

Experiment 08

Study of packet sniffer tools Wireshark - a. Observer performance in promiscuous and non-promiscuous modes. b. Show the packets can be traced based on different filters.

Roll No.	37
Name	Mikil Lalwani
Class	D15-B
Subject	Security Lab
LO Mapped	LO3: Explore the different network reconnaissance tools to gather information about networks

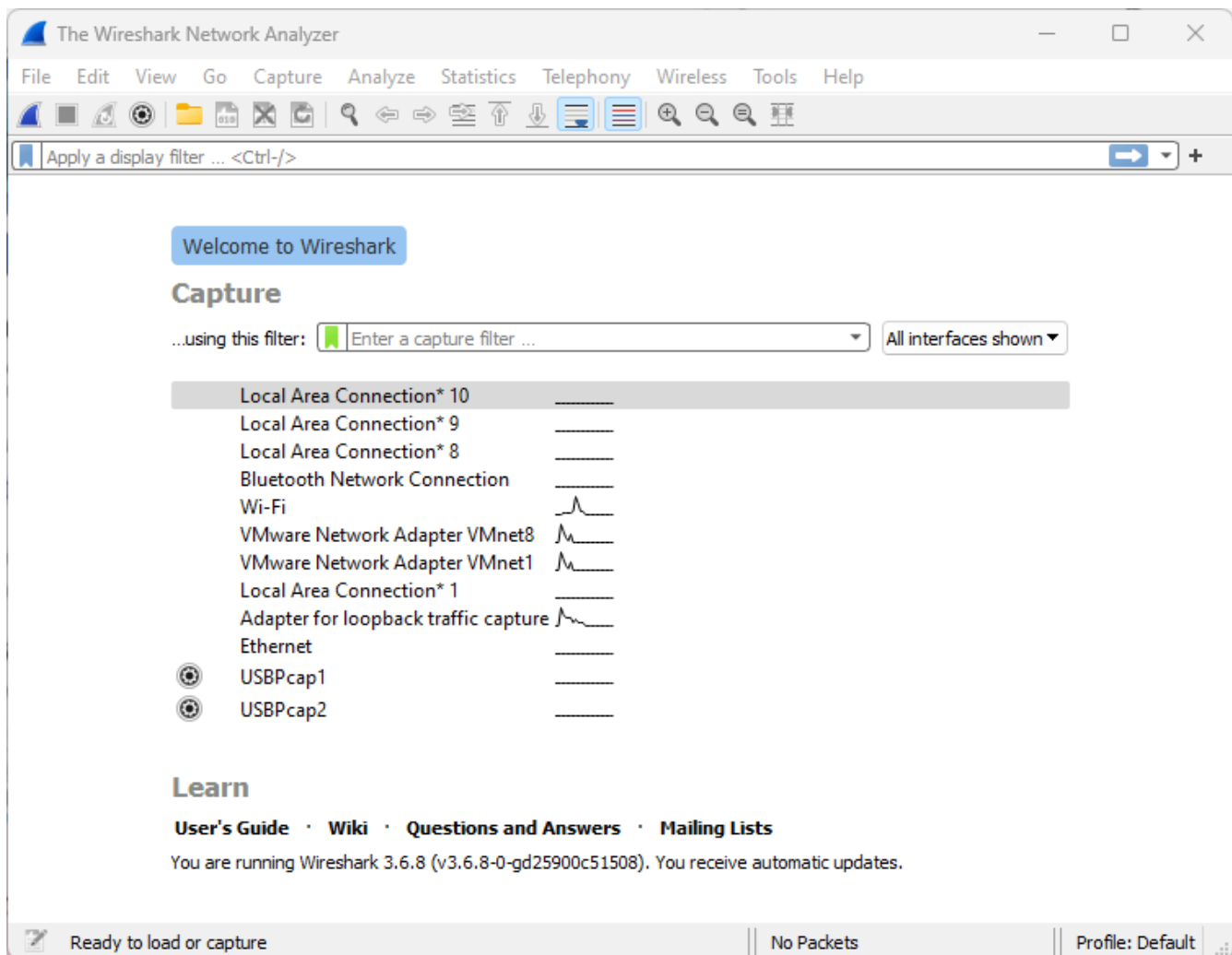
Aim:

Study of packet sniffer tools Wireshark: -

- a. Observer performance in promiscuous as well as non-promiscuous mode.
- b. Show the packets can be traced based on different filters.

Methods:

1. After downloading and installing Wireshark, you launch it and click the name of an interface under the interface list to start capturing packets on that interface. For example, if you want to capture traffic on the wireless network, you click your wireless interface. You can configure advanced features by clicking capture options.



2. As soon as you click the interface's name, you'll see the packets start to appear in real-time. Wireshark captures each packet sent to or from your system. If you're capturing on a wireless interface and have promiscuous mode enabled in your capture options, you'll also see other packets on the network.

The image shows a Wireshark packet capture window titled "Capturing from Wi-Fi". The interface includes a menu bar (File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help) and a toolbar with various icons for capture and analysis. A display filter bar shows "Apply a display filter ... <Ctrl-/>".

The packet list pane displays the following packets:

No.	Time	Source	Destination	Protocol	Length	Info
207	7.413032	2401:4900:1720:6930...	2404:6800:4009:80d:...	QUIC	96	Protected Payload (KP0),
208	7.440750	2404:6800:4009:80d:...	2401:4900:1720:6930...	QUIC	182	Protected Payload (KP0)
209	7.466032	2404:6800:4009:80d:...	2401:4900:1720:6930...	QUIC	91	Protected Payload (KP0)
210	7.466212	2401:4900:1720:6930...	2404:6800:4009:80d:...	QUIC	95	Protected Payload (KP0),
211	7.521229	2404:6800:4009:80d:...	2401:4900:1720:6930...	QUIC	1031	Protected Payload (KP0)
212	7.521229	2404:6800:4009:80d:...	2401:4900:1720:6930...	QUIC	87	Protected Payload (KP0)
213	7.521614	2401:4900:1720:6930...	2404:6800:4009:80d:...	QUIC	97	Protected Payload (KP0),
214	7.547450	2401:4900:1720:6930...	2404:6800:4009:80d:...	QUIC	95	Protected Payload (KP0),
215	7.601111	2404:6800:4009:80d:...	2401:4900:1720:6930...	QUIC	87	Protected Payload (KP0)
216	8.366137	2401:4900:1720:6930...	2404:6800:4009:82f:...	UDP	95	49234 → 443 Len=33
217	8.402522	2404:6800:4009:82f:...	2401:4900:1720:6930...	UDP	88	443 → 49234 Len=26

The packet details pane for the selected packet (Frame 1) shows:

- > Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface \Device\NPF_{EB2226AB-3F...
- > Ethernet II, Src: 66:70:e4:2b:33:f2 (66:70:e4:2b:33:f2), Dst: IntelCor_cb:e9:9f (e0:2b:e9:cb:e9:9f)
- > Internet Protocol Version 6, Src: 2404:6800:4009:82f::200e, Dst: 2401:4900:1720:6930:e550:66f4:8aad:1019
- > User Datagram Protocol, Src Port: 443, Dst Port: 49234
- > Data (36 bytes)

The packet bytes pane shows the raw data in hexadecimal and ASCII:

```

0000  e0 2b e9 cb e9 9f 66 70  e4 2b 33 f2 86 dd 60 00  .+...fp .+3...`
0010  00 00 00 2c 11 3a 24 04  68 00 40 09 08 2f 00 00  ...,$. h@../
0020  00 00 00 00 20 0e 24 01  49 00 17 20 69 30 e5 50  ...$. I.. i.P
0030  66 f4 8a ad 10 19 01 bb  c0 52 00 2c 62 d8 48 48  f..... R.,b.HH
0040  1f dd cf d4 90 ac 7f 32  db eb 7f af f5 a0 32 c3  .....2 .....2
0050  48 9b de fe 82 6a 6c 38  09 8a 7d be 75 a8 f7 09  H....j18 ..}.u...
0060  3b f7                                     ;
  
```

The status bar at the bottom indicates "Wi-Fi: <live capture in progress>" and "Packets: 217 · Displayed: 217 (100.0%) | Profile: Default".

- Click the stop capture button near the top left corner of the window when you want to stop capturing traffic.

The image shows a Wireshark packet capture window titled "Capturing from Wi-Fi". The interface includes a menu bar (File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help) and a toolbar with various icons. A status bar at the top indicates "Apply Stop capturing packets".

The main packet list displays the following data:

No.	Time	Source	Destination	Protocol	Length	Info
1099	47.997195	2404:6800:4009:832:...	2401:4900:1720:6930:...	TLSv1.2	198	Application Data
1100	47.999047	2401:4900:1720:6930:...	2404:6800:4009:832:...	TLSv1.2	109	Application Data
1101	47.999128	2401:4900:1720:6930:...	2404:6800:4009:832:...	TLSv1.2	109	Application Data
1102	48.024807	2404:6800:4009:832:...	2401:4900:1720:6930:...	TCP	74	443 → 60043 [ACK] Seq=559
1103	48.035461	2404:6800:4009:832:...	2401:4900:1720:6930:...	TCP	74	443 → 60043 [ACK] Seq=559
1104	48.382175	2401:4900:1720:6930:...	2404:6800:4009:82f:...	UDP	95	49234 → 443 Len=33
1105	48.433276	2404:6800:4009:82f:...	2401:4900:1720:6930:...	UDP	88	443 → 49234 Len=26
1106	50.041794	2401:4900:1720:6930:...	2404:6800:4009:82f:...	UDP	95	49234 → 443 Len=33
1107	50.112483	2404:6800:4009:82f:...	2401:4900:1720:6930:...	UDP	88	443 → 49234 Len=26
1108	50.984624	192.168.9.71	151.101.154.202	TCP	55	[TCP Keep-Alive] 60262 →
1109	51.035475	151.101.154.202	192.168.9.71	TCP	66	[TCP Keep-Alive ACK] 443

The detailed view of the selected packet (Frame 1) shows the following structure:

- Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface \Device\NPF_{EB2226AB-3F...}
- Ethernet II, Src: 66:70:e4:2b:33:f2 (66:70:e4:2b:33:f2), Dst: IntelCor_cb:e9:9f (e0:2b:e9:cb:e9:9f)
- Internet Protocol Version 6, Src: 2404:6800:4009:82f::200e, Dst: 2401:4900:1720:6930:e550:66f4:8aad:1019
- User Datagram Protocol, Src Port: 443, Dst Port: 49234
- Data (36 bytes)

The packet bytes are displayed in hexadecimal and ASCII format:

```

0000  e0 2b e9 cb e9 9f 66 70  e4 2b 33 f2 86 dd 60 00  .+...fp +3...`
0010  00 00 00 2c 11 3a 24 04  68 00 40 09 08 2f 00 00  .,.,$. h@./..
0020  00 00 00 00 20 0e 24 01  49 00 17 20 69 30 e5 50  .$. I. i.P
0030  66 f4 8a ad 10 19 01 bb  c0 52 00 2c 62 d8 48 48  f.....R.,b.HH
0040  1f dd cf d4 90 ac 7f 32  db eb 7f af f5 a0 32 c3  .....2.....2.
0050  48 9b de fe 82 6a 6c 38  09 8a 7d be 75 a8 f7 09  H...j18 ..}.u...
0060  3b f7                                     ;.

```

The status bar at the bottom indicates "Wi-Fi: <live capture in progress>" and "Packets: 1109 · Displayed: 1109 (100.0%) | Profile: Default".

- Wireshark uses colors to help you identify the types of traffic at a glance. By default, green is TCP traffic, dark blue is DNS traffic, light blue is UDP traffic, and black identifies TCP packets with problems — for example, they could have been delivered out-of-order.

Wireshark packet capture window showing a filtered list of DNS packets. The filter 'dns' is applied. The packet list shows various DNS queries and responses between 192.168.9.71 and 192.168.9.167. Packet 15 is selected, and its details are shown below the list. The details pane shows the structure of the DNS query: Ethernet II, Internet Protocol Version 4, User Datagram Protocol, and Domain Name System (query). The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
15	0.135030	192.168.9.71	192.168.9.167	DNS	77	Standard query 0xa199 A e
16	0.135190	192.168.9.71	192.168.9.167	DNS	77	Standard query 0xa976 AAA
20	0.202532	192.168.9.167	192.168.9.71	DNS	93	Standard query response 0
21	0.203729	192.168.9.167	192.168.9.71	DNS	134	Standard query response 0
187	7.179561	192.168.9.71	192.168.9.167	DNS	76	Standard query 0xb9e2 A b
188	7.179617	192.168.9.71	192.168.9.167	DNS	76	Standard query 0x72ca AAA
189	7.271627	192.168.9.71	192.168.9.167	DNS	76	Standard query 0x72ca AAA
190	7.271627	192.168.9.71	192.168.9.167	DNS	76	Standard query 0xb9e2 A b
192	7.288313	192.168.9.167	192.168.9.71	DNS	92	Standard query response 0
193	7.288870	192.168.9.167	192.168.9.71	DNS	127	Standard query response 0
837	28.099884	192.168.9.71	192.168.9.167	DNS	91	Standard query 0x4588 A s
838	28.100096	192.168.9.71	192.168.9.167	DNS	91	Standard query 0x3033 AAA

> Frame 15: 77 bytes on wire (616 bits), 77 bytes captured (616 bits) on interface \Device\NPF_{EB2226AB-3...}

> Ethernet II, Src: IntelCor_cb:e9:9f (e0:2b:e9:cb:e9:9f), Dst: 66:70:e4:2b:33:f2 (66:70:e4:2b:33:f2)

> Internet Protocol Version 4, Src: 192.168.9.71, Dst: 192.168.9.167

> User Datagram Protocol, Src Port: 57284, Dst Port: 53

> Domain Name System (query)

0000 66 70 e4 2b 33 f2 e0 2b e9 cb e9 9f 08 00 45 00 fp+3...+E

0010 00 3f 94 9a 00 00 80 11 00 00 c0 a8 09 47 c0 a8 .?.....G

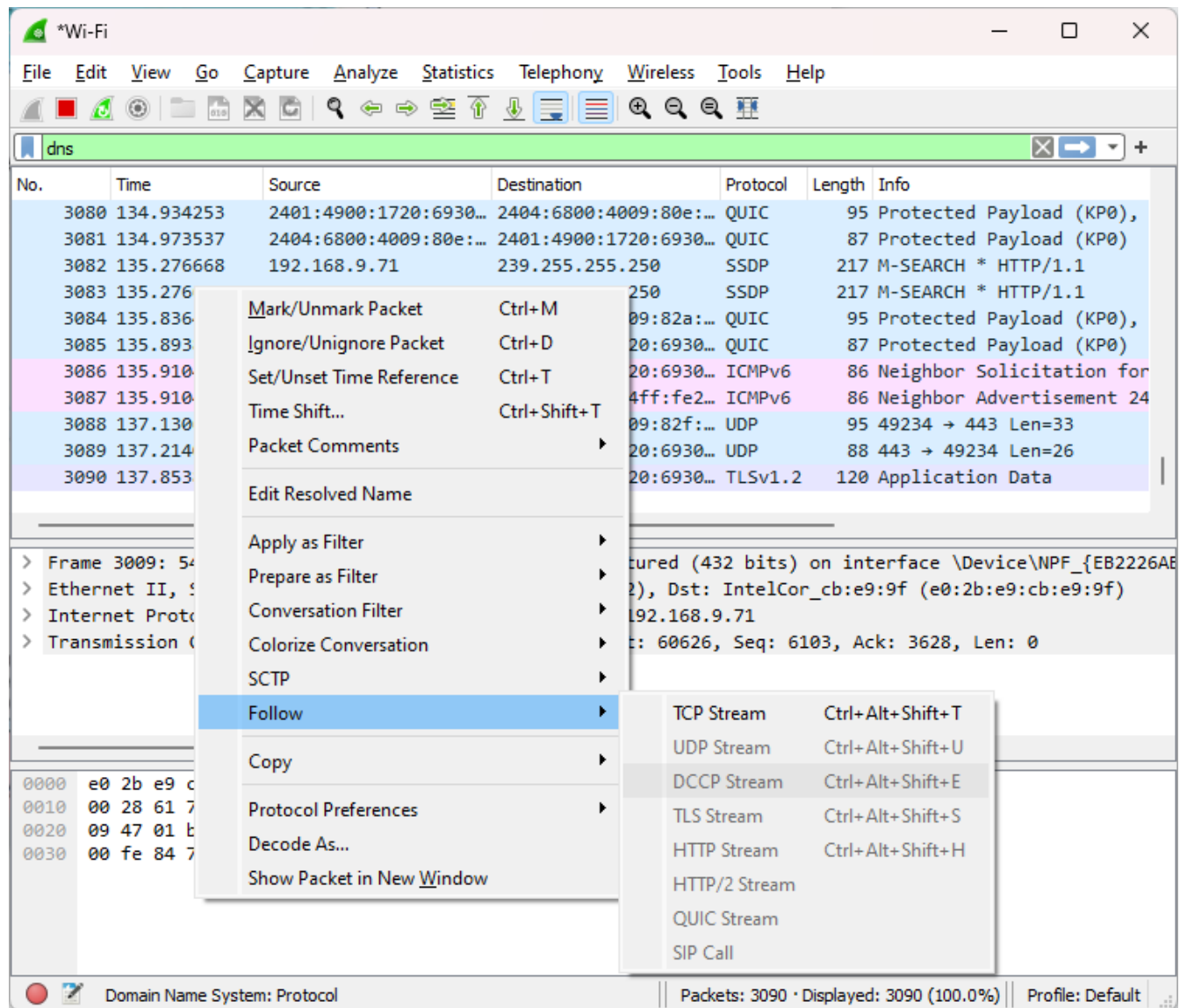
0020 09 a7 df c4 00 35 00 2b 94 7b a1 99 01 00 00 015+ .{.....

0030 00 00 00 00 00 04 65 32 63 33 03 67 63 70 04e 2c3.gcp

0040 67 76 74 32 03 63 6f 6d 00 00 01 00 01 gvt2.com

Domain Name System: Protocol | Packets: 1982 · Displayed: 24 (1.2%) | Profile: Default

5. Filtering Packets If you're trying to inspect something specific, such as the traffic a program sends when phoning home, it helps to close down all other applications using the network so you can narrow down the traffic. Still, you'll likely have a large number of packets to sift through. That's where Wireshark's filters come in.
The most basic way to apply a filter is by typing it into the filter box at the top of the window and clicking Apply (or pressing Enter). For example, type "DNS" and you'll see only DNS packets. When you start typing, Wireshark will help you autocomplete your filter.
6. Another interesting thing you can do is right-click a packet and click on Follow TCP Stream.



7. You can see full conversation between client and server.



8. Close the window and you can see a filter has been applied automatically. Wireshark is showing you the packets that make up the conversation.

The image shows a Wireshark packet capture window titled '*Wi-Fi'. The interface includes a menu bar (File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help) and a toolbar with various icons for packet capture and analysis. The packet list pane shows a table of captured packets:

No.	Time	Source	Destination	Protocol	Length	Info
2561	97.149588	192.168.9.71	20.85.30.134	TLSv1.2	100	Application Data
2562	97.149605	192.168.9.71	20.85.30.134	TLSv1.2	239	Application Data
2565	97.414269	20.85.30.134	192.168.9.71	TCP	54	443 → 60315 [ACK] Seq=166
2566	97.428397	20.85.30.134	192.168.9.71	TLSv1.2	100	Application Data
2567	97.428543	20.85.30.134	192.168.9.71	TLSv1.2	96	Application Data
2568	97.428543	20.85.30.134	192.168.9.71	TLSv1.2	131	Application Data
2569	97.428578	192.168.9.71	20.85.30.134	TCP	54	60315 → 443 [ACK] Seq=608
3101	142.440489	192.168.9.71	20.85.30.134	TCP	55	[TCP Keep-Alive] 60315 →
3102	142.693904	20.85.30.134	192.168.9.71	TCP	66	[TCP Keep-Alive ACK] 443
3783	159.150261	192.168.9.71	20.85.30.134	TLSv1.2	125	Application Data
3784	159.150436	192.168.9.71	20.85.30.134	TLSv1.2	100	Application Data
3785	159.150532	192.168.9.71	20.85.30.134	TLSv1.2	257	Application Data

Below the packet list, the details pane for packet 3785 is expanded, showing the following information:

- > Frame 3785: 257 bytes on wire (2056 bits), 257 bytes captured (2056 bits) on interface \Device\NPF_{EB22...}
- > Ethernet II, Src: IntelCor_cb:e9:9f (e0:2b:e9:cb:e9:9f), Dst: 66:70:e4:2b:33:f2 (66:70:e4:2b:33:f2)
- > Internet Protocol Version 4, Src: 192.168.9.71, Dst: 20.85.30.134
- > Transmission Control Protocol, Src Port: 60315, Dst Port: 443, Seq: 725, Ack: 331, Len: 203
- > Transport Layer Security

At the bottom, the packet bytes pane displays the raw data in hexadecimal and ASCII format:

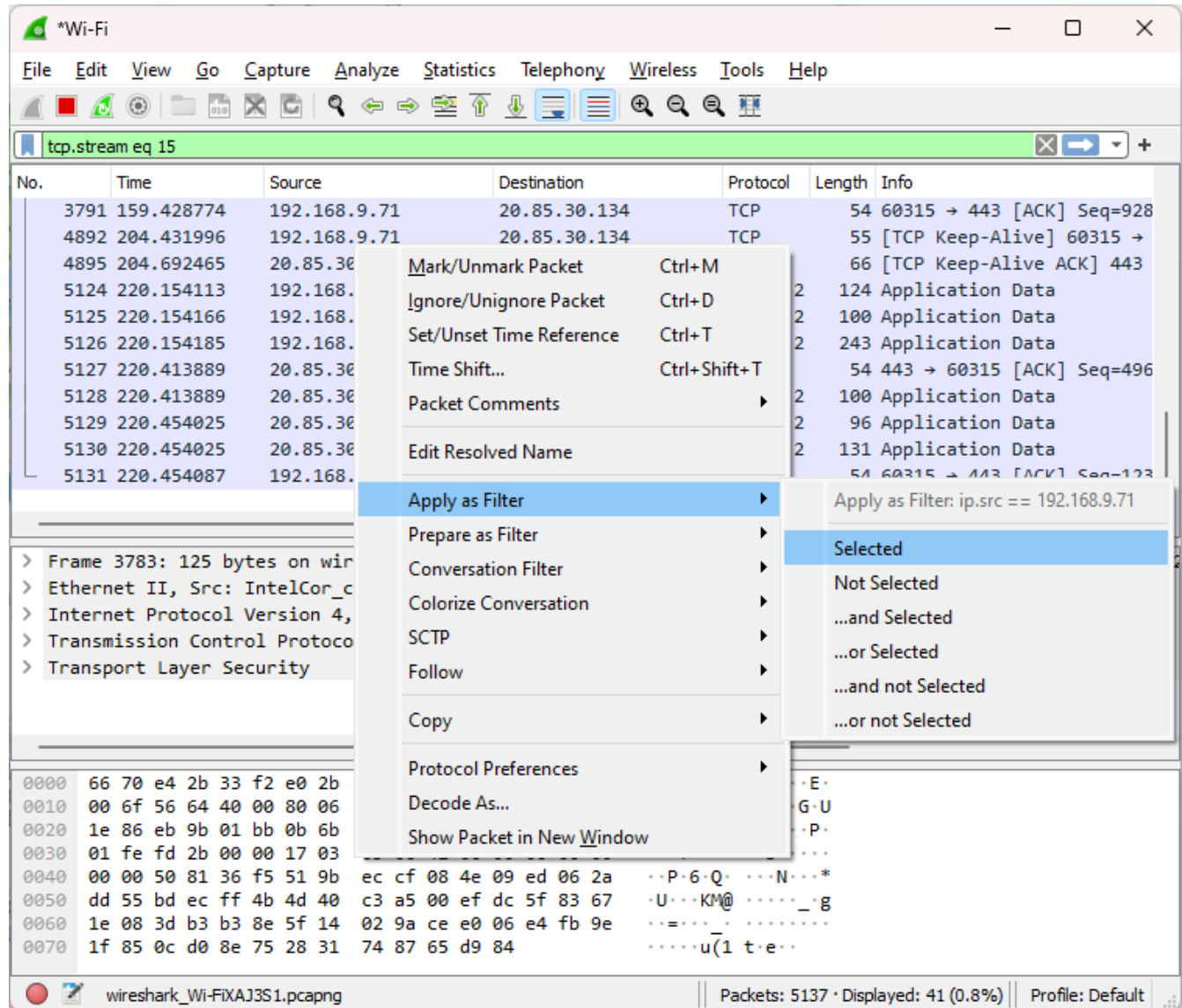
```

0000  66 70 e4 2b 33 f2 e0 2b e9 cb e9 9f 08 00 45 00  fp+3...+ .....E
0010  00 f3 56 66 40 00 80 06 00 00 c0 a8 09 47 14 55  ..Vf@... ..G.U
0020  1e 86 eb 9b 01 bb 0b 6b bc f3 b5 fa 15 f7 50 18  ....k .....P
0030  01 fe fd af 00 00 17 03 03 00 c6 00 00 00 00 00  .....
0040  00 00 52 64 d4 9b 46 fe 76 45 88 dd 3c 59 24 74  ..Rd..F..vE..<Y$t
0050  80 27 2b ef e4 9e ff db 1f 37 4f 7d 30 b7 70 7a  .'+.....70}0.pz
0060  c7 86 1b 05 1e e8 7a 70 c9 79 16 26 60 a2 db e7  ....zp..y.&`...
0070  75 0b 09 98 0a 98 46 60 1a 58 ae 46 ca 6d a4 44  u.....F`..X.F..m.D
  
```

The status bar at the bottom indicates the file name 'wireshark_Wi-FIXAJ3S1.pcapng', the total number of packets 'Packets: 4226', the number of displayed packets 'Displayed: 31 (0.7%)', and the profile 'Profile: Default'.

9. Inspecting Packets

Click a packet to select it and you can dig down to view its details.



10. Wireshark is an extremely powerful tool, and this tutorial is just scratching the surface of what you can do with it. Professionals use it to debug network protocol implementations, examine security problems and inspect network protocol internals.

Conclusion:

In this experiment, we analyze various packet sniffing tools that monitor network traffic transmitted between legitimate users or in the network. The packet sniffer is a network monitoring tool. It has opted for network monitoring, traffic analysis, troubleshooting, Packet gripping, message, protocol analysis, penetration testing, and many other purposes.