

Ex No: 14a 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:

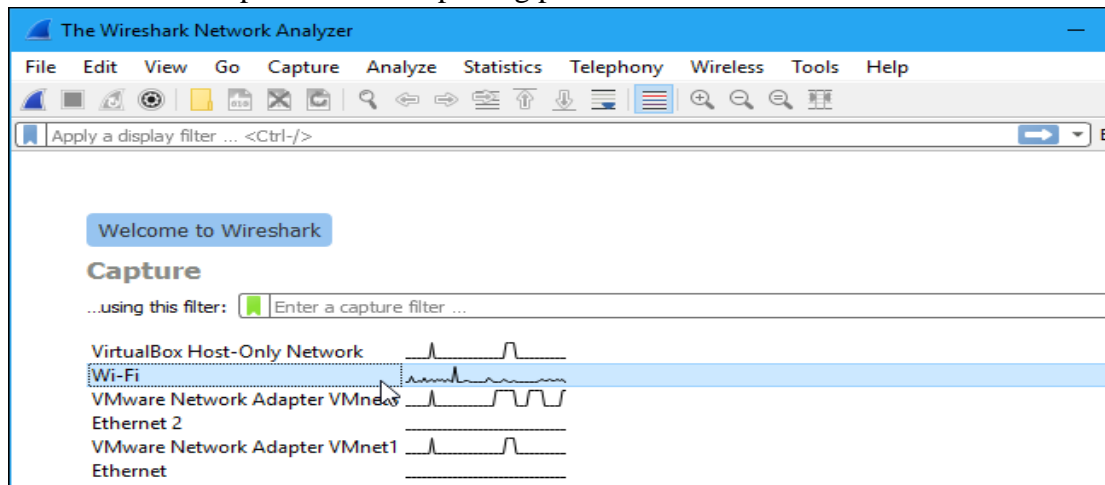
- Network administrators: troubleshoot network problems
- Network security engineers: examine security problems
- Developers: debug protocol implementations
- People: learn **network protocol internals**

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.

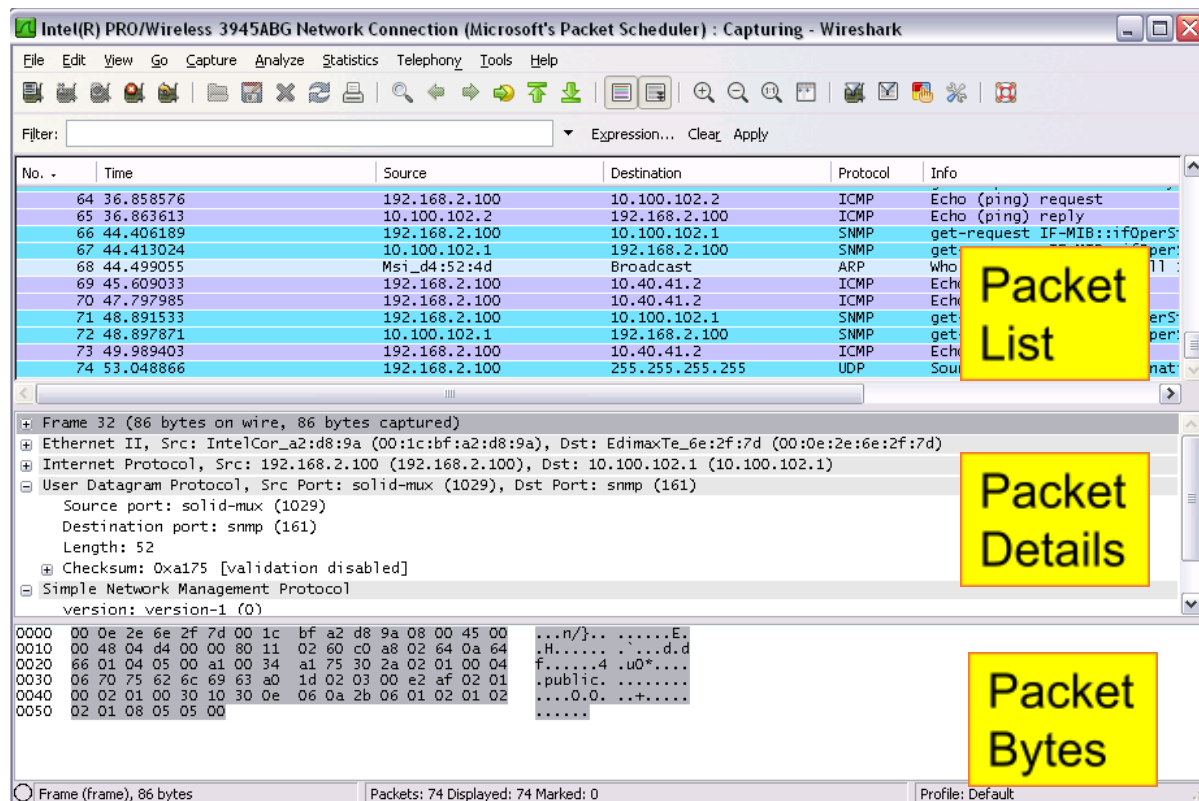
Capturing Packets

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” and “Packet Bytes” panes.

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.

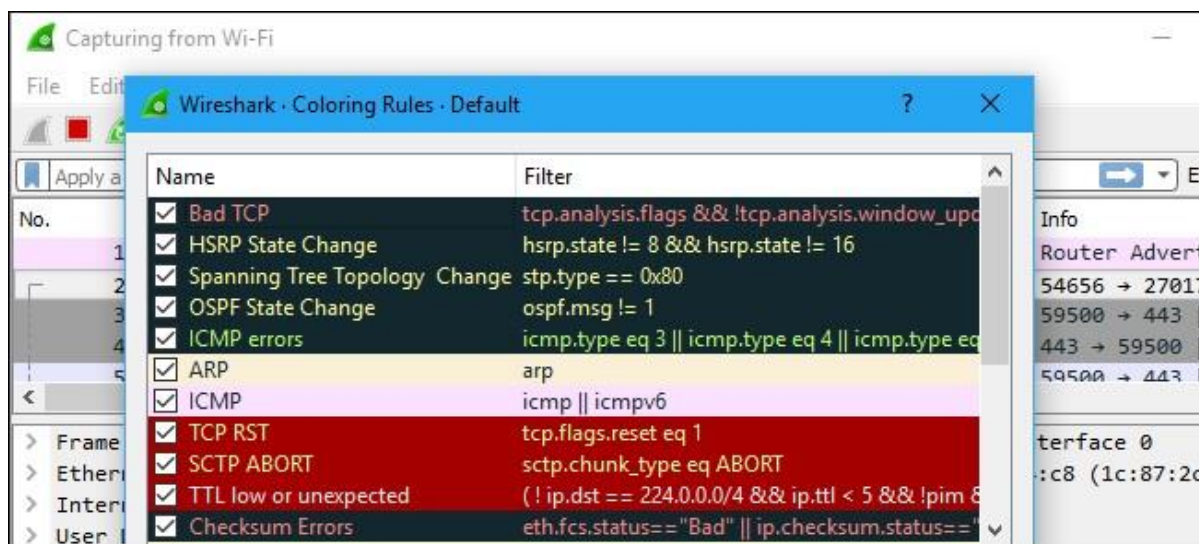
The “Packet Bytes” Pane

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.

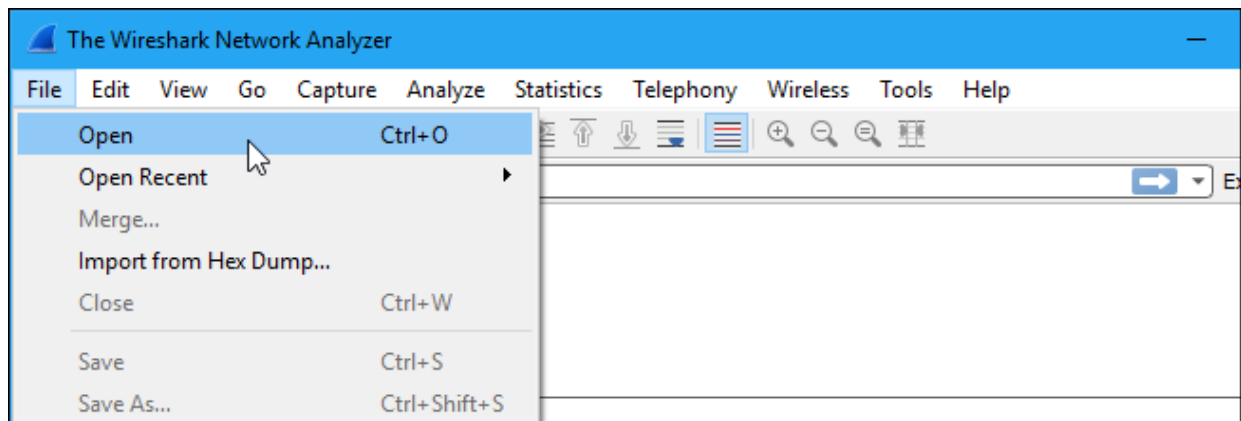
To view exactly what the color codes mean, click View > Coloring Rules. You can also customize and modify the coloring rules from here, if you like.



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.

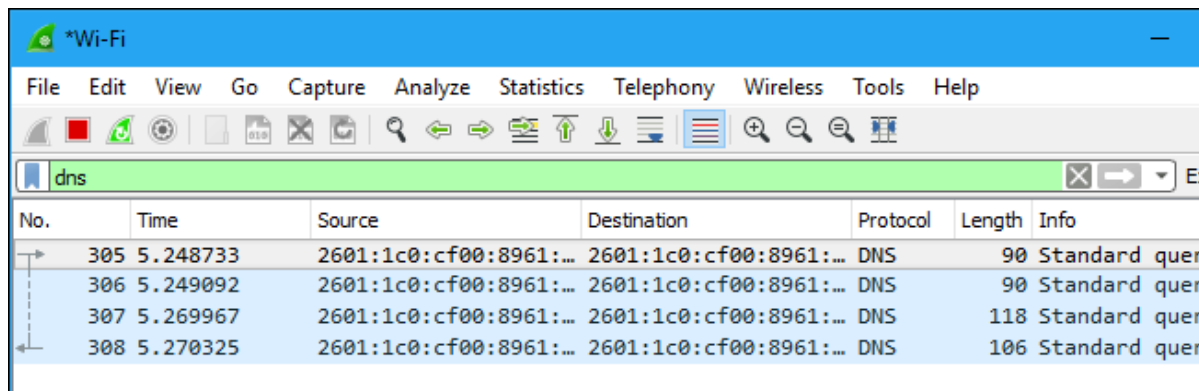
You can also save your own captures in Wireshark and open them later. Click File > Save to save your captured packets.



Filtering Packets

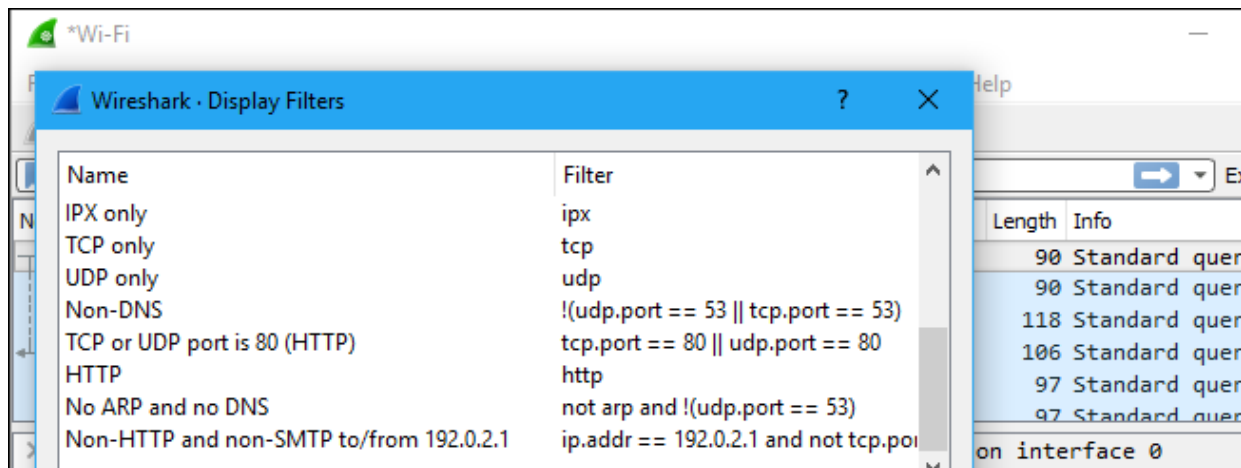
If you're trying to inspect something specific, such as the traffic a program sends when phoning home, it helps to close down all other applications using the network so you can narrow down the traffic. Still, you'll likely have a large amount of packets to sift through. That's where Wireshark's filters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



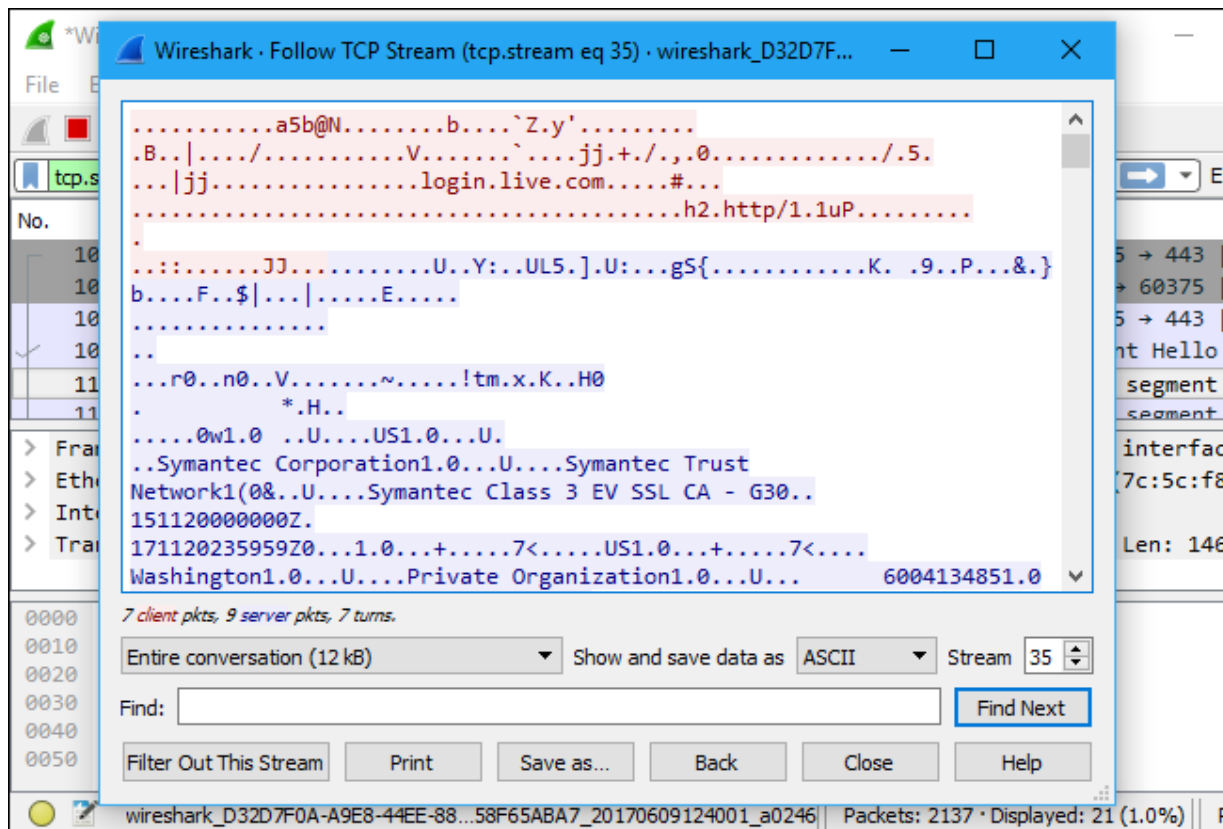
You can also click Analyze > Display Filters to choose a filter from among the default filters included in Wireshark. From here, you can add your own custom filters and save them to easily access them in the future.

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

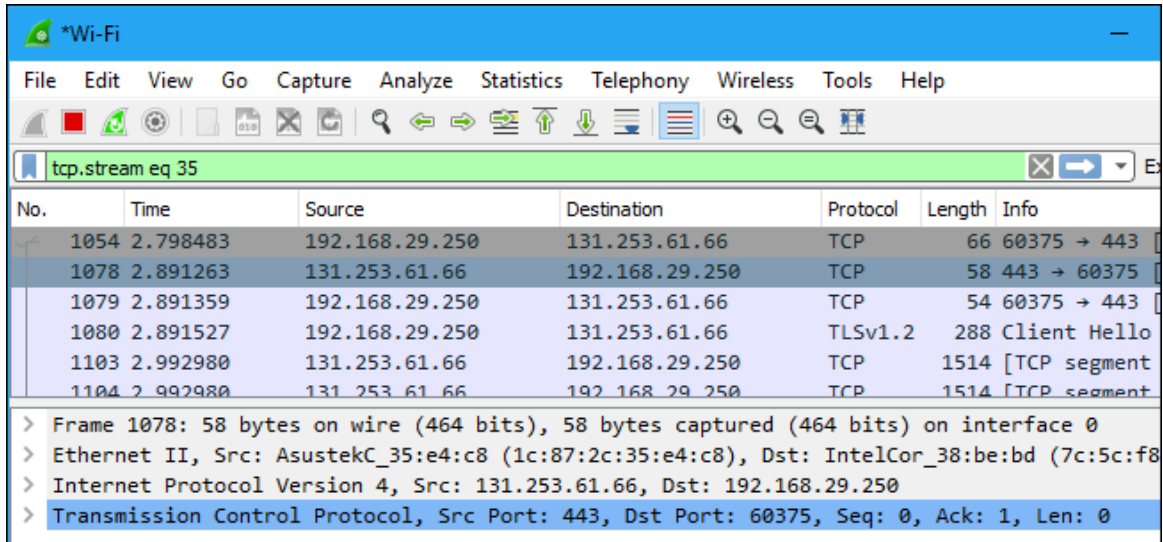


Another interesting thing you can do is right-click a packet and select Follow > TCP Stream.

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.



The image shows the Wireshark interface with a filter 'tcp.stream eq 35' applied. The packet list shows several packets, with packet 1078 selected. The details pane for packet 1078 is expanded, showing the frame, Ethernet II header, Internet Protocol Version 4 header, and Transmission Control Protocol header.

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.

The image shows the Wireshark network protocol analyzer interface. The title bar indicates the capture is on a Wi-Fi interface. The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains icons for various functions like capture, analysis, and display. The packet list pane shows a table of captured packets:

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

The details pane for frame 1054 is expanded, showing the following information:

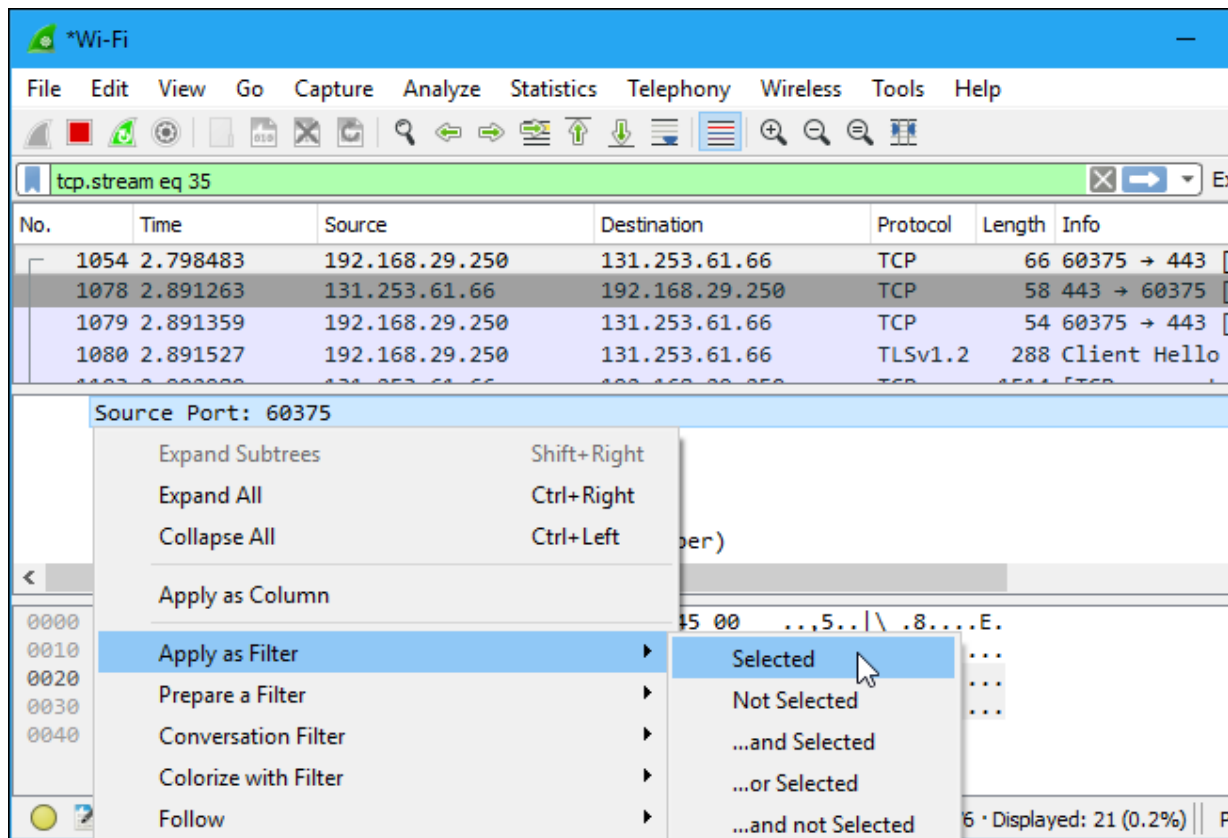
- Frame 1054: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
- Interface id: 0 (\Device\NPF_{D32D7F0A-A9E8-44EE-88DC-DFD58F65ABA7})
- Encapsulation type: Ethernet (1)
- Arrival Time: Jun 9, 2017 12:40:04.140141000 Pacific Daylight Time
- [Time shift for this packet: 0.000000000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

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

```
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.]@... O.....
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  ..
```

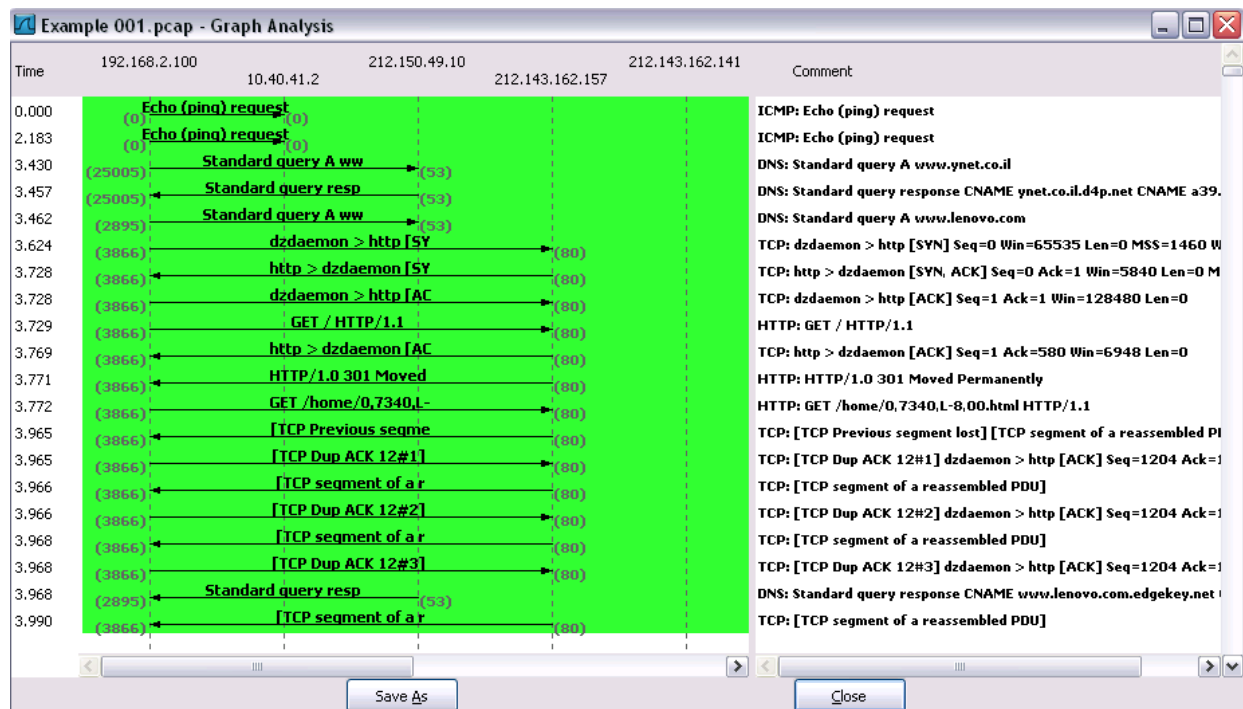
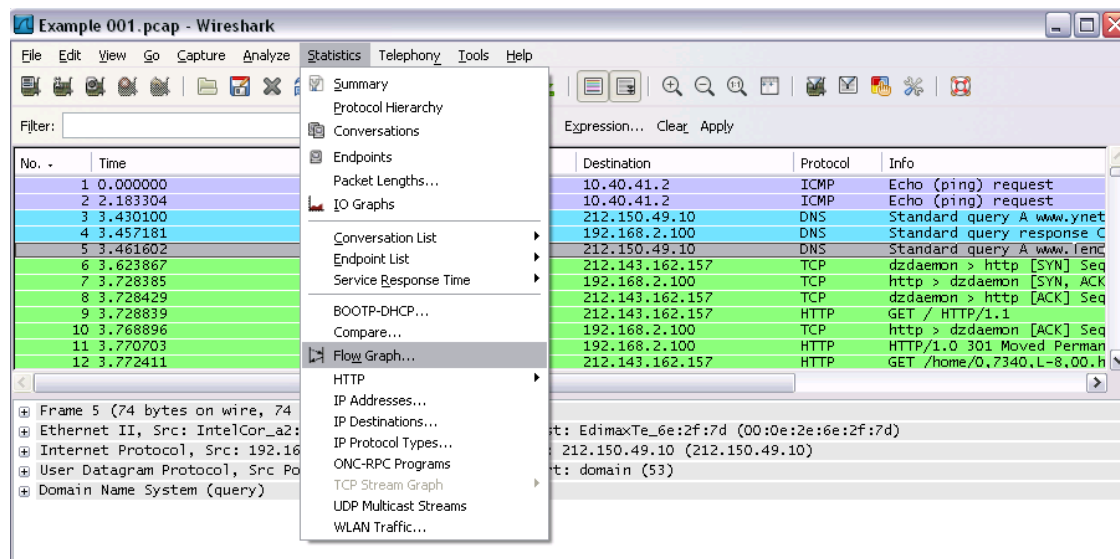
The status bar at the bottom shows the filter 'Encapsulation type (frame.encap_type)' and statistics: 'Packets: 8136 · Displayed: 21 (0.3%)'.

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

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. Below the menu is a toolbar with icons for common actions. The main window is divided into three panes: Packet List, Packet Details, and Packet Bytes.

Packet List Pane: Shows a list of 100 captured packets. The columns are No., Time, Source, Destination, Protocol, Length, and Info. The list includes various protocols such as DNS (Standard query response, Standard query), ARP (who has), ICMP (Echo (ping)), and UDP (NTP).

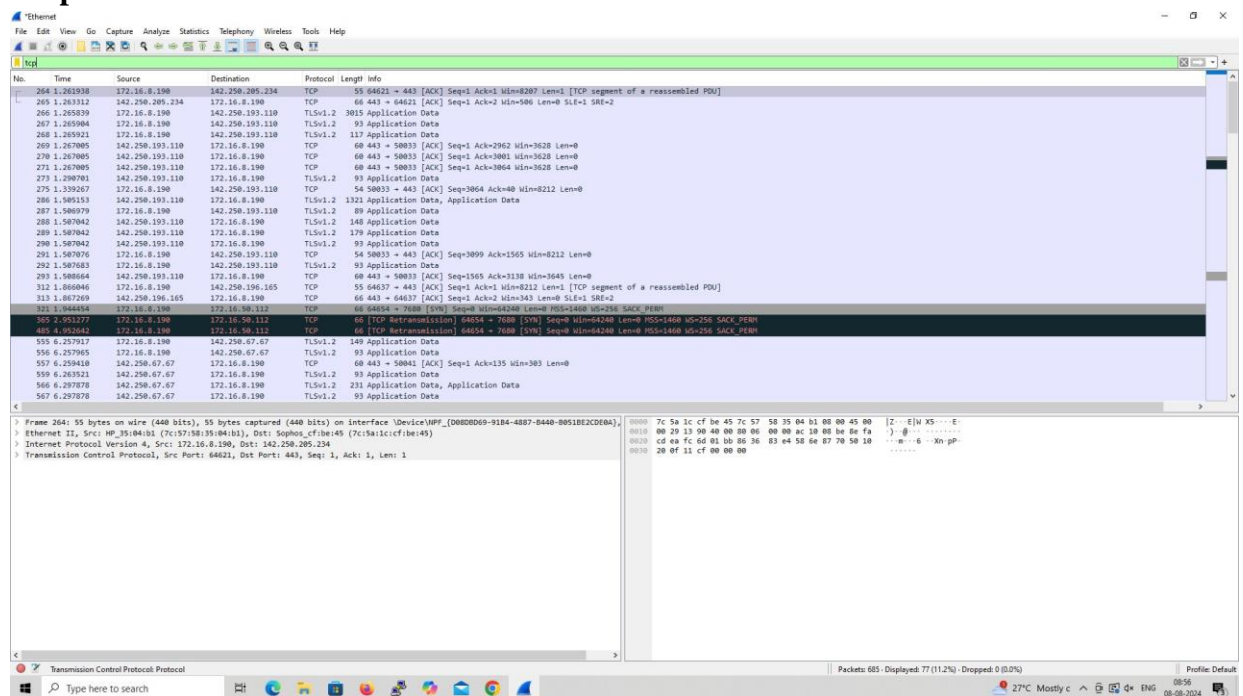
Packet Details Pane: Shows the hierarchical structure of the selected packet (No. 100). The tree view includes Ethernet II (Type: IPv4), Internet Protocol Version 4 (Source: 172.16.10.171, Destination: 172.16.10.171), and User Datagram Protocol (Source Port: 56919, Destination Port: 1900).

Packet Bytes Pane: Shows the raw data of the selected packet in hexadecimal and ASCII format. The data is displayed in a table with columns for offset, hexadecimal, and ASCII.

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.

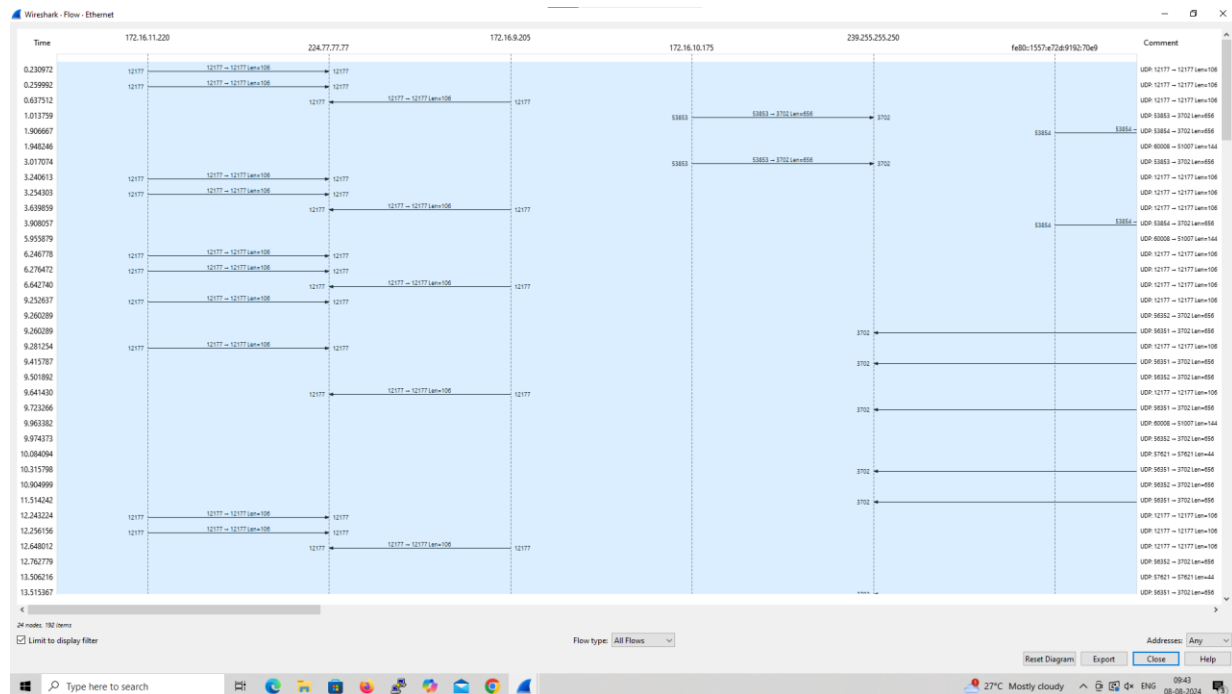
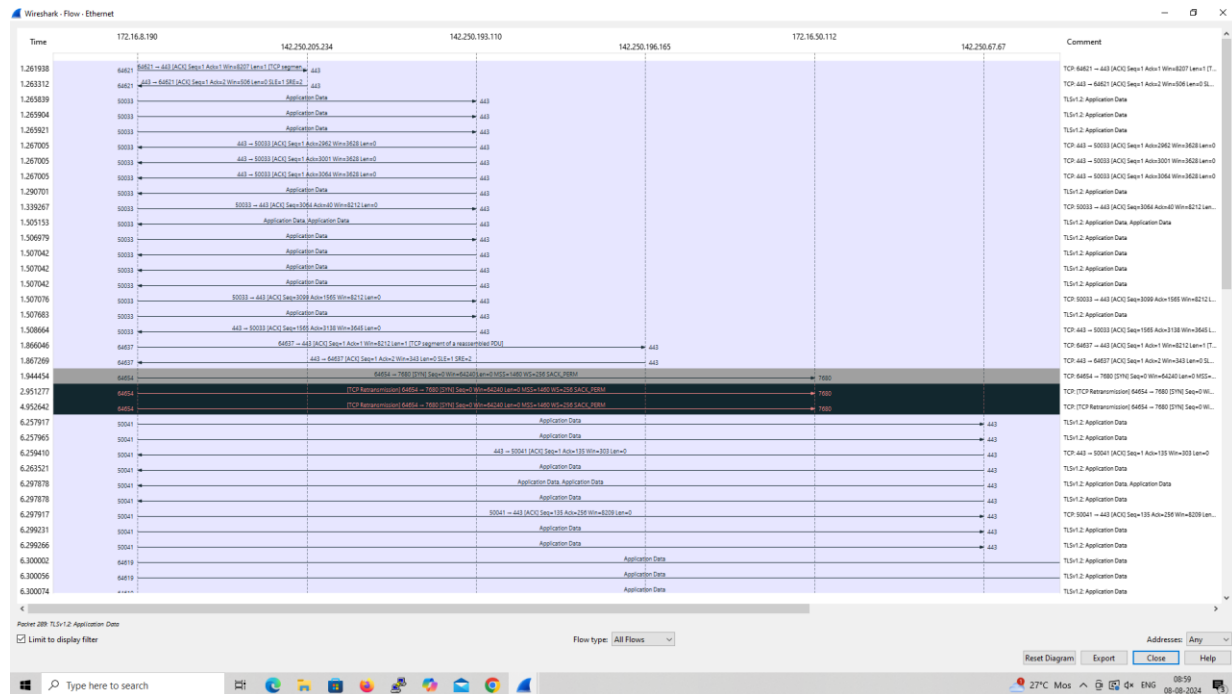
Output:



The image shows a Wireshark packet capture window. The top pane displays a list of captured packets, all of which are UDP packets from 172.16.11.220 to 224.77.77.77. The middle pane shows the details of the selected packet (No. 1902), including Ethernet II, Internet Protocol Version 4, User Datagram Protocol, and Data (106 bytes). The bottom pane shows the raw packet data in hexadecimal and ASCII. The status bar at the bottom indicates that 3154 packets are displayed, representing 0.2% of the capture.

The image shows a detailed view of packet 1196 in Wireshark. The packet is an Ethernet II frame from ASUSTeKComputer (08:0b:b0:c3:65:65) to IPoEcast_4d:4d:4d (01:00:5e:dd:dd:dd). It is an Internet Protocol Version 4 packet from 172.16.9.74 to 224.77.77.77. The User Datagram Protocol section shows a source port of 60008 and a destination port of 51007. The data section contains 144 bytes of raw data. The status bar at the bottom indicates that the packet is 1196 bytes long, with a source of 172.16.9.74, a destination of 224.77.77.77, a protocol of UDP, a length of 106, and information about the ports 60008 and 51007.

Flow Graph output

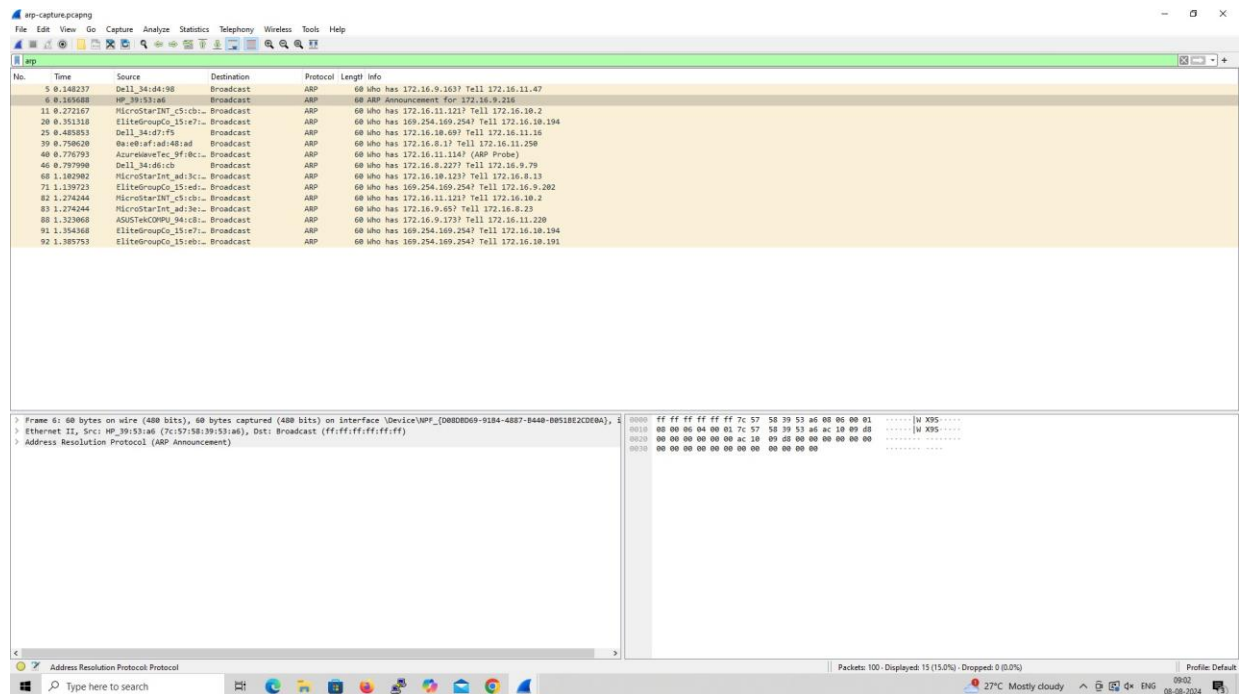


3.Create a Filter to display only ARP packets and inspect the packets.

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



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

The screenshot displays the Wireshark network protocol analyzer interface. The top pane shows a list of captured packets, with DNS queries and responses highlighted. The middle pane shows the details of the selected packet (No. 1521), including Ethernet II, Internet Protocol Version 4, User Datagram Protocol, and Domain Name System (query). The bottom pane shows the raw packet data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
1521	15.175484	172.16.8.190	172.16.8.1	DNS	78	Standard query 0x1ad A edge.microsoft.com
1522	15.175597	172.16.8.190	172.16.8.1	DNS	78	Standard query 0xb0b0 HTTPS edge.microsoft.com
1524	15.177324	172.16.8.1	172.16.8.190	DNS	181	Standard query response 0x1ad A edge.microsoft.com CNAME edge-microsoft-com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedge.net A 13.107.21.239 A 204.79.197.239
1525	15.177324	172.16.8.1	172.16.8.190	DNS	149	Standard query response 0xb0b0 HTTPS edge.microsoft.com CNAME edge-microsoft-com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedge.net
1590	15.170866	172.16.8.190	172.16.8.1	DNS	78	Standard query 0xb0b9 A edge.microsoft.com
1591	15.171175	172.16.8.190	172.16.8.1	DNS	78	Standard query 0xb05d HTTPS edge.microsoft.com
1592	15.171940	172.16.8.1	172.16.8.190	DNS	181	Standard query response 0xb2e9 A edge.microsoft.com CNAME edge-microsoft-com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedge.net A 13.107.21.239 A 204.79.197.239
1593	15.171940	172.16.8.1	172.16.8.190	DNS	149	Standard query response 0xb05d HTTPS edge.microsoft.com CNAME edge-microsoft-com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedge.net
1606	15.172996	172.16.8.190	172.16.8.1	DNS	92	Standard query 0xb2ef A edge-mobile-static.azureedge.net
1607	15.170154	172.16.8.190	172.16.8.1	DNS	92	Standard query 0xb453 HTTPS edge-mobile-static.azureedge.net
1680	15.171030	172.16.8.1	172.16.8.190	DNS	245	Standard query response 0xb453 HTTPS edge-mobile-static.azureedge.net CNAME edge-mobile-static.afd.azureedge.net CNAME azureedge-t-prod.trafficmanager.net CNAME shed.dual-low-s-part-0030.
1689	15.171030	172.16.8.1	172.16.8.190	DNS	261	Standard query response 0xb2ef A edge-mobile-static.azureedge.net CNAME edge-mobile-static.afd.azureedge.net CNAME azureedge-t-prod.trafficmanager.net CNAME shed.dual-low-s-part-0030.t-00
1629	15.170850	172.16.8.190	172.16.8.1	DNS	72	Standard query 0xd9d A www.bing.com
1630	15.170899	172.16.8.190	172.16.8.1	DNS	72	Standard query 0xb3e7 HTTPS www.bing.com
1631	15.171570	172.16.8.1	172.16.8.190	DNS	193	Standard query response 0xb3e7 HTTPS www.bing.com CNAME www.bing.com.trafficmanager.net CNAME www.bing.com.edgekey.net CNAME e86383.dscx.akamaiedge.net
1632	15.171570	172.16.8.1	172.16.8.190	DNS	337	Standard query response 0xd9d A www.bing.com CNAME www.bing.com.trafficmanager.net CNAME www.bing.com.edgekey.net CNAME e86383.dscx.akamaiedge.net A 23.223.244.123 A 23.223.244.131 A
1698	16.846668	172.16.8.190	172.16.8.1	DNS	78	Standard query 0xb3ab A edge.microsoft.com
1699	16.846740	172.16.8.190	172.16.8.1	DNS	78	Standard query 0xbdc8 HTTPS edge.microsoft.com
1700	16.847677	172.16.8.1	172.16.8.190	DNS	149	Standard query response 0xbdc8 HTTPS edge.microsoft.com CNAME edge-microsoft-com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedge.net
1701	16.847677	172.16.8.1	172.16.8.190	DNS	181	Standard query response 0xb3ab A edge.microsoft.com CNAME edge-microsoft-com.dual-a-0036.a-msedge.net CNAME dual-a-0036.a-msedge.net A 13.107.21.239 A 204.79.197.239
1932	17.720700	172.16.8.190	172.16.8.1	DNS	71	Standard query 0xbef1 A ntp.msn.com
1933	17.720999	172.16.8.190	172.16.8.1	DNS	71	Standard query 0xb512 HTTPS ntp.msn.com
1935	17.727890	172.16.8.1	172.16.8.190	DNS	138	Standard query response 0xb512 HTTPS ntp.msn.com CNAME www-msn-com-a-0003.a-msedge.net CNAME a-0003.a-msedge.net
1936	17.727890	172.16.8.1	172.16.8.190	DNS	146	Standard query response 0xbef1 A ntp.msn.com CNAME www-msn-com-a-0003.a-msedge.net CNAME a-0003.a-msedge.net A 204.79.197.283
1952	17.758437	172.16.8.190	172.16.8.1	DNS	71	Standard query 0xf1b A ntp.msn.com
1953	17.759553	172.16.8.1	172.16.8.190	DNS	146	Standard query response 0xf1b A ntp.msn.com CNAME www-msn-com-a-0003.a-msedge.net CNAME a-0003.a-msedge.net A 204.79.197.283
2045	17.975412	172.16.8.190	172.16.8.1	DNS	87	Standard query 0xb0bf A img-s-msn-com.akamaized.net
2046	17.975632	172.16.8.190	172.16.8.1	DNS	87	Standard query 0xb07f HTTPS img-s-msn-com.akamaized.net
2047	17.975862	172.16.8.190	172.16.8.1	DNS	84	Standard query 0x7e0b A sb.scorecardresearch.com
2048	17.975922	172.16.8.190	172.16.8.1	DNS	84	Standard query 0xb0e6 HTTPS sb.scorecardresearch.com
2049	17.976081	172.16.8.190	172.16.8.1	DNS	71	Standard query 0xb470 A th.bing.com
2050	17.976140	172.16.8.190	172.16.8.1	DNS	71	Standard query 0x7bde HTTPS th.bing.com
2051	17.976471	172.16.8.1	172.16.8.190	DNS	128	Standard query response 0xb07f HTTPS img-s-msn-com.akamaized.net CNAME a1834.dscg2.akamai.net
2052	17.976471	172.16.8.1	172.16.8.190	DNS	148	Standard query response 0x7e0b A sb.scorecardresearch.com A 18.161.216.181 A 18.161.216.63 A 18.161.216.23 A 18.161.216.37
2053	17.976471	172.16.8.1	172.16.8.190	DNS	152	Standard query response 0xb0bf A img-s-msn-com.akamaized.net CNAME a1834.dscg2.akamai.net A 23.215.215.104 A 23.215.215.187

Frame 1521: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface 'DeviceNPF{0800D069-9184-48B7-8448-B951B2CDEA} (Ethernet II, Src: HP_35104162, Dst: 172.16.8.190) (Ethernet II, Src: Intel_Ethernet Adapter, Dst: 172.16.8.190)

Ethernet II, Src: HP_35104162, Dst: 172.16.8.190, Protocol: 0x0800 (Internet Protocol Version 4), Length: 60

Internet Protocol Version 4, Src: 172.16.8.190, Dst: 172.16.8.1

User Datagram Protocol, Src Port: 51966, Dst Port: 53

Domain Name System (query)

0000 7c 5e 1c cf be 45 7c 57 58 35 04 b1 00 00 45 00 [2] ... [1] X5 ... E ...
0001 00 40 a3 03 00 00 00 11 00 00 ec 18 00 be ec 18 00 00 ... S, i
0002 00 01 ca fe 00 35 00 2c 69 1d 1a cd 01 00 00 01
0003 00 00 00 00 00 00 04 65 64 67 65 09 6d 69 63 72
0004 6f 73 6f 66 74 03 63 6f 6d 00 00 01 00 01
0005 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

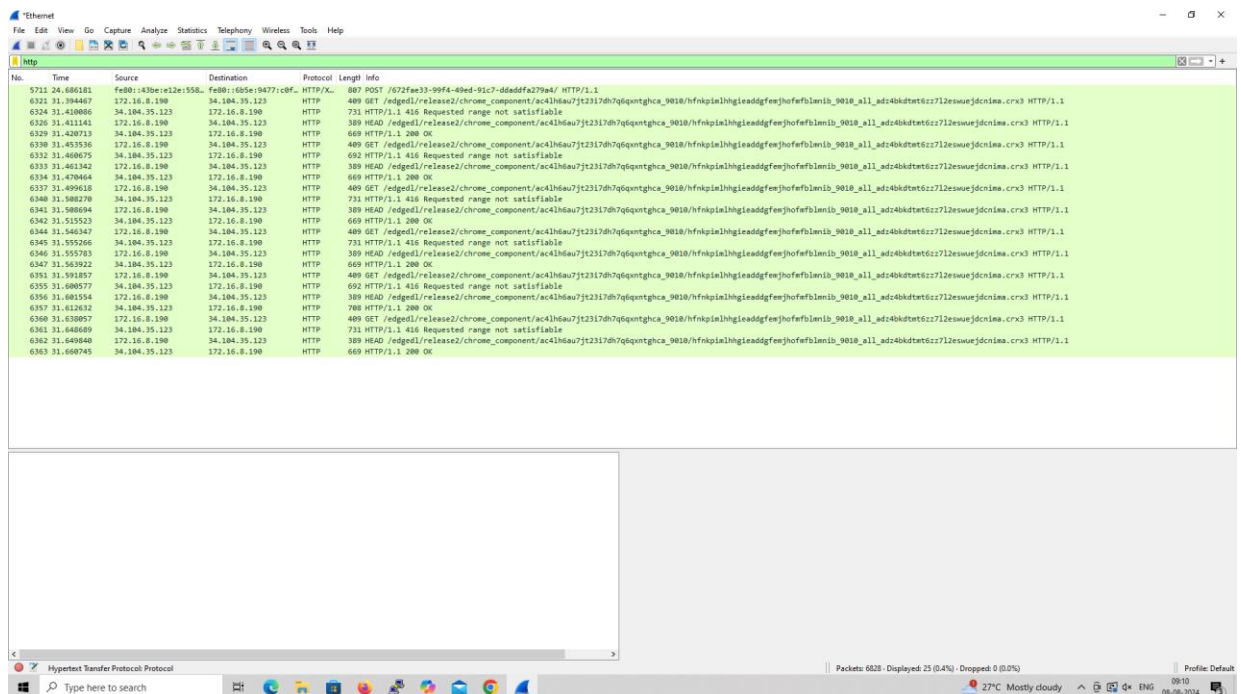
[illegible]

5. Create a Filter to display only HTTP packets and inspect the packets

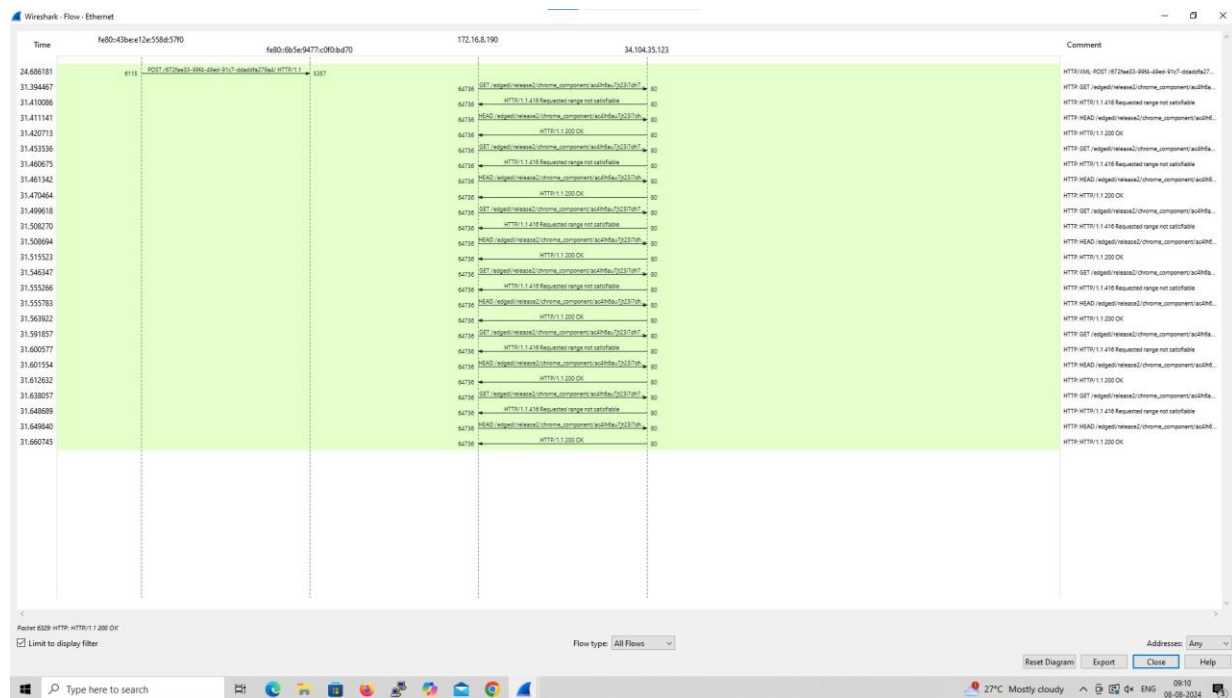
Procedure

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

Output



Flow Graph output

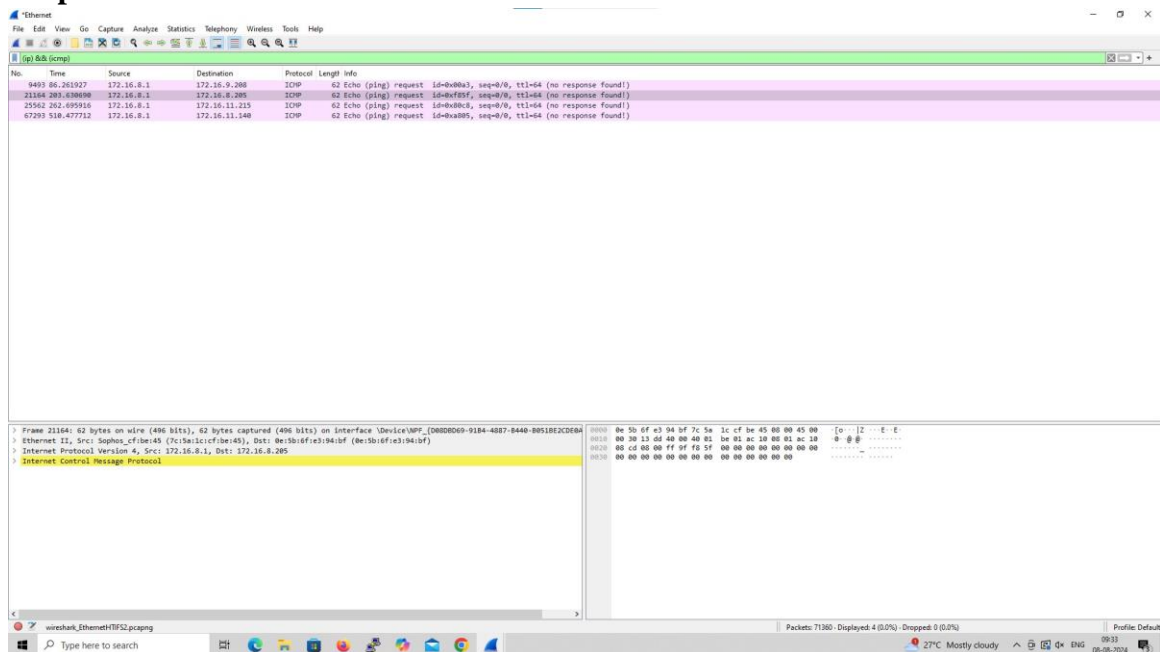


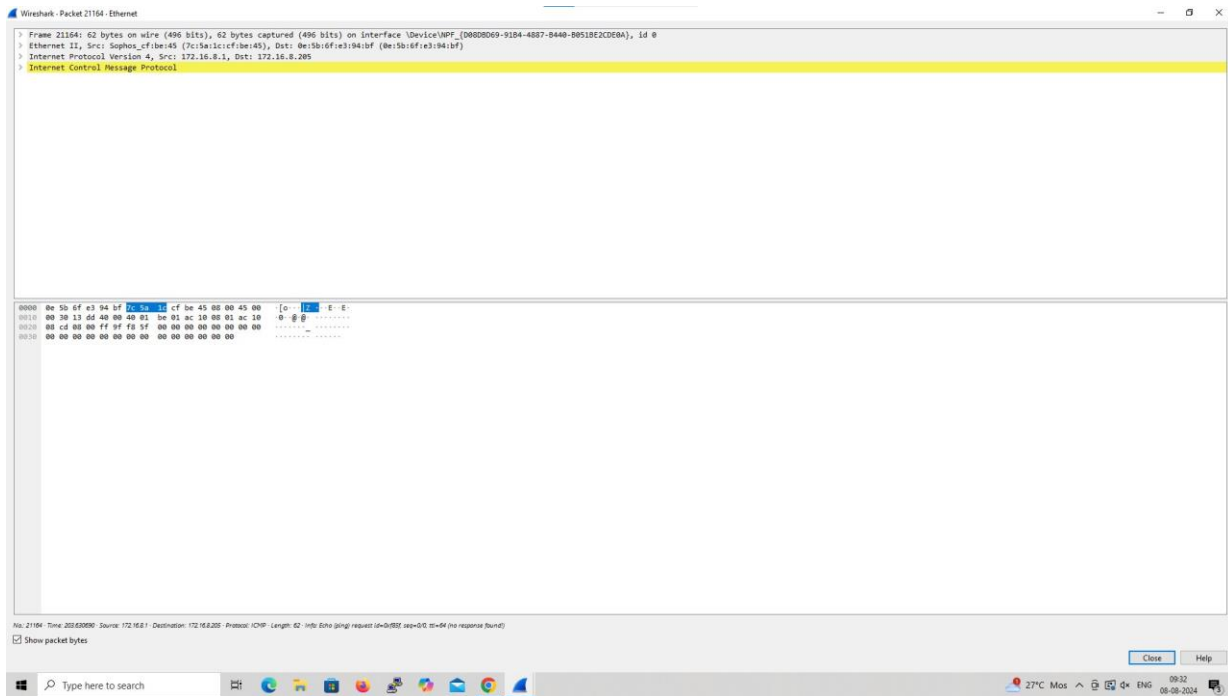
6. Create a Filter to display only IP/ICMP packets and inspect the packets.

Procedure

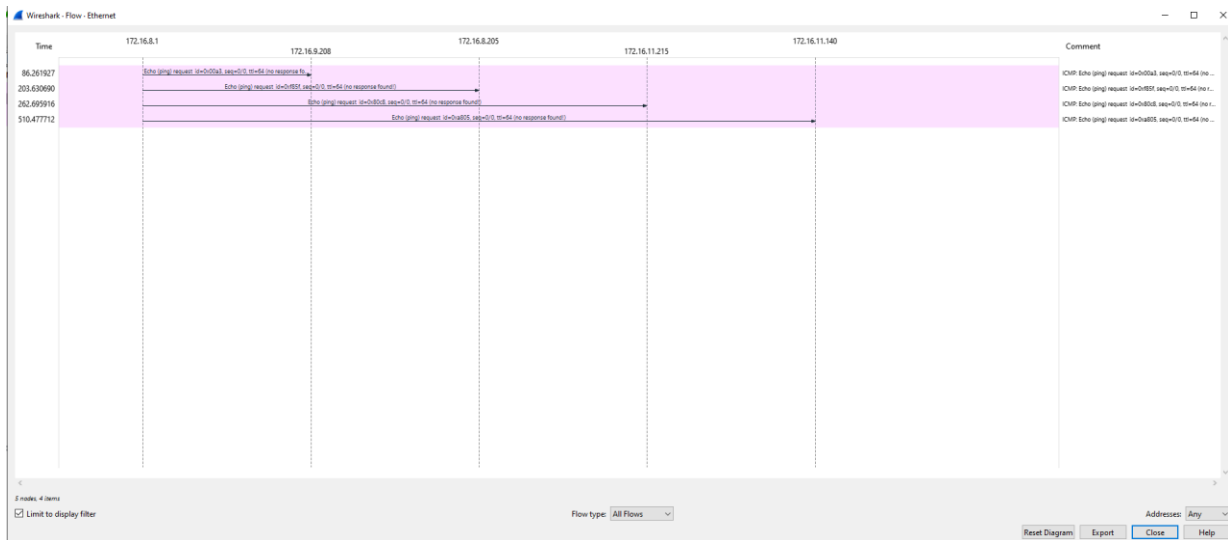
- ☐ Select Local Area Connection in Wireshark.
- ☐ Go to capture ☐ option
- ☐ Select stop capture automatically after 100 packets.
- ☐ Then click Start capture.
- ☐ Search ICMP/IP packets in search bar.
- ☐ Save the packets

Output





Flow Graph output



7. Create a Filter to display only DHCP 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 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, and Tools. The main window is divided into three panes: Packet List, Packet Details, and Packet Bytes.

Packet List: A table showing captured packets. The filter bar at the top is set to 'dhcp'. The table columns are No., Time, Source, Destination, Protocol, Length, and Info. The first five packets are DHCP requests and discoveries from 192.168.1.100 to 192.168.1.1.

No.	Time	Source	Destination	Protocol	Length	Info
549	4.923348	0.0.0.0	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0x586a891d
825	7.216266	0.0.0.0	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0x58a3d667
5518	47.791599	0.0.0.0	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0x2bcb3aee
7892	69.839681	0.0.0.0	255.255.255.255	DHCP	340	DHCP Discover - Transaction ID 0xf13cf3d1
7976	70.850412	0.0.0.0	255.255.255.255	DHCP	358	DHCP Request - Transaction ID 0xf13cf3d1
8211	75.389591	0.0.0.0	255.255.255.255	DHCP	364	DHCP Request - Transaction ID 0x34ab6d3d

Packet Details: The selected packet (No. 549) is expanded, showing the following layers:

- Ethernet II, Src: MicrostarTWT, c5cb93 (d8bbcl3cb93), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
- Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255
- User Datagram Protocol, Src Port: 68, Dst Port: 67
- Dynamic Host Configuration Protocol (Request)

Packet Bytes: The raw data of the selected packet is displayed in hexadecimal and ASCII. The first few bytes are ff ff ff ff, which correspond to the broadcast MAC address ff:ff:ff:ff:ff:ff.

Wireshark - Packet 19046 - Ethernet

> Frame 19046: 364 bytes on wire (2912 bits), 364 bytes captured (2912 bits) on interface \Device\NPF_{D80D0D09-91B4-4087-B040-B051B2CDE0A}, Id 0
> Ethernet II, Src: AzureWaveTee_9f:8c:75 (10:68:38:9f:8c:75), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol version 4, Src: 0.0.0.0, Dst: 255.255.255.255
> User Datagram Protocol, Src Port: 68, Dst Port: 67
> Dynamic Host Configuration Protocol (Request)

```
0000  ff ff ff ff ff 10 68 38 9f 8c 75 00 00 45 00  ....h B u E-
0010  01 5e a7 e5 00 00 00 11 91 aa 00 00 00 ff ff  .A.....$!
0020  ff ff 00 44 00 43 01 4a ff 97 01 01 00 2d 21  ..D C 2 .....
0030  9f 7f 00 00 00 00 00 00 00 00 00 00 00 00  ....
0040  00 00 00 00 00 10 68 38 9f 8c 75 00 00 00 00  ....h B u
0050  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0070  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0080  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0090  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00b0  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00c0  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00d0  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00e0  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
00f0  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0100  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0110  00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
0120  10 68 38 9f 8c 75 12 04 ac 10 00 72 0c 0f 4c  .hB u2...P LA
0130  50 54 4f 50 2d 4f 39 4b 47 53 53 54 43 51 12  .TOP-ORk GEEStQ
0140  00 00 4c 41 50 54 4f 50 2d 4f 39 4b 47 53 53 54  .LAPTOP-ORkGEEStQ
0150  43 3c 00 46 53 46 54 20 35 2e 30 3f 0e 01 03 06  Ck PStT 5.07...
0160  0f 1f 21 20 2c 2e 2f 20 70 f9 fc ff  .f...y
```

No. 19046, Time: 180.694333, Source: 0.0.0.0, Destination: 255.255.255.255, Protocol: DHCP, Length: 364, Info: DHCP Request, Transaction ID: 0A2A1987

☒ Show packet bytes

Close Help

Type here to search

27°C Mostly clou 09:21 08-08-2024