



# Wireshark

*Submitted by Aleena Anna Plessey*

Reg no: 21BCE5767

Course Program: B.Tech

Course code: BCSE308L

Course Title: Computer Networks

Batch:2021-25

*Submitted to: Dr. Swaminathan A*

*Submitted on 09/07/2023*

## Problem Statement:

12.1. Capture the packets using Wireshark and depict the TCP connection establishment and termination process.

12.2. Filter the TCP packets that contain the request to terminate.

12.3. Depict the flowgraph and the I/O graph of TCP, UDP, ICMP, ARP, and TLS.

## Aim and Objective:

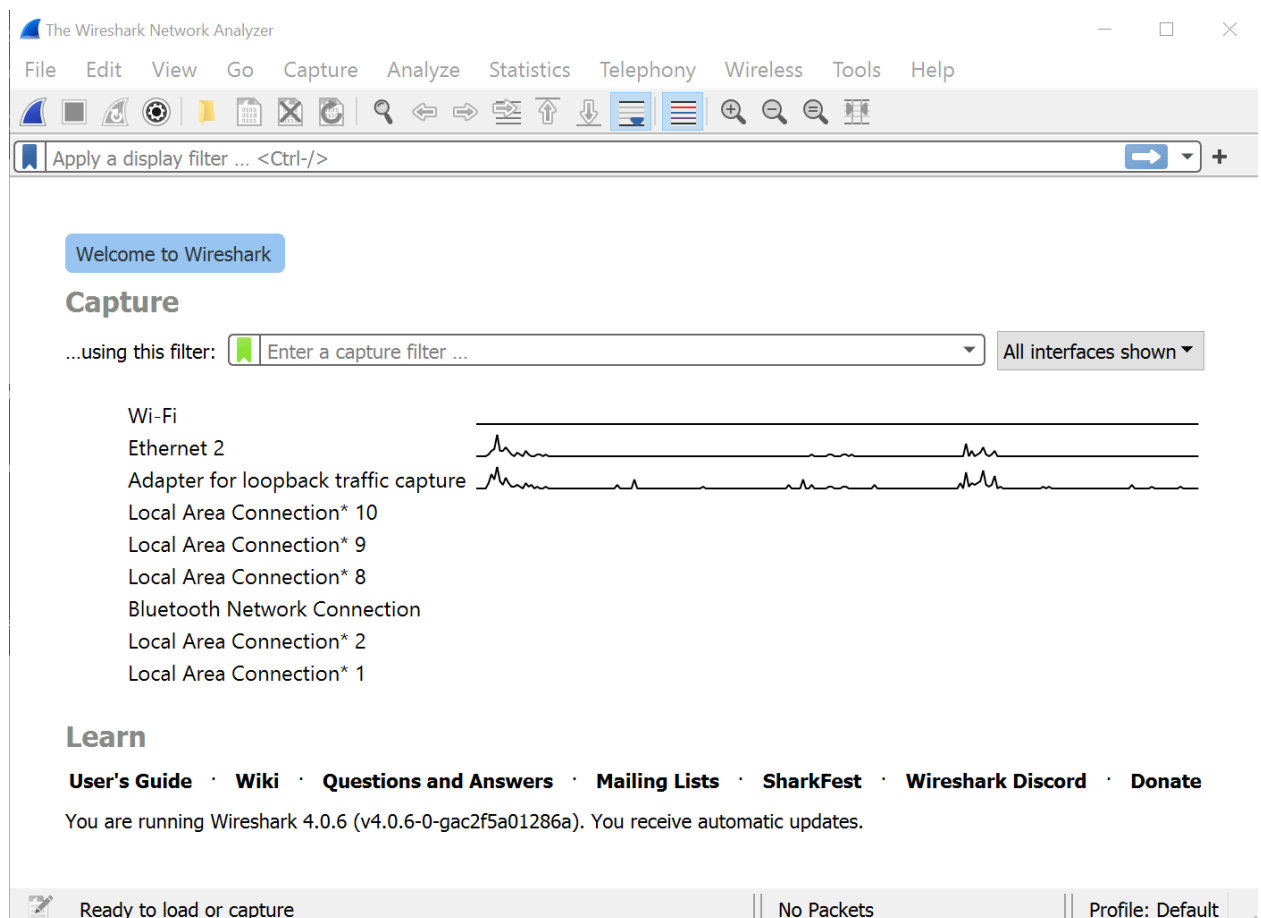
To understand the main use case of Wireshark and to understand its features and learn to capture packets to view the different properties of TCP packets.

## Solutions:

### 12.1. Capture the packets using Wireshark and depict the TCP connection establishment and termination process

#### Procedure:

1) Open Wireshark whilst connected to a network and use the appropriate interface(Ethernet/WiFi)



2) Press start and capture the packets for a few 5 to 10 minutes.

3) Press stop once enough packets have been gathered.

4) In the top right corner search for tcp only and filter out the packets.

The image shows the Wireshark interface with the 'Capture Filters' dialog box open. The dialog box has two columns: 'Filter Name' and 'Filter Expression'. The 'TCP only' filter is selected, with the expression 'tcp'. Other filters listed include 'No Broadcast and no Multicast', 'No ARP', 'IPv4 only', 'IPv4 address 192.0.2.1', 'IPv6 only', 'IPv6 address 2001:db8::1', 'UDP only', 'Non-DNS', 'TCP or UDP port 80 (HTTP)', 'HTTP TCP port (80)', 'No ARP and no DNS', and 'Non-HTTP and non-SMTP to/from www.wireshark.org'. The background shows a packet list with columns: No., Time, Source, Destination, Protocol, Length, Info. The first packet is selected, showing details for Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bit).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	2401:4900:2314:5e8b::	2401:4900:4007:82b::	TCP	74	63361
2	0.000000					
3	0.000000					
4	0.330000					
5	0.330000					
6	0.330000					
7	0.330000					
8	0.330000					
9	0.330000					
10	0.330000					
11	0.340000					
12	0.340000					
13	0.360000					
14	0.370000					
15	0.370000					
16	0.400000					

Wireshark - Capture Filters

Filter Name	Filter Expression
No Broadcast and no Multicast	not broadcast and not multicast
No ARP	not arp
IPv4 only	ip
IPv4 address 192.0.2.1	host 192.0.2.1
IPv6 only	ip6
IPv6 address 2001:db8::1	host 2001:db8::1
<b>TCP only</b>	<b>tcp</b>
UDP only	udp
Non-DNS	not port 53
TCP or UDP port 80 (HTTP)	port 80
HTTP TCP port (80)	tcp port http
No ARP and no DNS	not arp and port not 53
Non-HTTP and non-SMTP to/from www.wireshark.org	not port 80 and not port 25 and h

OK Cancel

Output:

The image shows the Wireshark interface with the packet list and packet details. The packet list shows a list of packets with columns: No., Time, Source, Destination, Protocol, Length, Info. The first packet is selected, showing details for Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bit). The packet details show the Ethernet II, Internet Protocol Version 6, and Transmission Control Protocol (TCP) layers. The TCP layer shows the source port 63361 and destination port 80. The packet list shows a list of packets with columns: No., Time, Source, Destination, Protocol, Length, Info. The first packet is selected, showing details for Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bit).

No.	Time	Source	Destination	Protocol	Length	Info
28	2.014861	192.168.71.215	152.195.38.76	TCP	54	50136 → 80 [FIN, ACK] Seq=1 Ack=1 Win=510 Len=0
29	2.014906	192.168.71.215	152.195.38.76	TCP	54	50345 → 80 [FIN, ACK] Seq=1 Ack=1 Win=512 Len=0
30	2.087038	152.195.38.76	192.168.71.215	TCP	54	80 → 50346 [FIN, ACK] Seq=1 Ack=2 Win=137 Len=0
31	2.087038	152.195.38.76	192.168.71.215	TCP	54	80 → 50136 [FIN, ACK] Seq=1 Ack=2 Win=152 Len=0
32	2.087109	192.168.71.215	152.195.38.76	TCP	54	50346 → 80 [ACK] Seq=2 Ack=2 Win=512 Len=0
33	2.087220	192.168.71.215	152.195.38.76	TCP	54	50136 → 80 [ACK] Seq=2 Ack=2 Win=510 Len=0

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bit)

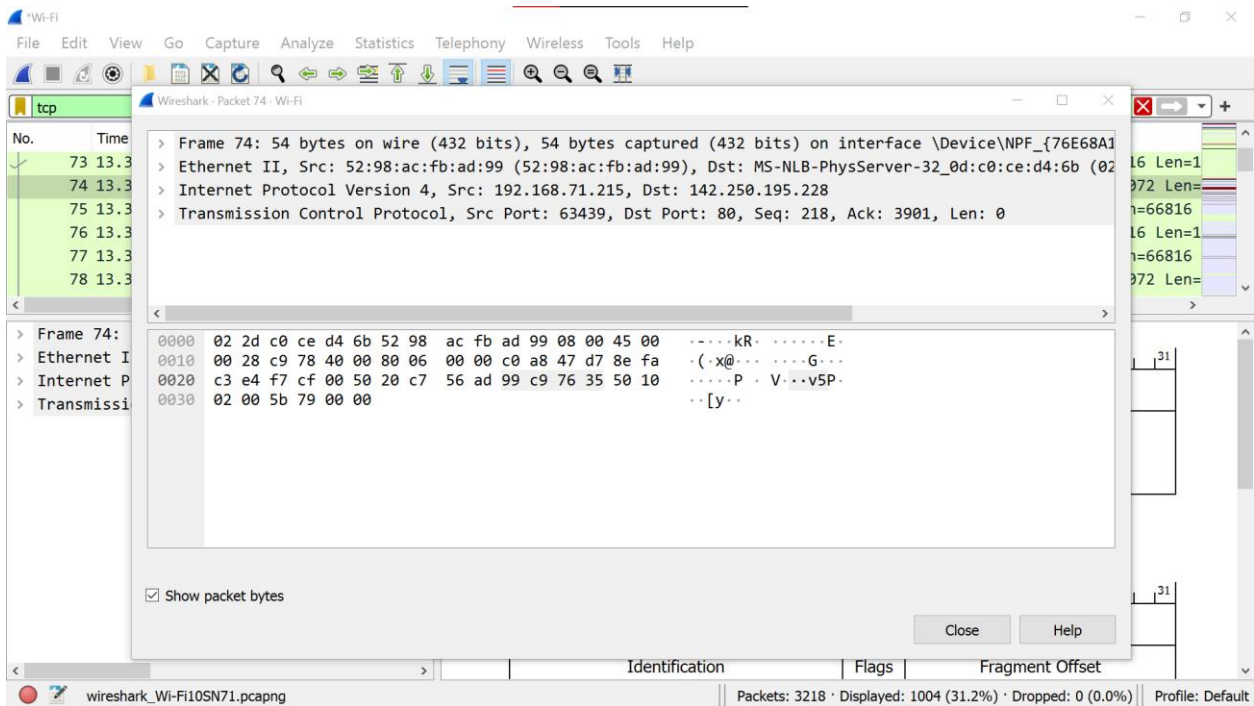
Ethernet II, Src: 52:98:ac:fb:ad:99 (52:98:ac:fb:ad:99), Dst: MS

Internet Protocol Version 6, Src: 2401:4900:2314:5e8b:1d15:c48d

Transmission Control Protocol, Src Port: 63361, Dst Port: 80, Seq=1, Ack=1, Win=510, Len=0

wireshark\_Wi-Fi10SN71.pcapng

Packets: 3218 · Displayed: 1004 (31.2%) · Dropped: 0 (0.0%) · Profile: Default

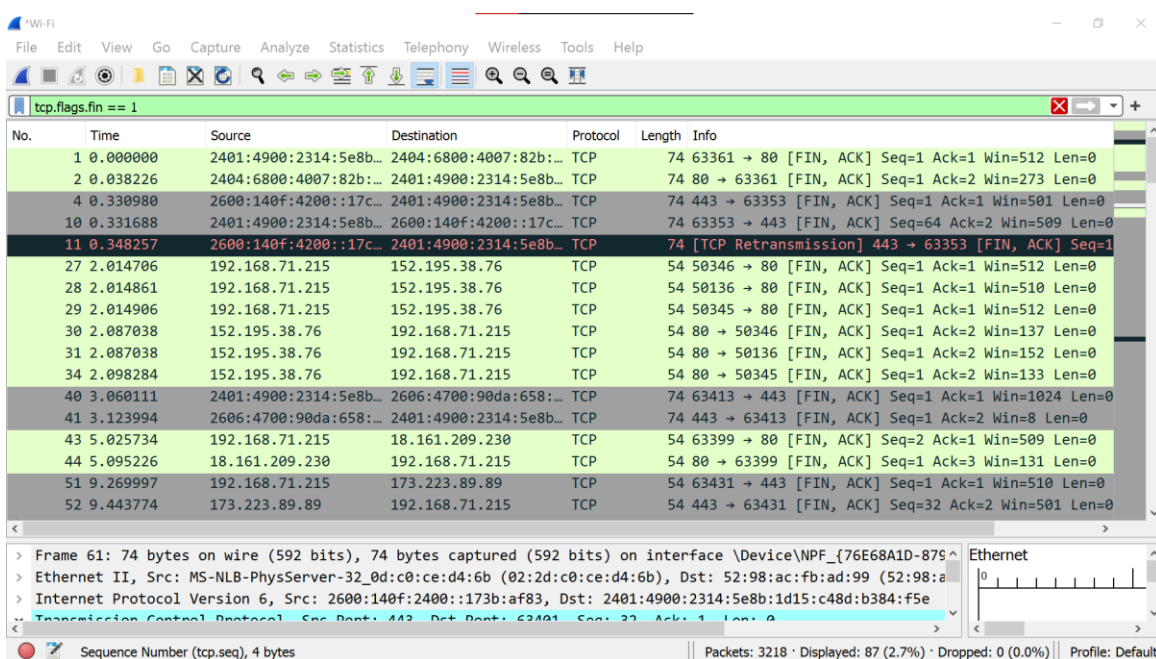


## 12.2 Filter the TCP packets that contain the request to terminate.

### Procedure:

- 1) Open Wireshark whilst connected to a network and use the appropriate interface (Ethernet/WiFi)
- 2) Press start and capture the packets for a few 5 to 10 minutes.
- 3) Press stop once enough packets have been gathered.
- 4) In the top right corner set a filter "tcp.flags.fin==1" which will aid in particularly searching for tcp packets that have sent a request to terminate. These packets are usually grayed out.

### Output:



We can see that all the packets displayed on the screen have FIN in their information status indicating that they have been sent to request to terminate.

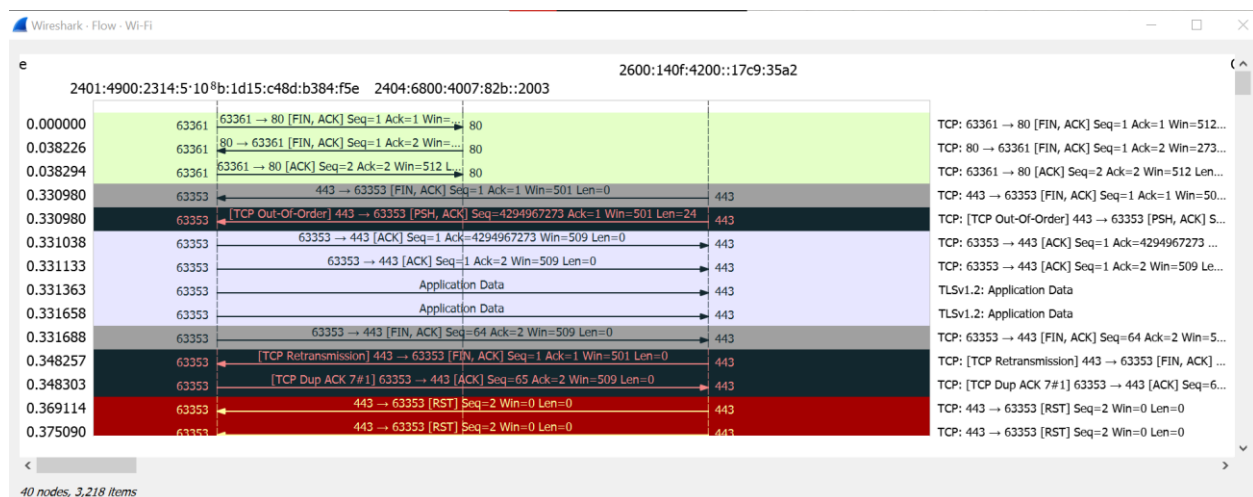
### 12.3. Depict the flowgraph and the I/O graph of TCP, UDP, ICMP, ARP, and TLS.

#### Procedure:

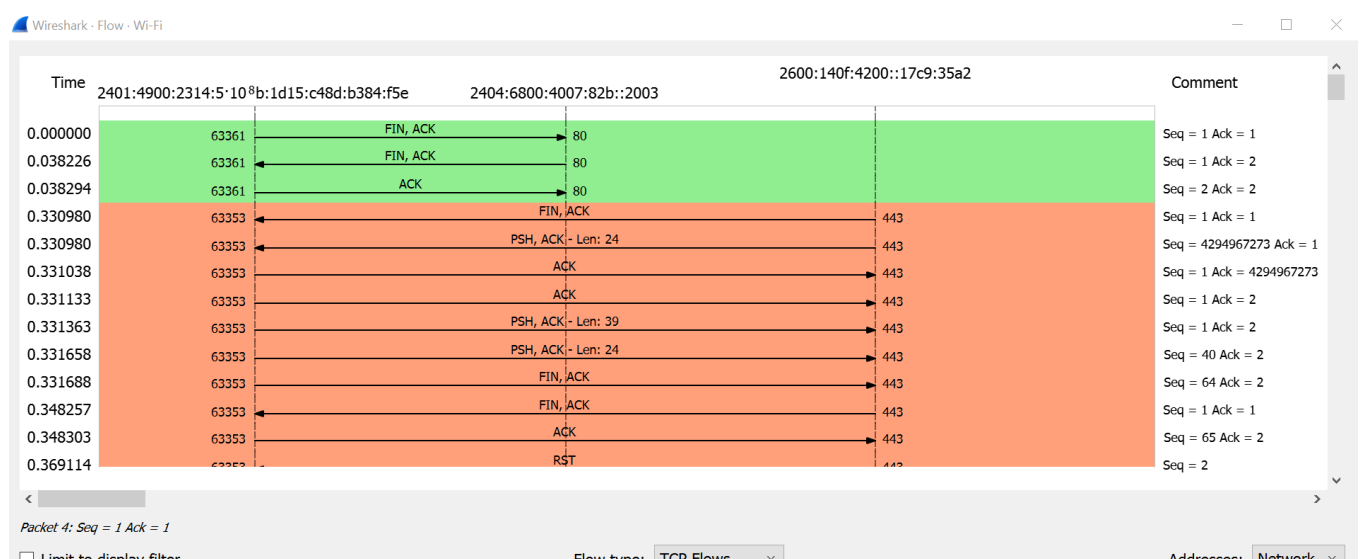
- 1) Open Wireshark whilst connected to a network and use the appropriate interface (Ethernet/WiFi)
- 2) Press start and capture the packets for a few 5 to 10 minutes.
- 3) Press stop once enough packets have been gathered.
- 4) Go to statistics in the menu bar and select flow graph.
- 5) From there choose what packets to filter by to see the flow graph in each protocol
- 6) Similarly got to statistics->IO graph to see the IO graphs of the packets you require.

#### Output:

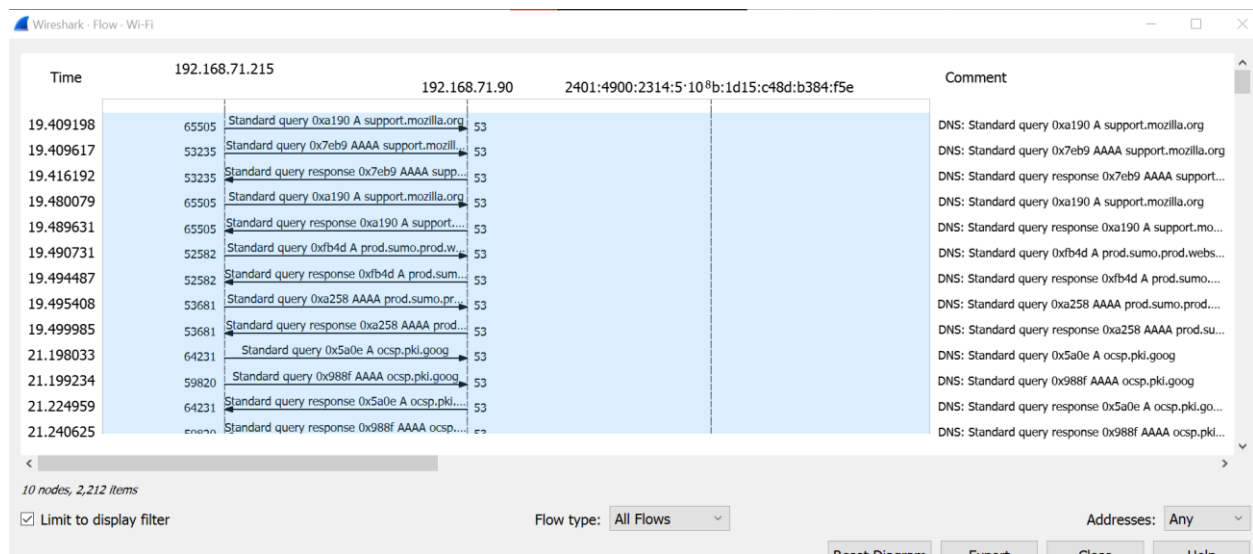
#### Overall



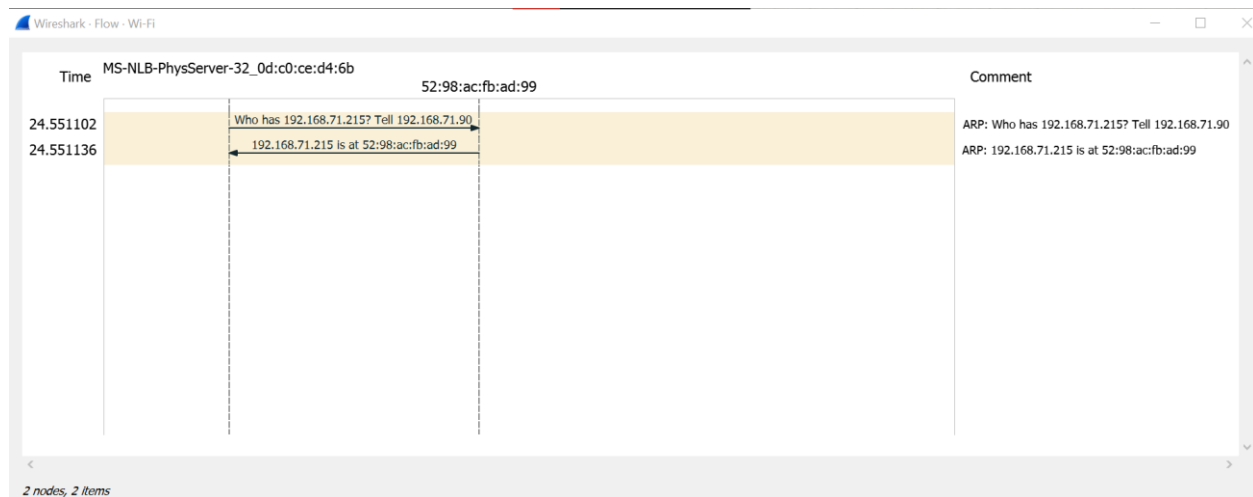
#### TCP



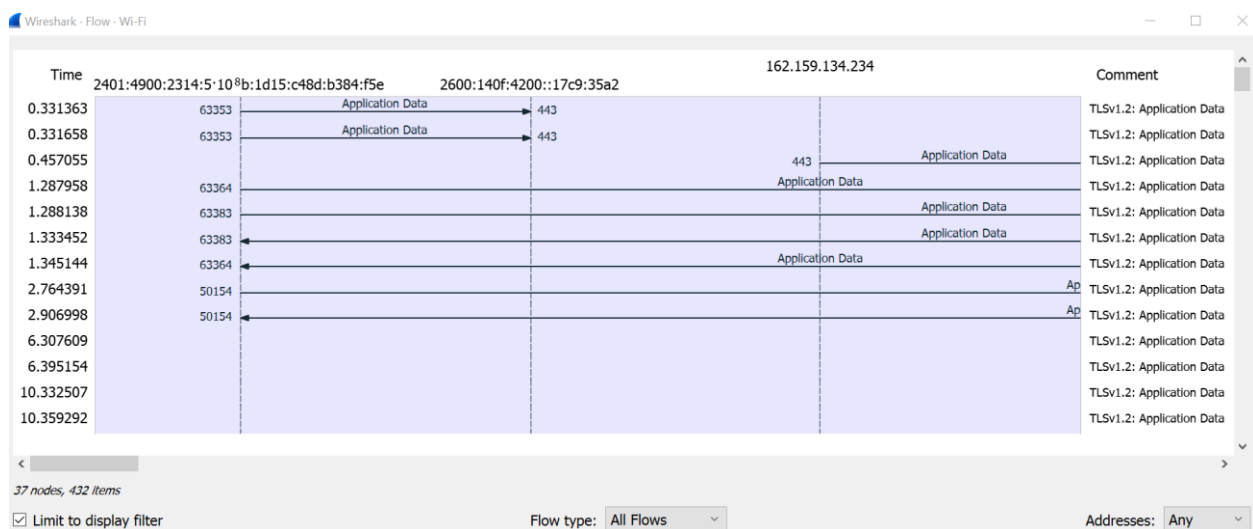
## UDP



## ARP



## TLS



## ICMP

No packets complying to ICMP were available in the packets that were collected as shown below.

\*Wi-Fi

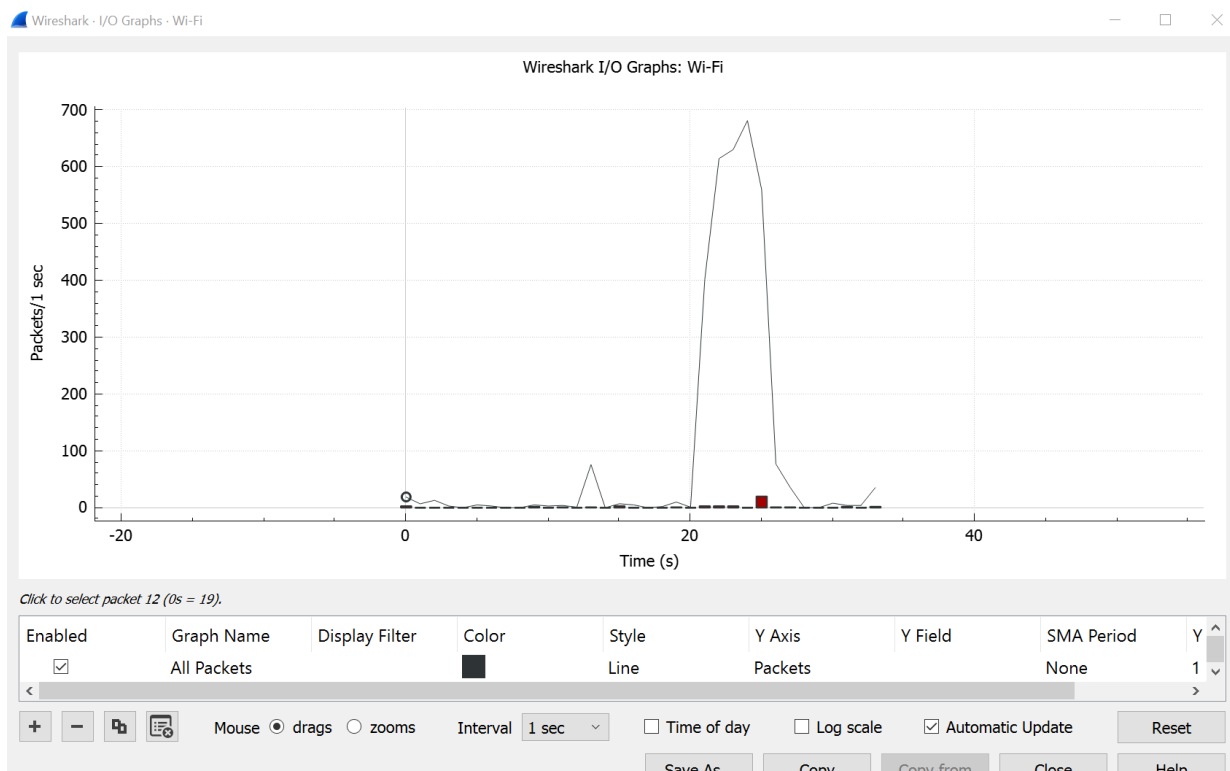
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help



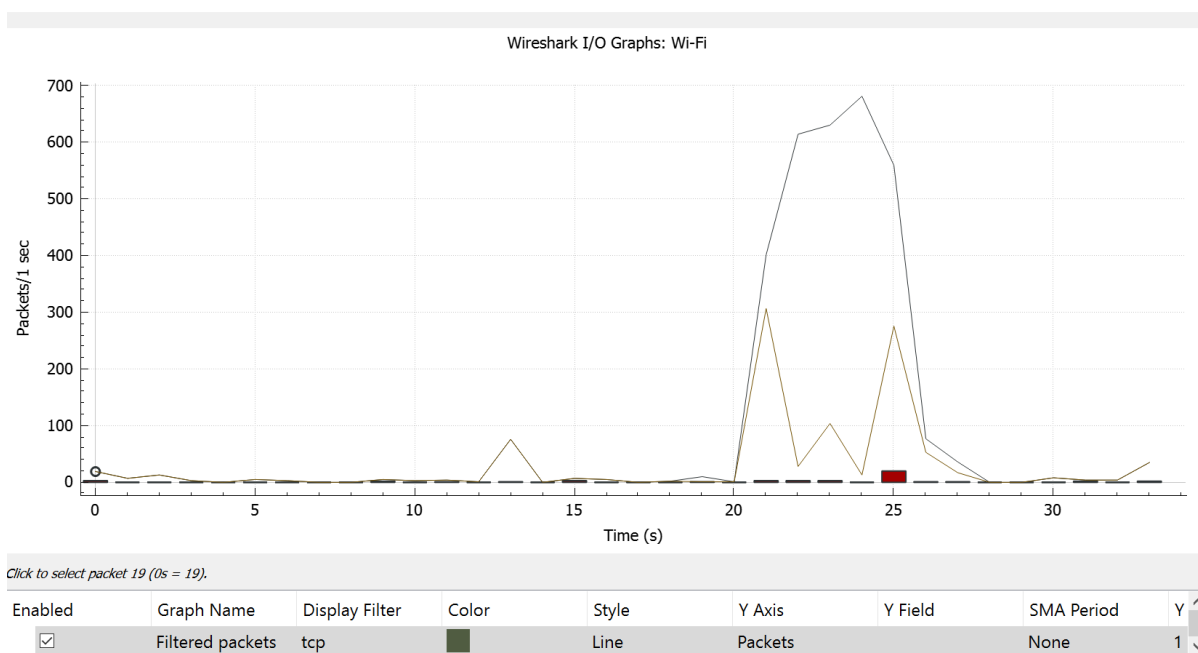
icmp

No.	Time	Source	Destination	Protocol	Length	Info
-----	------	--------	-------------	----------	--------	------

## IO Graph



## TCP packets



**Result:**

Through these experiments I have successfully gained practical knowledge on how to effectively use Wireshark for capturing PDUs and filtering them according to the various demands as well as viewing and retrieving flow and IO graphs for different commonly used protocols.