**"Wireshark Essentials: Learn, Capture, Analyze"**

**What is Wireshark:**

"Wireshark is used to monitor and analyze how our network communicates with other networks. By capturing the data packets that flow through the network, we can understand what data is being sent, how it's sent, and what happens to it. This helps us learn what’s happening behind the scenes of the internet and applications."

**Installation on Kali Linux**

Wireshark is free and open-source software. On Kali Linux, it usually comes pre-installed by default, so there's no need to install it manually. You can check its presence by running:

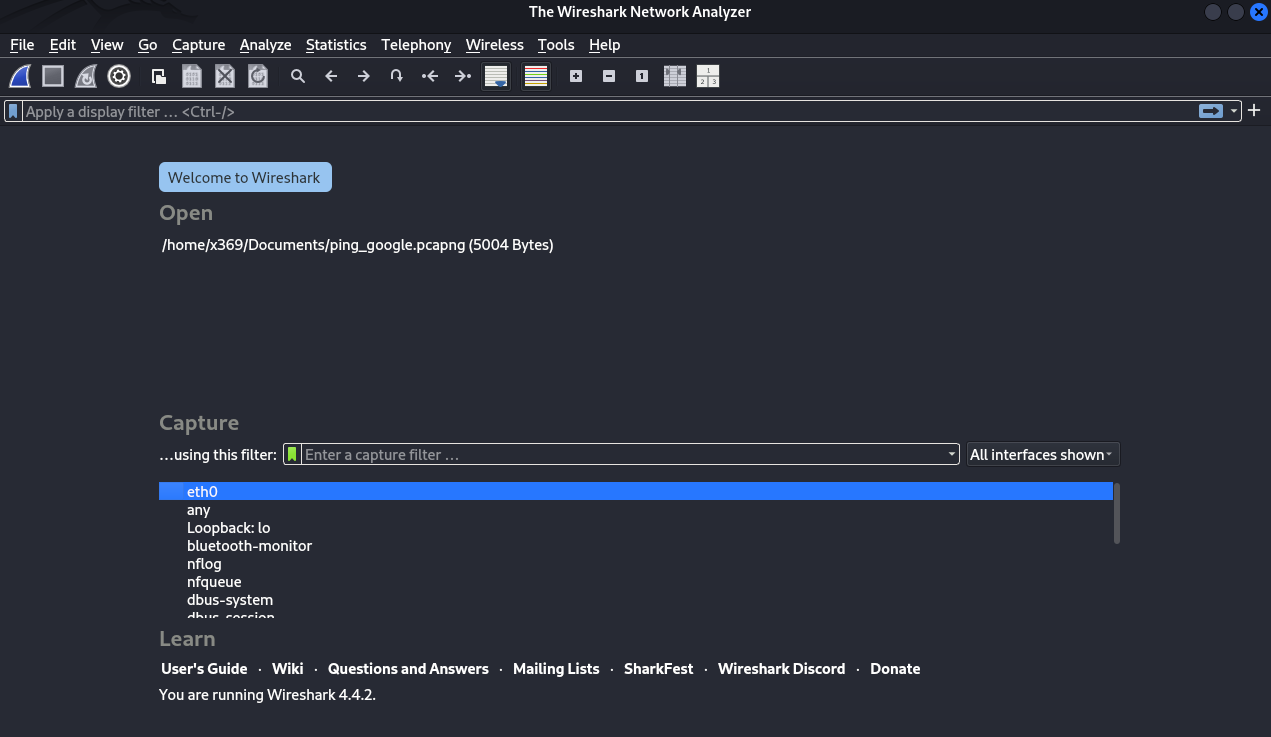
>> wireshark -- (you can use the command in Kali Linux to open the wireshark)

If for any reason it’s not installed, you can install it using:

>> sudo apt update -- (use the command to update kali machine)

>> sudo apt install wireshark -- (use this command to install wireshark)

Wireshark User Interface:

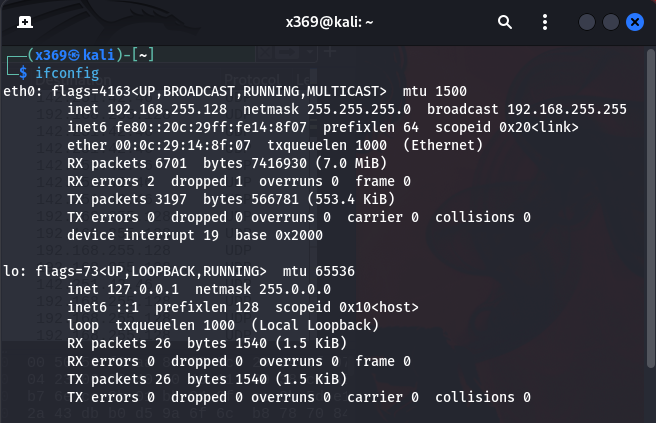


Select the network you are using and it starts capturing the data packets

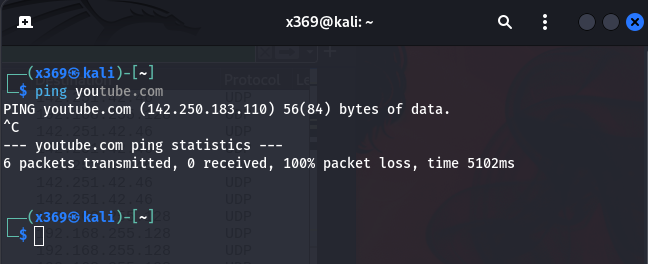
First, check both your IP address and the target IP address

Capturing Network Traffic:

Check the IP address of your system



**Check the IP address of your target address**



**Filters in Wireshark:**

Wireshark filters help you focus on specific traffic by filtering packets based on:

**IP Address Filters**  
Used to filter traffic by source, destination, or any matching IP address

>> ip.addr == 192.168.255.128

**Protocol Filters**

Used to filter packets by network protocols such as TCP, UDP, ICMP, HTTP, HTTPS, DNS, ARP, and others.

**Port Filters**

Used to capture or analyze packets based on specific TCP or UDP port numbers.

>> tcp.port == 80

>> tcp. port == 443

**MAC Address Filters**

Used to isolate traffic from specific physical network interfaces.

**Domain Name Filters**

Used to track DNS queries or TLS handshake extensions related to specific domains.

>> dns.qry.name == "youtube.com"

>> tls.handshake.extensions\_server\_name contains "youtube"

**Application Layer Filters**

Used to filter packets by application-level attributes like HTTP methods, headers, or cookies.

**Packet Content Filters**

Used to search for specific strings or binary values within the packet payload.

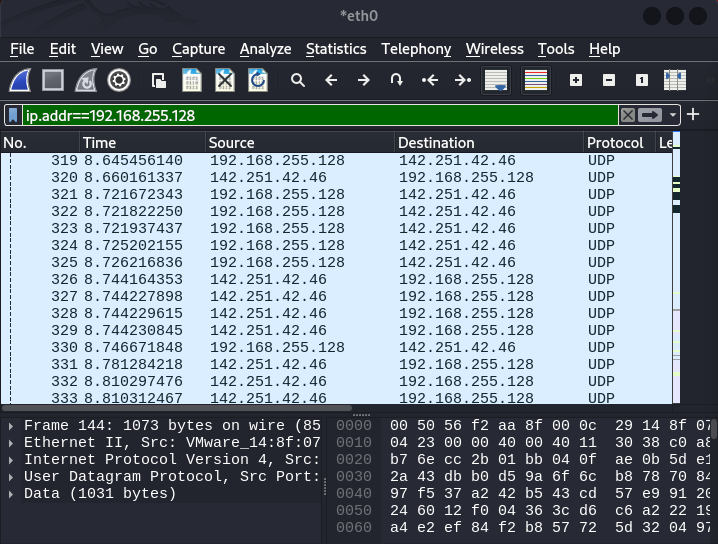
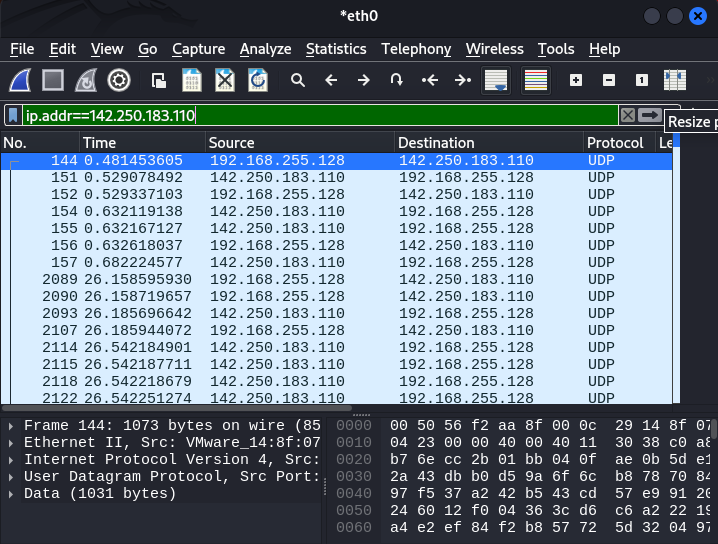
**Combined Logic Filters**

Used to apply multiple filters together using logical operators like AND (&&), OR (||), and NOT (!).

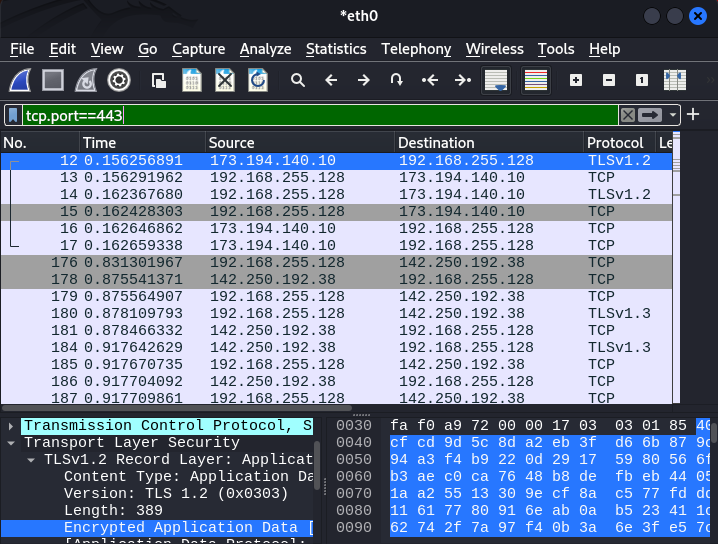
>> (ip.addr == 192.168.255.128) && (ip.addr == 142.250.196.78)

**Practical**

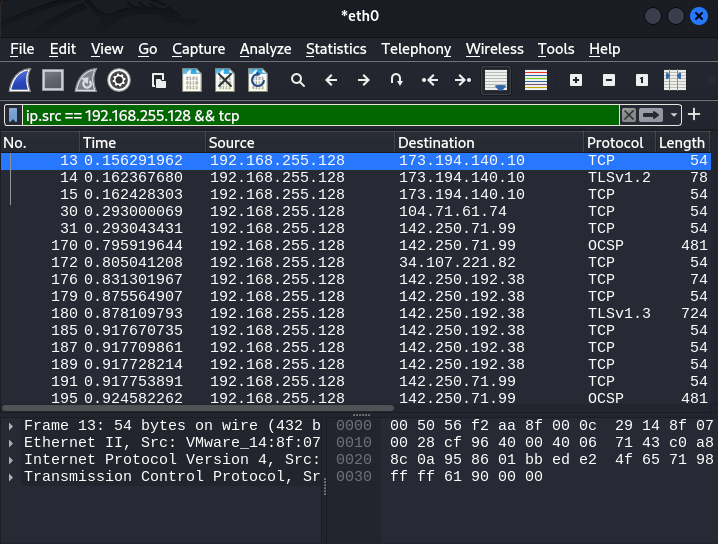
Filter by your System IP

Filter by Target Address 

**Filter by port**

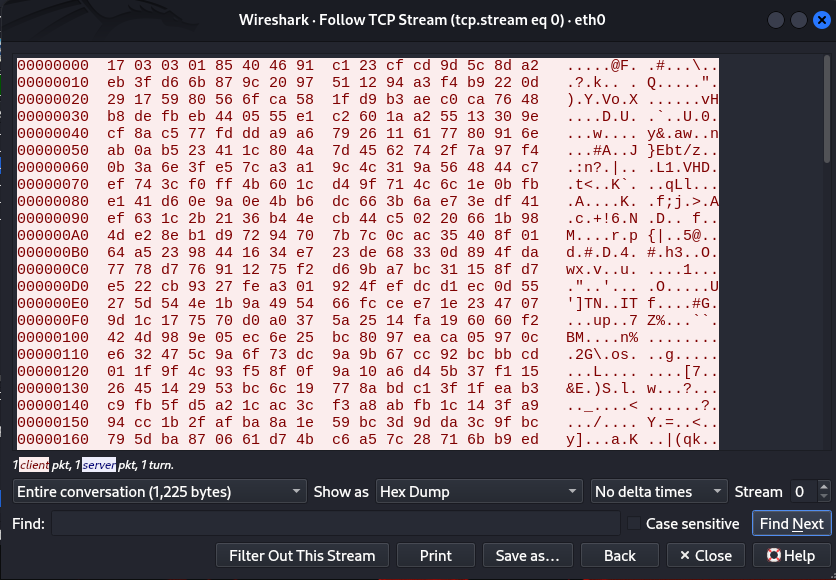


**Combined Filter**



**Analyzing Packets:**

To analyze a data packet after applying filters, select the packet, right-click on it, and choose "Follow → TCP Stream". This opens the raw communication between client and server. If the data is not encrypted (e.g., plain HTTP), you can read and analyze the actual content directly



We have different options to choose to see data type like raw , hex and more

**Wireshark Use Cases**

Network Troubleshooting

Identify slow network performance, dropped packets, or connection failures.

Security Analysis

Detect suspicious traffic, malware activity, port scans, and intrusion attempts.

Protocol Analysis

Understand how protocols like TCP, UDP, DNS, HTTP, TLS, etc., work in real time.

Traffic Monitoring

Monitor live traffic to or from specific IP addresses, ports, or applications.

Packet Inspection

View full packet details including headers, payloads, flags, and timing.

Application Debugging

Debug how applications communicate over the network (especially web, API calls).

DNS Resolution Tracking

Analyze DNS requests and responses to diagnose resolution issues.

Bandwidth Analysis

Identify heavy traffic sources or bandwidth-consuming applications.

Capture File Export

Save .pcap files for offline analysis or sharing with others.

Training and Education

Learn networking concepts, protocol structures, and cybersecurity fundamentals.