Network Traffic Analysis Report — Task 5

File Analysed: networkTraffic.pcapng
Tool Used: Wireshark
Date: *27-10-2025*

1. Objective

To capture live network packets using Wireshark and analyse common network protocols to understand how data flows across different layers of the TCP/IP model.
This task enhances packet analysis, protocol identification, and troubleshooting skills.

2. Methodology

1. Installed and opened **Wireshark** on the system.
2. Selected the **Wi-Fi network interface** for packet capturing.
3. Started capture and generated traffic by:
    - Visiting websites (google.com, youtube.com, etc.)
    - Sending ping requests to external servers.
4. Stopped the capture after around one minute.
5. Applied protocol filters: dns, http, tcp, tls, icmp, arp, quic, udp.
6. Saved the file as networkTraffic.pcapng for further analysis.

3. Protocols Observed

Protocol	**Layer**	**Function**	**Example Observation**
DNS (Domain Name System)	Application	Resolves domain names to IP addresses	Query for google.com, amazon.in and responses
TCP (Transmission Control Protocol)	Transport	Ensures reliable data transfer between devices	Three-way handshake (SYN, SYN-ACK, ACK)
HTTP (Hypertext Transfer Protocol)	Application	Transfers unencrypted web content	GET requests to websites

QUIC (Quick UDP Internet Connections)	Application/Transport	Modern protocol combining transport and encryption (used for HTTP/3)	QUIC packets replacing TCP+TLS over port 443
ICMP (Internet Control Message Protocol)	Network	Used for diagnostics like ping	Echo request and reply packets
ARP (Address Resolution Protocol)	Data Link	Maps IP addresses to MAC addresses	ARP requests between local devices

4. Key Observations

- **DNS queries** occurred before HTTP and QUIC communications — resolving domain names first.
- **QUIC** packets replaced separate UDP or TCP layers since browsers used **HTTP/3 over QUIC** (built on top of UDP).
- **TLS** and **QUIC** packets were encrypted — only handshake and metadata visible, not payload.
- **ICMP** packets confirmed network connectivity through echo requests and replies.
- **ARP** activity showed devices on the LAN exchanging MAC address information.

5. Summary of Findings and Packet Details

- Captured approximately **52581 packets** in one minute of network activity.
- Identified multiple protocols: **DNS, TCP, HTTP, TLS, UDP (QUIC), ICMP, ARP.**
- **Most traffic** consisted of QUIC packets (modern encrypted HTTP/3 connections).
- **DNS lookups** preceded web traffic.
- **TLS and QUIC** handled encryption and session security.
- **ICMP packets** verified connectivity during ping tests.
- Local traffic included **ARP** and **IGMP** exchanges between host and gateway.

6. Analysis Summary

Category	**Observation**
Total Packets Captured	52581
Most Active Protocols	QUIC, TCP, DNS, TLS

Encryption	Present (TLS/QUIC)
Local Network Traffic	ARP and IGMP packets
Transport Layer	QUIC replaced TCP/UDP for HTTP/3 traffic
Web Traffic	Encrypted HTTPS and HTTP/3 packets
Ping Activity	ICMP Echo Request and Reply observed

7. Outcome

- Learned to **capture and filter** live packets using Wireshark.
- Understood how **different protocols** (DNS, HTTP, TLS, QUIC, ICMP, ARP, IGMP) work together.
- Observed how **modern browsers use QUIC/HTTP3** instead of TCP/TLS.
- Enhanced understanding of **encrypted vs unencrypted traffic**.
- Developed skills in **protocol analysis** and **network troubleshooting**.

8. Conclusion

The network traffic analysis successfully captured and decoded real-time internet communication.
The findings demonstrated that modern web traffic relies heavily on **QUIC over UDP**, providing faster, encrypted connections compared to traditional TCP/TLS-based traffic.
Additional protocols such as **DNS, ICMP, & ARP** showed underlying network operations and system-level interactions.
This exercise provided valuable insights into packet flow, network behaviour, and secure communication.