Network Packet Capture and Analysis Using Wireshark

1. Introduction

Network packet capture and analysis is an essential technique used in network security, monitoring, and troubleshooting. By examining packets transmitted over a network, it is possible to identify protocols in use, understand packet structure, and detect suspicious or abnormal traffic patterns.
In this experiment, **Wireshark** is used to capture and analyze network packets related to **HTTP, DNS, and TCP communication**, along with identifying unusual or malformed packets.

---

2. Aim

To capture and analyze network packets using Wireshark in order to:

- Identify HTTP, DNS, and TCP protocols
- Analyze packet structure and TCP flags
- Detect suspicious or malformed network packets

---

3. Tools Used

- **Wireshark**
- **Operating System:** Windows / Linux
- **Network Interface:** Ethernet / Wi-Fi

---

4. Methodology / Procedure

1. Wireshark was installed and launched on the system.
2. The active network interface was selected.
3. Packet capture was started.
4. Network activity such as browsing websites and domain name resolution was generated.
5. Display filters were applied to isolate specific protocols.
6. Captured packets were analyzed in detail using Wireshark’s packet inspection feature.

7. Suspicious or unusual packets were identified based on abnormal flags or patterns.

---

5. Packet Capture and Analysis

5.1 HTTP Traffic Analysis

Display Filter Used:

http

Observation:
HTTP traffic was captured showing **GET** and **POST** requests. The packets revealed source and destination IP addresses, requested URLs, and HTTP headers.

Packet Structure:

- Ethernet Header
- IP Header
- TCP Header
- HTTP Application Layer Data

📌 *Screenshot:* HTTP packets captured in Wireshark

5.2 DNS Traffic Analysis

Display Filter Used:

dns

Observation:
DNS query and response packets were captured. Domain name resolution requests were visible, using UDP protocol on port 53.

Packet Structure:

- Transaction ID
- Query Name
- Query Type (A record)
- Response IP Address

📌 *Screenshot:* DNS query and response packets

5.3 TCP Handshake Analysis

Display Filter Used:

tcp

Observation:
The TCP **three-way handshake** was clearly observed:

1. SYN – Connection request
2. SYN-ACK – Acknowledgement from server
3. ACK – Confirmation from client

TCP Flags Identified:

- SYN
- ACK
- FIN

📌 *Screenshot:* TCP three-way handshake packets

6. Identification of Suspicious or Unusual Packets

Indicators of Suspicious Traffic:

- Multiple SYN packets without corresponding ACK responses
- Unusual TCP flag combinations
- High number of DNS requests in a short time interval

Possible Causes:

- Network scanning
- Misconfigured applications
- Potential malicious activity such as DoS attempts

📌 *Screenshot:* Unusual or malformed packets

7. Conclusion

In this experiment, Wireshark was successfully used to capture and analyze network packets. HTTP, DNS, and TCP handshake packets were examined to understand protocol behavior and packet structure. TCP flags were analyzed to observe connection establishment and termination. Additionally, suspicious packet patterns were identified, demonstrating the importance of packet analysis in network security and threat detection.

---

8. Deliverables

- Wireshark screenshots for HTTP, DNS, TCP, and suspicious packets
- Short packet analysis report