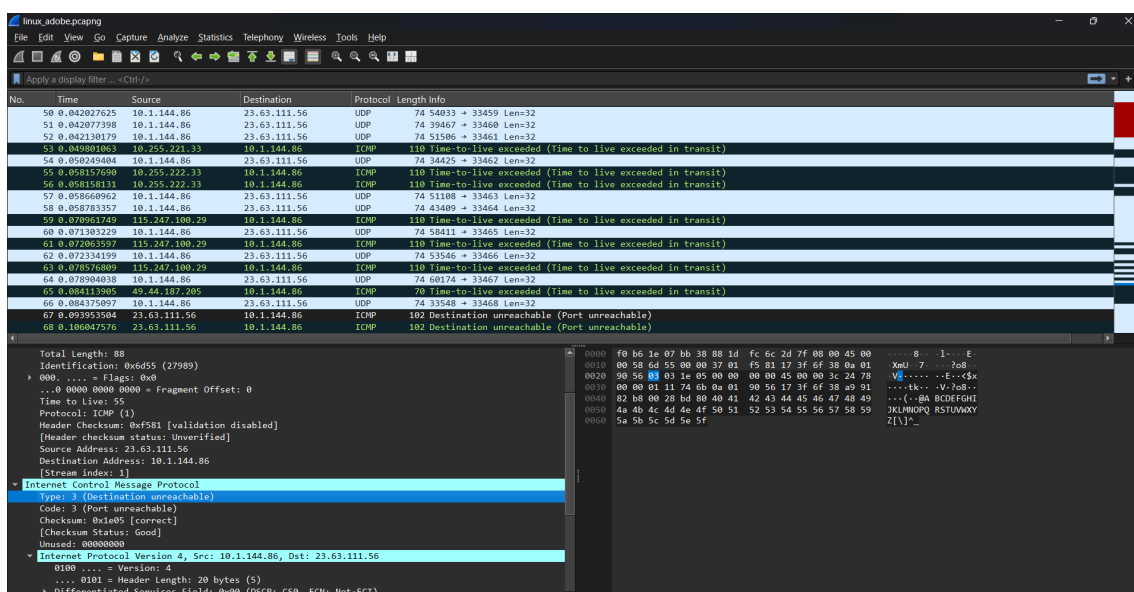


## Linux:

- Intermediate hop: This is similar to the Windows tracer, as we get an ICMP Time-to-live exceeded message when TTL becomes 0. (Type:11, Code:0)

|    |             |               |              |      |  |
|----|-------------|---------------|--------------|------|--|
| 20 | 0.021612804 | 10.1.144.86   | 23.63.111.56 | UDP  | 74 51159 → 33449 Len=32  |
| 21 | 0.023823640 | 172.16.4.7    | 10.1.144.86  | ICMP | 102 Time-to-live exceeded (Time to live exceeded in transit)               |
| 22 | 0.023824476 | 172.16.4.7    | 10.1.144.86  | ICMP | 102 Time-to-live exceeded (Time to live exceeded in transit)               |
| 23 | 0.023824593 | 172.16.4.7    | 10.1.144.86  | ICMP | 102 Time-to-live exceeded (Time to live exceeded in transit)               |
| 24 | 0.024293992 | 10.1.144.86   | 23.63.111.56 | UDP  | 74 45702 → 33450 Len=32  |
| 25 | 0.024381312 | 10.1.144.86   | 23.63.111.56 | UDP  | 74 53466 → 33451 Len=32  |
| 26 | 0.024474965 | 10.1.144.86   | 23.63.111.56 | UDP  | 74 35301 → 33452 Len=32  |
| 27 | 0.025272885 | 10.1.144.3    | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 28 | 0.025273598 | 10.1.144.3    | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 29 | 0.025273764 | 10.1.144.3    | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 30 | 0.025312107 | 10.117.81.253 | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 31 | 0.025312960 | 10.117.81.253 | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 32 | 0.025313129 | 10.117.81.253 | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 33 | 0.026454762 | 10.1.144.86   | 10.0.136.7   | DNS  | 83 Standard query 0xe943 PTR 3.144.1.10.in-addr.arpa                       |
| 34 | 0.027987434 | 14.139.98.1   | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 35 | 0.027987981 | 14.139.98.1   | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 36 | 0.027988108 | 14.139.98.1   | 10.1.144.86  | ICMP | 70 Time-to-live exceeded (Time to live exceeded in transit)                |
| 37 | 0.030345653 | 10.0.136.7    | 10.1.144.86  | DNS  | 83 Standard query response 0xe943 No such name PTR 3.144.1.10.in-addr.arpa |
| 38 | 0.031071262 | 10.1.144.86   | 10.0.136.7   | DNS  | 83 Standard query 0xb246 PTR 7.4.16.172.in-addr.arpa                       |

- Final hop: For this, we get an ICMP Destination Unreachable message because the packet reaches before the TTL expires, but the host has no application listening on that port. (Type:3, Code:3)



## Q5

Suppose a firewall blocks UDP traffic but allows ICMP — how would this affect the results of Linux traceroute vs. Windows tracert?

Ans.

### Windows

- As Windows tracert uses ICMP probes, it will act normally as expected.

### Linux

- As Linux traceroute uses UDP probes, there will be some abnormalities.
- Probes act normally up to the hop just before the firewall.
- From this hop onwards, the UDP probes would be dropped by the firewall. So, they never reach their destination, nor do their TTLs expire, leading to no ICMP reply.
- Traceroute probably shows \*\*\* for these packets because no reply is received.
- The blocked packets may generate an ICMP Destination Unreachable message, or the firewall may silently drop them.

## References

- [Superuser](#)
- [Stack Exchange](#)