

## ***Task 5: Capture and Analyze Network Traffic Using Wireshark***

***Author: Anshikaa Pahilajani***

***Cyber Security Internship — Elevate Labs***

***Date: 29 September 2025***

## Objective

Capture live network packets using Wireshark and identify basic protocols and traffic types to gain hands-on packet analysis skills and protocol awareness.

## Tools Used

- **Wireshark (latest version)** — for capturing and analyzing packets
- **TraceWrangler Beta 0.6.9 (64-bit)** — for anonymizing sensitive information in the packet capture
- **Windows Command Prompt** — for ping tests

## Procedure / Steps

### Step 1: Start Packet Capture

1. Open **Wireshark**
2. Select the active network interface (Wi-Fi/Ethernet)
3. Click **Start Capture**

### Step 2: Generate Traffic

- Open Command Prompt
- Run commands like:
  - ping -4 google.com
- Open a website in the browser to generate HTTP/TLS/DNS traffic

### Step 3: Stop Capture

- After about 1 minute of traffic, click **Stop Capture** in Wireshark

### Step 4: Identify Your IP

- Wireshark → **Statistics** → **Endpoints** → **IPv4 tab**
- Note your IP for filtering (10.0.0.2 - anonymized IP representing my machine)

## Step 5: Apply Filters (one at a time)

- Show only your packets:
- ip.addr == 10.0.0.2
- Filter by protocol:
- icmp && ip.addr == 10.0.0.2
- tcp && ip.addr == 10.0.0.2
- udp && ip.addr == 10.0.0.2
- dns && ip.addr == 10.0.0.2
- tls && ip.addr == 10.0.0.2

## Step 6: Take Screenshots

### 1. Outgoing Packets

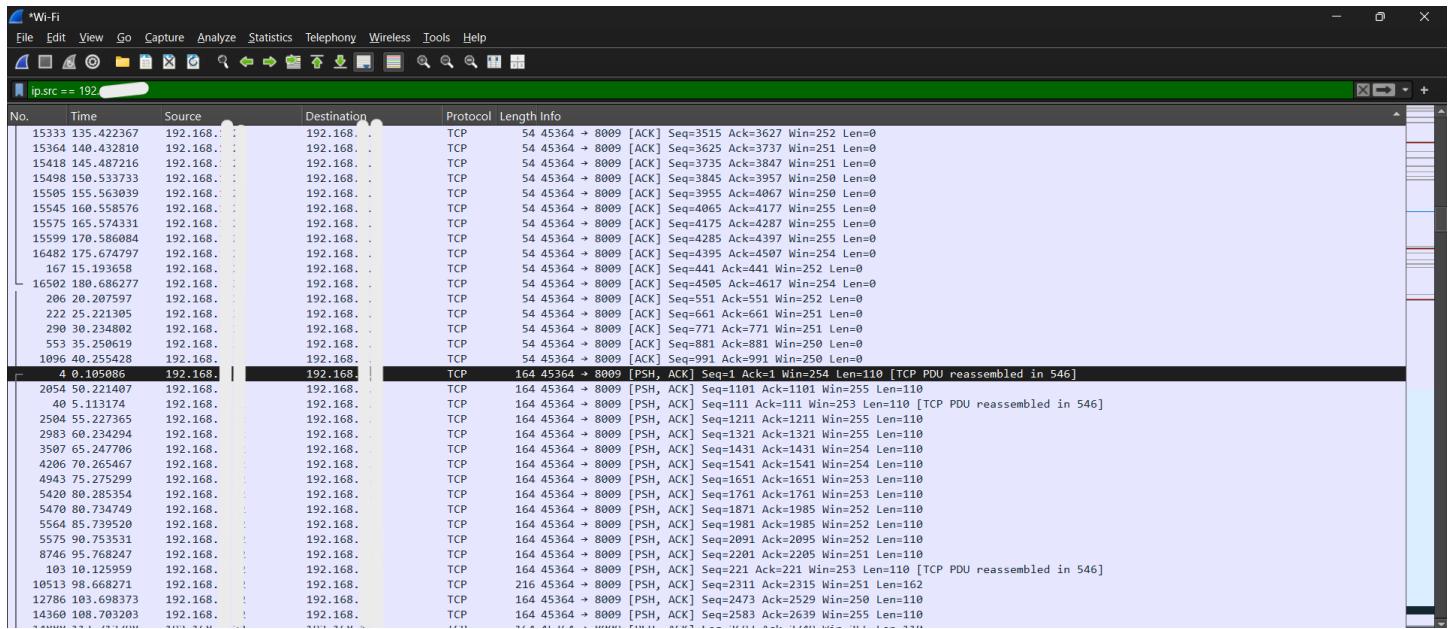


Figure 1 - Outgoing network traffic from the host

**Description:** Shows packets initiated by the host including TCP SYN requests, ICMP ping requests, and UDP packets. Source and destination IPs are anonymized to protect sensitive information.

## 2. Incoming Packets

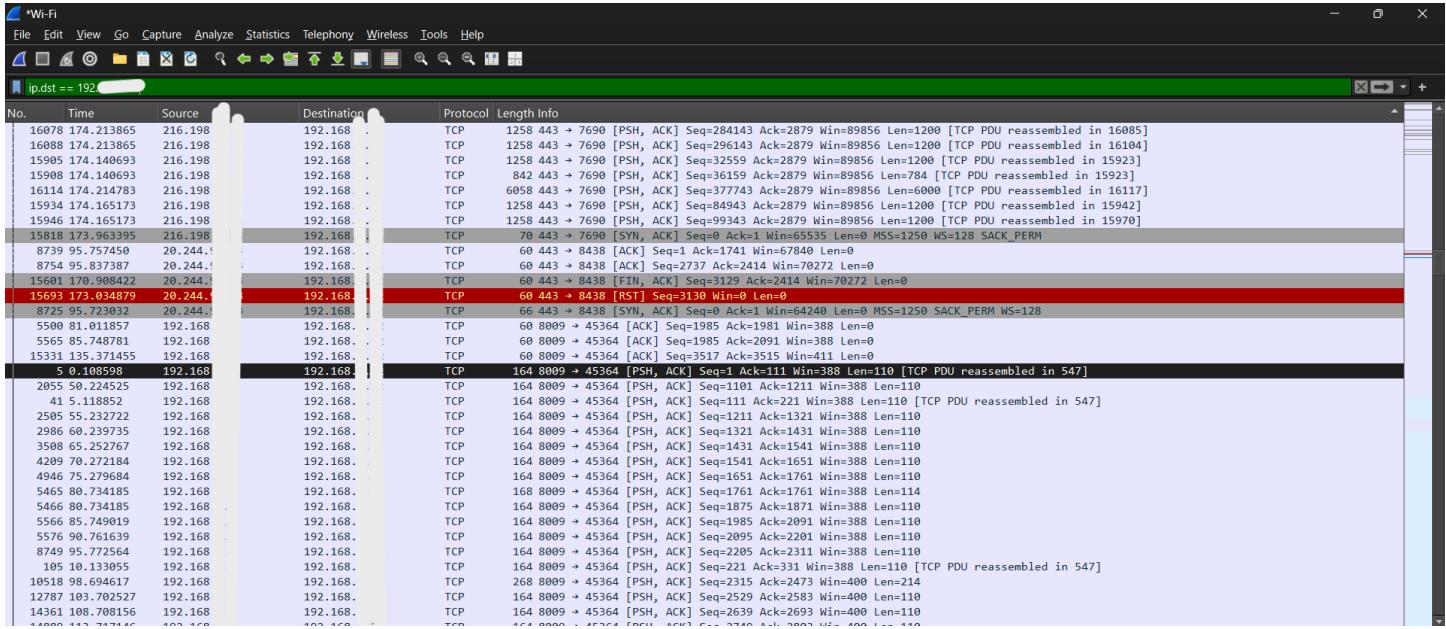


Figure 2- Incoming traffic received by the host

**Description:** Shows responses from other devices, including TCP ACKs and ICMP replies. Source IPs from external hosts are anonymized while the destination host is partially shown as 192.168.x.x.

## 3. ICMP Packets

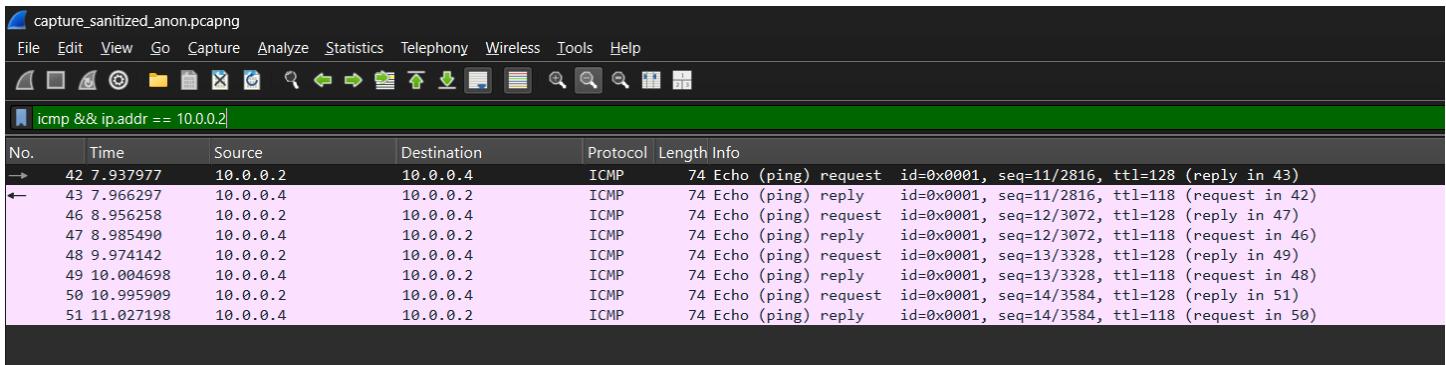


Figure 3 - ICMP ping requests and replies

**Description:** Demonstrates how the host tests connectivity using ICMP packets. Source and destination IPs have been anonymized for privacy.

## 4. TCP Packets

The screenshot shows a Wireshark capture window with the following details:

- File menu:** File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help.
- Toolbar:** Standard icons for opening files, saving, zooming, and filtering.
- Search bar:** tcp && ip.addr == 10.0.0.2 |
- Table Headers:** No., Time, Source, Destination, Protocol, Length Info.
- Data Rows:** Numerous TCP packets are listed, showing the flow of data from source 10.0.0.1 to destination 10.0.0.2. Key flags include SYN, ACK, PSH, and FIN. Sequence numbers (Seq) and Acknowledgment numbers (Ack) are visible, along with window sizes (Win) and lengths (Len).

Figure 4 - TCP traffic showing connection establishment

**Description:** Highlights TCP flags like SYN, ACK, and PSH, showing reliable connection setup and data transfer. IP addresses and ports are anonymized.

## 5. TLS Packets

The screenshot shows a Wireshark capture window with the following details:

- File menu:** File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help.
- Toolbar:** Standard icons for opening files, saving, zooming, and filtering.
- Search bar:** tls && ip.addr == 192.168. [REDACTED]
- Table Headers:** No., Time, Source, Destination, Protocol, Length Info.
- Data Rows:** Two TLSv1.2 sessions are shown. Session 156 (Host to 18.211. [REDACTED]) has a length of 124 bytes and is labeled "Application Data". Session 160 (18.211. [REDACTED] to Host) has a length of 121 bytes and is also labeled "Application Data". Both sessions are using TLSv1.2.

Figure 5 - TLS-encrypted traffic from the host

**Description:** Demonstrates secure communication using TLS. IPs are anonymized and payloads are hidden to maintain confidentiality.

## 6. UDP Packets

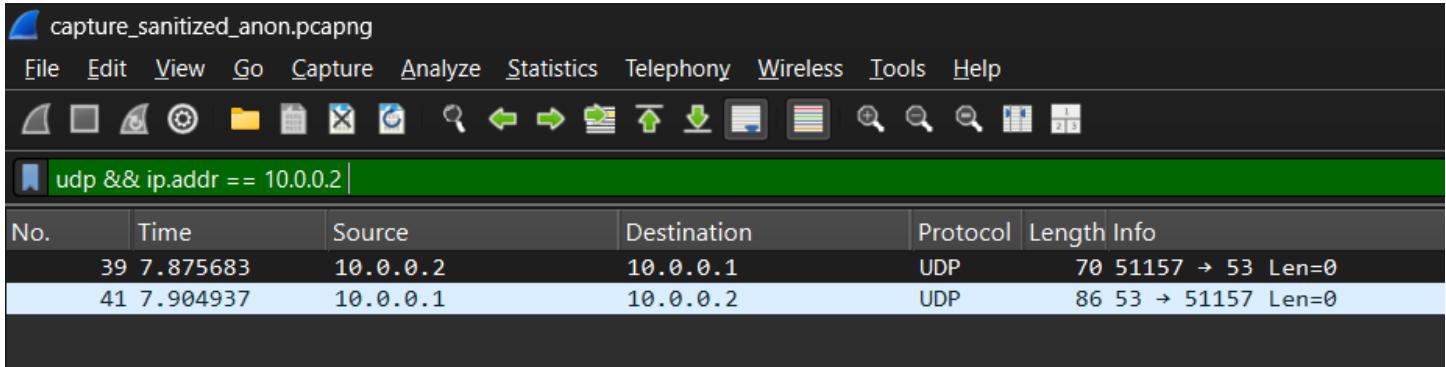


Figure 6 - UDP traffic from the host

**Description:** Shows connectionless communication using UDP. IPs and port numbers are anonymized while packet lengths and protocol info remain visible.

## 7. Ping Command Output

```
C:\Users\anshi>ping -4 google.com

Pinging google.com [142.251....] with 32 bytes of data:
Reply from 142.251.    : bytes=32 time=28ms TTL=118
Reply from 142.251.    : bytes=32 time=29ms TTL=118
Reply from 142.251.    : bytes=32 time=30ms TTL=118
Reply from 142.251.    : bytes=32 time=31ms TTL=118

Ping statistics for 142.251.    :
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 28ms, Maximum = 31ms, Average = 29ms
```

Figure 7 - Host sending ICMP ping requests to a remote server

**Description:** Shows connectivity testing using the ping command in the terminal.

## Observations / Protocol Analysis

Protocol	Description	Example / Packet Count
ICMP	Used for ping requests and replies to test connectivity	Echo request/reply packets visible in capture
TCP	Connection-oriented transport protocol; ensures reliable delivery	SYN, ACK, FIN packets observed for DNS/TLS traffic
UDP	Connectionless transport protocol	DNS query responses observed
DNS	Domain Name System; resolves hostnames to IPs	Queries to ssl.gstatic.com and google.com
TLS	Encryption protocol for secure communications	TLSv1.2 packets seen for HTTPS connections

### Notes:

- Packet counts are approximate based on filtered capture.
- Anonymization replaced all real IPs with 10.0.0.X for safety.

## Key Learnings

1. Wireshark allows **live packet capture** and filtering by protocol, IP, and port.
2. Understanding packet headers helps identify **protocol types and flow**.
3. Anonymization tools like TraceWrangler ensure sensitive info is **protected before sharing**.
4. Hands-on packet analysis improves **network troubleshooting skills**.