**Network Traffic Diagnosis Using Wireshark**

**Objective:** The objective of this project is to use Wireshark to capture network traffic, analyze the captured data, and identify patterns, anomalies, or security threats.

**Step 1: Setting Up Wireshark**

Before analyzing network traffic, I had to install **Wireshark**, a tool that allows us to see all the data moving in and out of a computer or network.

**Actions Taken:**

1. **Downloading Wireshark**
   * I visited the official **Wireshark website** (<https://www.wireshark.org/>) and downloaded the software.
   * Installed it on my computer by following the on-screen instructions.
2. **Launching Wireshark**
   * After installation, I opened Wireshark.
   * It displayed a list of network interfaces (Wi-Fi, Ethernet, etc.), which are the connections my computer uses to communicate over the internet.
3. **Choosing the Right Network Interface**
   * Since I was using Wi-Fi, I selected my **Wi-Fi network** for monitoring.
   * For a wired connection, I would have selected **Ethernet** instead.
4. **Starting Packet Capture**
   * I clicked the **Start Capturing Packets** button to begin monitoring the network.
   * At this point, Wireshark started collecting real-time data of all the internet activity happening on my computer.

**Step 2: Capturing Network Traffic**

To analyze meaningful data, I needed to generate network activity.

**Actions Taken:**

1. **Browsing a Website**
   * I opened my web browser and visited a website (e.g., <https://www.example.com>).
   * This allowed me to capture **HTTP and HTTPS traffic**, which is the data exchanged between my computer and the website.
2. **Sending an Email**
   * I sent a test email to capture **email-related network traffic**.
   * If the email was unencrypted, Wireshark would allow me to see its details.
3. **Accessing a Shared Resource**
   * I accessed a shared file on another computer in the network to capture **file-sharing traffic**.
4. **Stopping the Capture**
   * After collecting enough data, I clicked the **Stop Capturing Packets** button in Wireshark.
   * Now, I had a list of all the network packets (small pieces of data being transferred) during my activities.

**Step 3: Filtering Captured Traffic**

The captured data contained thousands of packets, which was too much information to analyze manually. So, I used **filters** to focus on specific types of data.

**Actions Taken:**

1. **Filtering for Web Traffic (HTTP/HTTPS)**
   * I typed **http** in the filter bar and pressed **Enter**.
   * This showed only web-related data, helping me see how my browser communicated with websites.
2. **Filtering by IP Address**
   * I used **ip.addr == 192.168.1.1** to view only packets related to a specific device on my network.
   * This was useful to check if an unknown device was communicating with my computer.
3. **Filtering by Port Number**
   * I applied **tcp.port == 80** to focus on HTTP traffic (unencrypted web traffic).
   * If I wanted to see secure traffic (HTTPS), I could use **tcp.port == 443**.

By applying these filters, I was able to focus only on relevant data instead of looking through thousands of unrelated packets.

**Step 4: Analyzing Packet Details**

After filtering, I started examining individual packets to understand how data was flowing.

**Actions Taken:**

1. **Checking Source and Destination IP Addresses**
   * Each packet had a **source IP** (where it came from) and a **destination IP** (where it was going).
   * This helped me track which websites or servers my computer was communicating with.
2. **Looking at Protocol Details**
   * I checked whether the packets contained **HTTP, DNS, or TCP data**.
   * For example, when I visited a website, I saw **DNS queries** (requests to convert a website name into an IP address).
3. **Identifying Anomalies**
   * I looked for any **unusual patterns**, such as:
     + **Repeated failed connections** (which could indicate a hacking attempt).
     + **Unexpected IP addresses** (which might show someone trying to access my network).

**Step 5: Identifying Security Concerns**

Finally, I checked for potential security risks that could indicate **data leaks** or **cyberattacks**.

**Actions Taken:**

1. **Detecting Unencrypted Data**
   * I noticed some **HTTP traffic**, meaning data was being sent in an unprotected format.
   * This is dangerous because **hackers can intercept and read it**.
2. **Unexpected IP Addresses**
   * I found some packets communicating with an unknown IP address.
   * This could mean my computer was sending data to an unauthorized server, possibly a malware infection.
3. **High Traffic Volume**
   * I noticed a sudden increase in data transfer, which could indicate a **Denial of Service (DoS) attack** or malware sending out data.

To mitigate these risks, I recommended:

* **Using only HTTPS websites** for secure browsing.
* **Blocking suspicious IP addresses** on my router.
* **Running an antivirus scan** to check for malware.