**Analyse Protocols:**

**Step 1: Start Wireshark**

Before we start capturing traffic, we need to ensure Wireshark is installed and running:

* Open your Ubuntu terminal by searching for "Terminal" in the applications menu or pressing **Ctrl + Alt + T**.
* Type the following command to start Wireshark:

Copy code

**sudo wireshark**

* + This command starts Wireshark with administrative privileges, allowing you to capture network traffic.

**Step 2: Capture Various Protocols' Traffic**

Now, let's capture traffic for different protocols:

**Capture HTTPS Traffic (Secure Web Traffic):**

HTTPS traffic is encrypted web traffic used for secure communication, typically on port 443.

Go

Copy code

**sudo tcpdump -i <interface> -w https\_traffic.pcap 'port 443'**

* Replace **<interface>** with the name of your network interface (e.g., eth0).
* This command captures HTTPS traffic on port 443 and saves it to a file named **https\_traffic.pcap**.

**Capture IPsec Traffic (Secure Network Communication):**

IPsec is a protocol suite for secure internet protocol (IP) communication.

goCopy code

**sudo tcpdump -i <interface> -w ipsec\_traffic.pcap 'ip proto 50 or ip proto 51'**

* Replace **<interface>** with the name of your network interface.
* This command captures IPsec traffic and saves it to a file named **ipsec\_traffic.pcap**.

**Capture SSH Traffic (Secure Shell Communication):**

SSH is a cryptographic network protocol used for secure data communication.

Copy code

**sudo tcpdump -i <interface> -w ssh\_traffic.pcap 'port 22'**

* Replace **<interface>** with the name of your network interface.
* This command captures SSH traffic on port 22 and saves it to a file named **ssh\_traffic.pcap**.

**Capture WPA/WPA2 Traffic (Wi-Fi Protected Access):**

WPA/WPA2 is a security protocol used to secure wireless networks.

Copy code

**sudo tcpdump -i <wireless\_interface> -w wpa\_traffic.pcap 'type mgt subtype assoc-req or type mgt subtype assoc-resp'**

* Replace **<wireless\_interface>** with the name of your wireless network interface (e.g., wlan0).
* This command captures WPA/WPA2 traffic and saves it to a file named **wpa\_traffic.pcap**.

**Capture DNSSEC Traffic (Secure DNS Communication):**

DNSSEC is a set of extensions to DNS that provides security to DNS responses.

Copy code

**sudo tcpdump -i <interface> -w dnssec\_traffic.pcap 'port 53'**

* \Replace **<interface>** with the name of your network interface.
* This command captures DNSSEC traffic on port 53 and saves it to a file named **dnssec\_traffic.pcap**.

**Capture OAuth Traffic (OAuth Authentication Communication):**

OAuth is an open standard for access delegation, commonly used for authorization.

Copy code

**sudo tcpdump -i <interface> -w oauth\_traffic.pcap 'port 443 and (tcp[((tcp[12] & 0xf0) >> 2):1] = 0x16 or tcp[((tcp[12] & 0xf0) >> 2):1] = 0x80)'**

* Replace **<interface>** with the name of your network interface.
* This command captures OAuth traffic on port 443 and saves it to a file named **oauth\_traffic.pcap**.

**Step 3: Analyze Captured Traffic in Wireshark**

Once you've captured traffic for each protocol, you can analyze it using Wireshark:

* Open Wireshark if it's not already open.
* Go to "File" > "Open" and navigate to the location where you saved the captured files.
* Select the file you want to analyze (e.g., **https\_traffic.pcap**) and click "Open".
* Wireshark will display the captured packets for analysis. You can explore different packets to understand the security mechanisms embedded in each protocol.