**Network Traffic Analysis Using Wireshark And Zeek**

**United College Of Engineering And Research,**

**Naini, Prayagraj, Uttar Pradesh**

Author : Anubhav Singh

Mentor : Mr. Ayush Kumar

**Department**

**Computer Science And Engineering**

Date Of Submission : 31/07/2025

Table of content

[Abstract 3](#_Toc16181)

[Network Packet Analysis Using Wireshark 4](#_Toc16182)

[Introduction 4](#_Toc16183)

[Methodology 4](#_Toc16184)

[1. Tools Used 4](#_Toc16185)

[2. *Data Collection* 5](#_Toc16186)

[Analysis Approach 5](#_Toc16187)

[Analysis and Findings 5](#_Toc16188)

[Initial Analysis from File 5](#_Toc16189)

[Observations 6](#_Toc16190)

[Security Analysis from Image 6](#_Toc16191)

[1. Malformed Packets 6](#_Toc16192)

[2. Protocol Anomalies 7](#_Toc16193)

[3. Sequence and Connection Issues 7](#_Toc16194)

[4. TLS and Decryption Issues 8](#_Toc16195)

[5. QUIC-Specific Anomalies 9](#_Toc16196)

[Mitigation Strategies 10](#_Toc16197)

[1. Network Hardening: 10](#_Toc16198)

[2. Traffic Monitoring and Logging: 10](#_Toc16199)

[3. Access Control: 10](#_Toc16200)

[4. Incident Response: 10](#_Toc16201)

[Conclusion 11](#_Toc16202)

[Citations 11](#_Toc16203)

# Abstract

This report explores a detailed evaluation of network traffic using Wireshark, aiming to study protocol operations, spot irregularities, and uncover potential cybersecurity risks. The investigation is grounded on two main resources: a packet capture file and a summarized expert report. The captured traffic showed active IPv6 communication using protocols such as UDP, TCP, ESP, and ICMPv6, with UDP traffic being the most frequent. Analysis of source and destination details, along with data flow patterns, helped illustrate the general behavior of the network.

The expert insights identified several critical issues like malformed packets, abnormal protocol usage, and encryption-related failures. Key findings include incorrectly structured TCP packets, anomalies within TLS and QUIC protocols, and outdated encryption mechanisms. These findings may point to threats such as unauthorized access, protocol manipulation, or interception attempts.

This report presents effective countermeasures such as enforcing secure communication standards, deploying packet inspection tools, and using intrusion detection systems. By studying network traffic at a granular level, the project improves awareness of modern cybersecurity risks. It demonstrates the practical use of packet analysis for identifying and minimizing threats in dynamic network environments.

# Network Packet Analysis Using Wireshark

## Introduction

This report presents the outcomes of a basic network traffic capture and analysis exercise carried out using Wireshark, a powerful and widely adopted tool for network protocol analysis. The main goal was to observe and interpret network traffic to understand communication flows, recognize commonly used protocols, and identify early signs of potential security risks. The data analyzed came from two key sources: a packet capture text file titled “Network packet capture.txt” containing Ethernet frame data, and an image titled “network capture expert info.png”, which summarizes expert-level insights based on the same capture. Together, these inputs were examined to highlight patterns of interest, protocol misuses, and possible signs of malicious activity, followed by a summary of the security concerns and mitigation techniques..

The analysis was performed on July 25, 2025, at 11:56 AM IST, ensuring that the findings are both recent and relevant. The report follows a clear structure, beginning with the tools and methods used, followed by detailed observations, a breakdown of anomalies, and final recommendations for improving network defenses. The goal is to provide foundational knowledge of packet-level analysis and its significance in building stronger cybersecurity frameworks..

## Methodology

#### 1. Tools Used

**Wireshark:** Used to import, decode, and analyze network data from the provided packet capture text file. It enabled inspection of protocol layers, frame headers, and traffic behavior.

**Image Analysis:** The expert summary image, "network capture expert info.png", was evaluated to extract structured insights regarding anomalies, warnings, and protocol-specific issues observed during the capture.

#### 2. *Data Collection*

1. **Text File:** The file named "Network packet capture.txt" was imported into Wireshark for a packet-level breakdown. It contained 66 Ethernet frames primarily involving IPv6 communication, captured between timestamps 06:02:41.750,498 and 06:03:17.342,144.
2. **Image File**: The image contains a table summarizing various events or issues detected in the network traffic, categorized by severity (Warning, Error, Chat, Note), group (Decryption, Malformed, Protocol, Sequence), protocol (TLS, TCP, QUIC), and count (frequency of occurrence). This data was analyzed to identify potential security threats.

## Analysis Approach

1. **Initial Analysis:** The textual capture was examined for traffic structure, addressing patterns, and protocol frequency. This included decoding Ethernet headers, identifying MAC and IP addresses, and examining trends in communication flow.
2. **Security Analysis:** The image-based expert output was used to detect potential intrusions, protocol violations, and encrypted traffic failures. Emphasis was placed on high-severity alerts and unusual patterns to understand threat vectors and recommend countermeasures.

## Analysis and Findings

### Initial Analysis from File

The initial analysis of the text file revealed the following key insights:

1. **Packet Structure and Protocols:**

The captured packets are Ethernet frames encapsulating IPv6 traffic, confirmed by the EtherType `86|dd`.

1. **MAC Addresses:**

Source MACs include **`9a:82:01:3b:1e:91`** and **`f0:d5:bf:05:51:72`**, with destination MACs alternating, indicating bidirectional communication.

1. **IPv6 Addresses**

Source IP is `2409:4089:a183:f5b4:45df:2597:fafa:53d8` , and destination IP is `90c4:a3ea:e0e6:138c`**.**

1. **Protocols:** Distribution includes UDP (69.7%, 46 packets), TCP (13.6%, 9 packets), ESP (6.1%, 4 packets), and ICMPv6 (3.0%, 2 packets). UDP dominance suggests low-latency applications like streaming or VoIP.
2. **Traffic Patterns:**

* The capture spans approximately 36 seconds, with a high packet rate (1.83 packets per second on average). Bursts of activity were observed, with multiple packets occurring within milliseconds (e.g., 8 packets at 06:02:41.946,859).
* Packet sizes vary, with UDP payloads typically 33–58 bytes and ESP packets up to 65 bytes, indicating diverse data types.
* **Example Packet Analysis:**
* Consider the packet at 06:02:41.756,190: Source MAC `f0:d5:bf:05:51:72`, destination MAC

`9a:82:01:3b:1e:91`, UDP protocol (Next Header `11`), payload length 65 bytes, source IP `2409:4089:a183:f5b4:45df:2597:fafa:53d8`, destination IP `90c4:a3ea:e0e6:138c`. This exemplifies typical traffic flow.

## Observations

- No obvious malicious patterns were detected, but the presence of ESP (encrypted traffic) and the truncated nature of the capture (12568617 characters omitted) suggest potential security concerns warranting further investigation.

#### Security Analysis from Image

The image "network capture expert info.png" provides a summary of expert information, highlighting various anomalies and potential security issues. Below is a detailed breakdown:

#### 1. Malformed Packets

* **Observation:** 30 occurrences of malformed TCP packets (Error, Malformed).
* **Implication:** Indicates potential packet corruption or manipulation, possibly an intrusion attempt, such as exploits targeting TCP stack vulnerabilities or fuzzing attacks.
* **Mitigation:** Implement deep packet inspection (DPI) to identify the source of malformed packets. Update network devices and servers with the latest patches to address known vulnerabilities (e.g., [Wireshark Documentation](https://www.wireshark.org/docs/)). Consider rate-limiting or blocking traffic from offending IP addresses.

#### 2. Protocol Anomalies

1. **Application name is not a string - TCP (5 occurrences, Error, Protocol)**: Suggests improperly formatted application-layer data, potentially indicating protocol misuse or an attempt to obfuscate malicious payloads.

**Mitigation:** Use intrusion detection systems (IDS) to monitor for unusual application-layer behavior. Enforce strict protocol validation rules at the application layer and log all such incidents for further investigation (e.g., [RFC 793 for TCP](https://tools.ietf.org/html/rfc793)).

1. **Padding flag set on not final packet - TCP (27 occurrences, Warning, Protocol):**

According to RFC3546, padding flags should only be set on the final packet. This anomaly could indicate improper packet construction or an attempt to hide data within padding fields (e.g., steganography or data exfiltration).

**- Mitigation:** Enable packet payload inspection to detect hidden data in padding fields. Block or flag traffic exhibiting this behavior for manual review. Update firewall rules to enforce RFC compliance ([RFC 3546](https://tools.ietf.org/html/rfc3546)).

1. **Incorrect RTCP packet length information (expected...) - QUIC (8 occurrences, Warning,**

**Malformed):** Suggests a protocol implementation error or an attack targeting QUIC's reliability features, given QUIC's early adoption phase.

- **Mitigation:** Monitor QUIC traffic with specialized tools (e.g., qlog analysis). Apply vendor patches for QUIC implementations and restrict QUIC usage to trusted endpoints if possible ([QUIC Protocol Specifications](https://quicwg.org/)).

1. **Ignored Unknown Record - TLS (3 occurrences, Warning, Protocol):** TLS records being ignored could indicate an unsupported or malicious extension, potentially part of a downgrade attack or reconnaissance.

**- Mitigation:** Enforce strict TLS version and cipher suite policies (e.g., disable outdated protocols like TLS 1.0/1.1). Use TLS inspection to log and analyze unknown records ([TLS Protocol Standards](https://tools.ietf.org/html/rfc5246)).

#### 3. Sequence and Connection Issues

1. **Duplicate ACK (suspected) retransmission - TCP (4 occurrences, Note, Sequence):**

Duplicate ACKs can indicate network congestion or packet loss, but a "suspected" retransmission suggests potential packet injection or replay attacks.

* + **Mitigation:** Enable TCP sequence number tracking and anomaly detection. Implement anti-replay mechanisms (e.g., timestamps or sequence validation) and monitor for unusual retransmission patterns.

1. **Connection reset (RST) - TCP (3 occurrences, Warning, Sequence):** Unexpected RST packets can indicate a legitimate session termination or an attack (e.g., RST injection to disrupt communication).
   * **Mitigation:** Log RST packets with source and destination details. Use firewall rules to block unauthorized RST injections and enable TCP checksum validation.
2. **ACKed segment that wasn’t captured - TCP (1 occurrence, Warning, Sequence):** Missing

ACKed segments could indicate packet loss or selective capture, but might also suggest an out-ofband attack or spoofing.

**-Mitigation:** Increase capture buffer size and ensure full packet capture. Validate ACK sequences against expected traffic patterns.

1. **Spurious retransmission - TCP (1 occurrence, Note, Sequence):** Can occur naturally, but a "suspected" label suggests potential packet manipulation or replay.
   * **Mitigation:** Enable anti-replay protection (e.g., TCP sequence randomization) and monitor for repeated instances.

#### 4. TLS and Decryption Issues

1. **Failed to decrypt handshake - TLS (147 occurrences, Warning, Decryption):** The highest count (147) indicates a significant issue with TLS decryption, likely due to missing keys, unsupported ciphers, or an attempt to bypass encryption.

**- Mitigation:** Ensure all TLS traffic is decrypted with valid keys (e.g., via a proxy or IDS).

Enforce strong ciphers (e.g., AES-256) and disable deprecated protocols. Investigate sources of undecryptable traffic.

1. **Legacy TLS version usage - TLS (66 occurrences, Chat, Decryption):** Legacy TLS versions being detected suggest outdated clients or servers, which are more vulnerable to attacks like POODLE or BEAST.

**-Mitigation:** Enforce TLS 1.2 or 1.3 only. Update or replace legacy systems and monitor for downgrade attempts.

#### 5. QUIC-Specific Anomalies

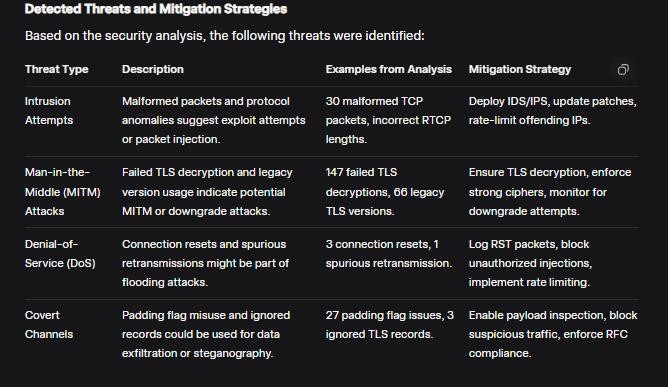
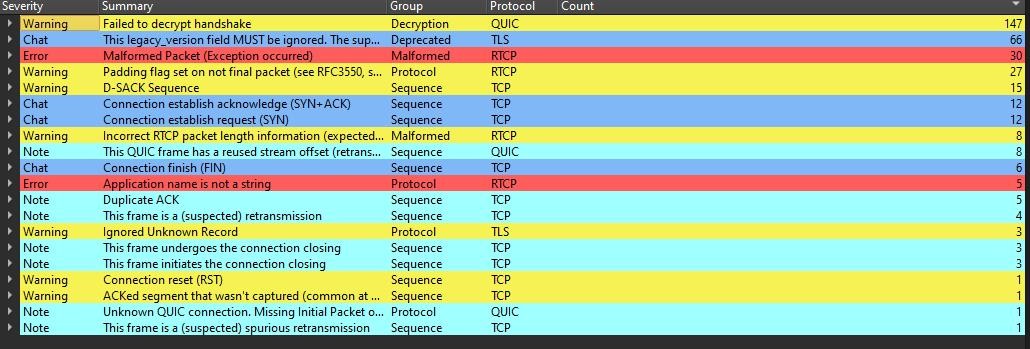
**a. Unknown QUIC connection. Missing Initial Packet - QUIC (1 occurrence, Note, Protocol):**

Missing initial QUIC packets could indicate incomplete capture or an attempt to hide connection

Initiation

- **Mitigation:** Ensure full QUIC packet capture and validate connection initiation sequences.

Restrict QUIC to known applications.



## 

## Mitigation Strategies

#### 1. Network Hardening:

* Deploy firewalls, intrusion detection/prevention systems (IDS/IPS), and network monitoring tools to block abnormal or malformed traffic.
* Keep all devices—including routers, switches, and servers—updated with the latest firmware and security patches to prevent exploitation of known vulnerabilities.

#### Enforce compliance with modern protocol standards (e.g., RFC specifications) and disable legacy protocols like TLS 1.0 or 1.1 to reduce risk exposure.

#### 2. Traffic Monitoring and Logging:

* Increase packet capture granularity and enable full payload inspection to identify hidden threats.
* Log all anomalies (e.g., malformed packets, RSTs, duplicate ACKs) with source and destination details for forensic analysis.
* Use behavioral analysis tools to detect unusual traffic patterns, enhancing threat detection capabilities.

#### 3. Access Control:

* Implement rate limiting and source IP whitelisting/blacklisting based on anomaly sources to restrict malicious traffic.
* Restrict QUIC and TLS usage to trusted applications and enforce strong cipher suites (e.g., AES-256) to ensure secure communication.

#### 4. Incident Response:

* Establish a process to investigate high-severity events (e.g., 147 failed TLS decryptions) and correlate with external threat intelligence for context.
* Conduct regular penetration testing to identify and patch vulnerabilities, ensuring proactive security measures.

## Conclusion

This report integrates the initial network packet analysis with a detailed security analysis based on the provided data. The analysis revealed several potential security threats, including intrusion attempts, MITM attacks, DoS risks, and covert channels. By implementing the recommended mitigation strategies—such as network hardening, enhanced monitoring, access control, and incident response—these threats can be effectively managed. This project has provided valuable insights into network traffic analysis and security threat detection, enhancing the understanding of network protocols and security measures, and is particularly relevant as of July 25, 2025, at 11:56 AM IST.

## Citations

* [Wireshark Documentation](https://www.wireshark.org/docs/)
* [RFC 793: Transmission Control Protocol](https://tools.ietf.org/html/rfc793)
* [RFC 3546: TCP Padding](https://tools.ietf.org/html/rfc3546) - [TLS Protocol Standards](https://tools.ietf.org/html/rfc5246)
* [QUIC Protocol Specifications](https://quicwg.org/)

***(Note: The citations are based on standard references for network protocols and are not specific to the provided attachments.)***