

19/08/24

Hemanth kumar.A

231901010

Ex No:4a STUDY OF WIRESHARK TOOL FOR PACKET SNIFFING

AIM:

To study packet sniffing concepts using Wireshark Tool.

DESCRIPTION:

Wireshark, a network analysis tool formerly known as Ethereal, captures packets in real time and display them in human-readable format. Wireshark includes filters, color coding, and other features that let you dig deep into network traffic and inspect individual packets. You can use Wireshark to inspect a suspicious program's network traffic, analyze the traffic flow on your network, or troubleshoot network problems.

What we can do with Wireshark:

- Capture network traffic
- Decode packet protocols using dissectors
- Define filters – capture and display
- Watch smart statistics
- Analyze problems
- Interactively browse that traffic

Wireshark used for:

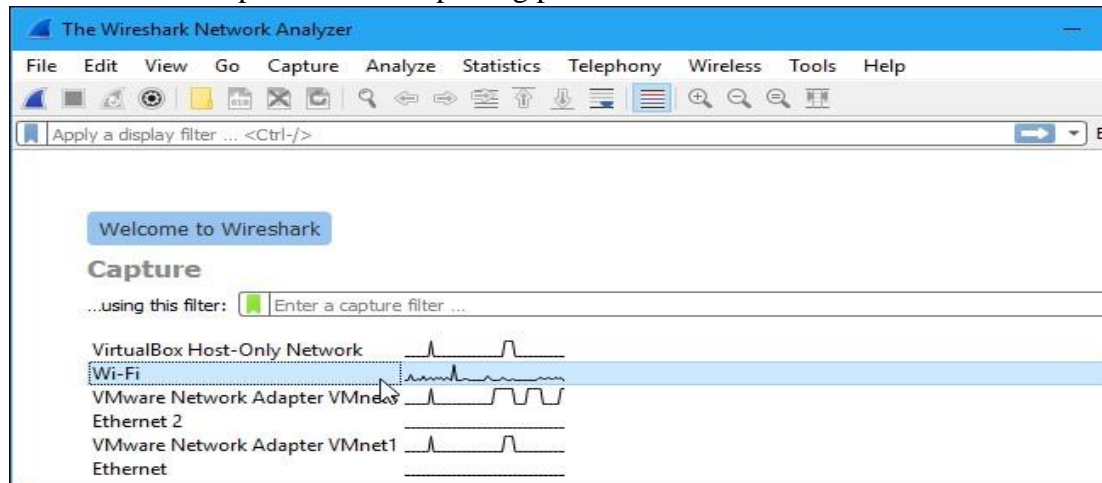
- Network administrators: troubleshoot network problems
- Network security engineers: examine security problems
- Developers: debug protocol implementations
- People: learn **network protocol internals**

Getting Wireshark

Wireshark can be downloaded for Windows or macOS from [its official website](#). For Linux or another UNIX-like system, Wireshark will be found in its package repositories. For Ubuntu, Wireshark will be found in the Ubuntu Software Center.

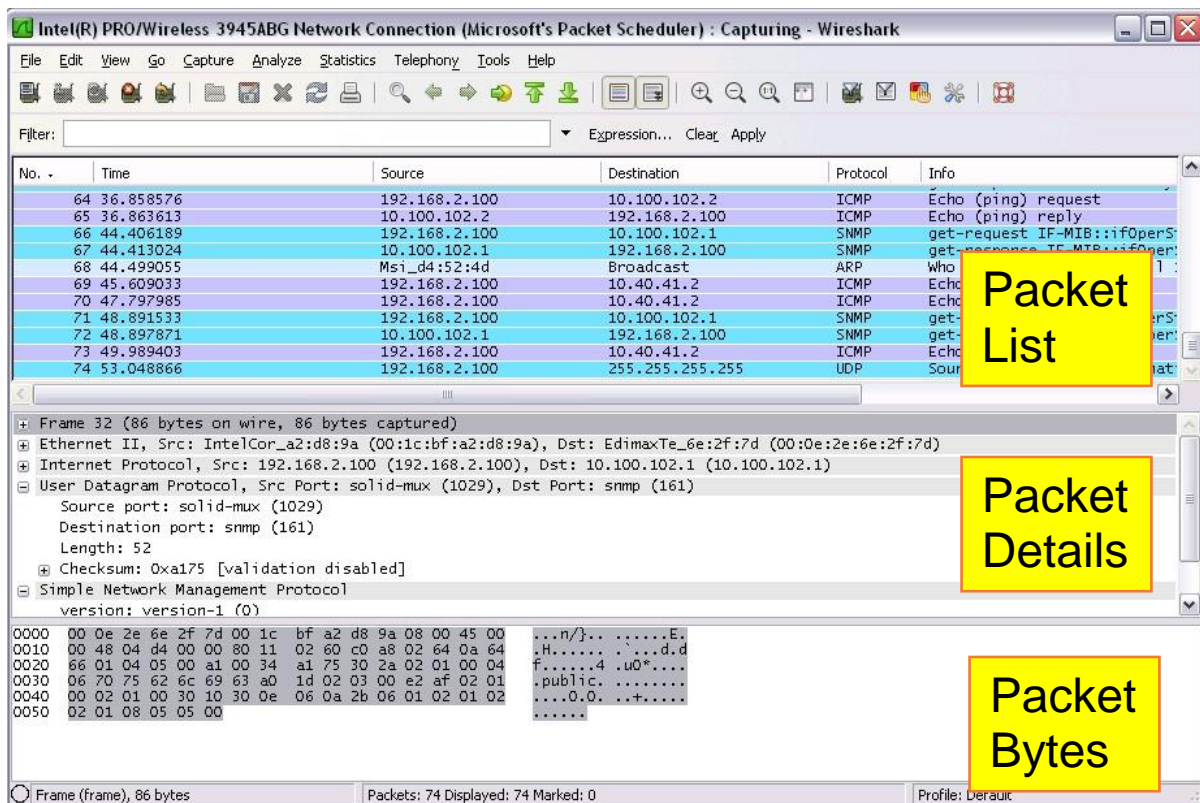
Capturing Packets

After downloading and installing Wireshark, launch it and double-click the name of a network interface under Capture to start capturing packets on that interface



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 have promiscuous mode enabled—it's enabled by default—you'll also see all the other packets on the network instead of only packets addressed to your network adapter. To check if promiscuous mode is enabled, click Capture > Options and verify the ☒ Enable promiscuous mode on all interfaces checkbox is activated at the bottom of this window.



Click the red —Stop button near the top left corner of the window when you want to stop capturing traffic.

The “Packet List” Pane

The packet list pane displays all the packets in the current capture file. The —Packet List pane Each line in the packet list corresponds to one packet in the capture file. If you select a line in this pane, more details will be displayed in the —Packet Details pane and —Packet Bytes pane.

The “Packet Details” Pane

The packet details pane shows the current packet (selected in the —Packet List pane) in a more detailed form. This pane shows the protocols and protocol fields of the packet selected in the —Packet List pane. The protocols and fields of the packet shown in a tree which can be expanded and collapsed.

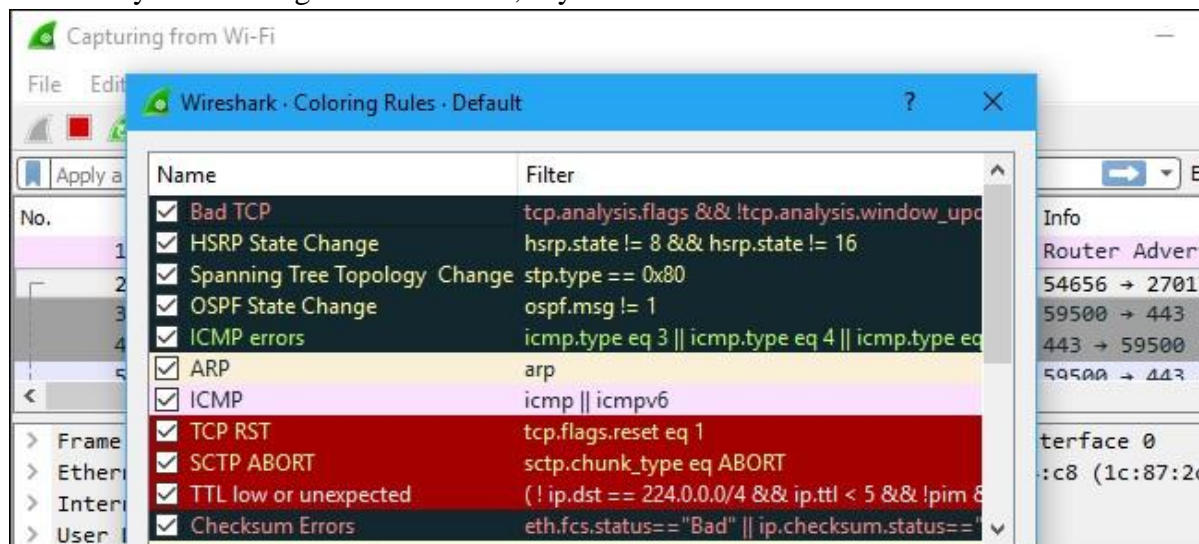
The “Packet Bytes” Pane

The packet bytes pane shows the data of the current packet (selected in the —Packet List pane) in a hexdump style.

Color Coding

You’ll probably see packets highlighted in a variety of different colors. Wireshark uses colors to help you identify the types of traffic at a glance. By default, light purple is TCP traffic, light blue is UDP traffic, and black identifies packets with errors—for example, they could have been delivered out of order.

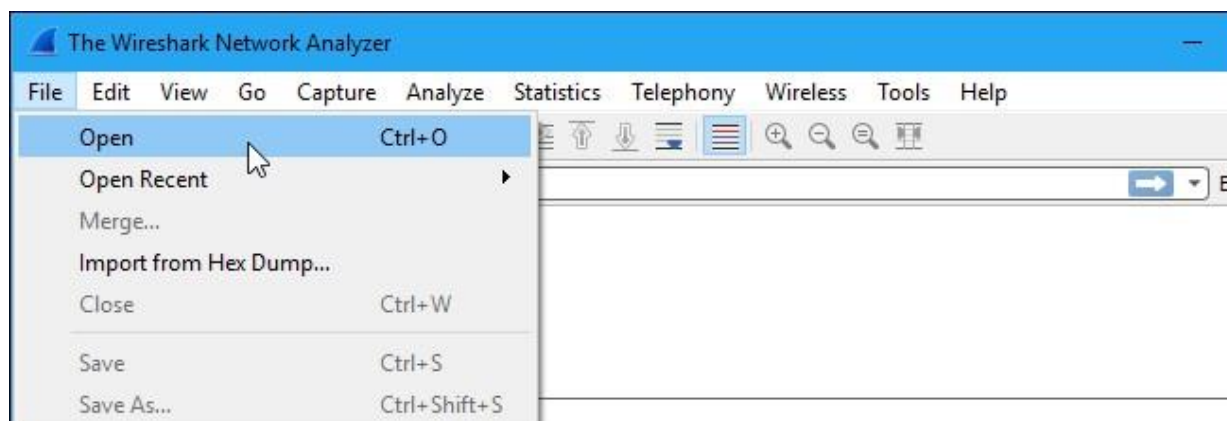
To view exactly what the color codes mean, click View > Coloring Rules. You can also customize and modify the coloring rules from here, if you like.



Sample Captures

If there’s nothing interesting on your own network to inspect, Wireshark’s wiki has you covered. The wiki contains a [page of sample capture files](#) that you can load and inspect. Click File > Open in Wireshark and browse for your downloaded file to open one.

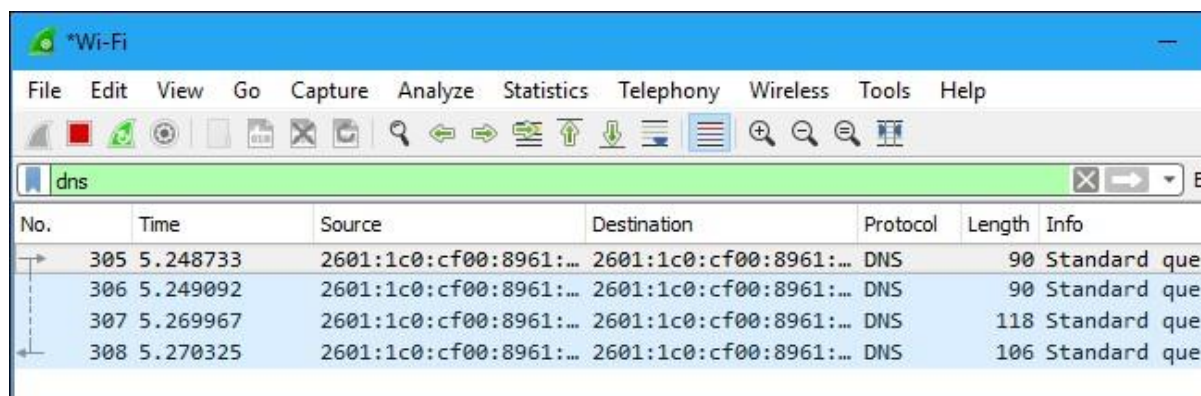
You can also save your own captures in Wireshark and open them later. Click File > Save to save your captured packets.



Filtering Packets

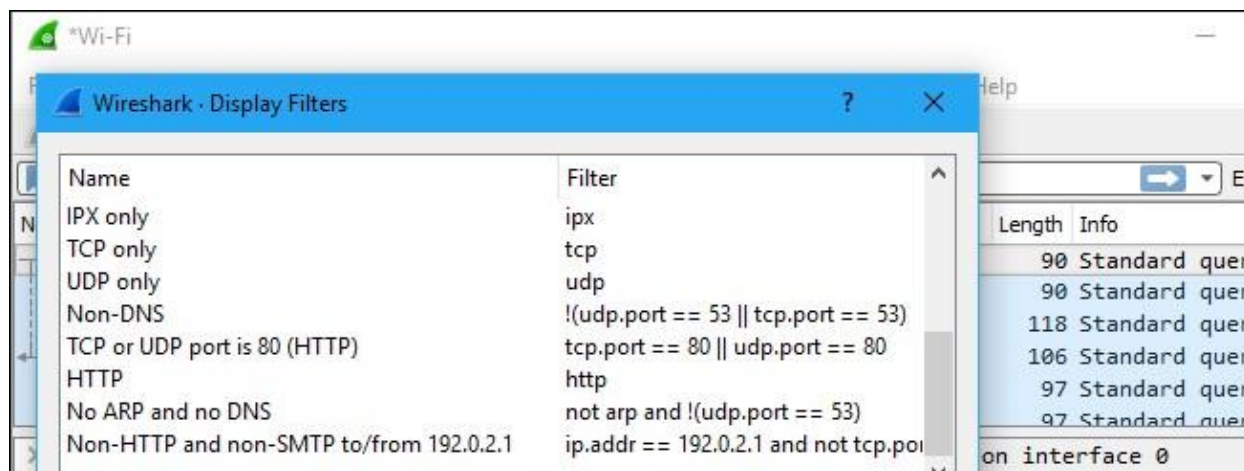
If you're trying to inspect something specific, such as the traffic a program sends when phoning home, it helps to close down all other applications using the network so you can narrow down the traffic. Still, you'll likely have a large amount of packets to sift through. That's where Wireshark's filters 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.



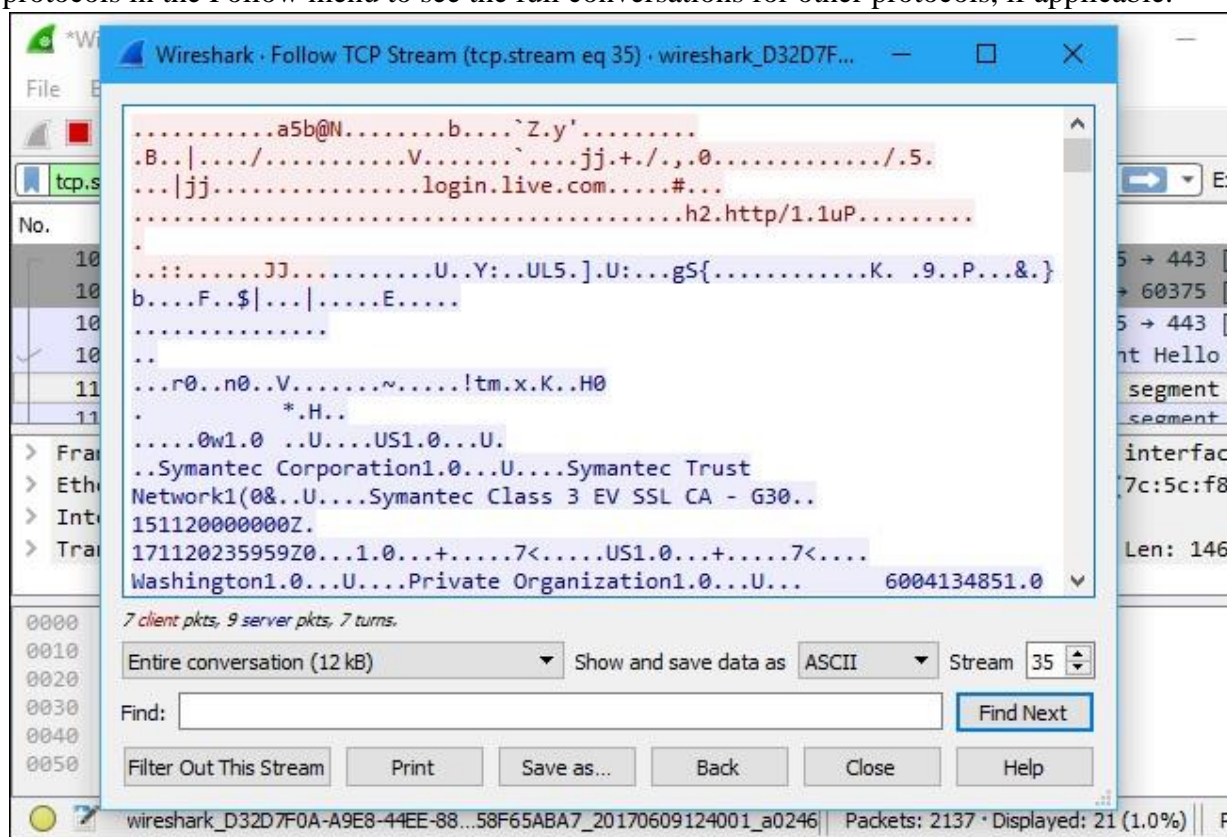
You can also click Analyze > Display Filters to choose a filter from among the default filters included in Wireshark. From here, you can add your own custom filters and save them to easily access them in the future.

For more information on Wireshark's display filtering language, read the [Building display filter expressions](#) page in the official Wireshark documentation.



Another interesting thing you can do is right-click a packet and select Follow > TCP Stream.

You'll see the full TCP conversation between the client and the server. You can also click other protocols in the Follow menu to see the full conversations for other protocols, if applicable.



Close the window and you'll find a filter has been applied automatically. Wireshark is showing you the packets that make up the conversation.

No.	Time	Source	Destination	Protocol	Length	Info
1054	2.798483	192.168.29.250	131.253.61.66	TCP	66	60375 → 443
1078	2.891263	131.253.61.66	192.168.29.250	TCP	58	443 → 60375
1079	2.891359	192.168.29.250	131.253.61.66	TCP	54	60375 → 443
1080	2.891527	192.168.29.250	131.253.61.66	TLSv1.2	288	Client Hello
1103	2.992980	131.253.61.66	192.168.29.250	TCP	1514	[TCP segment
1104	2.992980	131.253.61.66	192.168.29.250	TCP	1514	[TCP segment

> Frame 1078: 58 bytes on wire (464 bits), 58 bytes captured (464 bits) on interface 0
 > Ethernet II, Src: AsustekC_35:e4:c8 (1c:87:2c:35:e4:c8), Dst: IntelCor_38:be:bd (7c:5c:f8
 > Internet Protocol Version 4, Src: 131.253.61.66, Dst: 192.168.29.250
 > Transmission Control Protocol, Src Port: 443, Dst Port: 60375, Seq: 0, Ack: 1, Len: 0

Inspecting Packets

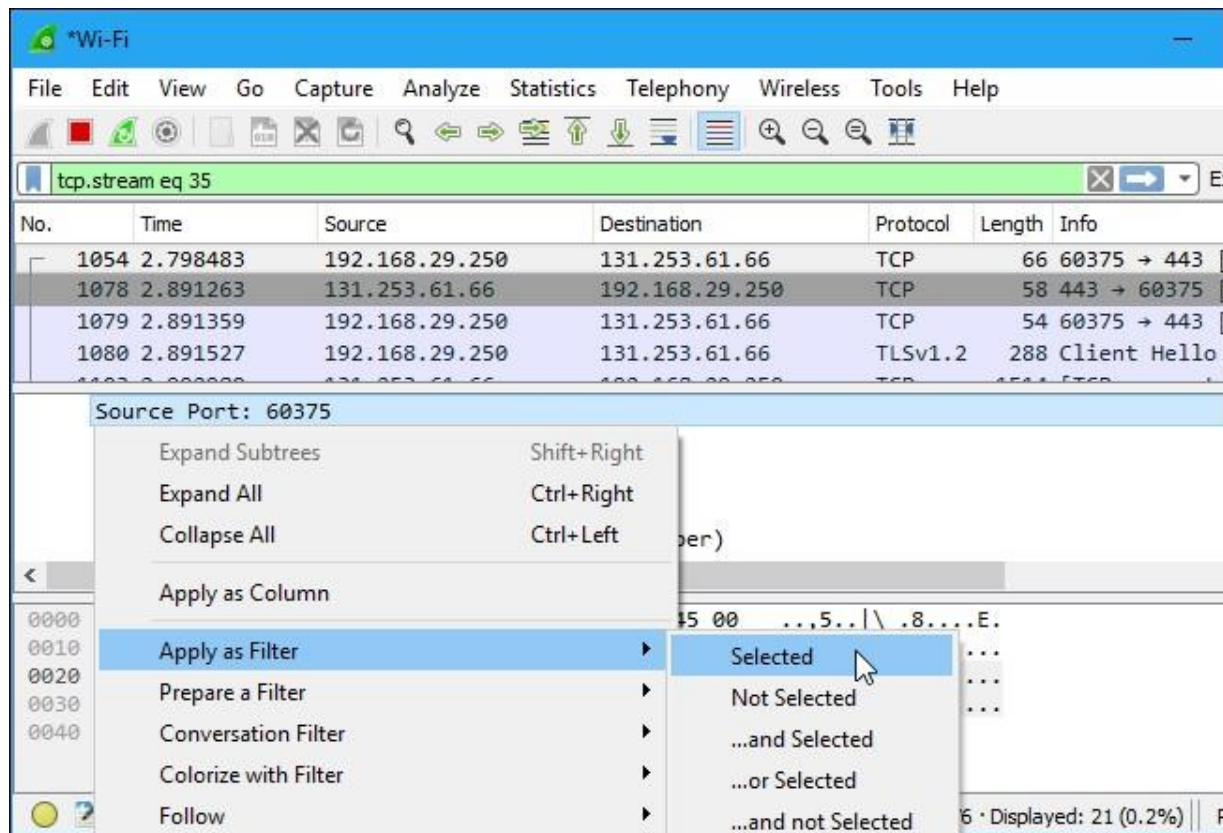
Click a packet to select it and you can dig down to view its details.

> Frame 1054: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
 Interface id: 0 (\Device\NPF_{D32D7F0A-A9E8-44EE-88DC-DFD58F65ABA7})
 Encapsulation type: Ethernet (1)
 Arrival Time: Jun 9, 2017 12:40:04.140141000 Pacific Daylight Time
 [Time shift for this packet: 0.000000000 seconds]
 Epoch Time: 1497037204.140141000 seconds

0000	1c 87 2c 35 e4 c8 7c 5c f8 38 be bd 08 00 45 00	..,5.. \ .8....E.
0010	00 34 0b 5d 40 00 80 06 4f 85 c0 a8 1d fa 83 fd	.4.]@... 0.....
0020	3d 42 eb d7 01 bb 22 52 7b 69 00 00 00 00 80 02	=B...."R {i.....
0030	fa f0 48 ef 00 00 02 04 05 b4 01 03 03 08 01 01	..H.....
0040	04 02	..

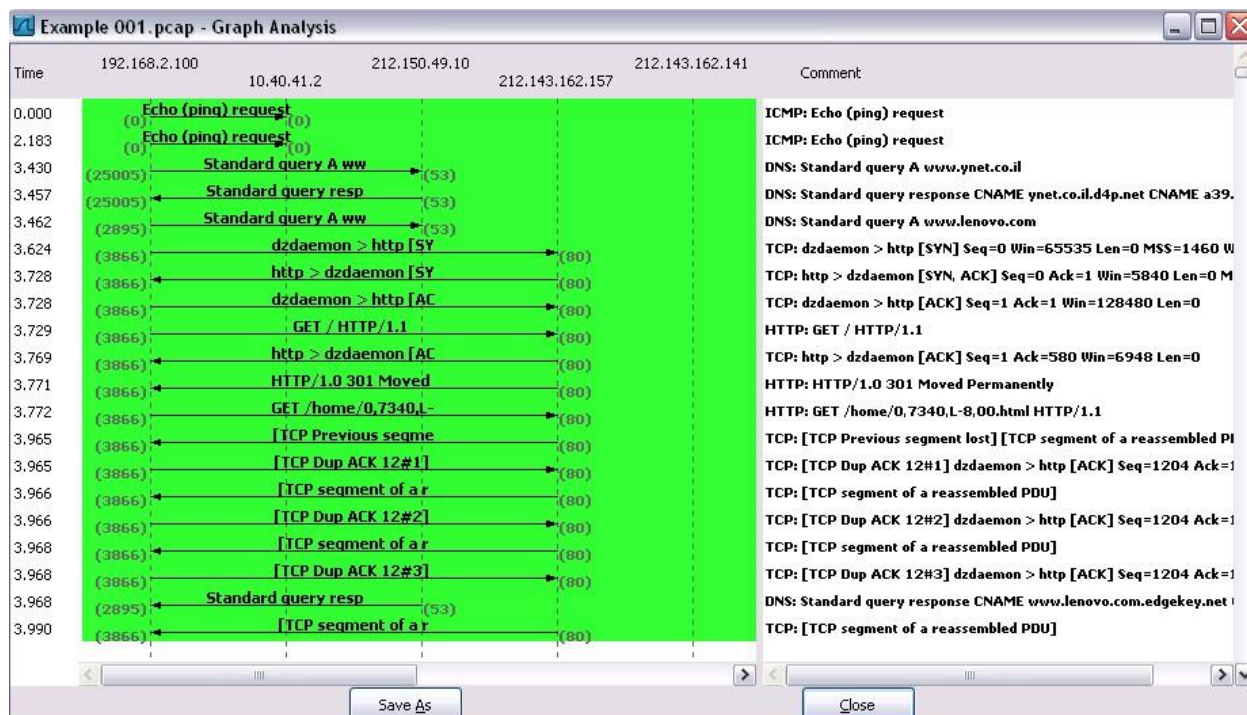
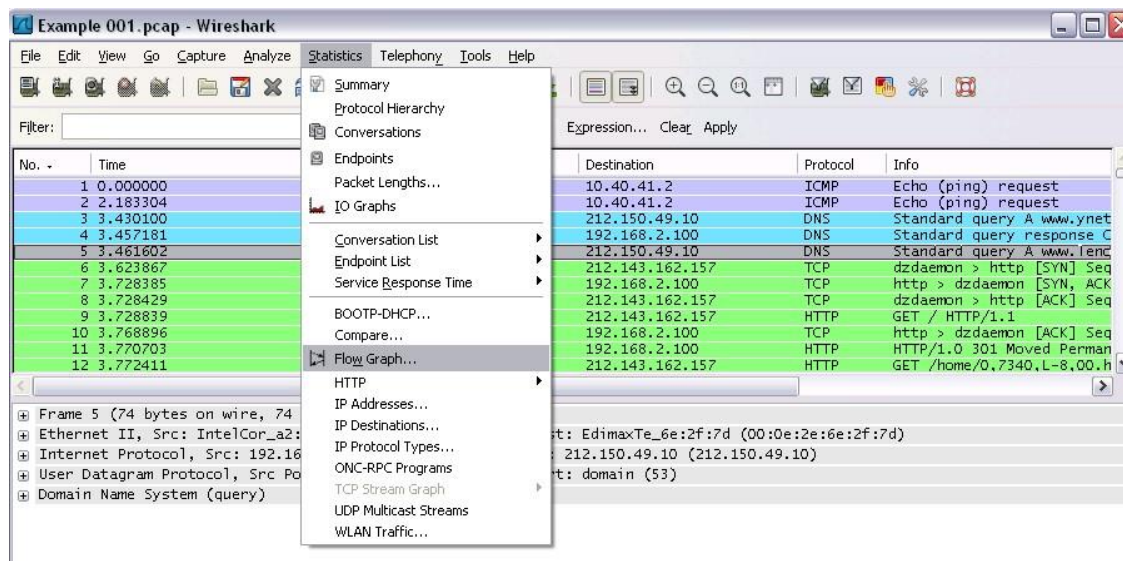
Encapsulation type (frame.encap_type) | Packets: 8136 · Displayed: 21 (0.3%)

You can also create filters from here — just right-click one of the details and use the Apply as Filter submenu to create a filter based on it.



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.

Flow Graph: Gives a better understanding of what we see.



Ex No: 14 b

PACKET SNIFFING USING WIRESHARK


AIM:

To capture, save, filter and analyze network traffic on TCP / UDP / IP / HTTP / ARP /DHCP /ICMP /DNS using Wireshark Tool

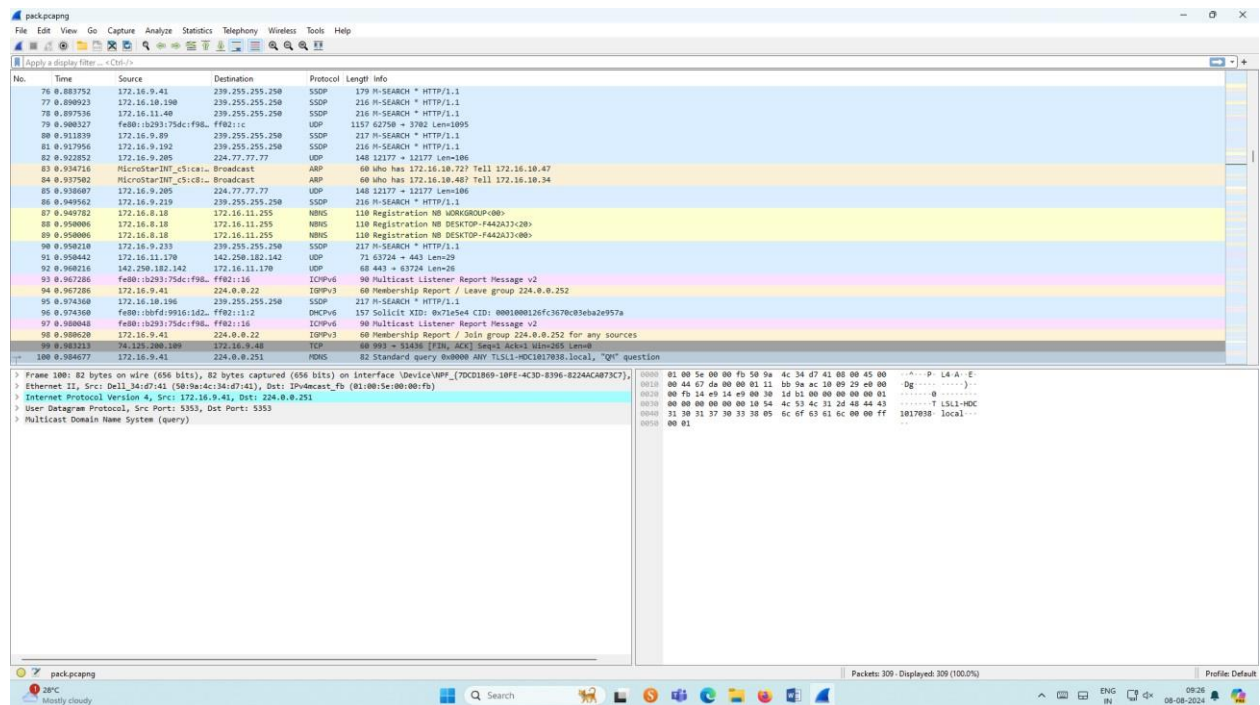
Exercises

1. Capture 100 packets from the Ethernet: IEEE 802.3 LAN Interface and save it.

Procedure



- Select Local Area Connection in Wireshark.
- Go to capture  option
- Select stop capture automatically after 100 packets.
- Then click Start capture. ➤ Save the packets.

Output

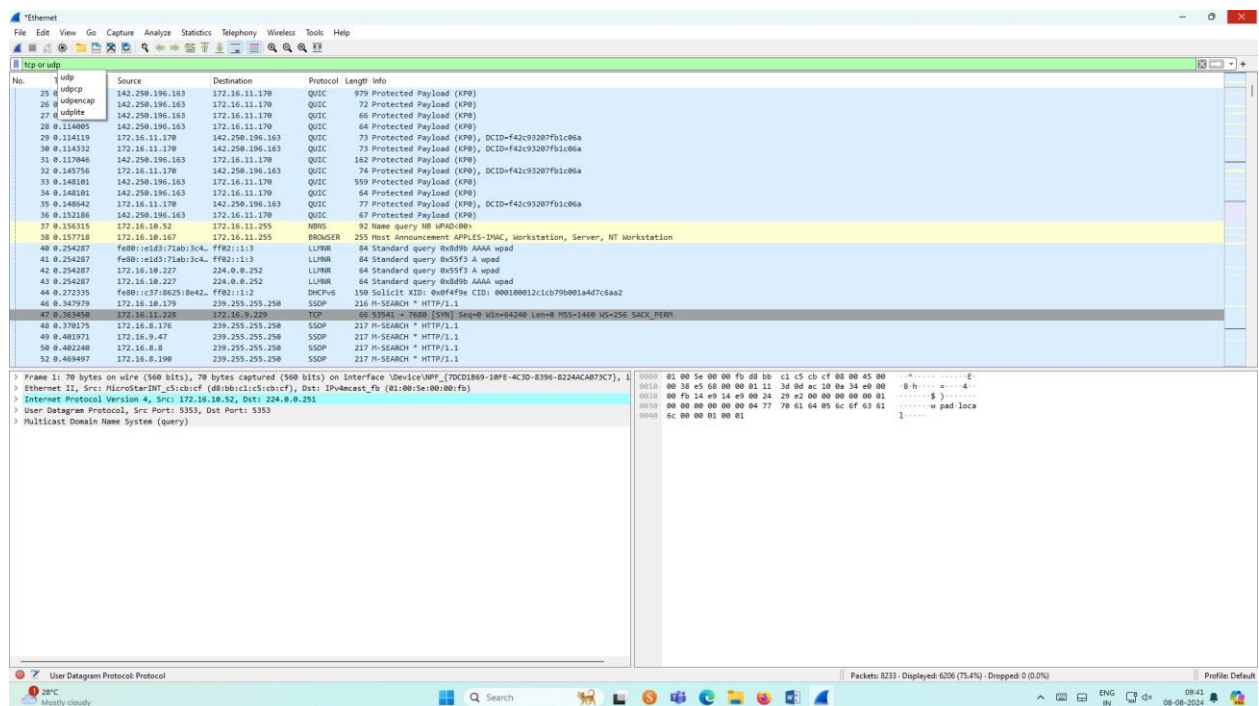


2. Create a Filter to display only TCP/UDP packets, inspect the packets and provide the flow graph.

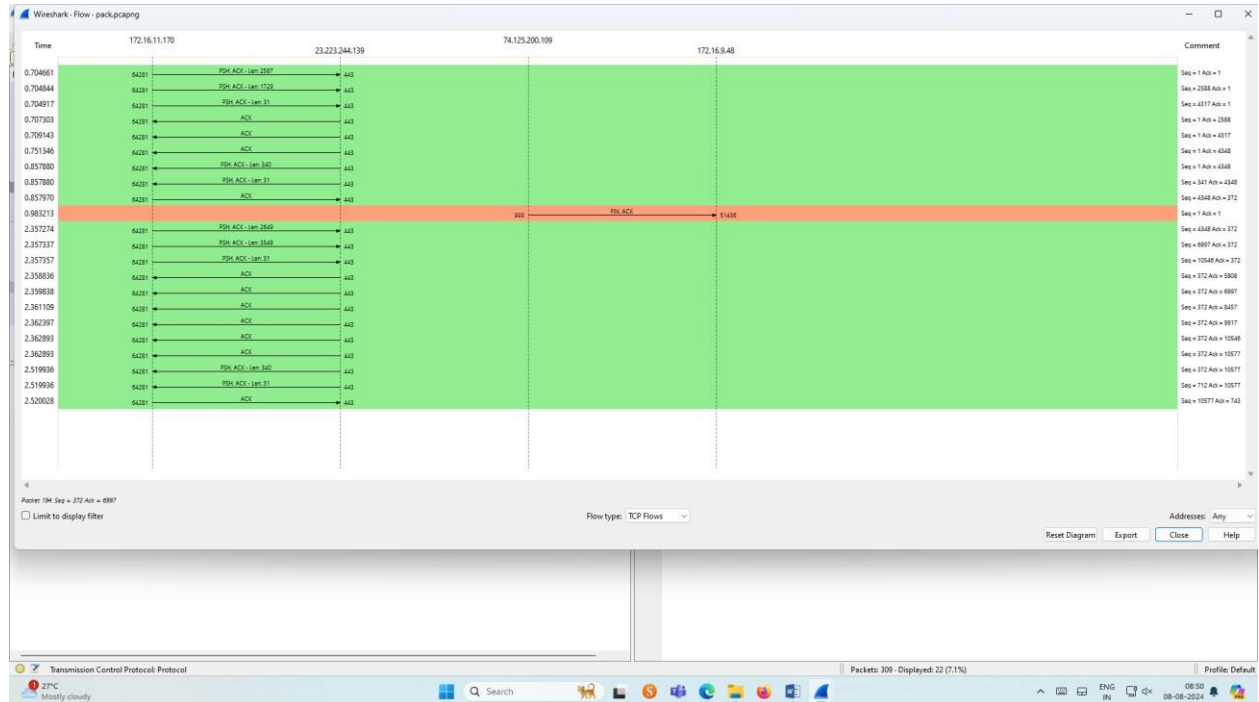
Procedure

- Select Local Area Connection in Wireshark.
- Go to capture  option
- Select stop capture automatically after 100 packets.
- Then click Start capture.
- Search TCP packets in search bar.
- To see flow graph click Statistics  Flow graph. ➤
- Save the packets.

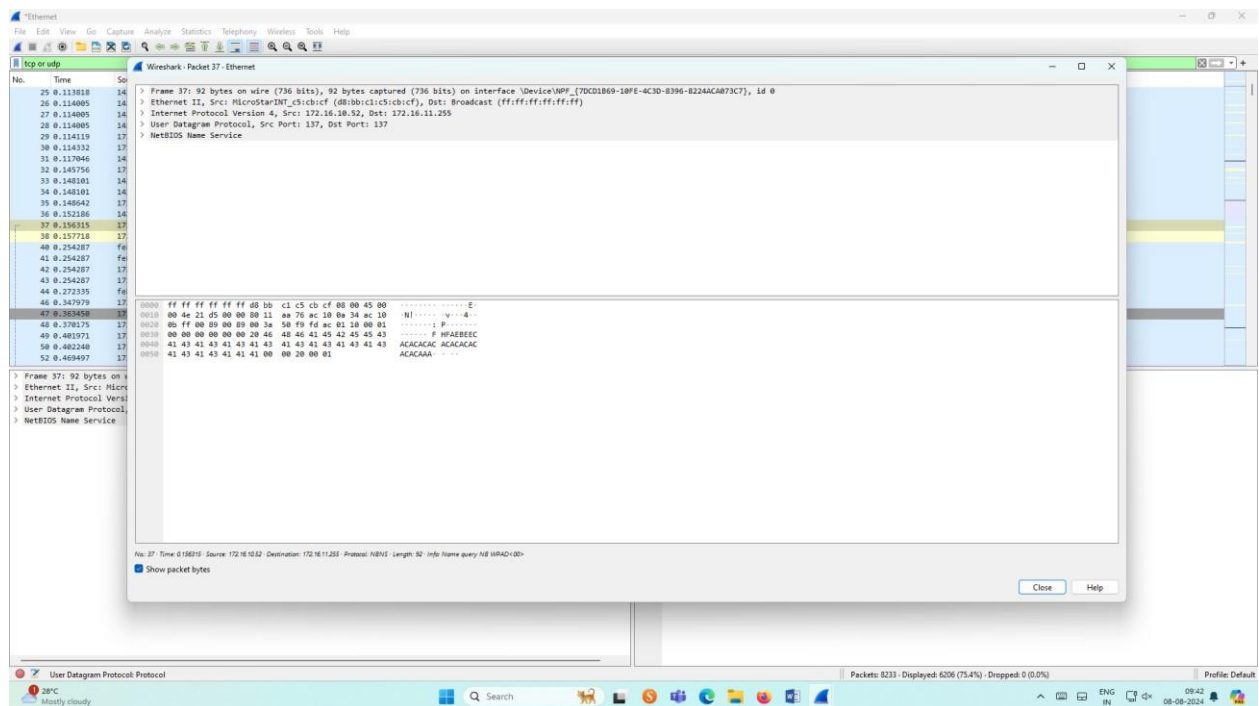
Output:



Flow Graph output




Inspecting the packets

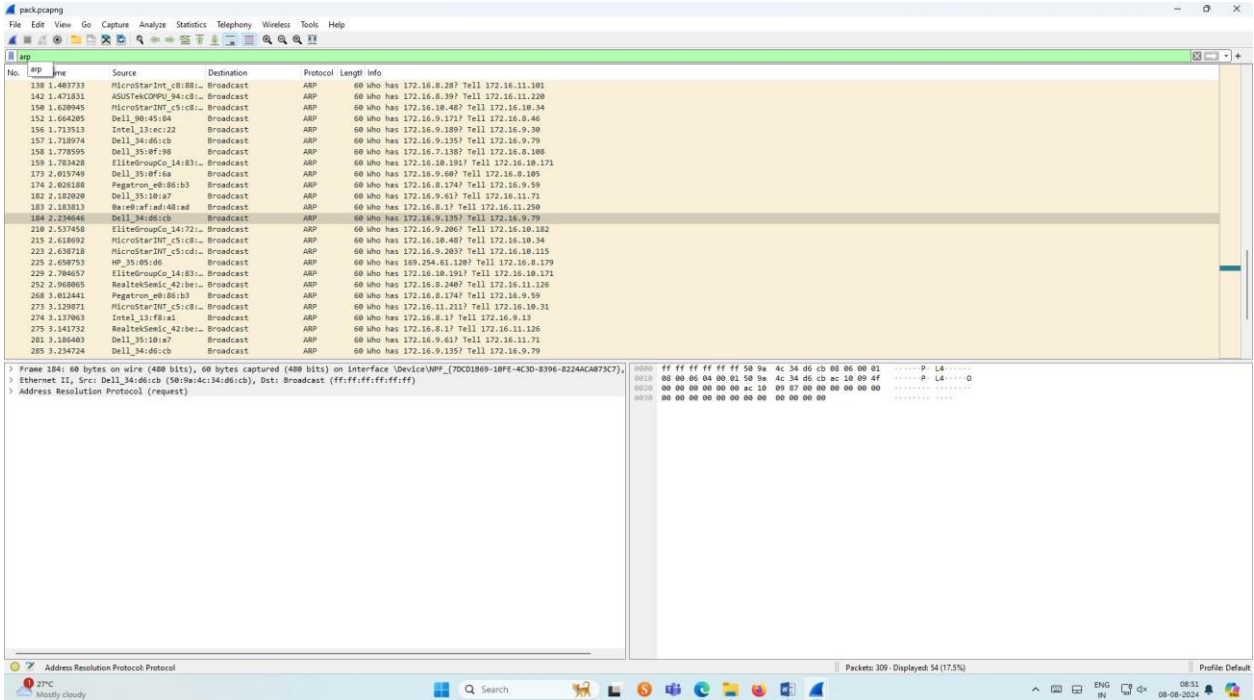


3.Create a Filter to display only ARP packets and inspect the packets.

Procedure

- Select Local Area Connection in Wireshark.
- Go to capture  option
- Select stop capture automatically after 100 packets.
- Then click Start capture.
- Search ARP packets in search bar.
- Save the packets.

Output

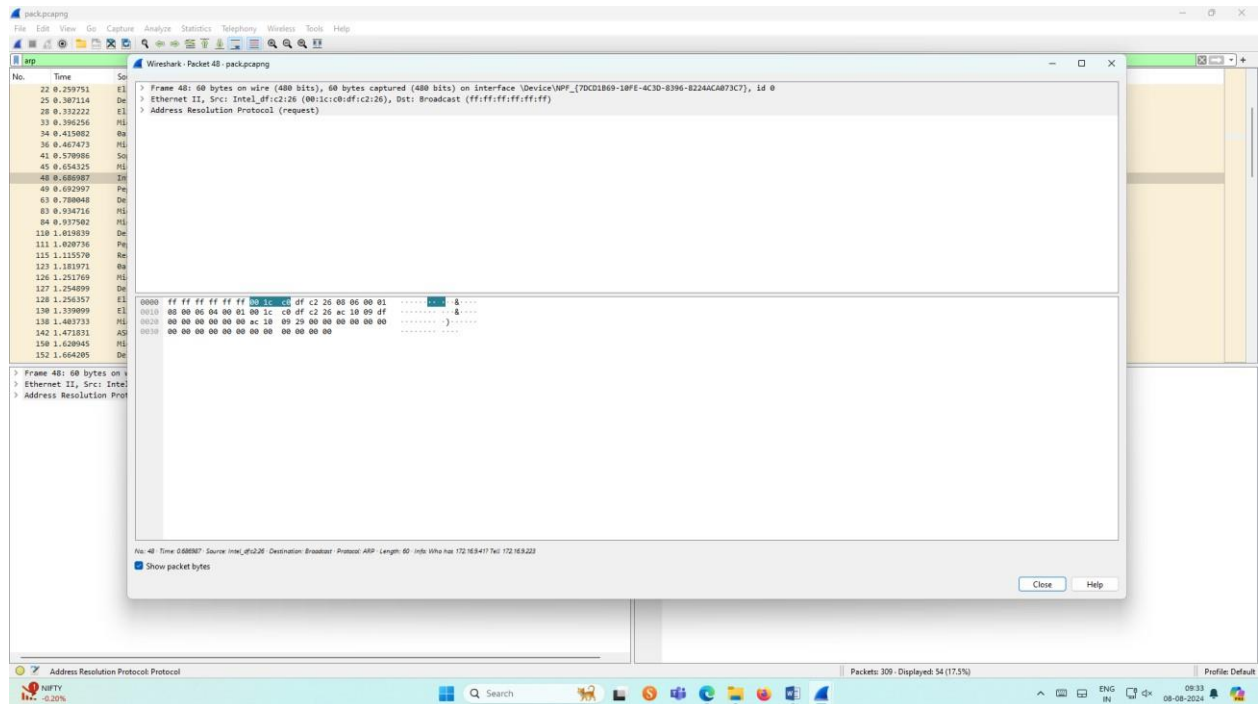


The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains icons for file operations, capture control, and analysis. The main window is divided into three panes:

- Packet List Pane:** Shows a list of captured packets. The first 100 packets are ARP requests, all with source MAC address '08:00:27:00:00:00' and destination MAC address 'ff:ff:ff:ff:ff:ff'. The list includes columns for No., Time, Source, Destination, Protocol, Length, and Info.
- Packet Details Pane:** Displays the details of the selected packet (No. 100). It shows the Ethernet II header (Source: 08:00:27:00:00:00, Destination: ff:ff:ff:ff:ff:ff) and the Address Resolution Protocol (ARP) section (Request).
- Packet Bytes Pane:** Shows the raw data of the selected packet in hexadecimal and ASCII format.



The status bar at the bottom indicates that 309 packets are displayed, representing 17.5% of the total capture. The system tray shows the date and time as 08:51 on 08-08-2024.

Inspecting the packets



4.Create a Filter to display only DNS packets and provide the flow graph.

Procedure

- Select Local Area Connection in Wireshark.
- Go to capture  option
- Select stop capture automatically after 100 packets.
- Then click Start capture.
- Search DNS packets in search bar.
- To see flow graph click Statistics  Flow graph.
- Save the packets.

Output

dns

No.	dns	Source	Destination	Protocol	Length	Info
34		172.16.11.178	172.16.8.1	DNS	79	Standard query 8xc945 A fp-vp.azureedge.net
35	4.824856	172.16.11.178	172.16.8.1	DNS	79	Standard query 8xc945 A fp-vp.azureedge.net
36	4.838791	172.16.8.1	172.16.11.178	DNS	146	Standard query response 8xc945 A fp-vp.azureedge.net CNAME fp-vp.ec.azureedge.net CNAME cs9.xpc.v8cdn.net A 117.18.232.200
37	4.838791	172.16.8.1	172.16.11.178	DNS	146	Standard query response 8xc945 A fp-vp.azureedge.net CNAME fp-vp.ec.azureedge.net CNAME cs9.xpc.v8cdn.net A 117.18.232.200

Frame 373: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NPF_{70CD1869-10FE-4C3D-8396-8224ACAB73C7} 1

Ethernet II, Src: HP_39:1e:d9 (7c:57:58:39:1e:d9), Dst: Sophos_cf:be:45 (7c:5a:1c:cf:be:45)

Internet Protocol Version 4, Src: 172.16.11.178, Dst: 172.16.8.1

User Datagram Protocol, Src Port: 51988, Dst Port: 53

Domain Name System (query)

0000 7c 5a 1c cf be 45 7c 57 58 39 1e d9 08 00 45 00 [Z E]w X9...E

0010 00 41 6d 38 00 00 00 11 00 00 ac 10 0b aa ac 10 -And-

0020 00 01 c3 14 00 15 00 2d 6c 8a c5 45 01 00 00 01 ...S - 1-E....

0030 00 00 00 00 00 05 68 70 2d 76 70 09 61 7a 75f p-vp.azu

0040 72 65 65 64 67 65 63 66 65 74 00 00 01 00 01 reedge n et:....

Domain Name System Protocol

Packets: 1562 - Displayed 4 (0.3%)

Profile: Default

The screenshot displays the Wireshark network protocol analyzer interface. The top section shows a list of captured packets. Packet 29 is selected, which is an ARP request from 172.16.11.107 to 172.16.8.11. The middle pane shows the details of this packet, including the Ethernet II header, the Internet Protocol (IP) header, and the ARP request payload. The bottom pane shows the raw bytes of the packet in hexadecimal and ASCII. The interface is in English, and the packet list is filtered to show only ARP packets.

Select Local Area Connection in Wireshark.

Go to capture  option

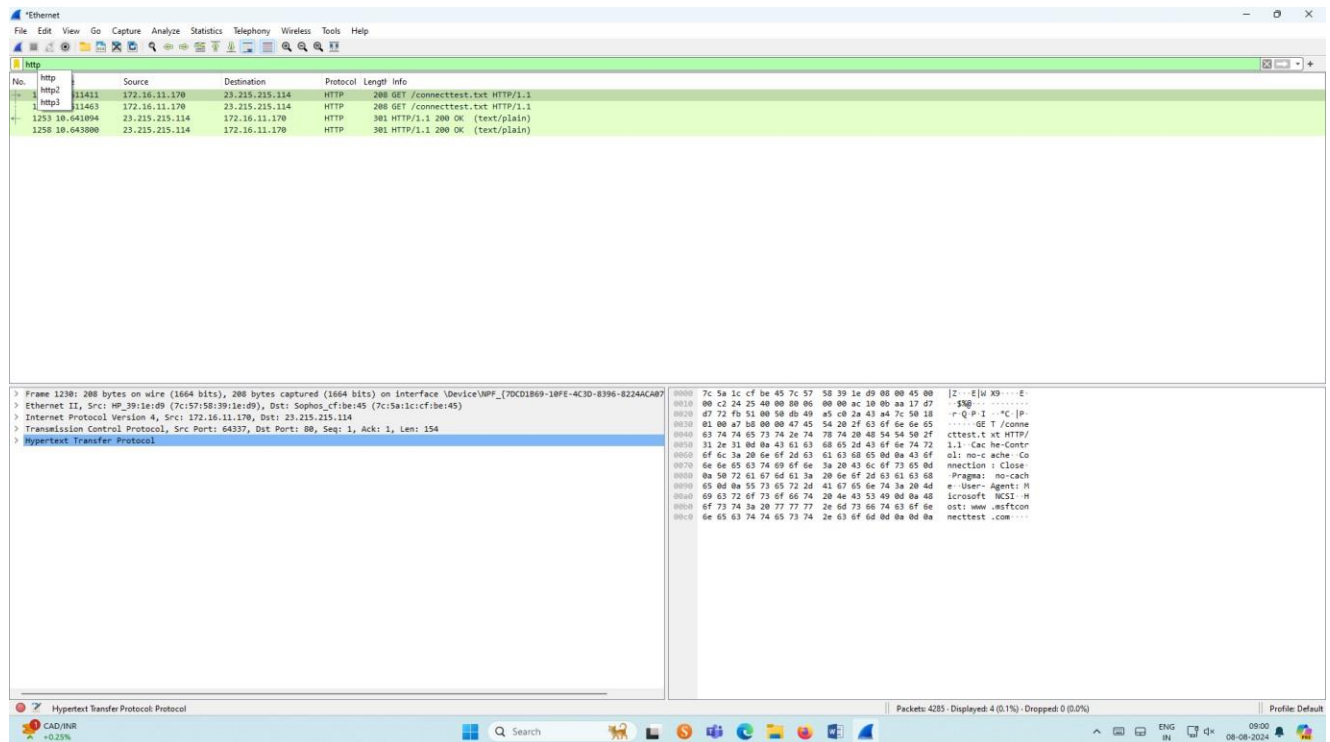
Select stop capture automatically after 100 packets.

Then click Start capture.

➤ Search HTTP packets in the search bar. ➤

Save the packets.

Output



The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, and Tools. The main window is divided into three panes:

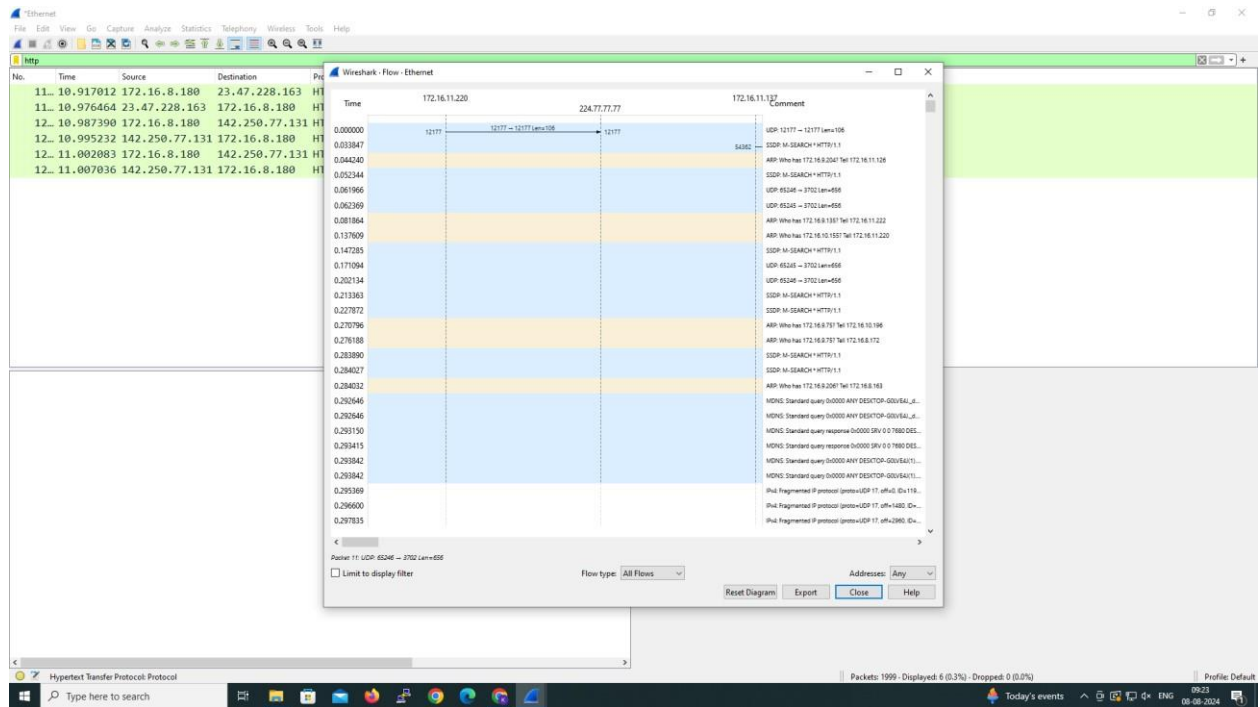
- Packet List:** Shows a list of captured packets. The first three packets are HTTP GET requests to /connecttest.txt. The fourth packet is an HTTP 200 OK response.
- Packet Details:** Provides a hierarchical view of the selected packet (No. 4). It shows the Ethernet II header, Internet Protocol Version 4 header, Transmission Control Protocol header, and the Hypertext Transfer Protocol section.
- Packet Bytes:** Displays the raw data of the selected packet in hexadecimal and ASCII.

The bottom status bar indicates that 4285 packets were captured, 4 were displayed (0.1%), and 0 were dropped (0.0%). The system clock shows 09:00 on 08-08-2024.

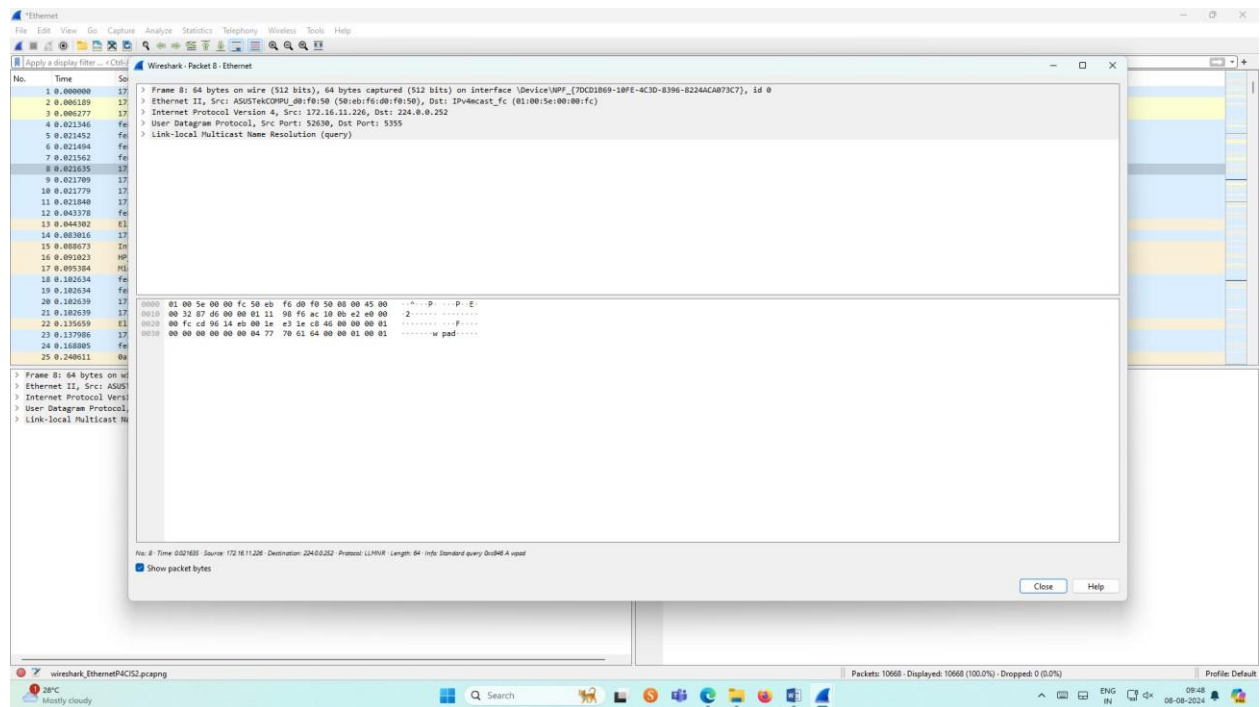
Procedure



Flow Graph output



Inspecting the packets



6. Create a Filter to display only IP/ICMP packets and inspect the packets.

Select Local Area Connection in Wireshark.

Go to capture  option

Select stop capture automatically after 100 packets.

Then click Start capture.

- Search ICMP/IP packets in search bar.
- Save the packets

Output

File Edit View Go Capture Analyze Statistics Telephony Windows Tools Help

icmp or ping

No.	Time	Source	Destination	Protocol	Length	Info
306	3.389477	142.250.182.78	172.16.11.170	UDP	70	443 → 65183 Len=28
307	3.390806	172.17.217.31.282	172.16.11.170	UDP	69	443 → 50415 Len=27
309	3.415817	142.250.182.78	172.16.11.170	UDP	66	443 → 65183 Len=28
16	0.197335	172.16.8.18	172.16.11.255	NBNS	110	Registration NB DESKTOP-F442A73-080
17	0.197335	172.16.8.18	172.16.11.255	NBNS	110	Registration NB DESKTOP-F442A73-280
18	0.197335	172.16.8.18	172.16.11.255	NBNS	110	Registration NB WORKGROUP-080
35	0.467287	172.16.8.18	172.16.11.255	NBNS	92	Name query NB DESKTOP-H93F858-1c1
87	0.949782	172.16.8.18	172.16.11.255	NBNS	110	Registration NB WORKGROUP-080
88	0.950806	172.16.8.18	172.16.11.255	NBNS	110	Registration NB DESKTOP-F442A73-280
89	0.950806	172.16.8.18	172.16.11.255	NBNS	110	Registration NB DESKTOP-F442A73-080
92	1.217692	172.16.8.18	172.16.11.255	NBNS	92	Name query NB DESKTOP-H93F858-1c1
295	3.351537	172.16.9.74	172.16.11.255	UDP	186	60808 → 51807 Len=144
300	3.389313	172.16.11.170	172.16.8.18	TCP	60	893 → 34368 [RST,ACK] Seq=1 Ack=1 Win=255 Len=0
304	3.386154	172.16.11.170	172.17.217.31.282	UDP	1288	50415 → 443 Len=1246
305	3.386282	172.16.11.170	172.17.217.31.282	UDP	994	50415 → 443 Len=992
94	0.907286	172.16.9.41	224.0.0.22	IGMPv3	60	Membership Report / Leave group 224.0.0.252
98	0.908628	172.16.9.41	224.0.0.22	IGMPv3	60	Membership Report / Join group 224.0.0.252 for any sources
139	1.461079	172.16.9.41	224.0.0.22	IGMPv3	60	Membership Report / Join group 224.0.0.252 for any sources
167	1.912286	169.254.0.14	224.0.0.22	IGMPv3	60	Membership Report / Leave group 224.0.0.252 for any sources
168	1.912286	169.254.0.14	224.0.0.22	IGMPv3	60	Membership Report / Join group 224.0.0.252 for any sources
231	2.768294	169.254.0.14	224.0.0.22	IGMPv3	60	Membership Report / Join group 224.0.0.252 for any sources
233	2.768149	169.254.0.14	224.0.0.22	IGMPv3	60	Membership Report / Leave group 224.0.0.252
256	2.954657	172.16.9.41	224.0.0.22	IGMPv3	60	Membership Report / Leave group 224.0.0.252
254	2.970187	172.16.9.41	224.0.0.22	IGMPv3	60	Membership Report / Join group 224.0.0.252 for any sources
26	0.327343	172.16.8.225	224.0.0.251	NBNS	85	Standard query 0x0000 PTR microsoft_mcc_tcp_local, "QI" question

Frame 4: 71 bytes on wire (568 bits), 71 bytes captured (568 bits) on interface vnicuppp7 (70C1D89-18FE-4C3D-8396-E224ACAB73C7), 1 interface 11, Src: HP 30c1a6 (7c57:58:10:0:0:0), Dst: Sophos cf4bfe45 (7c5a:1c:cf:be:45:7c) 58 39 1e 49 08 00 45 00 | 2 - E | u X9 - - E -

Internet Protocol Version 4, Src: 172.16.11.170, Dst: 142.250.182.142

User Datagram Protocol, Src Port: 63724, Dst Port: 443

Data (29 bytes)

0000 7c 5a 1c cf be 45 7c 57 58 39 1e 49 08 00 45 00 | 2 - E | u X9 - - E -

0010 00 39 c8 10 40 00 08 11 00 00 ec 10 0a 0e fe | 9 - |

0020 b6 8e f8 ec 01 b0 00 25 fd 79 52 e2 03 d1 c1 57 | X yd - - W

0030 a5 fa 90 c7 c0 fe 8f 70 71 01 01 92 bd cd d3 ab | ((((((

0040 1f 71 17 01 94 70 34 | q - - - - -

Internet Protocol Version 4 Protocol

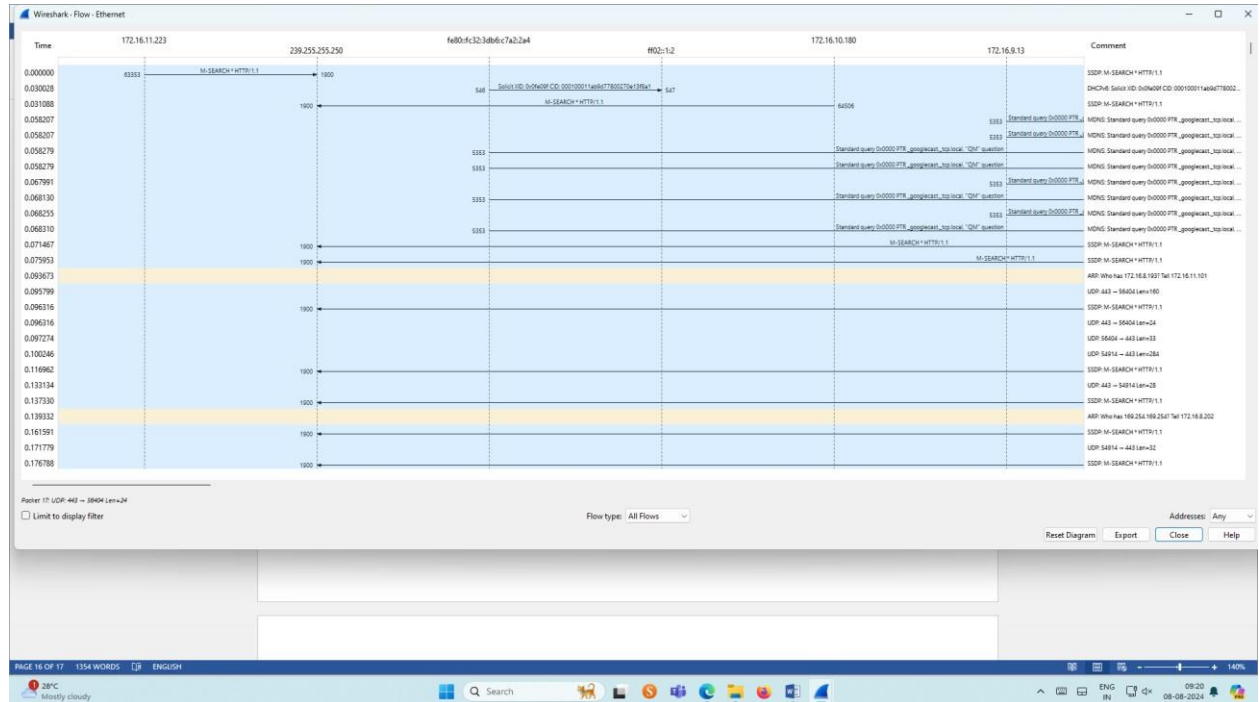
ENC -

08-08-2024

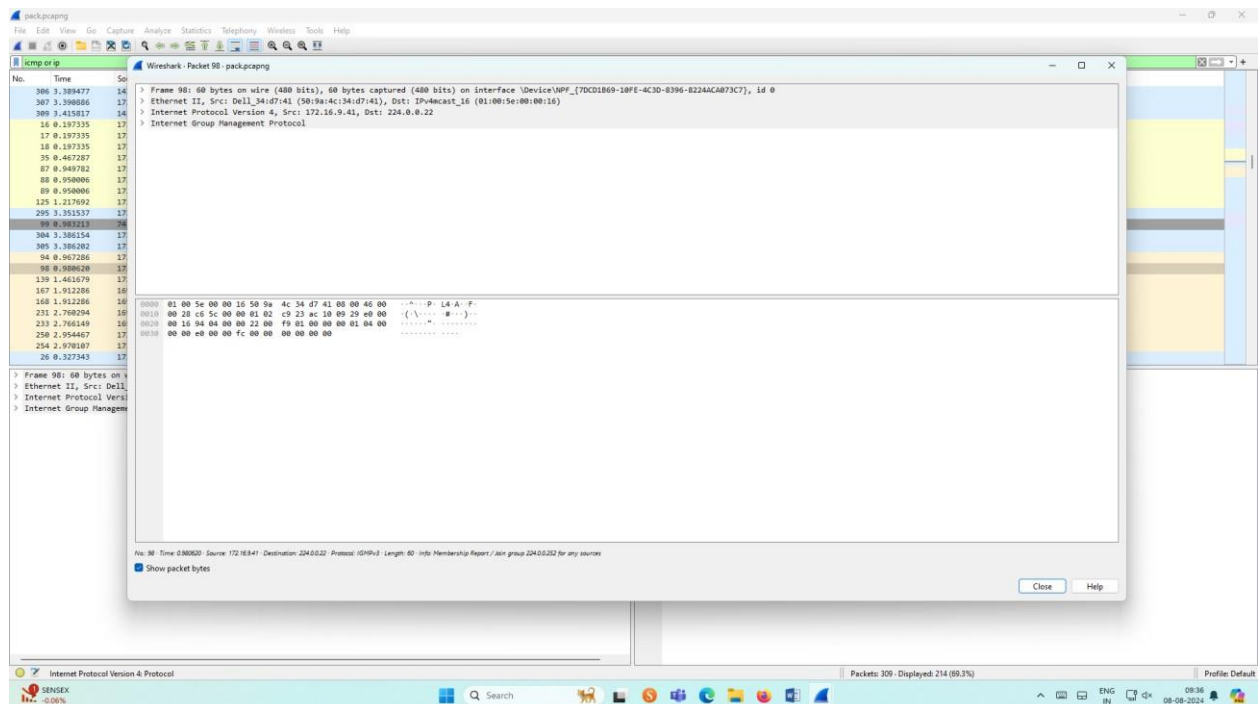
Packets: 309 - Displayed: 214 (69.3%)

Profile Default

Flow Graph output



Inspecting the packets




Procedure



7. Create a Filter to display only DHCP packets and inspect the packets.

Select Local Area Connection in Wireshark.

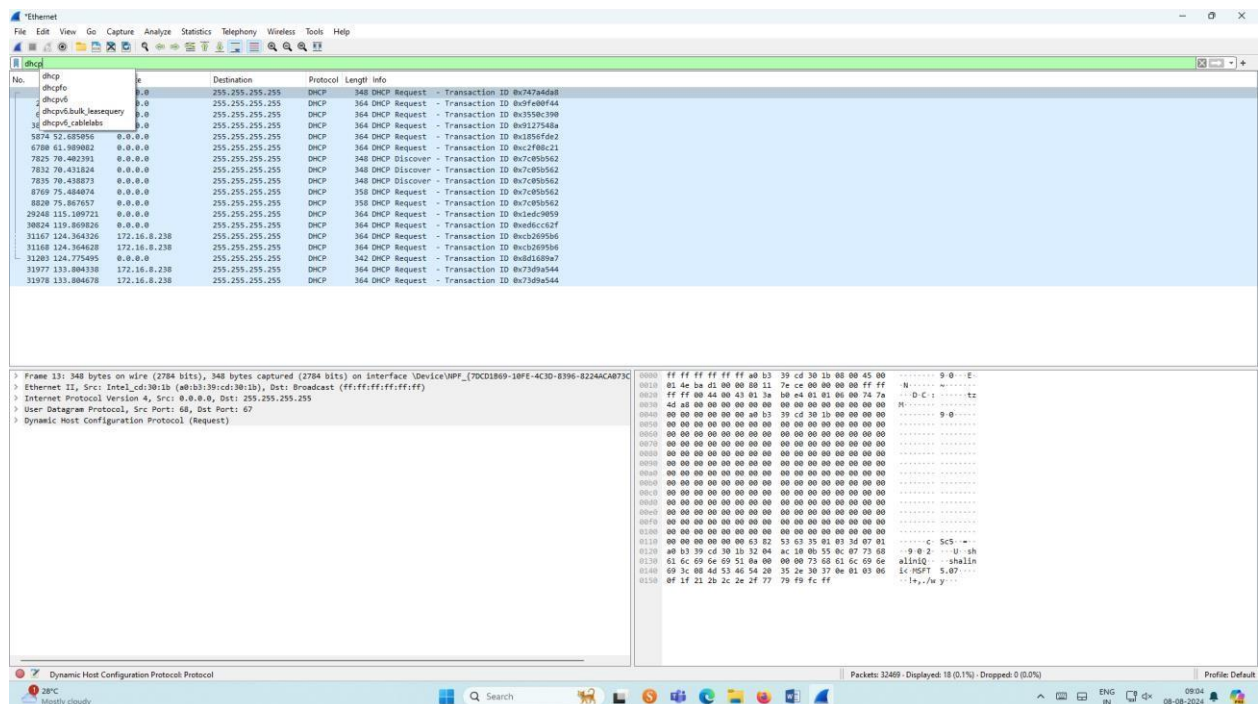
Go to capture  option

Select stop capture automatically after 100 packets.

Then click Start capture.

- Search DHCP packets in search bar.
- Save the packets

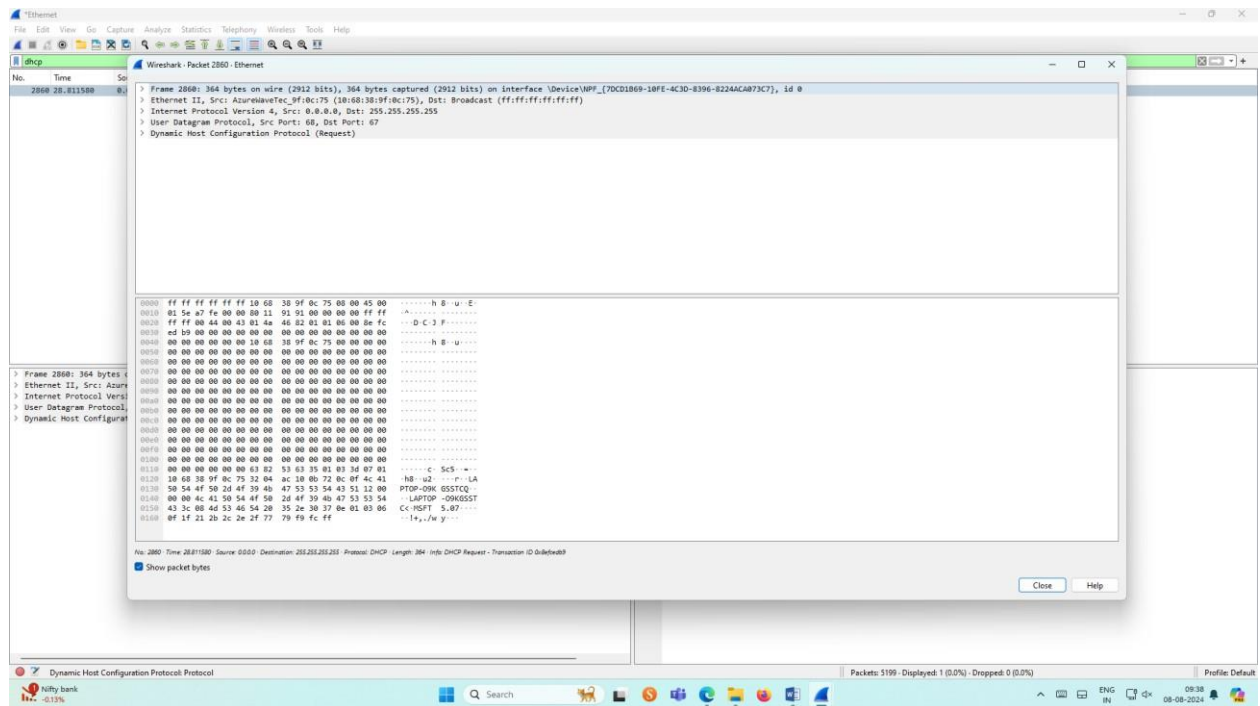
Output



The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains icons for various functions like opening files, saving, and capturing. The main window is divided into three panes:

- Packet List:** Shows a list of captured packets. The first few packets are DHCP requests and discoveries. The columns are No., Time, Source, Destination, Protocol, Length, and Info.
- Packet Details:** Shows the hierarchical structure of the selected packet (Frame 13). It includes Ethernet II, Internet Protocol Version 4, User Datagram Protocol, and Dynamic Host Configuration Protocol (Request).
- Packet Bytes:** Shows the raw data of the selected packet in hexadecimal and ASCII format.

The status bar at the bottom indicates that 12400 packets were captured, 18 (0.1%) were displayed, and 0 (0.0%) were dropped. The profile is set to Default.



**RESULT:
THE GIVEN**