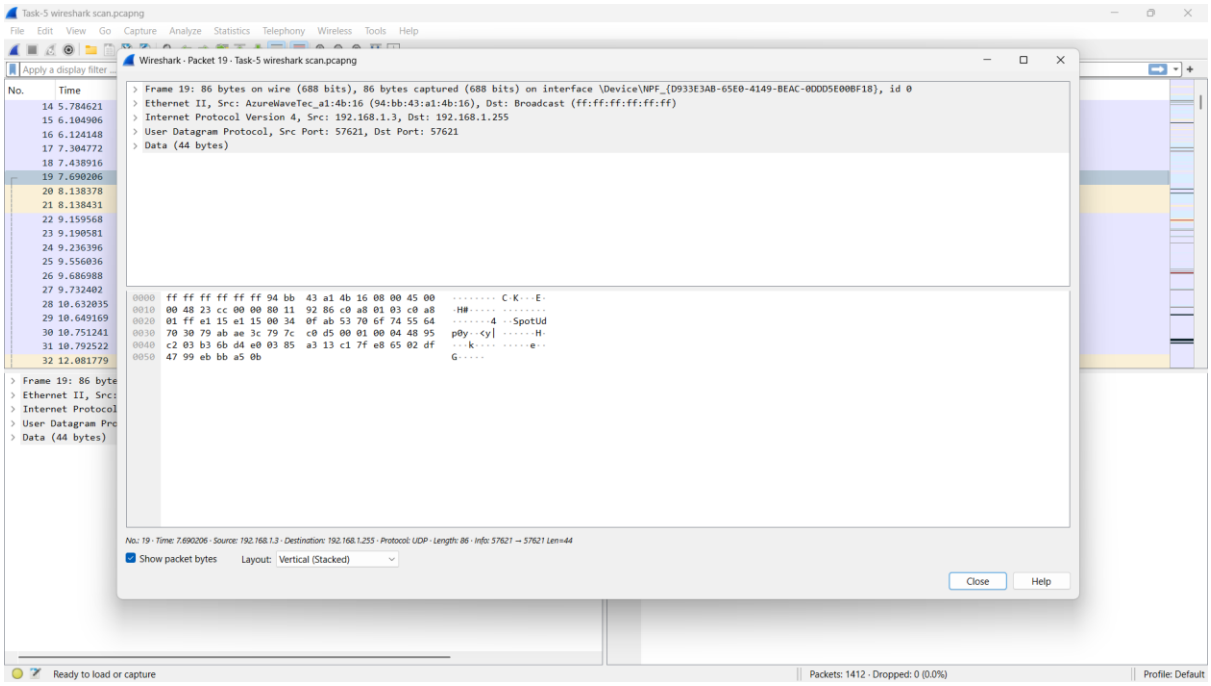
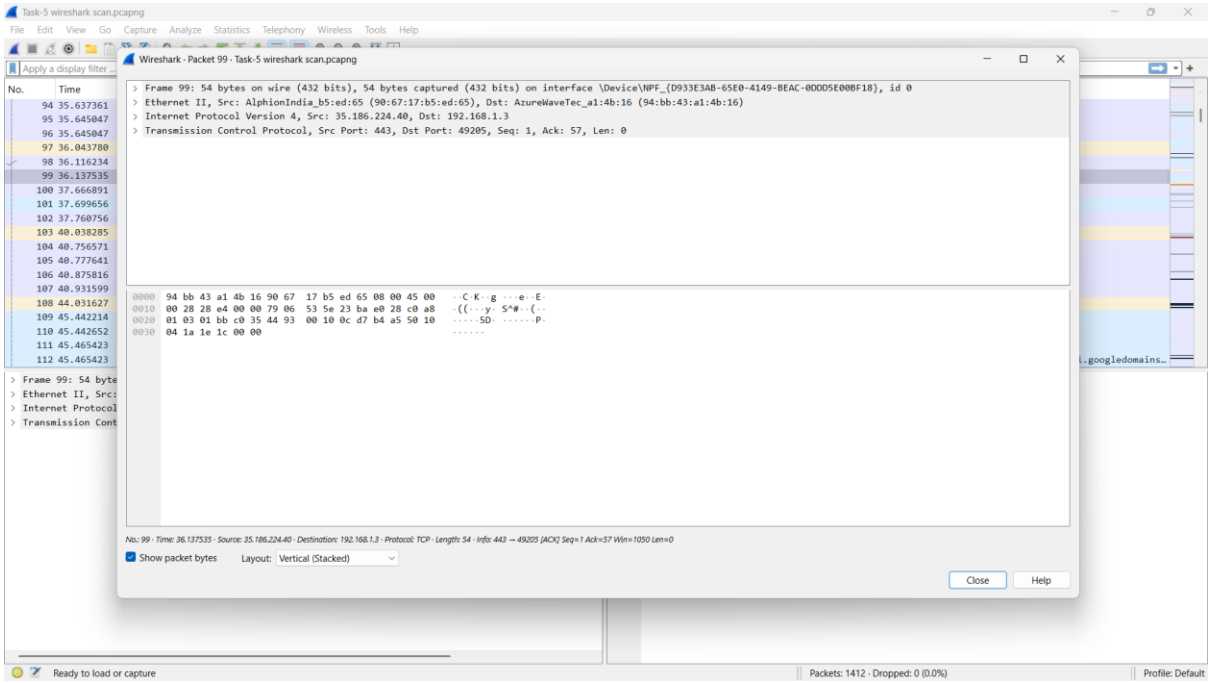
TASK-5

Capture and Analyze Network Traffic Using Wirwshark

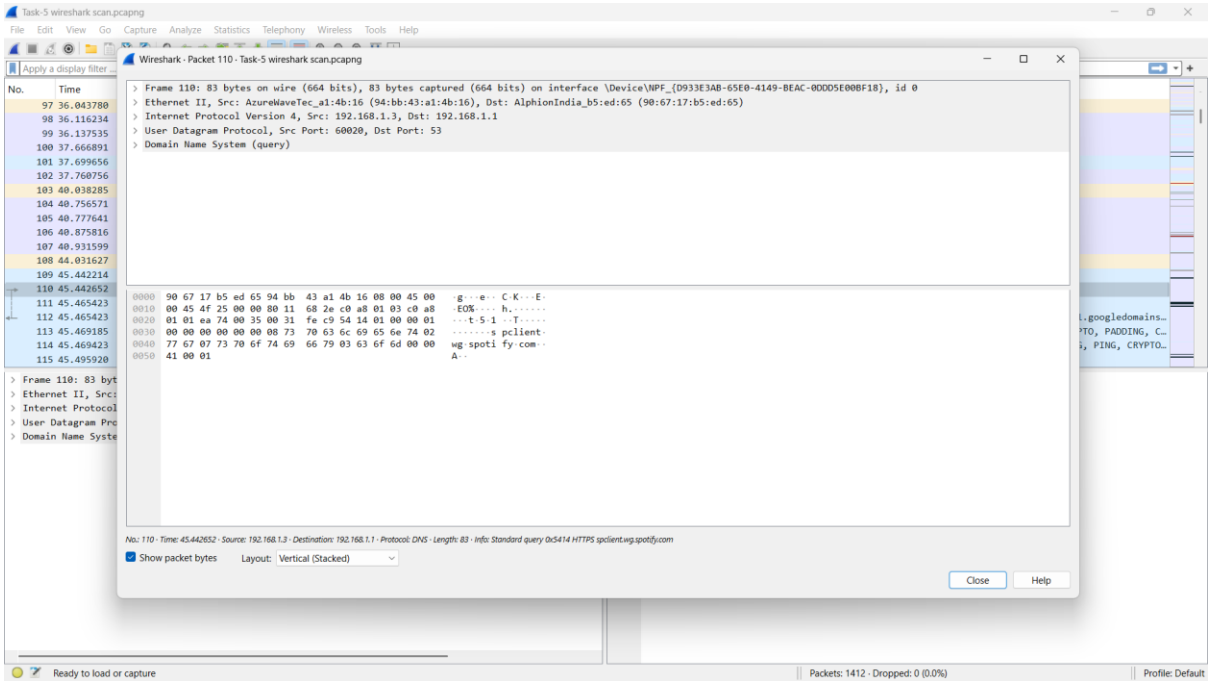
1. UDP



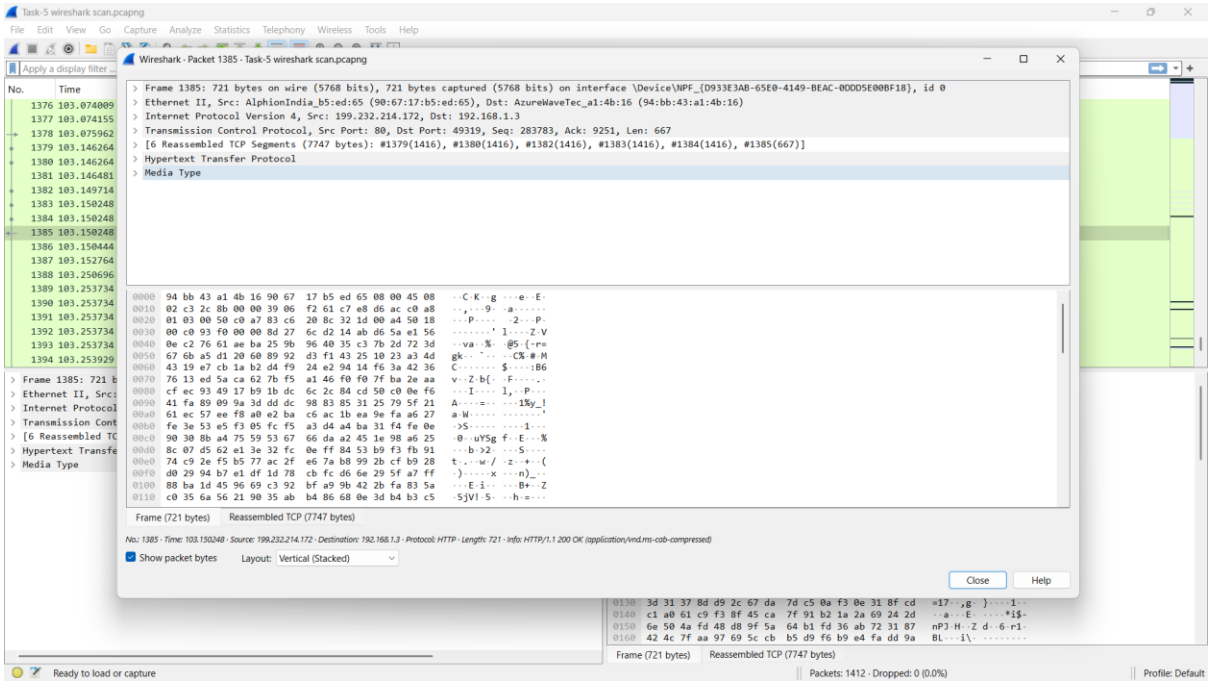
2. TCP:



3. DNS


Wireshark · Packet 110 · Task-5 wireshark scan.pcapng
Frame 110: 83 bytes on wire (664 bits), 83 bytes captured (664 bits) on interface \Device\NPF_{D933E3AB-65E0-4149-BEAC-0DDD5E00BF18}, id 0
Ethernet II, Src: AzureWaveTec_a1:4b:16 (94:bb:43:a1:4b:16), Dst: AlphionIndia_b5:ed:65 (90:67:17:b5:ed:65)
Internet Protocol Version 4, Src: 192.168.1.3, Dst: 192.168.1.1
User Datagram Protocol, Src Port: 60020, Dst Port: 53
Domain Name System (query)
Show packet bytes
Layout: Vertical (Stacked)
Close
Help

4. HTTP


Wireshark · Packet 1385 · Task-5 wireshark scan.pcapng
Frame 1385: 721 bytes on wire (5768 bits), 721 bytes captured (5768 bits) on interface \Device\NPF_{D933E3AB-65E0-4149-BEAC-0DDD5E00BF18}, id 0
Ethernet II, Src: AlphionIndia_b5:ed:65 (90:67:17:b5:ed:65), Dst: AzureWaveTec_a1:4b:16 (94:bb:43:a1:4b:16)
Internet Protocol Version 4, Src: 199.232.214.172, Dst: 192.168.1.3
Transmission Control Protocol, Src Port: 80, Dst Port: 49319, Seq: 283783, Ack: 9251, Len: 667
[6 Reassembled TCP Segments (7747 bytes): #1379(1416), #1380(1416), #1382(1416), #1383(1416), #1384(1416), #1385(667)]
Hypertext Transfer Protocol
Media Type
Frame (721 bytes)
Reassembled TCP (7747 bytes)
Show packet bytes
Layout: Vertical (Stacked)
Close
Help

Wireshark Network Packet Analysis Report
Task: Task-5 Network Scan
Analyst: [Sandeep]
Date: [June 2 2025]

---

1. Objective

To analyze the captured packets in the provided .pcapng file and evaluate the network communication behaviors, protocol usage, and endpoint interactions based on selected packet samples.

---

2. Tools Used

- **Wireshark v4.x**
- **Capture File**: Task-5 wireshark scan.pcapng
- **Operating System**: Windows

---

3. Packet Analysis Summary

Packet 19 – UDP Broadcast

- **Protocol**: UDP
- **Source IP**: 192.168.1.3
- **Destination IP**: 192.168.1.255 (Broadcast)
- **Source Port**: 57621
- **Destination Port**: 57621
- **Data**: 44 bytes
- **Details**:
  The source device is sending a UDP broadcast message, likely for service discovery or local peer communication (e.g., Spotify's local device detection).

Packet 99 – TCP Acknowledgment

- **Protocol**: TCP
- **Source IP**: 35.186.224.40 (likely external, possibly Google Cloud)
- **Destination IP**: 192.168.1.3 (local device)

- **Source Port**: 443 (HTTPS)
- **Destination Port**: 49205
- **Info**: Acknowledgment packet (ACK)
- **Details**:
  This packet acknowledges receipt of previous data, with Seq=1, Ack=57. No payload data is present (Len=0), indicating a TCP-level handshake or keep-alive.

Packet 110 – DNS Query

- **Protocol**: DNS over UDP
- **Source IP**: 192.168.1.3
- **Destination IP**: 192.168.1.1 (DNS server/gateway)
- **Source Port**: 60020
- **Destination Port**: 53
- **Query**: spclient.wg.spotify.com
- **Details**:
  This DNS query is resolving a Spotify subdomain, indicating a connection attempt to the Spotify client services.

Packet 1385 – HTTP Response

- **Protocol**: HTTP over TCP
- **Source IP**: 199.232.214.172
- **Destination IP**: 192.168.1.3
- **Source Port**: 80
- **Destination Port**: 49319
- **Status**: HTTP/1.1 200 OK
- **Content-Type**: application/vnd.ms-cab-compressed
- **Details**:
  This is a successful HTTP response containing compressed data. The size and type suggest it could be a Windows Update or a system-related download.

---

4. Key Observations

- DNS resolution and HTTP requests indicate typical client-server communication with cloud services and music streaming (Spotify).

- TCP ACK packets confirm established and maintained connections.
- UDP broadcast suggests service discovery or local device interaction behavior, possibly linked to media streaming or syncing.
- Compressed data from a public IP likely points to OS or application updates.

---

5. Recommendations

- Ensure DNS traffic is filtered and monitored for security and performance.
- Confirm whether large HTTP transfers (like in packet 1385) are expected; investigate unknown destinations.
- Monitor broadcast traffic for excessive activity, which may indicate misconfigurations or unwanted software behavior.

---

6. Attachments

- **1** – UDP Broadcast packet
- **2**– TCP ACK Packet
- 3– DNS Query to Spotify
- **4** – HTTP 200 OK with compressed data

7.Conclusion

The packet capture analysis reveals standard network behavior primarily involving DNS resolution, HTTP communication, and TCP session management. The observed traffic includes:

- Legitimate DNS queries to services like Spotify.
- Valid HTTP responses containing compressed data, likely related to software updates.
- TCP acknowledgments indicating stable connections.
- UDP broadcast traffic consistent with local network service discovery.

No signs of malicious or suspicious behavior were identified in the selected packet samples. However, continuous monitoring is recommended to ensure all external communications and broadcasts align with organizational policies and user activity. This analysis serves as a foundational review of network health and connectivity patterns.