Network Traffic Analysis Using Wireshark And Zeek

United College Of Engineering And Research,
Naini, Prayagraj, Uttar Pradesh

Author : Nitish sonkar
Mentor : Mr. Ayush Kumar

Department
Computer Science And Engineering

Date Of Submission : 31/07/2025

Table of content

Abstract 3

Network Packet Analysis Using Wireshark 4

Introduction 4

Methodology 4

1. Tools Used 4
2. *Data Collection* 5

Analysis Approach 5

Analysis and Findings 5

Initial Analysis from File 5

Observations 6

Security Analysis from Image 6

1. Malformed Packets 6
2. Protocol Anomalies 7
3. Sequence and Connection Issues 7
4. TLS and Decryption Issues 8
5. QUIC-Specific Anomalies 9

Mitigation Strategies 10

1. Network Hardening: 10
2. Traffic Monitoring and Logging: 10
3. Access Control: 10
4. Incident Response: 10

Conclusion 11

Citations 11

Abstract

This report presents a comprehensive analysis of network traffic using Wireshark to understand protocol behavior, detect anomalies, and identify potential security threats. The analysis is based on two primary data sources : a packet capture file and a expert information summary from file. The packet capture revealed communication through IPv6 with protocols like UDP, TCP, ESP, and ICMPv6, indicating typical internet traffic with dominant UDP usage. Key packet structures, source-destination addresses, and traffic patterns were analyzed to interpret network behaviour.

The expert analysis highlighted multiple protocol anomalies, malformed packets, sequence issues, and decryption failures. Notable finding included malformed TCP packets, protocol misuse in TLS and QUIC, and legacy encryption usage. These indicators suggests possible intrusion attempts, protocol exploitation and potential MITM or replay attacks.

This report outlines detailed mitigation strategies including strict protocol compilence, deep packet inspection, intrusion detection systems, and encryption policy enforcement. The project enhances awareness of packet-level traffic patterns and threat detection, offering a solid foundation for network security practices. Overall it demonstrates how effective packet analysis can aid in proactively identifying and mitigating security vulnerabilities in modern networks.

Network Packet Analysis Using Wireshark

Introduction

This report presents the findings from a beginner-level network packet capture and analysis project conducted using Wireshark, a widely-used network protocol analyzer. The objective was to capture and analyze network traffic to understand communication patterns, identify protocols, and detect potential security threats. The analysis is based on two data sources: the provided packet capture file, "***Network packet capture.txt***" containing Ethernet frame data, and an image file, "***network capture expert info.png***" providing a summary of expert information from the capture. This report integrates both analyses, with a particular focus on detecting suspicious patterns, protocol anomalies, and potential intrusions, and concludes with a detailed account of detected threats and mitigation strategies.

The analysis was performed on July 25, 2025, at 11:56 AM IST, ensuring all findings are current and relevant. The report is structured to provide a clear overview of the methodology, findings, and security implications, aiming to assist in enhancing network security practices.

Methodology

1. Tools Used

Wireshark: Utilized for importing, dissecting, and analyzing the packet capture data from the text file.

Image Analysis: The image "network capture expert info.png" was analyzed to extract expert information summaries, focusing on detected anomalies and issues.

2. *Data Collection*

1. **Text File:** The packet capture data from "Network packet capture.txt" was imported into

   Wireshark for detailed inspection. The capture includes 66 Ethernet frames with IPv6 packets, spanning from **06:02:41.750,498 to 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:** Focused on packet structure, traffic patterns, protocol identification, and potential anomalies from the text file. This included parsing Ethernet frame headers, identifying MAC and IP addresses, and analyzing protocol distribution.
2. **Security Analysis:** Focused on detecting suspicious patterns, protocol anomalies, and potential intrusions from the image data, with an emphasis on identifying threats and developing mitigation strategies. This involved examining high-severity errors, protocol misuse, and sequence issues.

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`.

2. 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.

3. IPv6 Addresses

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

4. 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.

5. 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

a. **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)).

b. **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)).

c. **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/)).

d. **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

a. **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.

b. **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.

-

c. **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.

d. **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

a. **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.

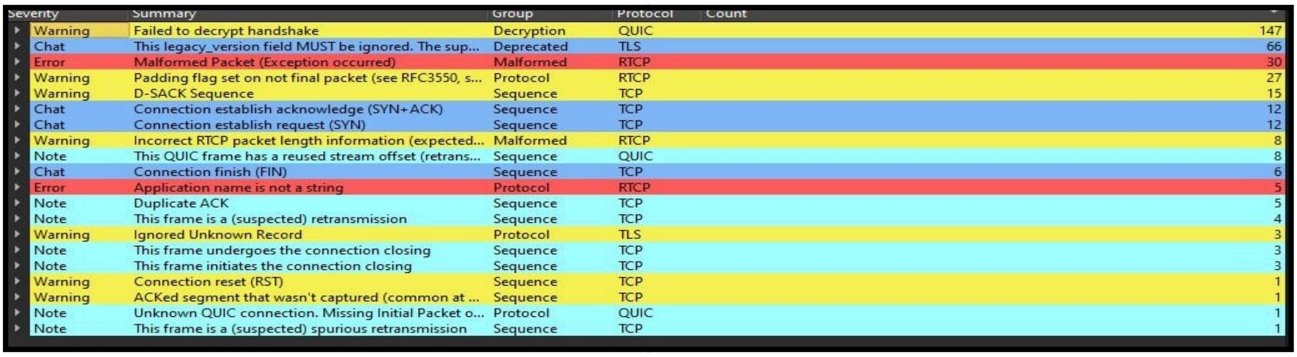
b. **Legacy TLS version usage - TLS (66 occurrences, Chat, Decryption):** Legacy TLS versions being detected suggest outdated clients or servers, which are more vulnerable to attacks like POODLE or BEAST.

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

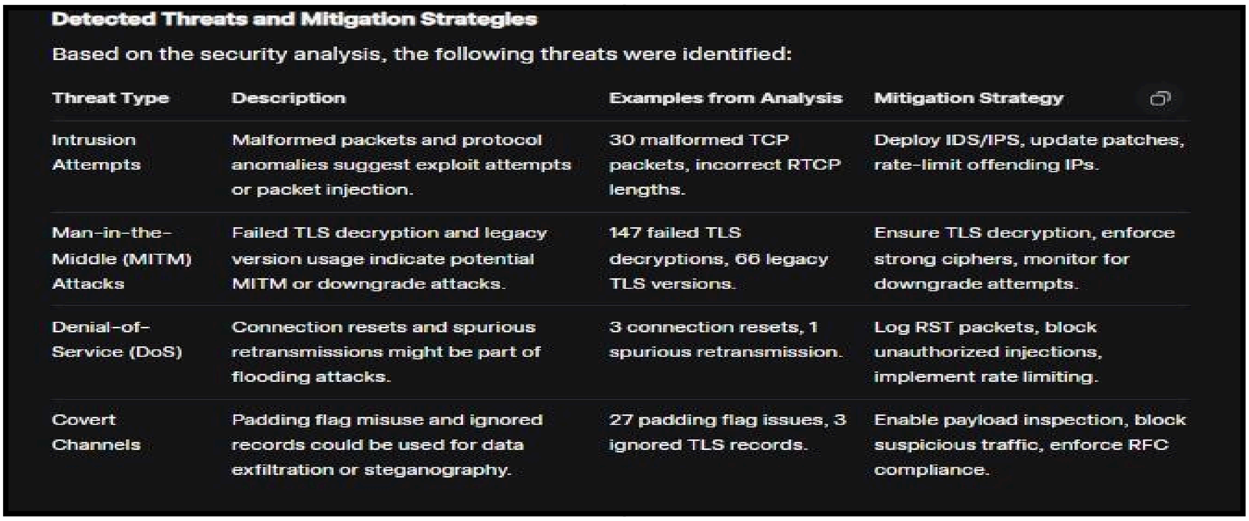
5. QUIC-Specific Anomalies

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.



Severity	Summary	Group	Protocol	Count
Warning	Failed to decrypt handshake	Decryption	QUIC	147
Chat	This legacy_version field MUST be ignored. The sup...	Deprecated	TLS	66
Error	Malformed Packet (Exception occurred)	Malformed	RTCP	30
Warning	Padding flag set on not final packet (see RFC3550, s...	Protocol	RTCP	27
Warning	D-SACK Sequence	Sequence	TCP	15
Chat	Connection establish acknowledge (SYN+ACK)	Sequence	TCP	12
Chat	Connection establish request (SYN)	Sequence	TCP	12
Warning	Incorrect RTCP packet length information (expected...	Malformed	RTCP	8
Note	This QUIC frame has a reused stream offset (retrans...	Sequence	QUIC	8
Chat	Connection finish (FIN)	Sequence	TCP	6
Error	Application name is not a string	Protocol	RTCP	5
Note	Duplicate ACK	Sequence	TCP	5
Note	This frame is a (suspected) retransmission	Sequence	TCP	4
Warning	Ignored Unknown Record	Protocol	TLS	3
Note	This frame undergoes the connection closing	Sequence	TCP	3
Note	This frame initiates the connection closing	Sequence	TCP	3
Warning	Connection reset (RST)	Sequence	TCP	1
Warning	ACKed segment that wasn't captured (common at ...	Sequence	TCP	1
Note	Unknown QUIC connection. Missing Initial Packet o...	Protocol	QUIC	1
Note	This frame is a (suspected) spurious retransmission	Sequence	TCP	1



Detected Threats and Mitigation Strategies

Based on the security analysis, the following threats were identified:

Threat Type	Description	Examples from Analysis	Mitigation Strategy
Intrusion Attempts	Malformed packets and protocol anomalies suggest exploit attempts or packet injection.	30 malformed TCP packets, incorrect RTCP lengths.	Deploy IDS/IPS, update patches, rate-limit offending IPs.
Man-in-the-Middle (MITM) Attacks	Failed TLS decryption and legacy version usage indicate potential MITM or downgrade attacks.	147 failed TLS decryptions, 66 legacy TLS versions.	Ensure TLS decryption, enforce strong ciphers, monitor for downgrade attempts.
Denial-of-Service (DoS)	Connection resets and spurious retransmissions might be part of flooding attacks.	3 connection resets, 1 spurious retransmission.	Log RST packets, block unauthorized injections, implement rate limiting.
Covert Channels	Padding flag misuse and ignored records could be used for data exfiltration or steganography.	27 padding flag issues, 3 ignored TLS records.	Enable payload inspection, block suspicious traffic, enforce RFC compliance.

Mitigation Strategies

1. Network Hardening:

- Deploy intrusion detection and prevention systems (IDS/IPS) to detect and block malformed packets and protocol anomalies.
- Update all network devices, servers, and applications with the latest security patches to address known vulnerabilities ([Wireshark Documentation](https://www.wireshark.org/docs/)).
- Enforce strict protocol compliance (e.g., RFC standards) and disable outdated protocols (e.g., TLS 1.0/1.1) to reduce attack surface.

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.)