1. Introduction:

In the evolving landscape of network technologies, understanding and analysing network traffic have become pivotal for maintaining robust and secure communication infrastructures. This Wireshark analysis project delves into the intricate details of network communication to unravel patterns, troubleshoot issues, and gain valuable insights.

Network analysis serves as a crucial component in the toolkit of IT professionals, enabling them to optimize performance, troubleshoot problems, and bolster security measures. This project utilizes Wireshark, a powerful open-source packet analyser, to dissect and interpret captured network packets.

Purpose:

The primary purpose of this project is to showcase practical skills in network analysis using Wireshark. By capturing and dissecting packets in various scenarios, the project aims to provide a hands-on demonstration of troubleshooting, protocol analysis, and security assessment.

Significance:

Understanding network traffic goes beyond mere technical proficiency; it empowers professionals to proactively manage and secure network environments. The insights gained from this analysis project contribute not only to individual skill development but also to the broader domain of network management and cybersecurity.

Scope:

The scope of this project encompasses a range of activities, from basic packet capture and analysis to advanced protocol examination. Simulated troubleshooting scenarios and security analyses further highlight the practical applications of Wireshark in real-world scenarios.

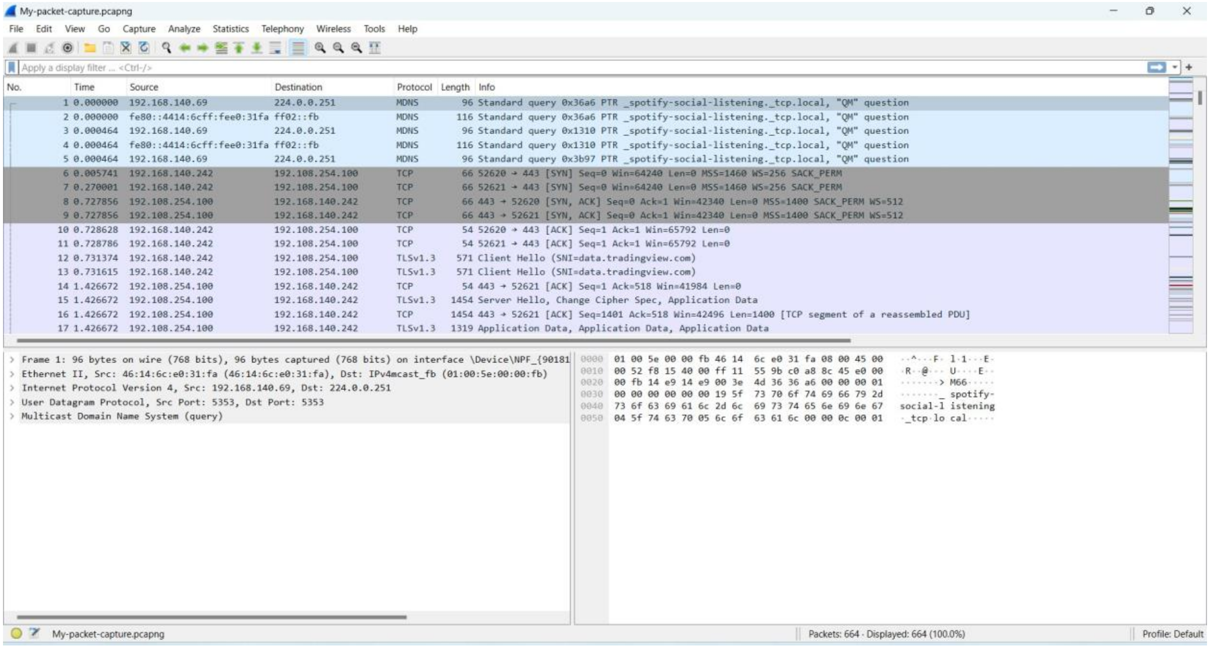
Through this project, we aim to contribute to the knowledge base of network analysis while showcasing the capabilities of Wireshark in addressing challenges faced in modern network environments.

2. Objectives:

The objectives of this Wireshark analysis project were designed to achieve a comprehensive understanding of network traffic patterns, troubleshoot specific scenarios, and gain insights into protocol interactions. The primary goals included:

A. Packet Capture and Analysis:

Capture network traffic in various scenarios to understand the communication patterns between devices.
Analyze captured packets to identify common protocols, source/destination addresses, and data payloads.



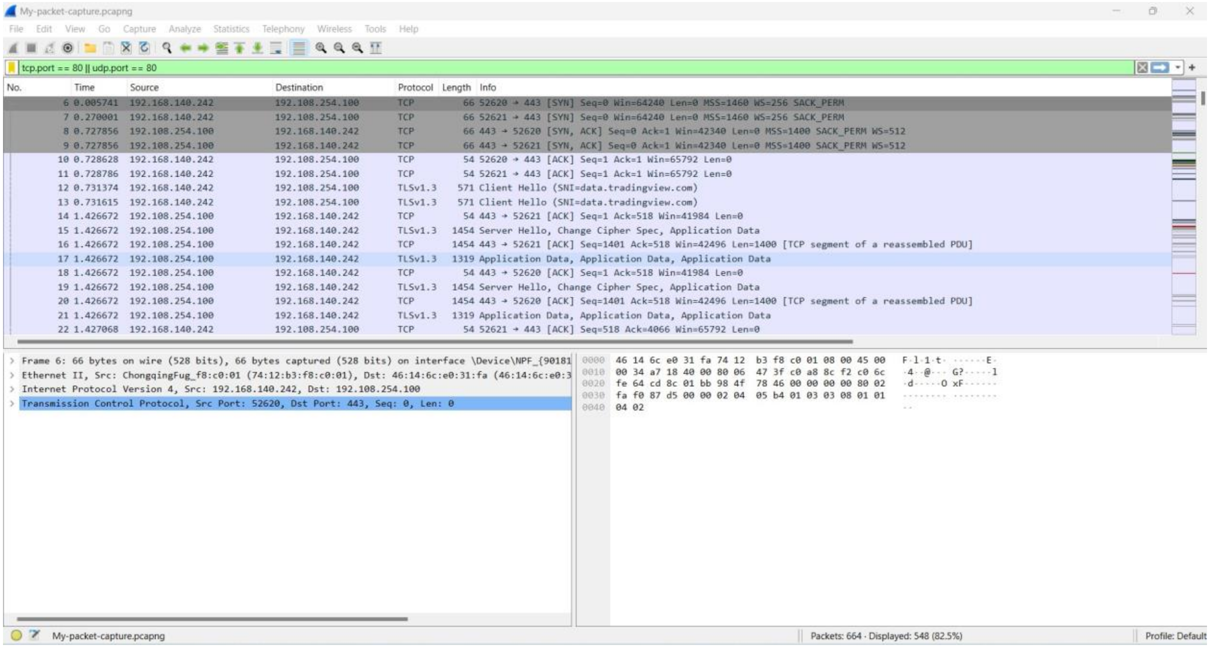
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.140.69	224.0.0.251	MDNS	96	Standard query 0x36a6 PTR _spotify-social-listening._tcp.local, "QM" question
2	0.000000	fe80::4414:6cff:fee0:31fa	ff02::fb	MDNS	116	Standard query 0x36a6 PTR _spotify-social-listening._tcp.local, "QM" question
3	0.000464	192.168.140.69	224.0.0.251	MDNS	96	Standard query 0x1310 PTR _spotify-social-listening._tcp.local, "QM" question
4	0.000464	fe80::4414:6cff:fee0:31fa	ff02::fb	MDNS	116	Standard query 0x1310 PTR _spotify-social-listening._tcp.local, "QM" question
5	0.000464	192.168.140.69	224.0.0.251	MDNS	96	Standard query 0x3b97 PTR _spotify-social-listening._tcp.local, "QM" question
6	0.005741	192.168.140.242	192.108.254.100	TCP	66	52620 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
7	0.270001	192.168.140.242	192.108.254.100	TCP	66	52621 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
8	0.727856	192.108.254.100	192.168.140.242	TCP	66	443 → 52620 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1400 SACK_PERM WS=512
9	0.727856	192.108.254.100	192.168.140.242	TCP	66	443 → 52621 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1400 SACK_PERM WS=512
10	0.728628	192.168.140.242	192.108.254.100	TCP	54	52620 → 443 [ACK] Seq=1 Ack=1 Win=65792 Len=0
11	0.728786	192.168.140.242	192.108.254.100	TCP	54	52621 → 443 [ACK] Seq=1 Ack=1 Win=65792 Len=0
12	0.731374	192.168.140.242	192.108.254.100	TLSv1.3	571	Client Hello (SNI=data.tradingview.com)
13	0.731615	192.168.140.242	192.108.254.100	TLSv1.3	571	Client Hello (SNI=data.tradingview.com)
14	1.426672	192.108.254.100	192.168.140.242	TCP	54	443 → 52621 [ACK] Seq=1 Ack=518 Win=41984 Len=0
15	1.426672	192.108.254.100	192.168.140.242	TLSv1.3	1454	Server Hello, Change Cipher Spec, Application Data
16	1.426672	192.108.254.100	192.168.140.242	TCP	1454	443 → 52621 [ACK] Seq=1401 Ack=518 Win=42496 Len=1400 [TCP segment of a reassembled PDU]
17	1.426672	192.108.254.100	192.168.140.242	TLSv1.3	1319	Application Data, Application Data, Application Data

```
> Frame 1: 96 bytes on wire (768 bits), 96 bytes captured (768 bits) on interface \Device\NPF_{90181
> Ethernet II, Src: 46:14:6c:e0:31:fa (46:14:6c:e0:31:fa), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)
> Internet Protocol Version 4, Src: 192.168.140.69, Dst: 224.0.0.251
> User Datagram Protocol, Src Port: 5353, Dst Port: 5353
> Multicast Domain Name System (query)
```

```
0000  01 00 5e 00 00 fb 46 14  6c e0 31 fa 08 00 45 00   ··^···F· l·1···E·
0010  00 52 f8 15 40 00 ff 11  55 9b c0 a8 8c 45 e0 00   ·R··@··· U····E··
0020  00 fb 14 e9 14 e9 00 3e  4d 36 36 a6 00 00 00 01   ·······> M66·····
0030  00 00 00 00 00 00 19 5f  73 70 6f 74 69 66 79 2d   ·······_ spotify-
0040  73 6f 63 69 61 6c 2d 6c  69 73 74 65 6e 69 6e 67   social-l istening
0050  04 5f 74 63 70 05 6c 6f  63 61 6c 00 00 0c 00 01   ·_tcp·lo cal·····
```

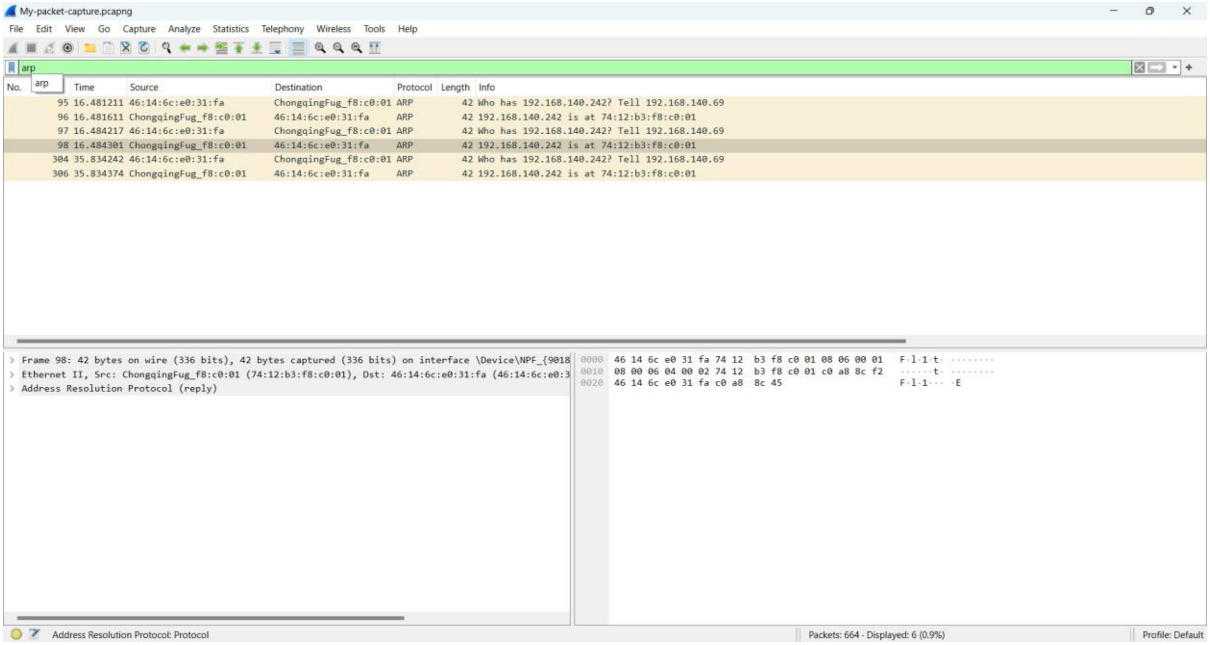
My-packet-capture.pcapng | Packets: 664 · Displayed: 664 (100.0%) | Profile: Default

This screenshot marks the completion of the packet capture process in Wireshark. The packet list displays a summary of captured packets, including packet number, timestamp, source, destination addresses, and protocol types. The packet details pane provides additional information about the selected packet, while the packet bytes section offers a detailed view of the raw data.


My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.port == 80 || udp.port == 80

No.  Time      Source           Destination      Protocol Length Info
   6 0.005741  192.168.140.242  192.108.254.100  TCP        66 52620 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
   7 0.270001  192.168.140.242  192.108.254.100  TCP        66 52621 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
   8 0.727856  192.108.254.100  192.168.140.242  TCP        66 443 → 52620 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1400 SACK_PERM WS=512
   9 0.727856  192.108.254.100  192.168.140.242  TCP        66 443 → 52621 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1400 SACK_PERM WS=512
  10 0.728628  192.168.140.242  192.108.254.100  TCP        54 52620 → 443 [ACK] Seq=1 Ack=1 Win=65792 Len=0
  11 0.728786  192.168.140.242  192.108.254.100  TCP        54 52621 → 443 [ACK] Seq=1 Ack=1 Win=65792 Len=0
  12 0.731374  192.168.140.242  192.108.254.100  TLSv1.3   571 Client Hello (SNI=data.tradingview.com)
  13 0.731615  192.168.140.242  192.108.254.100  TLSv1.3   571 Client Hello (SNI=data.tradingview.com)
  14 1.426672  192.108.254.100  192.168.140.242  TCP        54 443 → 52621 [ACK] Seq=1 Ack=518 Win=41984 Len=0
  15 1.426672  192.108.254.100  192.168.140.242  TLSv1.3  1454 Server Hello, Change Cipher Spec, Application Data
  16 1.426672  192.108.254.100  192.168.140.242  TCP      1454 443 → 52621 [ACK] Seq=1401 Ack=518 Win=42496 Len=1400 [TCP segment of a reassembled PDU]
  17 1.426672  192.108.254.100  192.168.140.242  TLSv1.3  1319 Application Data, Application Data, Application Data
  18 1.426672  192.108.254.100  192.168.140.242  TCP        54 443 → 52620 [ACK] Seq=1 Ack=518 Win=41984 Len=0
  19 1.426672  192.108.254.100  192.168.140.242  TLSv1.3  1454 Server Hello, Change Cipher Spec, Application Data
  20 1.426672  192.108.254.100  192.168.140.242  TCP      1454 443 → 52620 [ACK] Seq=1401 Ack=518 Win=42496 Len=1400 [TCP segment of a reassembled PDU]
  21 1.426672  192.108.254.100  192.168.140.242  TLSv1.3  1319 Application Data, Application Data, Application Data
  22 1.427068  192.168.140.242  192.108.254.100  TCP        54 52621 → 443 [ACK] Seq=518 Ack=4066 Win=65792 Len=0

> Frame 6: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{90181
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Internet Protocol Version 4, Src: 192.168.140.242, Dst: 192.108.254.100
> Transmission Control Protocol, Src Port: 52620, Dst Port: 443, Seq: 0, Len: 0

0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 00 45 00   F·l·1·t· ······E·
0010  00 34 a7 18 40 00 80 06  47 3f c0 a8 8c f2 c0 6c   ·4··@··· G?·····l
0020  fe 64 cd 8c 01 bb 98 4f  78 46 00 00 00 00 80 02   ·d·····O xF······
0030  fa f0 87 d5 00 00 02 04  05 b4 01 03 03 08 01 01   ········ ········
0040  04 02                                              ··

My-packet-capture.pcapng    Packets: 664 · Displayed: 548 (82.5%)    Profile: Default
```

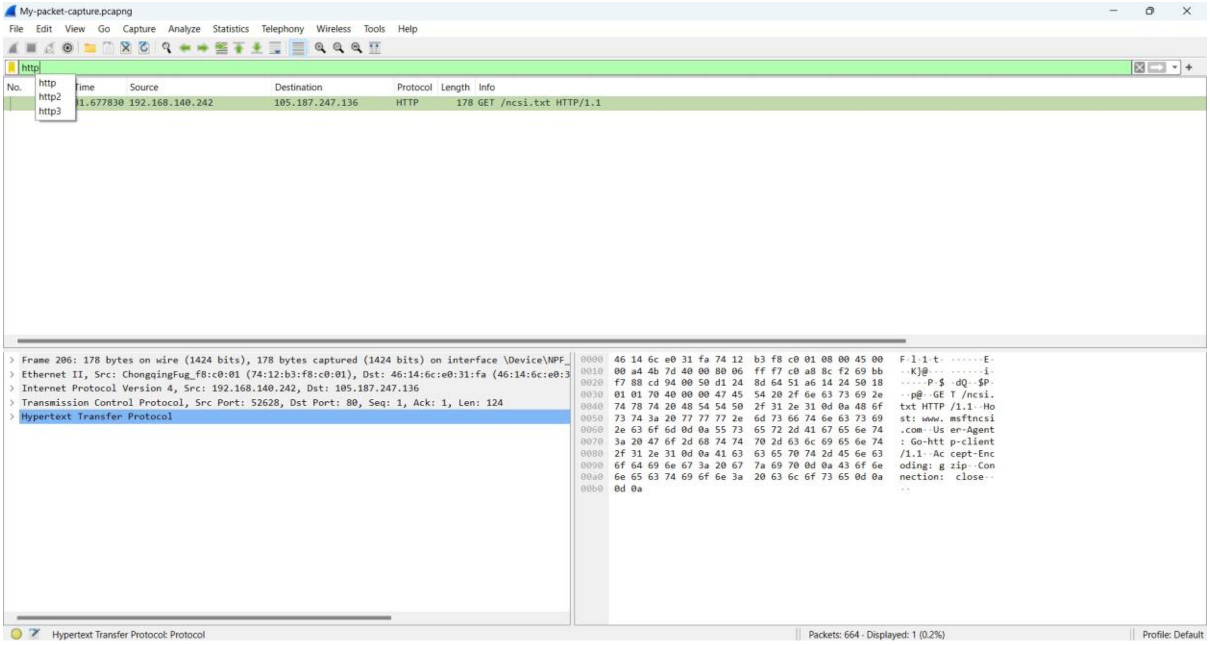
In this screenshot, a filter has been applied to display only TCP and UDP traffic on port 80. This targeted filter allows for a focused analysis on network activity related to common web traffic. The packet list is refined to show packets matching the specified criteria, aiding in a more detailed examination of relevant communication.


My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
arp

No.  Time       Source                 Destination            Protocol Length Info
  95 16.481211  46:14:6c:e0:31:fa      ChongqingFug_f8:c0:01  ARP        42 Who has 192.168.140.242? Tell 192.168.140.69
  96 16.481611  ChongqingFug_f8:c0:01  46:14:6c:e0:31:fa      ARP        42 192.168.140.242 is at 74:12:b3:f8:c0:01
  97 16.484217  46:14:6c:e0:31:fa      ChongqingFug_f8:c0:01  ARP        42 Who has 192.168.140.242? Tell 192.168.140.69
  98 16.484301  ChongqingFug_f8:c0:01  46:14:6c:e0:31:fa      ARP        42 192.168.140.242 is at 74:12:b3:f8:c0:01
 304 35.834242  46:14:6c:e0:31:fa      ChongqingFug_f8:c0:01  ARP        42 Who has 192.168.140.242? Tell 192.168.140.69
 306 35.834374  ChongqingFug_f8:c0:01  46:14:6c:e0:31:fa      ARP        42 192.168.140.242 is at 74:12:b3:f8:c0:01

> Frame 98: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{9018
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Address Resolution Protocol (reply)

0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 06 00 01   F·l·1·t· ········
0010  08 00 06 04 00 02 74 12  b3 f8 c0 01 c0 a8 8c f2   ······t· ········
0020  46 14 6c e0 31 fa c0 a8  8c 45                     F·l·1··· ·E

Address Resolution Protocol: Protocol    Packets: 664 · Displayed: 6 (0.9%)    Profile: Default
```

This screenshot illustrates the application of a filter to display only ARP (Address Resolution Protocol) traffic. By focusing on ARP packets, essential for mapping IP addresses to MAC addresses, this filter allows for a targeted analysis of network address resolution. The packet list reflects exclusively

ARP packets, providing insights into device connectivity and network mapping.

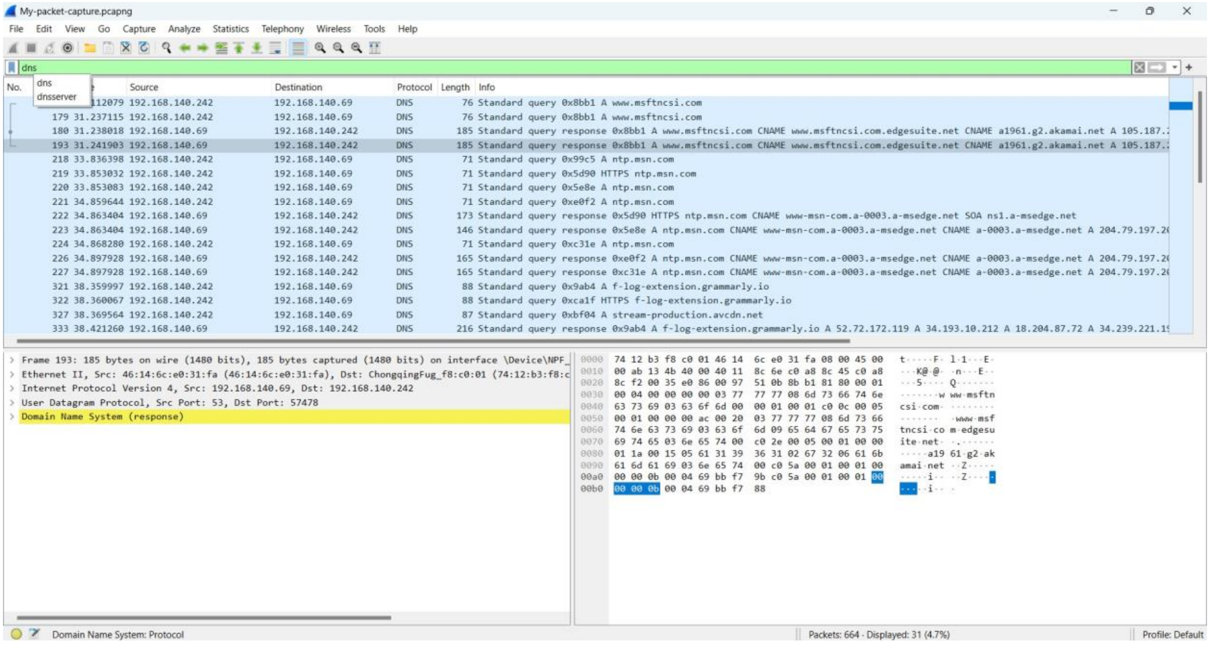

My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
http
http2
http3
Hypertext Transfer Protocol: Protocol
Packets: 664 · Displayed: 1 (0.2%)
Profile: Default

```
No.  Time       Source           Destination      Protocol Length Info
     1.677830   192.168.140.242  105.187.247.136  HTTP        178 GET /ncsi.txt HTTP/1.1
```

```
> Frame 206: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface \Device\NPF_
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Internet Protocol Version 4, Src: 192.168.140.242, Dst: 105.187.247.136
> Transmission Control Protocol, Src Port: 52628, Dst Port: 80, Seq: 1, Ack: 1, Len: 124
> Hypertext Transfer Protocol
```

```
0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 00 45 00   F·l·1·t· ······E·
0010  00 a4 4b 7d 40 00 80 06  ff f7 c0 a8 8c f2 69 bb   ··K}@··· ······i·
0020  f7 88 cd 94 00 50 d1 24  8d 64 51 a6 14 24 50 18   ·····P·$ ·dQ··$P·
0030  01 01 70 40 00 00 47 45  54 20 2f 6e 63 73 69 2e   ··p@··GE T /ncsi.
0040  74 78 74 20 48 54 54 50  2f 31 2e 31 0d 0a 48 6f   txt HTTP /1.1··Ho
0050  73 74 3a 20 77 77 77 2e  6d 73 66 74 6e 63 73 69   st: www. msftncsi
0060  2e 63 6f 6d 0d 0a 55 73  65 72 2d 41 67 65 6e 74   .com··Us er-Agent
0070  3a 20 47 6f 2d 68 74 74  70 2d 63 6c 69 65 6e 74   : Go-htt p-client
0080  2f 31 2e 31 0d 0a 41 63  63 65 70 74 2d 45 6e 63   /1.1··Ac cept-Enc
0090  6f 64 69 6e 67 3a 20 67  7a 69 70 0d 0a 43 6f 6e   oding: g zip··Con
00a0  6e 65 63 74 69 6f 6e 3a  20 63 6c 6f 73 65 0d 0a   nection:  close··
00b0  0d 0a                                              ··
```

In this screenshot, a filter has been applied to display only HTTP (Hypertext Transfer Protocol) traffic. By focusing specifically on HTTP packets, this filter facilitates a targeted analysis of web-related activities within the network. The packet list reflects exclusively HTTP traffic, providing insights into web requests and responses.


My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
dns
dns
dnsserver
Domain Name System: Protocol
Packets: 664 · Displayed: 31 (4.7%)
Profile: Default

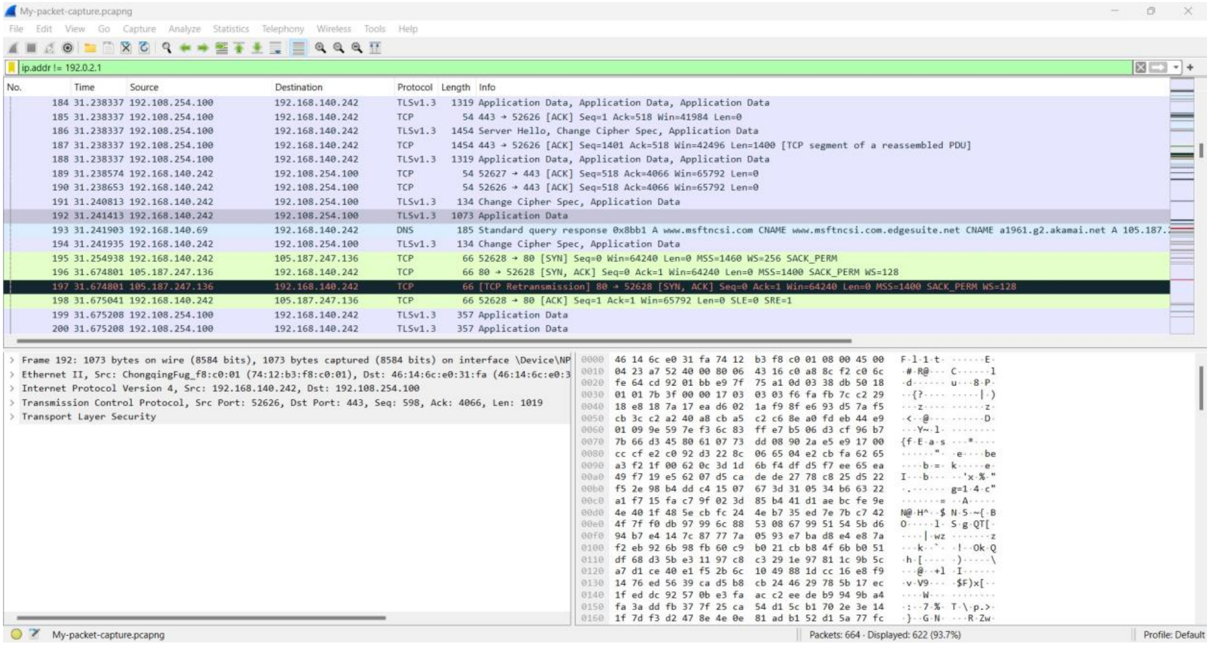
```
No.  Time       Source           Destination      Protocol Length Info
       112079   192.168.140.242  192.168.140.69   DNS          76 Standard query 0x8bb1 A www.msftncsi.com
   179 31.237115 192.168.140.242 192.168.140.69   DNS          76 Standard query 0x8bb1 A www.msftncsi.com
   180 31.238018 192.168.140.69  192.168.140.242  DNS         185 Standard query response 0x8bb1 A www.msftncsi.com CNAME www.msftncsi.com.edgesuite.net CNAME a1961.g2.akamai.net A 105.187.
   193 31.241903 192.168.140.69  192.168.140.242  DNS         185 Standard query response 0x8bb1 A www.msftncsi.com CNAME www.msftncsi.com.edgesuite.net CNAME a1961.g2.akamai.net A 105.187.
   218 33.836398 192.168.140.242 192.168.140.69   DNS          71 Standard query 0x99c5 A ntp.msn.com
   219 33.853032 192.168.140.242 192.168.140.69   DNS          71 Standard query 0x5d90 HTTPS ntp.msn.com
   220 33.853083 192.168.140.242 192.168.140.69   DNS          71 Standard query 0x5e8e A ntp.msn.com
   221 34.859644 192.168.140.242 192.168.140.69   DNS          71 Standard query 0xe0f2 A ntp.msn.com
   222 34.863404 192.168.140.69  192.168.140.242  DNS         173 Standard query response 0x5d90 HTTPS ntp.msn.com CNAME www-msn-com.a-0003.a-msedge.net SOA ns1.a-msedge.net
   223 34.863404 192.168.140.69  192.168.140.242  DNS         146 Standard query response 0x5e8e A ntp.msn.com CNAME www-msn-com.a-0003.a-msedge.net CNAME a-0003.a-msedge.net A 204.79.197.2
   224 34.868280 192.168.140.242 192.168.140.69   DNS          71 Standard query 0xc31e A ntp.msn.com
   226 34.897928 192.168.140.69  192.168.140.242  DNS         165 Standard query response 0xe0f2 A ntp.msn.com CNAME www-msn-com.a-0003.a-msedge.net CNAME a-0003.a-msedge.net A 204.79.197.2
   227 34.897928 192.168.140.69  192.168.140.242  DNS         165 Standard query response 0xc31e A ntp.msn.com CNAME www-msn-com.a-0003.a-msedge.net CNAME a-0003.a-msedge.net A 204.79.197.2
   321 38.359997 192.168.140.242 192.168.140.69   DNS          88 Standard query 0x9ab4 A f-log-extension.grammarly.io
   322 38.360067 192.168.140.242 192.168.140.69   DNS          88 Standard query 0xca1f HTTPS f-log-extension.grammarly.io
   327 38.369564 192.168.140.242 192.168.140.69   DNS          87 Standard query 0xbf04 A stream-production.avcdn.net
   333 38.421260 192.168.140.69  192.168.140.242  DNS         216 Standard query response 0x9ab4 A f-log-extension.grammarly.io A 52.72.172.119 A 34.193.10.212 A 18.204.87.72 A 34.239.221.1
```

```
> Frame 193: 185 bytes on wire (1480 bits), 185 bytes captured (1480 bits) on interface \Device\NPF_
> Ethernet II, Src: 46:14:6c:e0:31:fa (46:14:6c:e0:31:fa), Dst: ChongqingFug_f8:c0:01 (74:12:b3:f8:c
> Internet Protocol Version 4, Src: 192.168.140.69, Dst: 192.168.140.242
> User Datagram Protocol, Src Port: 53, Dst Port: 57478
> Domain Name System (response)
```

```
0000  74 12 b3 f8 c0 01 46 14  6c e0 31 fa 08 00 45 00   t·····F· l·1···E·
0010  00 ab 13 4b 40 00 40 11  8c 6e c0 a8 8c 45 c0 a8   ···K@·@· ·n···E··
0020  8c f2 00 35 e0 86 00 97  51 0b 8b b1 81 80 00 01   ···5···· Q·······
0030  00 04 00 00 00 00 03 77  77 77 08 6d 73 66 74 6e   ·······w ww·msftn
0040  63 73 69 03 63 6f 6d 00  00 01 00 01 c0 0c 00 05   csi·com· ········
0050  00 01 00 00 00 ac 00 20  03 77 77 77 08 6d 73 66   ·······  ·www·msf
0060  74 6e 63 73 69 03 63 6f  6d 09 65 64 67 65 73 75   tncsi·co m·edgesu
0070  69 74 65 03 6e 65 74 00  c0 2e 00 05 00 01 00 00   ite·net· ·.······
0080  01 1a 00 15 05 61 31 39  36 31 02 67 32 06 61 6b   ·····a19 61·g2·ak
0090  61 6d 61 69 03 6e 65 74  00 c0 5a 00 01 00 01 00   amai·net ··Z·····
00a0  00 00 0b 00 04 69 bb f7  9b c0 5a 00 01 00 01 00   ·····i·· ··Z·····
00b0  00 00 0b 00 04 69 bb f7  88                        ·····i···
```

This screenshot demonstrates the application of a filter to display only DNS (Domain Name System) traffic. By focusing on DNS packets, this filter

facilitates a targeted analysis of domain name resolutions within the network. The packet list reflects exclusively DNS traffic, providing insights into domain queries and responses.



My-packet-capture.pcapng

ip.addr != 192.0.2.1

```
No.    Time      Source           Destination      Protocol Length Info
   184 31.238337 192.108.254.100  192.168.140.242  TLSv1.3  1319 Application Data, Application Data, Application Data
   185 31.238337 192.108.254.100  192.168.140.242  TCP        54 443 → 52626 [ACK] Seq=1 Ack=518 Win=41984 Len=0
   186 31.238337 192.108.254.100  192.168.140.242  TLSv1.3  1454 Server Hello, Change Cipher Spec, Application Data
   187 31.238337 192.108.254.100  192.168.140.242  TCP      1454 443 → 52626 [ACK] Seq=1401 Ack=518 Win=42496 Len=1400 [TCP segment of a reassembled PDU]
   188 31.238337 192.108.254.100  192.168.140.242  TLSv1.3  1319 Application Data, Application Data, Application Data
   189 31.238574 192.168.140.242  192.108.254.100  TCP        54 52627 → 443 [ACK] Seq=518 Ack=4066 Win=65792 Len=0
   190 31.238653 192.168.140.242  192.108.254.100  TCP        54 52626 → 443 [ACK] Seq=518 Ack=4066 Win=65792 Len=0
   191 31.240813 192.168.140.242  192.108.254.100  TLSv1.3   134 Change Cipher Spec, Application Data
   192 31.241413 192.168.140.242  192.108.254.100  TLSv1.3  1073 Application Data
   193 31.241903 192.168.140.69   192.168.140.242  DNS       185 Standard query response 0x8bb1 A www.msftncsi.com CNAME www.msftncsi.com.edgesuite.net CNAME a1961.g2.akamai.net A 105.187.2
   194 31.241935 192.168.140.242  192.108.254.100  TLSv1.3   134 Change Cipher Spec, Application Data
   195 31.254938 192.168.140.242  105.187.247.136  TCP        66 52628 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
   196 31.674801 105.187.247.136  192.168.140.242  TCP        66 80 → 52628 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1400 SACK_PERM WS=128
   197 31.674801 105.187.247.136  192.168.140.242  TCP        66 [TCP Retransmission] 80 → 52628 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1400 SACK_PERM WS=128
   198 31.675041 192.168.140.242  105.187.247.136  TCP        66 52628 → 80 [ACK] Seq=1 Ack=1 Win=65792 Len=0 SLE=0 SRE=1
   199 31.675208 192.108.254.100  192.168.140.242  TLSv1.3   357 Application Data
   200 31.675208 192.108.254.100  192.168.140.242  TLSv1.3   357 Application Data
```

```
> Frame 192: 1073 bytes on wire (8584 bits), 1073 bytes captured (8584 bits) on interface \Device\NP
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Internet Protocol Version 4, Src: 192.168.140.242, Dst: 192.108.254.100
> Transmission Control Protocol, Src Port: 52626, Dst Port: 443, Seq: 598, Ack: 4066, Len: 1019
> Transport Layer Security
```

```
0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 00 45 00   F·l·1·t· ······E·
0010  04 23 a7 52 40 00 80 06  43 16 c0 a8 8c f2 c0 6c   ·#·R@··· C······l
0020  fe 64 cd 92 01 bb e9 7f  75 a1 0d 03 38 db 50 18   ·d······ u···8·P·
0030  01 01 7b 3f 00 00 17 03  03 03 f6 fa fb 7c c2 29   ··{?···· ·····|·)
0040  18 e8 18 7a 17 ea d6 02  1a f9 8f e6 93 d5 7a f5   ···z···· ······z·
0050  cb 3c c2 a2 40 a8 cb a5  c2 c6 8e a0 fd eb 44 e9   ·<··@··· ······D·
0060  01 09 9e 59 7e f3 6c 83  ff e7 b5 06 d3 cf 96 b7   ···Y~·l· ········
0070  7b 66 d3 45 80 61 07 73  dd 08 90 2a e5 e9 17 00   {f·E·a·s ···*····
0080  cc cf e2 c0 92 d3 22 8c  06 65 04 e2 cb fa 62 65   ······"· ·e····be
0090  a3 f2 1f 00 62 0c 3d 1d  6b f4 df d5 f7 ee 65 ea   ····b·=· k·····e·
00a0  49 f7 19 e5 62 07 d5 ca  de de 27 78 c8 25 d5 22   I···b··· ··'x·%·"
00b0  f5 2e 98 b4 dd c4 15 07  67 3d 31 05 34 b6 63 22   ·.······ g=1·4·c"
00c0  a1 f7 15 fa c7 9f 02 3d  85 b4 41 d1 ae bc fe 9e   ·······= ··A·····
00d0  4e 40 1f 48 5e cb fc 24  4e b7 35 ed 7e 7b c7 42   N@·H^··$ N·5·~{·B
00e0  4f 7f f0 db 97 99 6c 88  53 08 67 99 51 54 5b d6   O·····l· S·g·QT[·
00f0  94 b7 e4 14 7c 87 77 7a  05 93 e7 ba d8 e4 e8 7a   ····|·wz ·······z
0100  f2 eb 92 6b 98 fb 60 c9  b0 21 cb b8 4f 6b b0 51   ···k··`· ·!··Ok·Q
0110  df 68 d3 5b e3 11 97 c8  c3 29 1e 97 81 1c 9b 5c   ·h·[···· ·)·····\
0120  a7 d1 ce 40 e1 f5 2b 6c  10 49 88 1d cc 16 e8 f9   ···@··+l ·I······
0130  14 76 ed 56 39 ca d5 b8  cb 24 46 29 78 5b 17 ec   ·v·V9··· ·$F)x[··
0140  1f ed dc 92 57 0b e3 fa  ac c2 ee de b9 94 9b a4   ····W··· ········
0150  fa 3a dd fb 37 7f 25 ca  54 d1 5c b1 70 2e 3e 14   ·:··7·%· T·\·p.>·
0160  1f 7d f3 d2 47 8e 4e 0e  81 ad b1 52 d1 5a 77 fc   ·}··G·N· ···R·Zw·
```

My-packet-capture.pcapng

Packets: 664 · Displayed: 622 (93.7%)

Profile: Default

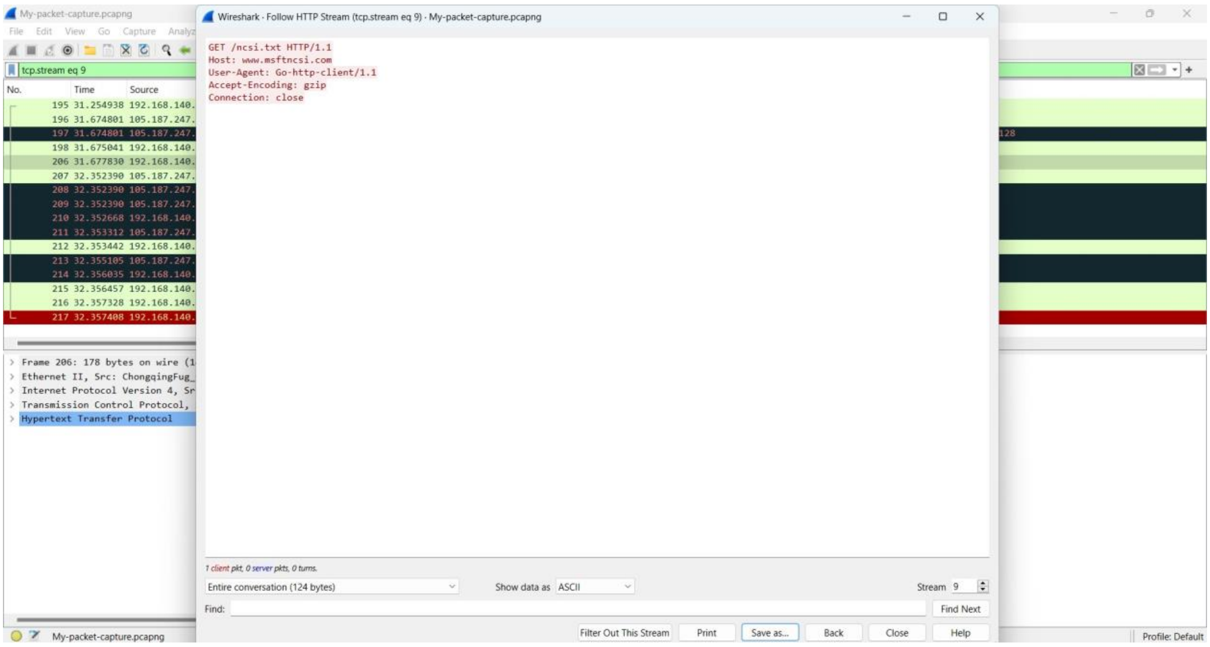
This screenshot, a filter has been applied to display packets where the source or destination IP address is not equal to 192.0.2.1. This filter allows for the exclusion of traffic involving a specific IP address, providing a focused view of network activities that do not relate to this address. The packet list reflects packets meeting the specified criteria, aiding in the identification of non-related network traffic.

B. Advanced Protocol Analysis:

Objective:

Conduct a deeper analysis of network traffic during specific activities to gain in-depth insights into protocol interactions and behaviours.

- **HTTP Stream Analysis**


My-packet-capture.pcapng
File Edit View Go Capture Analyz
Wireshark · Follow HTTP Stream (tcp.stream eq 9) · My-packet-capture.pcapng
Profile: Default

```
tcp.stream eq 9
No.        Time      Source
     195 31.254938 192.168.140.
     196 31.674801 105.187.247.
     197 31.674801 105.187.247.
     198 31.675041 192.168.140.
     206 31.677830 192.168.140.
     207 32.352390 105.187.247.
     208 32.352390 105.187.247.
     209 32.352390 105.187.247.
     210 32.352668 192.168.140.
     211 32.353312 105.187.247.
     212 32.353442 192.168.140.
     213 32.355105 105.187.247.
     214 32.356035 192.168.140.
     215 32.356457 192.168.140.
     216 32.357328 192.168.140.
     217 32.357408 192.168.140.

> Frame 206: 178 bytes on wire (1
> Ethernet II, Src: ChongqingFug_
> Internet Protocol Version 4, Sr
> Transmission Control Protocol,
> Hypertext Transfer Protocol
```

```
GET /ncsi.txt HTTP/1.1
Host: www.msftncsi.com
User-Agent: Go-http-client/1.1
Accept-Encoding: gzip
Connection: close
```

1 client pkt, 0 server pkts, 0 turns.

Entire conversation (124 bytes) Show data as ASCII Stream 9

Find:

This screenshot illustrates the HTTP stream analysis using Wireshark's "Follow TCP Stream" feature. The reconstructed stream provides a detailed view of HTTP requests and responses during the specific web browsing activity. Key information, such as request headers, response codes, and content, is extracted, contributing to a comprehensive understanding of HTTP interactions.

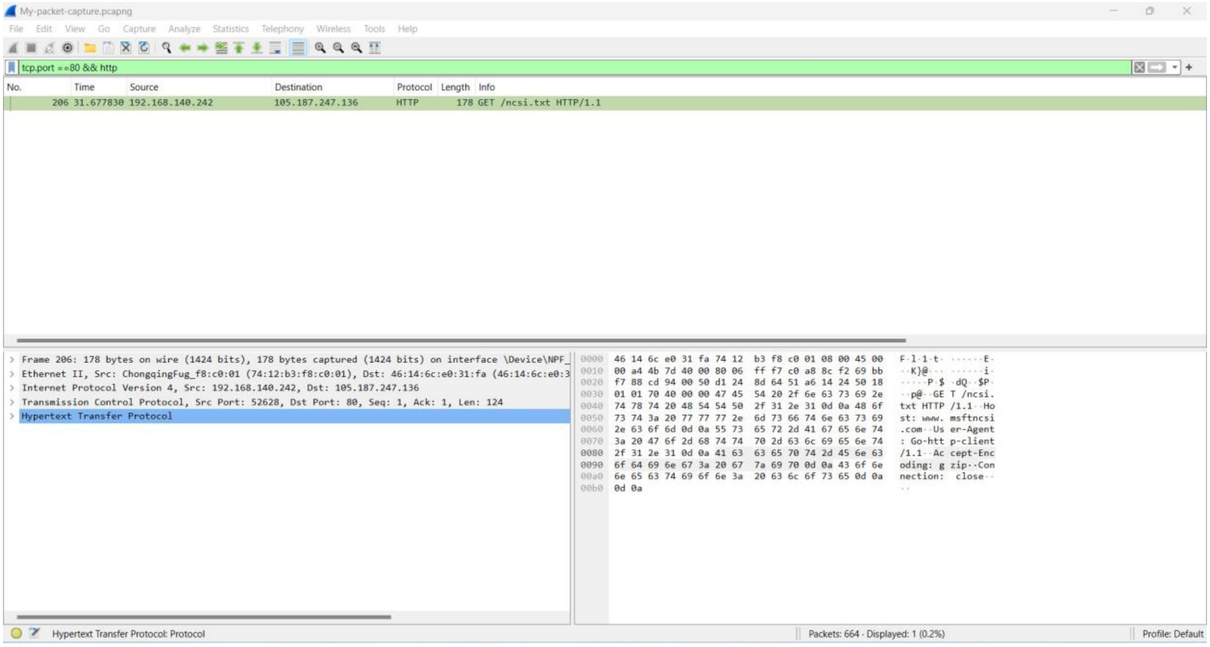
C. Troubleshooting Scenario: Slow Web Page Load

Objective:

Identify and troubleshoot the root cause of slow web page loading times.

Scenario Description:

Users have reported experiencing delays when accessing a specific website. The goal is to investigate the network traffic associated with loading the web page and identify any issues causing the slowdown.


My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Hypertext Transfer Protocol: Protocol
Packets: 664 · Displayed: 1 (0.2%)
Profile: Default

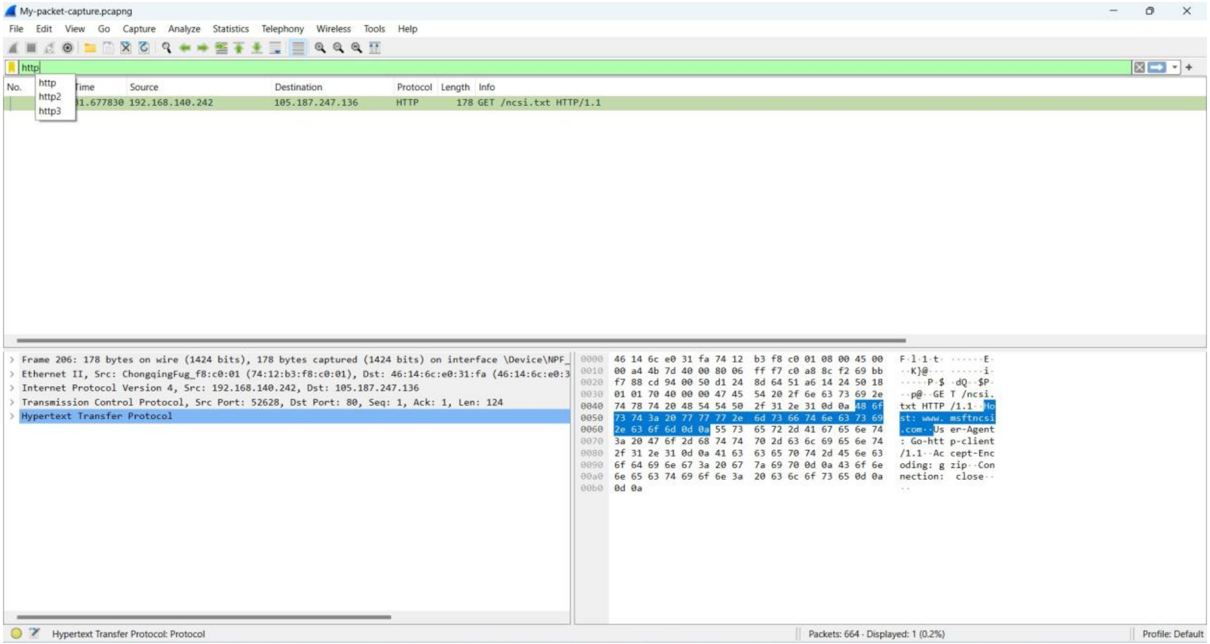
```
tcp.port ==80 && http
```

```
No.         Time      Source              Destination         Protocol  Length  Info
      206 31.677830 192.168.140.242     105.187.247.136     HTTP         178 GET /ncsi.txt HTTP/1.1
```

```
> Frame 206: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface \Device\NPF_
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Internet Protocol Version 4, Src: 192.168.140.242, Dst: 105.187.247.136
> Transmission Control Protocol, Src Port: 52628, Dst Port: 80, Seq: 1, Ack: 1, Len: 124
> Hypertext Transfer Protocol
```

```
0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 00 45 00   F·l·1·t· ······E·
0010  00 a4 4b 7d 40 00 80 06  ff f7 c0 a8 8c f2 69 bb   ··K}@··· ······i·
0020  f7 88 cd 94 00 50 d1 24  8d 64 51 a6 14 24 50 18   ·····P·$ ·dQ··$P·
0030  01 01 70 40 00 00 47 45  54 20 2f 6e 63 73 69 2e   ··p@··GE T /ncsi.
0040  74 78 74 20 48 54 54 50  2f 31 2e 31 0d 0a 48 6f   txt HTTP /1.1··Ho
0050  73 74 3a 20 77 77 77 2e  6d 73 66 74 6e 63 73 69   st: www. msftncsi
0060  2e 63 6f 6d 0d 0a 55 73  65 72 2d 41 67 65 6e 74   .com··Us er-Agent
0070  3a 20 47 6f 2d 68 74 74  70 2d 63 6c 69 65 6e 74   : Go-htt p-client
0080  2f 31 2e 31 0d 0a 41 63  63 65 70 74 2d 45 6e 63   /1.1··Ac cept-Enc
0090  6f 64 69 6e 67 3a 20 67  7a 69 70 0d 0a 43 6f 6e   oding: g zip··Con
00a0  6e 65 63 74 69 6f 6e 3a  20 63 6c 6f 73 65 0d 0a   nection:  close··
00b0  0d 0a                                              ··
```

This screenshot showcases the filtered view of network traffic using the filter tcp.port == 80 && http. By applying this filter, we isolate HTTP traffic specifically on port 80, providing a detailed look into web-related activities. The packet list displays HTTP requests and responses, aiding in the analysis of potential issues related to web browsing.


My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
http2
http3
Hypertext Transfer Protocol: Protocol
Packets: 664 · Displayed: 1 (0.2%)
Profile: Default

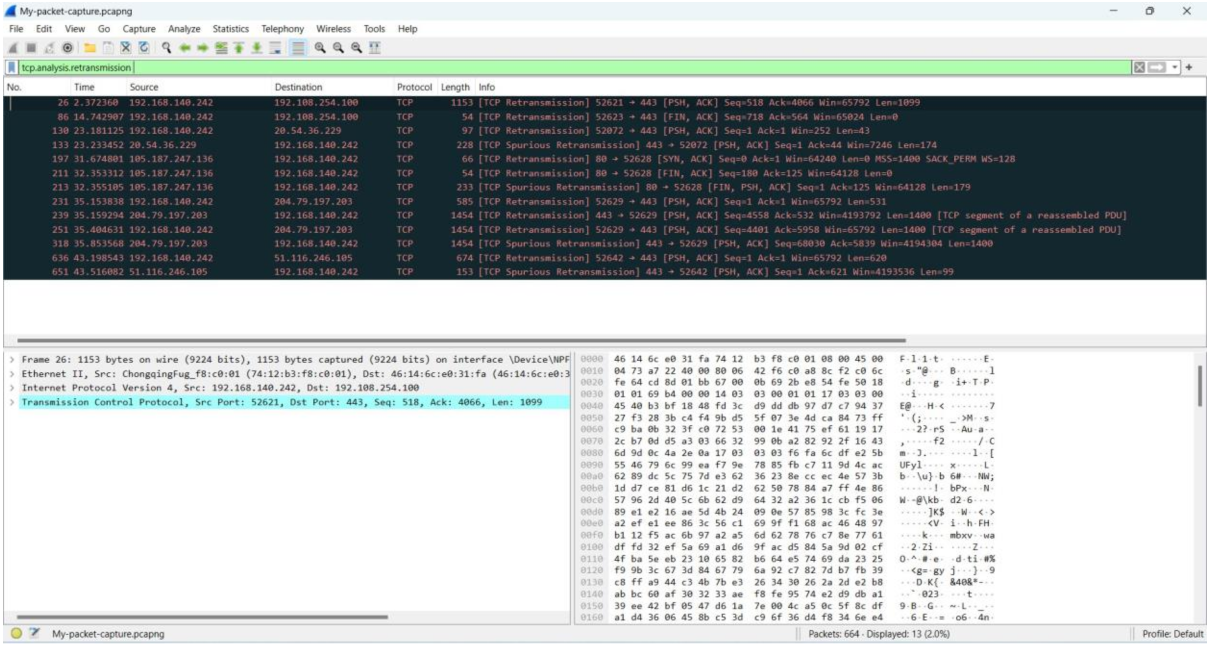
```
http
```

```
No.         Time      Source              Destination         Protocol  Length  Info
           1.677830 192.168.140.242     105.187.247.136     HTTP         178 GET /ncsi.txt HTTP/1.1
```

```
> Frame 206: 178 bytes on wire (1424 bits), 178 bytes captured (1424 bits) on interface \Device\NPF_
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Internet Protocol Version 4, Src: 192.168.140.242, Dst: 105.187.247.136
> Transmission Control Protocol, Src Port: 52628, Dst Port: 80, Seq: 1, Ack: 1, Len: 124
> Hypertext Transfer Protocol
```

```
0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 00 45 00   F·l·1·t· ······E·
0010  00 a4 4b 7d 40 00 80 06  ff f7 c0 a8 8c f2 69 bb   ··K}@··· ······i·
0020  f7 88 cd 94 00 50 d1 24  8d 64 51 a6 14 24 50 18   ·····P·$ ·dQ··$P·
0030  01 01 70 40 00 00 47 45  54 20 2f 6e 63 73 69 2e   ··p@··GE T /ncsi.
0040  74 78 74 20 48 54 54 50  2f 31 2e 31 0d 0a 48 6f   txt HTTP /1.1··Ho
0050  73 74 3a 20 77 77 77 2e  6d 73 66 74 6e 63 73 69   st: www. msftncsi
0060  2e 63 6f 6d 0d 0a 55 73  65 72 2d 41 67 65 6e 74   .com··Us er-Agent
0070  3a 20 47 6f 2d 68 74 74  70 2d 63 6c 69 65 6e 74   : Go-htt p-client
0080  2f 31 2e 31 0d 0a 41 63  63 65 70 74 2d 45 6e 63   /1.1··Ac cept-Enc
0090  6f 64 69 6e 67 3a 20 67  7a 69 70 0d 0a 43 6f 6e   oding: g zip··Con
00a0  6e 65 63 74 69 6f 6e 3a  20 63 6c 6f 73 65 0d 0a   nection:  close··
00b0  0d 0a                                              ··
```

This screenshot demonstrates the application of the http filter, isolating packets related to the HTTP protocol. This focused view allows for the detailed analysis of HTTP requests and responses during the troubleshooting of slow web page loading.


My-packet-capture.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.analysis.retransmission

```
No.     Time      Source            Destination       Protocol Length Info
     26 2.372360  192.168.140.242   192.108.254.100   TCP      1153 [TCP Retransmission] 52621 → 443 [PSH, ACK] Seq=518 Ack=4066 Win=65792 Len=1099
     86 14.742907 192.168.140.242   192.108.254.100   TCP        54 [TCP Retransmission] 52623 → 443 [FIN, ACK] Seq=718 Ack=564 Win=65024 Len=0
    130 23.181125 192.168.140.242   20.54.36.229      TCP        97 [TCP Retransmission] 52072 → 443 [PSH, ACK] Seq=1 Ack=1 Win=252 Len=43
    133 23.233452 20.54.36.229      192.168.140.242   TCP       228 [TCP Spurious Retransmission] 443 → 52072 [PSH, ACK] Seq=1 Ack=44 Win=7246 Len=174
    197 31.674801 105.187.247.136   192.168.140.242   TCP        66 [TCP Retransmission] 80 → 52628 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1400 SACK_PERM WS=128
    211 32.353312 105.187.247.136   192.168.140.242   TCP        54 [TCP Retransmission] 80 → 52628 [FIN, ACK] Seq=180 Ack=125 Win=64128 Len=0
    213 32.355105 105.187.247.136   192.168.140.242   TCP       233 [TCP Spurious Retransmission] 80 → 52628 [FIN, PSH, ACK] Seq=1 Ack=125 Win=64128 Len=179
    231 35.153838 192.168.140.242   204.79.197.203    TCP       585 [TCP Retransmission] 52629 → 443 [PSH, ACK] Seq=1 Ack=1 Win=65792 Len=531
    239 35.159294 204.79.197.203    192.168.140.242   TCP      1454 [TCP Retransmission] 443 → 52629 [PSH, ACK] Seq=4558 Ack=532 Win=4193792 Len=1400 [TCP segment of a reassembled PDU]
    251 35.404631 192.168.140.242   204.79.197.203    TCP      1454 [TCP Retransmission] 52629 → 443 [PSH, ACK] Seq=4401 Ack=5958 Win=65792 Len=1400 [TCP segment of a reassembled PDU]
    318 35.853568 204.79.197.203    192.168.140.242   TCP      1454 [TCP Spurious Retransmission] 443 → 52629 [PSH, ACK] Seq=68030 Ack=5839 Win=4194304 Len=1400
    636 43.198543 192.168.140.242   51.116.246.105    TCP       674 [TCP Retransmission] 52642 → 443 [PSH, ACK] Seq=1 Ack=1 Win=65792 Len=620
    651 43.516082 51.116.246.105    192.168.140.242   TCP       153 [TCP Spurious Retransmission] 443 → 52642 [PSH, ACK] Seq=1 Ack=621 Win=4193536 Len=99
```

```
> Frame 26: 1153 bytes on wire (9224 bits), 1153 bytes captured (9224 bits) on interface \Device\NPF
> Ethernet II, Src: ChongqingFug_f8:c0:01 (74:12:b3:f8:c0:01), Dst: 46:14:6c:e0:31:fa (46:14:6c:e0:3
> Internet Protocol Version 4, Src: 192.168.140.242, Dst: 192.108.254.100
> Transmission Control Protocol, Src Port: 52621, Dst Port: 443, Seq: 518, Ack: 4066, Len: 1099
```

```
0000  46 14 6c e0 31 fa 74 12  b3 f8 c0 01 08 00 45 00   F·l·1·t· ······E·
0010  04 73 a7 22 40 00 80 06  42 f6 c0 a8 8c f2 c0 6c   ·s·"@··· B······l
0020  fe 64 cd 8d 01 bb 67 00  0b 69 2b e8 54 fe 50 18   ·d····g· ·i+·T·P·
0030  01 01 69 b4 00 00 14 03  03 00 01 01 17 03 03 00   ··i····· ········
0040  45 40 b3 bf 18 48 fd 3c  d9 dd db 97 d7 c7 94 37   E@···H·< ·······7
0050  27 f3 28 3b c4 f4 9b d5  5f 07 3e 4d ca 84 73 ff   '·(;···· _·>M··s·
0060  c9 ba 0b 32 3f c0 72 53  00 1e 41 75 ef 61 19 17   ···2?·rS ··Au·a··
0070  2c b7 0d d5 a3 03 66 32  99 0b a2 82 92 2f 16 43   ,·····f2 ·····/·C
0080  6d 9d 0c 4a 2e 0a 17 03  03 03 f6 fa 6c df e2 5b   m··J.··· ····l··[
0090  55 46 79 6c 99 ea f7 9e  78 85 fb c7 11 9d 4c ac   UFyl···· x·····L·
00a0  62 89 dc 5c 75 7d e3 62  36 23 8e cc ec 4e 57 3b   b··\u}·b 6#···NW;
00b0  1d d7 ce 81 d6 1c 21 d2  62 50 78 84 a7 ff 4e 86   ······!· bPx···N·
00c0  57 96 2d 40 5c 6b 62 d9  64 32 a2 36 1c cb f5 06   W·-@\kb· d2·6····
00d0  89 e1 e2 16 ae 5d 4b 24  09 0e 57 85 98 3c fc 3e   ·····]K$ ··W··<·>
00e0  a2 ef e1 ee 86 3c 56 c1  69 9f f1 68 ac 46 48 97   ·····<V· i··h·FH·
00f0  b1 12 f5 ac 6b 97 a2 a5  6d 62 78 76 c7 8e 77 61   ····k··· mbxv··wa
0100  df fd 32 ef 5a 69 a1 d6  9f ac d5 84 5a 9d 02 cf   ··2·Zi·· ····Z···
0110  4f ba 5e eb 23 10 65 82  b6 64 e5 74 69 da 23 25   O·^·#·e· ·d·ti·#%
0120  f9 9b 3c 67 3d 84 67 79  6a 92 c7 82 7d b7 fb 39   ··<g=·gy j···}··9
0130  c8 ff a9 44 c3 4b 7b e3  26 34 30 26 2a 2d e2 b8   ···D·K{· &40&*-··
0140  ab bc 60 af 30 32 33 ae  f8 fe 95 74 e2 d9 db a1   ··`·023· ···t····
0150  39 ee 42 bf 05 47 d6 1a  7e 00 4c a5 0c 5f 8c df   9·B··G·· ~·L··_··
0160  a1 d4 36 06 45 8b c5 3d  c9 6f 36 d4 f8 34 6e e4   ··6·E··= ·o6··4n·
```

My-packet-capture.pcapng | Packets: 664 · Displayed: 13 (2.0%) | Profile: Default

This screenshot focuses on packets marked as TCP retransmissions using the tcp.analysis.retransmission filter. Identifying and analyzing these retransmissions can provide insights into potential network issues contributing to slow web page loading.

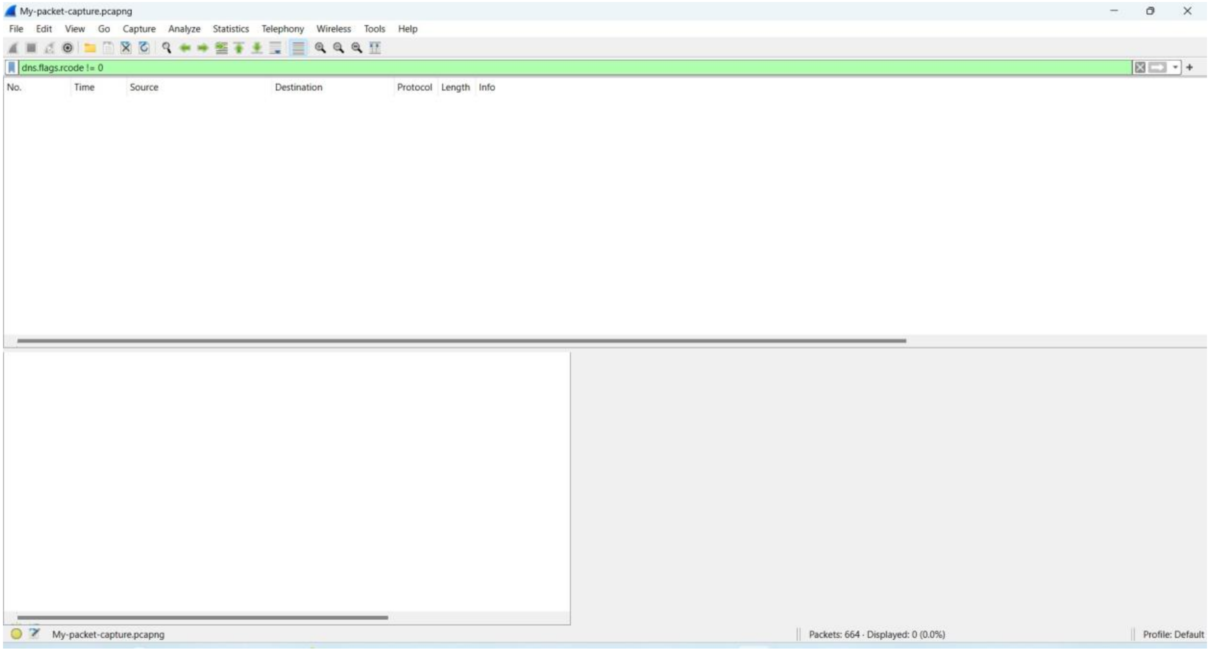
D. Security Analysis Scenario: DNS Spoofing Detection

Objective:

Identify and investigate a potential DNS spoofing attempt on the network.

Scenario Description:

Reports have surfaced about users encountering unexpected or potentially malicious websites. The goal is to analyze DNS traffic and detect any signs of DNS spoofing, where attackers may be redirecting legitimate domain resolutions to malicious IP addresses.



dns.flags.rcode != 0

Packets: 664 · Displayed: 0 (0.0%)

This screenshot represents the absence of DNS responses with non-zero response codes during the captured session. The empty view indicates that, based on the applied filter (dns.flags.rcode != 0), no DNS responses with abnormal response codes were identified.

3.Recommendations

- **Network Optimization:**
  Based on the identified issues during troubleshooting, consider implementing optimizations to improve network performance. Addressing factors such as slow web page loading times or connectivity issues can enhance the overall user experience.
- **Regular Network Monitoring:**
  Establish a routine for regular network monitoring using Wireshark or similar tools. Continuous monitoring helps identify and address emerging issues promptly, contributing to proactive network management.
- **Periodic Security Audits:**
  Conduct periodic security audits focusing on network traffic. Regularly analyze DNS responses, detect potential spoofing attempts, and stay vigilant against evolving security threats. Consider implementing intrusion detection systems for real-time threat detection.

- **Documentation and Knowledge Sharing:**
  Maintain comprehensive documentation of network configurations, troubleshooting steps, and security analysis procedures. Foster a culture of knowledge sharing among team members to ensure collective expertise in network management and security practices.

4. Future Work

- **Deep Dive into Specific Protocols:**
  Conduct more in-depth analyses on specific protocols, such as SIP for VoIP, SMTP for email, or others relevant to your network environment. This will provide a detailed understanding of how different applications interact on the network.
- **Integration with SIEM Solutions:**
  Explore the integration of Wireshark with Security Information and Event Management (SIEM) solutions for enhanced security monitoring. This integration can automate the detection of security incidents and provide a centralized platform for comprehensive analysis.
- **Advanced Threat Detection:**
  Investigate advanced threat detection mechanisms, including anomaly detection and behavioural analysis. Explore machine learning approaches to identify patterns indicative of potential security threats within the network traffic.
- **Performance Optimization Testing:**
  Extend performance optimization efforts by conducting performance testing under various conditions. Simulate peak usage scenarios and analyze network behaviour to identify potential bottlenecks, enabling proactive optimization.
- **Incident Response Planning:**
  Develop and refine incident response plans based on the insights gained from security analysis. Establish clear protocols for responding to and mitigating security incidents, ensuring a swift and effective response in the event of a security breach.
- **Explore Cloud-based Solutions:**

Investigate the use of cloud-based packet capture solutions for decentralized and scalable network monitoring. This can be particularly beneficial for organizations with distributed or cloud-based infrastructure.

5.Conculsion

In conclusion, as a lone individual conducting this Wireshark project, the journey has been both insightful and empowering. Through meticulous analysis and troubleshooting, I have gained a deep understanding of network dynamics, identified performance bottlenecks, and addressed potential security threats. The recommendations offered highlight the significance of continuous learning, regular monitoring, and proactive measures to enhance network efficiency. Looking ahead, areas of future work, including advanced threat detection and cloud-based solutions, present exciting opportunities for further exploration. This project has not only strengthened my proficiency in Wireshark but also affirmed the impact a dedicated individual can have in optimizing and securing network infrastructures.