Kingdom of Saudi Arabia **Royal Commission at Yanbu** **Colleges & Institutes Division**		المـمـلـكـة العربية الـسـعـودية الـهــيـئة الملـكـية بينبع قـطـاع الـكـلـيات والمعاهد
Yanbu University College **Computer Science & Engineering Dept.** **Information & Computer Technology Dept.**		كـلـية ينبع الجـــامــعية قسم علوم وهندسة الحاسب الآلي قسم تقنية المعلومات والحاسب الآلي

PROJECT

ACADEMIC YEAR 1444/1445 H (2022/2023 G), SEMESTER I (441)

Information and Computer Security NET363

Assigned Date:	Nov 2, 2022	**Submission Date** (draft)**:**	Nov 23, 2022

STUDENT ID & NAME:	**Fai Almutairi 3820190**

FOR INSTRUCTOR USE ONLY				GENERAL INSTRUCTIONS
Q. No.	**CLOs**	**MAX MARK**	**MARKS OBTAINED**	• This is a group project. Students will work in the same group they formed earlier. • Group leader is responsible for the submission. • Project document should be uploaded to the BlackBoard course web page. • Project should be submitted in MS WORD file format using the correct template. • Do not use any other text color except blue/black. • Plagiarized work will not be graded. • Late submission could be penalized.
	1.01	3		
	1.03	3		
	2.01	8		
	3.01	2		
	3.02	2		
	3.03	2		
TOTAL MARKS		**20**		

MARKED BY: HUMERA GHANI	**Signature:**
CHECKED BY:	**Signature:**

Introduction to Wireshark:

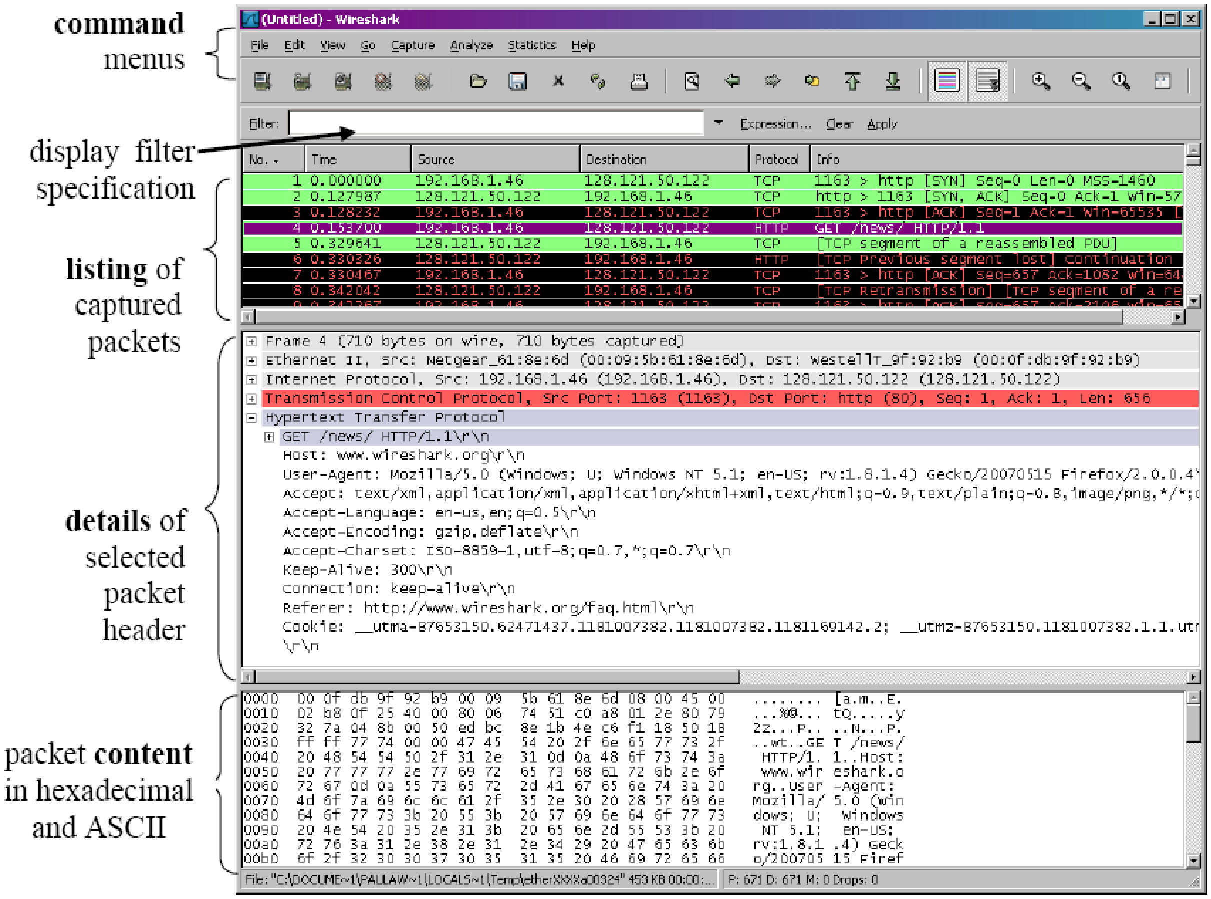
Wireshark is the world’s most popular network protocol analyzer. It has a rich and powerful feature set and runs on most computing platforms including Windows, OS X, Linux, and UNIX. Network professionals, security experts, developers, and educators around the world use it regularly. It is freely available as open source, and is released under the GNU General Public License version 2. It has been developed and maintained by a global team of protocol experts, and it is an example of a disruptive technology. Wireshark formerly used to be known as Ethereal. Wireshark is a free packet sniffer computer application. It is used for network troubleshooting, analysis, software and communications protocol development, and education. In June 2006 the project was renamed from Ethereal due to trademark issues. Wireshark has tools for capturing, viewing, and analysis of data packets. Wireshark has sophisticated wireless protocol analysis support to help administrators troubleshoot wireless networks. With the appropriate driver support, Wireshark can capture traffic” from the air” and decode it into a format that helps administrators track down issues that are causing poor performance, intermittent connectivity, and other common problems.

Basic Installation and Test Run

Wireshark allows us to view the content of messages sent/received from/by protocols at different levels of the protocol stack Wireshark is a free network protocol analyzer that runs on Windows, Linux/Unix and Mac computers It is an ideal packet analyzer including the ability to analyze hundreds of protocols and well-designed user interfaces. It works on computers using Ethernet, Token-Ring, FDDI, serial (PPP and SLIP), 802.11 wireless LAN, and ATM connectivity (if the operating system it is running on allows Wireshark). Run Wireshark

When you run the Wireshark program, the Wireshark GUI shown in Figure 2 will appear. Initially, no data will be displayed in different windows

Figure 1: Wireshark Graphical User Interface


command menus
display filter specification
listing of captured packets
details of selected packet header
packet content in hexadecimal and ASCII
(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply

```
No. -  Time        Source            Destination       Protocol  Info
    1 0.000000    192.168.1.46      128.121.50.122    TCP       1163 > http [SYN] Seq=0 Len=0 MSS=1460
    2 0.127987    128.121.50.122    192.168.1.46      TCP       http > 1163 [SYN, ACK] Seq=0 Ack=1 Win=57
    3 0.128232    192.168.1.46      128.121.50.122    TCP       1163 > http [ACK] Seq=1 Ack=1 Win=65535 [
    4 0.153700    192.168.1.46      128.121.50.122    HTTP      GET /news/ HTTP/1.1
    5 0.329641    128.121.50.122    192.168.1.46      TCP       [TCP segment of a reassembled PDU]
    6 0.330326    128.121.50.122    192.168.1.46      HTTP      [TCP Previous segment lost] Continuation
    7 0.330467    192.168.1.46      128.121.50.122    TCP       1163 > http [ACK] Seq=657 Ack=1082 Win=64
    8 0.342042    128.121.50.122    192.168.1.46      TCP       [TCP Retransmission] [TCP segment of a re
```

```
⊞ Frame 4 (710 bytes on wire, 710 bytes captured)
⊞ Ethernet II, Src: Netgear_61:8e:6d (00:09:5b:61:8e:6d), Dst: WestellT_9f:92:b9 (00:0f:db:9f:92:b9)
⊞ Internet Protocol, Src: 192.168.1.46 (192.168.1.46), Dst: 128.121.50.122 (128.121.50.122)
⊞ Transmission Control Protocol, Src Port: 1163 (1163), Dst Port: http (80), Seq: 1, Ack: 1, Len: 656
⊟ Hypertext Transfer Protocol
  ⊞ GET /news/ HTTP/1.1\r\n
    Host: www.wireshark.org\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.4) Gecko/20070515 Firefox/2.0.0.4
    Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;
    Accept-Language: en-us,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
    Keep-Alive: 300\r\n
    Connection: keep-alive\r\n
    Referer: http://www.wireshark.org/faq.html\r\n
    Cookie: __utma=87653150.62471437.1181007382.1181007382.1181169142.2; __utmz=87653150.1181007382.1.1.utr
    \r\n
```

```
0000  00 0f db 9f 92 b9 00 09  5b 61 8e 6d 08 00 45 00   ........ [a.m..E.
0010  02 b8 0f 25 40 00 80 06  74 51 c0 a8 01 2e 80 79   ...%@... tQ.....y
0020  32 7a 04 8b 00 50 ed bc  8e 1b 4e c6 f1 18 50 18   2z...P.. ..N...P.
0030  ff ff 77 74 00 00 47 45  54 20 2f 6e 65 77 73 2f   ..wt..GE T /news/
0040  20 48 54 54 50 2f 31 2e  31 0d 0a 48 6f 73 74 3a    HTTP/1. 1..Host:
0050  20 77 77 77 2e 77 69 72  65 73 68 61 72 6b 2e 6f    www.wir eshark.o
0060  72 67 0d 0a 55 73 65 72  2d 41 67 65 6e 74 3a 20   rg..User -Agent: 
0070  4d 6f 7a 69 6c 6c 61 2f  35 2e 30 20 28 57 69 6e   Mozilla/ 5.0 (Win
0080  64 6f 77 73 3b 20 55 3b  20 57 69 6e 64 6f 77 73   dows; U;  Windows
0090  20 4e 54 20 35 2e 31 3b  20 65 6e 2d 55 53 3b 20    NT 5.1;  en-US; 
00a0  72 76 3a 31 2e 38 2e 31  2e 34 29 20 47 65 63 6b   rv:1.8.1 .4) Geck
00b0  6f 2f 32 30 30 37 30 35  31 35 20 46 69 72 65 66   o/200705 15 Firef
```

File: "C:\DOCUME~1\PALLAW~1\LOCALS~1\Temp\etherXXXXa00324" 453 KB 00:00:... P: 671 D: 671 M: 0 Drops: 0

The Wireshark interface has five main components:

1. The command menu is the standard drop-down menu located at the top of the window Now we are interested in the File and Capture menus The File menu allows you to save the captured packet data or open the file containing the previously captured packet data and exit the Wireshark application The Capture menu allows you to start capturing packets.
2. The packet list window displays a one-line summary for each captured packet, including the packet number (specified by Wireshark; it's not the packet number included in the protocol's headers), and the time the packet was captured. , source and destination address of the packet, protocol type, and protocol-specific information contained in the packet The list of packages can be sorted by one of these categories by clicking on the column name The Protocol Type field lists the top-level protocol that sent or received this packet, i.e. the protocol that was the final source or sink for this packet

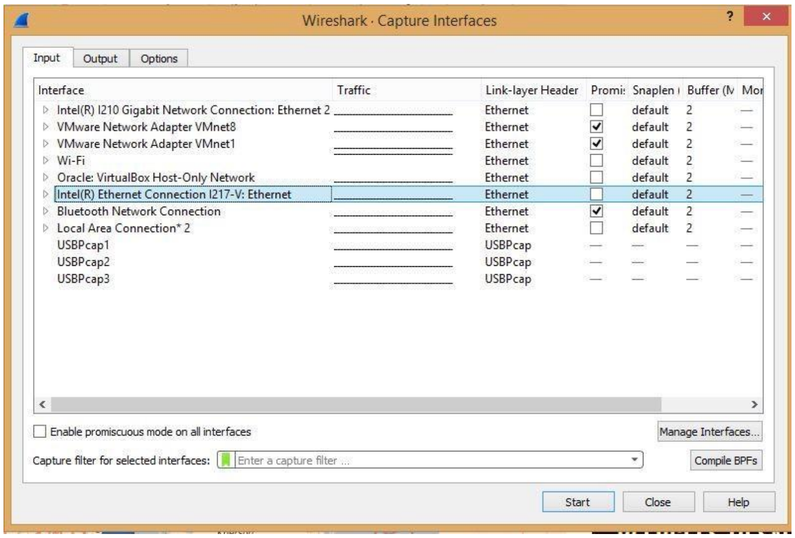
3. The Package Header Details window provides detailed information about the (highlighted) package selected in the Package List window (To select a package in the package list window, place the cursor over the package's one-line summary in the package list window and click the left mouse button.) These details include information about the Ethernet frame and the IP datagram it contains. The number of Ethernet and IP layer details displayed can be expanded or reduced by clicking the right- or down-pointing arrow to the left of the Ethernet frame or IP datagram line in the packet details window If the packet is transported over TCP or UDP, the TCP or UDP details will also be displayed, which can also be expanded or reduced. Finally, details about the higher-level protocol that sent or received this packet are also provided.
4. The packet content window displays the entire contents of the captured frame, both in ASCII and hexadecimal formats.
5. Near the top of Wireshark's GUI is the Display Packet Filter field, where a protocol name or other information can be entered to filter the information displayed in the packet display window. The package list (and thus the packet header). and the package contents window). In the example below, we'll use the show packet filter field to have Wireshark hide (not show) packets except those that match the HTTP message

Let’s taking Wireshark for a Test Run

Perform the following steps:

1. Start your favorite web browser, it will display the homepage you have selected.
2. Start the Wireshark software Initially, you will see a window similar to the one shown in Figure 2, except that no packet data is displayed in the Packet List, Packet Headers, or Packet Contents window, because Wireshark has not started capturing the packets package.
3. To start packet capture, select the Capture drop-down menu and select Options This will cause the "Wireshark: Capture Options" window to appear as shown in the image below

Figure 2: Capture Options



Interface	Traffic	Link-layer Header	Promi:	Snaplen (	Buffer (M	Mor
Intel(R) I210 Gigabit Network Connection: Ethernet 2		Ethernet	□	default	2	—
VMware Network Adapter VMnet8		Ethernet	☑	default	2	—
VMware Network Adapter VMnet1		Ethernet	☑	default	2	—
Wi-Fi		Ethernet	□	default	2	—
Oracle: VirtualBox Host-Only Network		Ethernet	□	default	2	—
Intel(R) Ethernet Connection I217-V: Ethernet		Ethernet	□	default	2	—
Bluetooth Network Connection		Ethernet	☑	default	2	—
Local Area Connection* 2		Ethernet	□	default	2	—
USBPcap1		USBPcap	—	—	—	—
USBPcap2		USBPcap	—	—	—	—
USBPcap3		USBPcap	—	—	—	—

□ Enable promiscuous mode on all interfaces

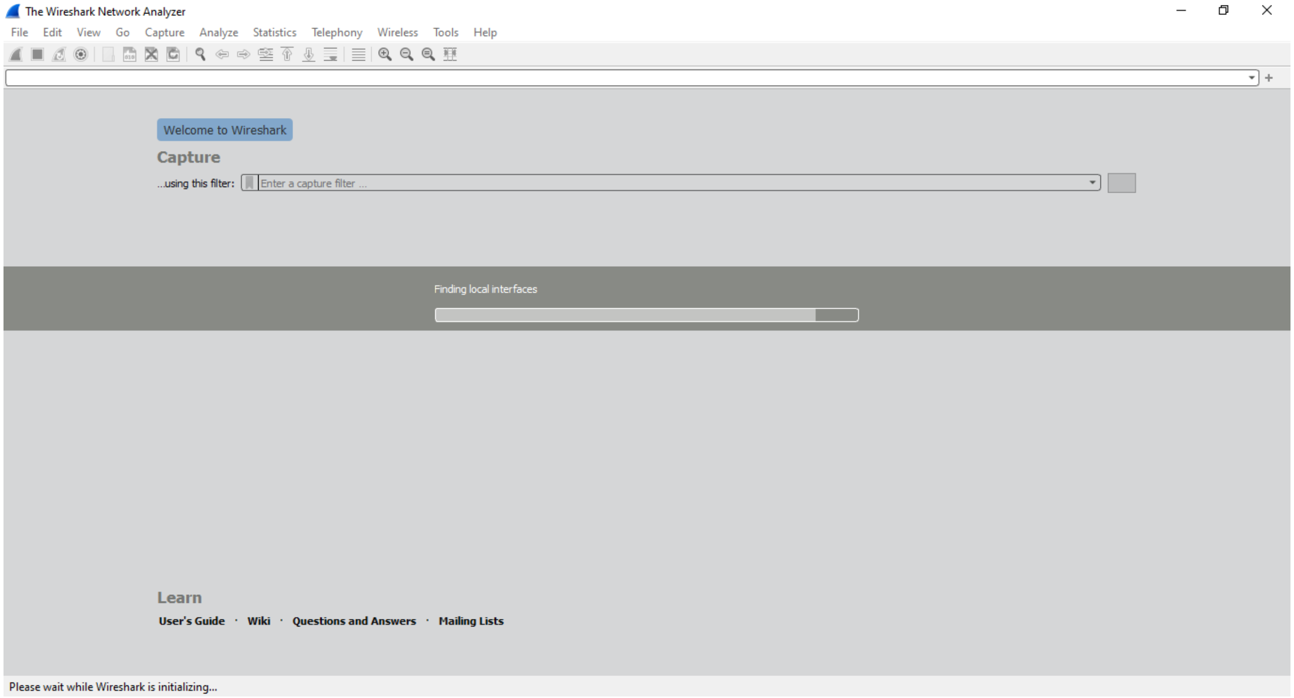
Capture filter for selected interfaces: Enter a capture filter ...

After selecting the network interface (or using the default one chosen by Wireshark), click Start Packet capture will now begin - all packets sent/received from/by your computer are now captured by Wireshark!

4. While Wireshark is running, enter the URL: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html and view this page in your browser To display this page, your browser will contact the HTTP server at gaia.cs.umass.edu and exchange HTTP messages with the server to download this page Ethernet frames containing these HTTP messages will be captured by Wireshark.
5. After your browser has displayed the INTRO-wireshark-file1.html page, stop capturing Wireshark packets by selecting stop in the Wireshark capture window Exchanging HTTP messages with the gaia.cs.umass.edu web server should appear somewhere in the list of captured packets. But there will be more package types displayed Even if the only action you take is to download a web page, it is clear that there are many other protocols running on your computer that are not visible to the user.
6. Type "http" (without quotes and lowercase - all protocol names are lowercase in Wireshark) in the filter specification window displayed at the top of the main Wireshark window Then select Apply (to the right of where you entered "http"). Therefore, only HTTP messages will be displayed in the package list window.

7. Select the first http message displayed in the package list window This should be an HTTP GET message sent from your computer to the gaia.cs.umass.edu HTTP server When you select the HTTP GET message, the Ethernet frame, IP datagram, TCP segment, and HTTP message header information are displayed in the packet headers window By clicking the plus and minus boxes on the left side of the packet details window, reduce the amount of Frame, Ethernet, Internet Protocol, and Transmission Control Protocol information displayed Maximize the amount of information displayed about the HTTP protocol
8. Remove Wireshark.

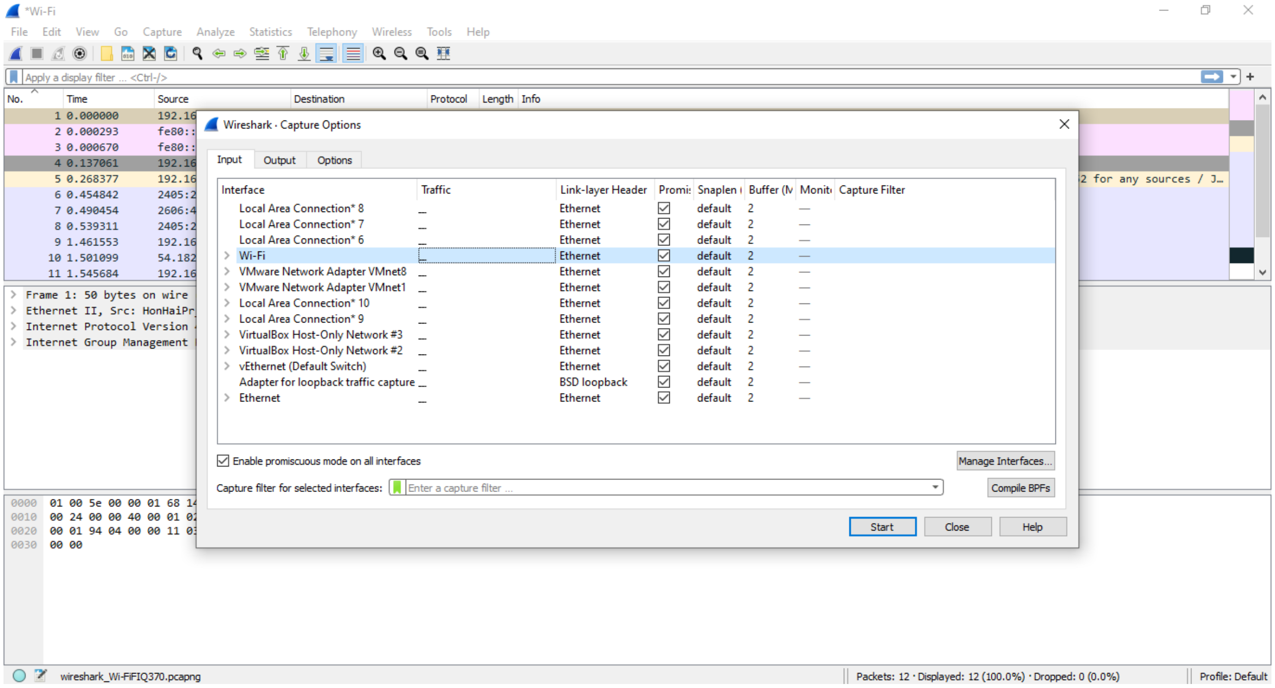
1) Running the Wireshark.


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists

Finding local interfaces

Please wait while Wireshark is initializing...

2)Choosing the adapter


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Wireshark · Capture Options
Input
Output
Options
Manage Interfaces…
Compile BPFs
Start
Close
Help

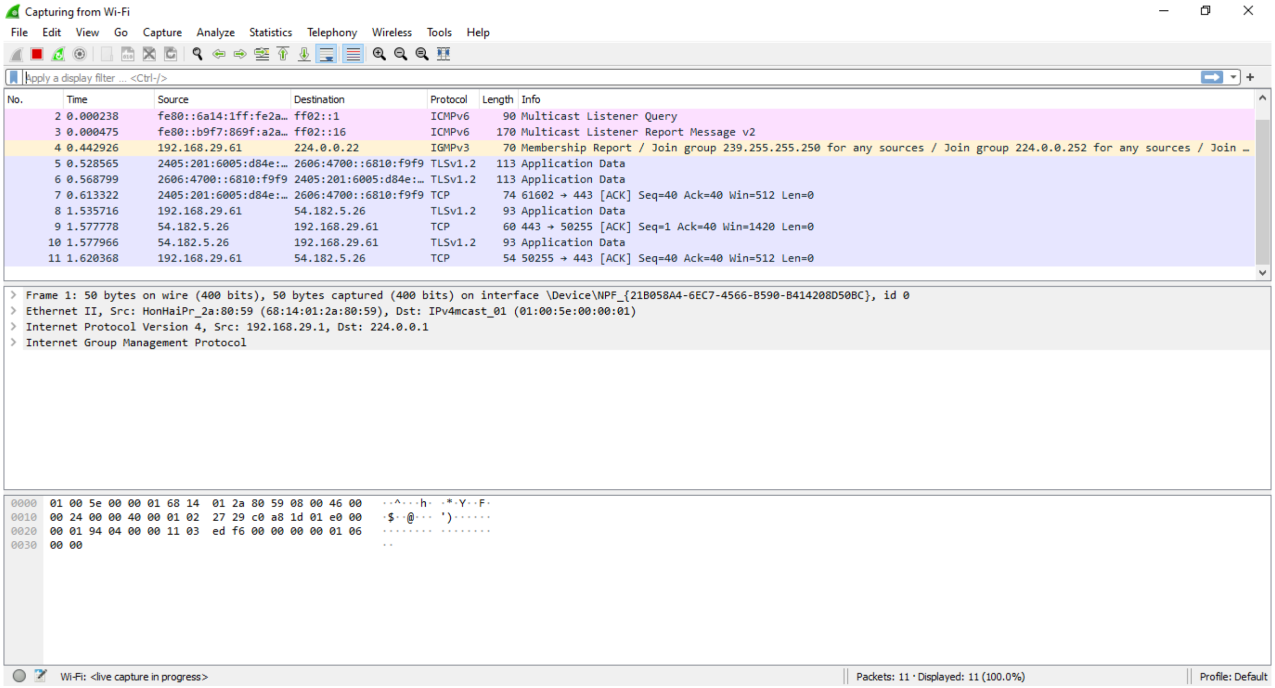
Interface	Traffic	Link-layer Header	Promi:	Snaplen (	Buffer (M	Monit	Capture Filter
Local Area Connection* 8	_	Ethernet	☑	default	2	—	
Local Area Connection* 7	_	Ethernet	☑	default	2	—	
Local Area Connection* 6	_	Ethernet	☑	default	2	—	
Wi-Fi	_	Ethernet	☑	default	2	—	
VMware Network Adapter VMnet8	_	Ethernet	☑	default	2	—	
VMware Network Adapter VMnet1	_	Ethernet	☑	default	2	—	
Local Area Connection* 10	_	Ethernet	☑	default	2	—	
Local Area Connection* 9	_	Ethernet	☑	default	2	—	
VirtualBox Host-Only Network #3	_	Ethernet	☑	default	2	—	
VirtualBox Host-Only Network #2	_	Ethernet	☑	default	2	—	
vEthernet (Default Switch)	_	Ethernet	☑	default	2	—	
Adapter for loopback traffic capture	_	BSD loopback	☑	default	2	—	
Ethernet	_	Ethernet	☑	default	2	—	

☑ Enable promiscuous mode on all interfaces

Capture filter for selected interfaces: Enter a capture filter …

wireshark_Wi-FiFIQ370.pcapng | Packets: 12 · Displayed: 12 (100.0%) · Dropped: 0 (0.0%) | Profile: Default

3) capturing packets


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

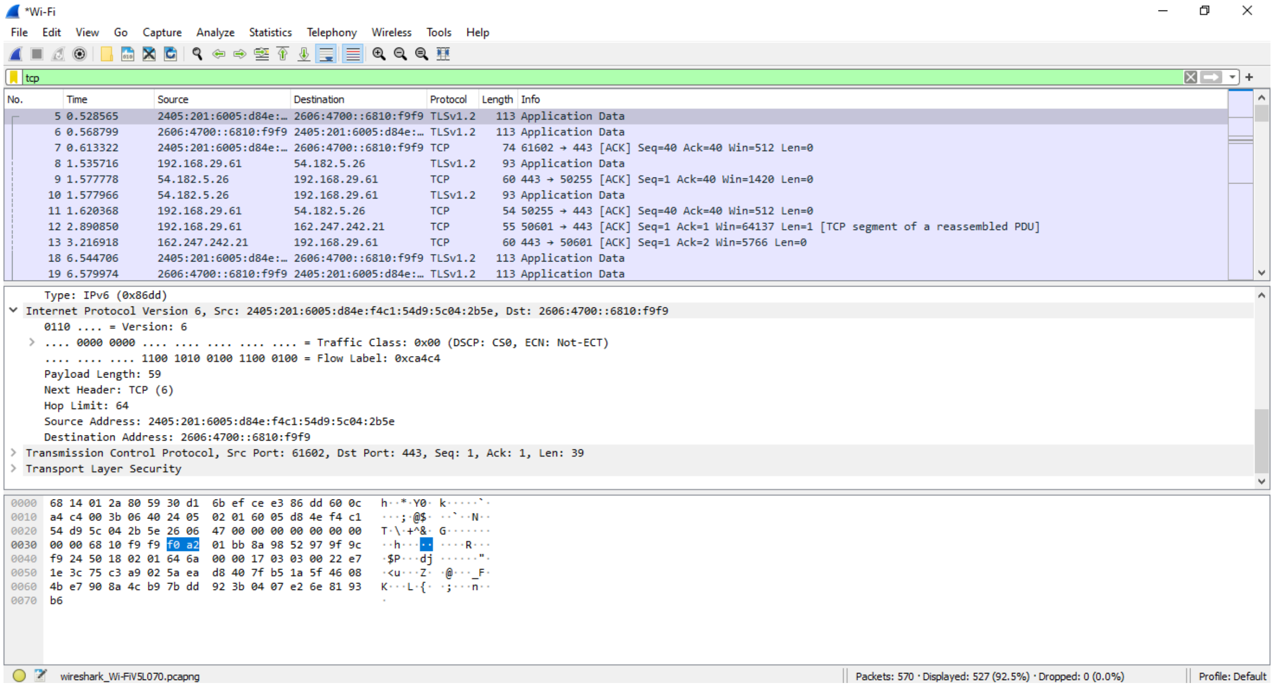
```
No.   Time      Source                 Destination            Protocol Length Info
    2 0.000238  fe80::6a14:1ff:fe2a…   ff02::1                ICMPv6     90 Multicast Listener Query
    3 0.000475  fe80::b9f7:869f:a2a…   ff02::16               ICMPv6    170 Multicast Listener Report Message v2
    4 0.442926  192.168.29.61          224.0.0.22             IGMPv3     70 Membership Report / Join group 239.255.255.250 for any sources / Join group 224.0.0.252 for any sources / Join …
    5 0.528565  2405:201:6005:d84e:…   2606:4700::6810:f9f9   TLSv1.2   113 Application Data
    6 0.568799  2606:4700::6810:f9f9   2405:201:6005:d84e:…   TLSv1.2   113 Application Data
    7 0.613322  2405:201:6005:d84e:…   2606:4700::6810:f9f9   TCP        74 61602 → 443 [ACK] Seq=40 Ack=40 Win=512 Len=0
    8 1.535716  192.168.29.61          54.182.5.26            TLSv1.2    93 Application Data
    9 1.577778  54.182.5.26            192.168.29.61          TCP        60 443 → 50255 [ACK] Seq=1 Ack=40 Win=1420 Len=0
   10 1.577966  54.182.5.26            192.168.29.61          TLSv1.2    93 Application Data
   11 1.620368  192.168.29.61          54.182.5.26            TCP        54 50255 → 443 [ACK] Seq=40 Ack=40 Win=512 Len=0
```

```
> Frame 1: 50 bytes on wire (400 bits), 50 bytes captured (400 bits) on interface \Device\NPF_{21B058A4-6EC7-4566-B590-B414208D50BC}, id 0
> Ethernet II, Src: HonHaiPr_2a:80:59 (68:14:01:2a:80:59), Dst: IPv4mcast_01 (01:00:5e:00:00:01)
> Internet Protocol Version 4, Src: 192.168.29.1, Dst: 224.0.0.1
> Internet Group Management Protocol
```

```
0000  01 00 5e 00 00 01 68 14  01 2a 80 59 08 00 46 00   ··^···h· ·*·Y··F·
0010  00 24 00 00 40 00 01 02  27 29 c0 a8 1d 01 e0 00   ·$··@··· ')······
0020  00 01 94 04 00 00 11 03  ed f6 00 00 00 00 01 06   ········ ········
0030  00 00                                              ··
```

Wi-Fi: <live capture in progress> | Packets: 11 · Displayed: 11 (100.0%) | Profile: Default

4)After applying different filter of protocols.


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

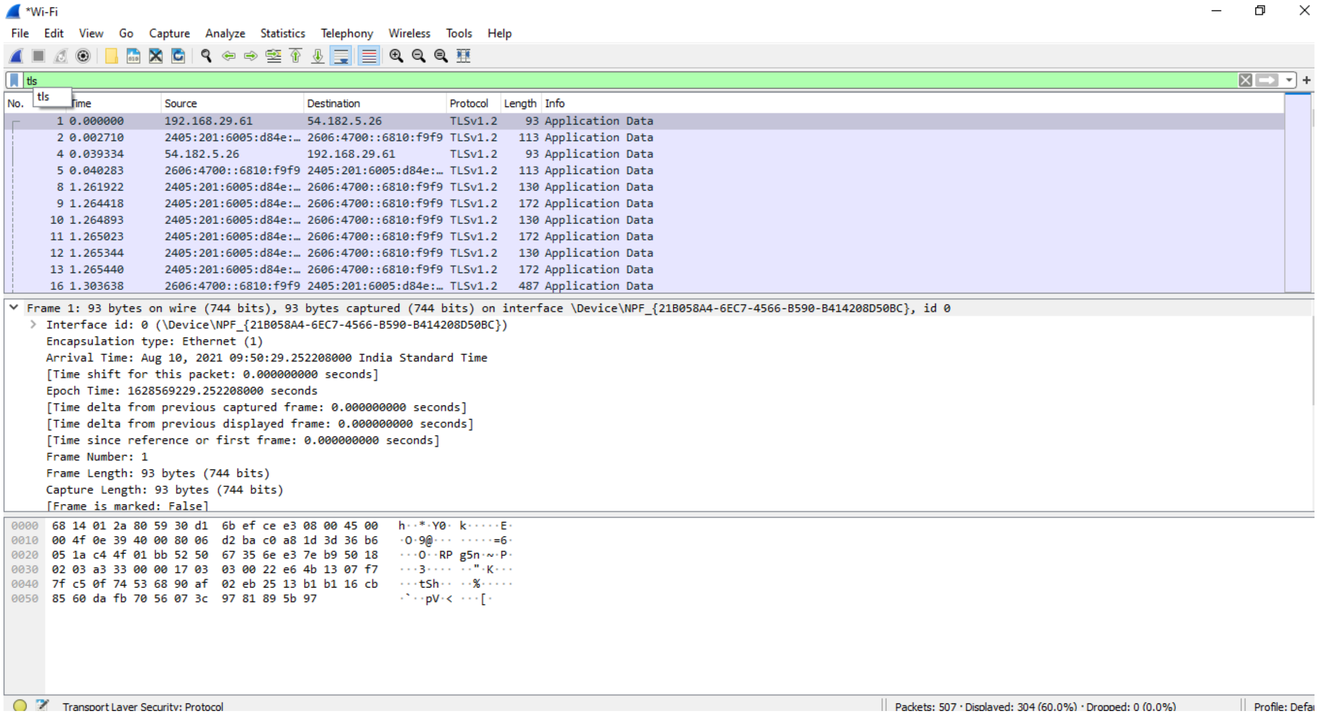
```
tcp

No.  Time      Source                Destination           Protocol Length Info
   5 0.528565  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  113 Application Data
   6 0.568799  2606:4700::6810:f9f9  2405:201:6005:d84e:… TLSv1.2  113 Application Data
   7 0.613322  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TCP       74 61602 → 443 [ACK] Seq=40 Ack=40 Win=512 Len=0
   8 1.535716  192.168.29.61         54.182.5.26           TLSv1.2   93 Application Data
   9 1.577778  54.182.5.26           192.168.29.61         TCP       60 443 → 50255 [ACK] Seq=1 Ack=40 Win=1420 Len=0
  10 1.577966  54.182.5.26           192.168.29.61         TLSv1.2   93 Application Data
  11 1.620368  192.168.29.61         54.182.5.26           TCP       54 50255 → 443 [ACK] Seq=40 Ack=40 Win=512 Len=0
  12 2.890850  192.168.29.61         162.247.242.21        TCP       55 50601 → 443 [ACK] Seq=1 Ack=1 Win=64137 Len=1 [TCP segment of a reassembled PDU]
  13 3.216918  162.247.242.21        192.168.29.61         TCP       60 443 → 50601 [ACK] Seq=1 Ack=2 Win=5766 Len=0
  18 6.544706  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  113 Application Data
  19 6.579974  2606:4700::6810:f9f9  2405:201:6005:d84e:… TLSv1.2  113 Application Data

    Type: IPv6 (0x86dd)
  Internet Protocol Version 6, Src: 2405:201:6005:d84e:f4c1:54d9:5c04:2b5e, Dst: 2606:4700::6810:f9f9
    0110 .... = Version: 6
  > .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
    .... .... .... 1100 1010 0100 1100 0100 = Flow Label: 0xca4c4
    Payload Length: 59
    Next Header: TCP (6)
    Hop Limit: 64
    Source Address: 2405:201:6005:d84e:f4c1:54d9:5c04:2b5e
    Destination Address: 2606:4700::6810:f9f9
> Transmission Control Protocol, Src Port: 61602, Dst Port: 443, Seq: 1, Ack: 1, Len: 39
> Transport Layer Security

0000  68 14 01 2a 80 59 30 d1  6b ef ce e3 86 dd 60 0c   h··*·Y0· k·····`·
0010  a4 c4 00 3b 06 40 24 05  02 01 60 05 d8 4e f4 c1   ···;·@$· ··`··N··
0020  54 d9 5c 04 2b 5e 26 06  47 00 00 00 00 00 00 00   T·\·+^&· G·······
0030  00 00 68 10 f9 f9 f0 a2  01 bb 8a 98 52 97 9f 9c   ··h····· ····R···
0040  f9 24 50 18 02 01 64 6a  00 00 17 03 03 00 22 e7   ·$P···dj ······"·
0050  1e 3c 75 c3 a9 02 5a ea  d8 40 7f b5 1a 5f 46 08   ·<u···Z· ·@···_F·
0060  4b e7 90 8a 4c b9 7b dd  92 3b 04 07 e2 6e 81 93   K···L·{· ·;···n··
0070  b6                                                  ·

wireshark_Wi-FiV5L070.pcapng    Packets: 570 · Displayed: 527 (92.5%) · Dropped: 0 (0.0%)    Profile: Default
```

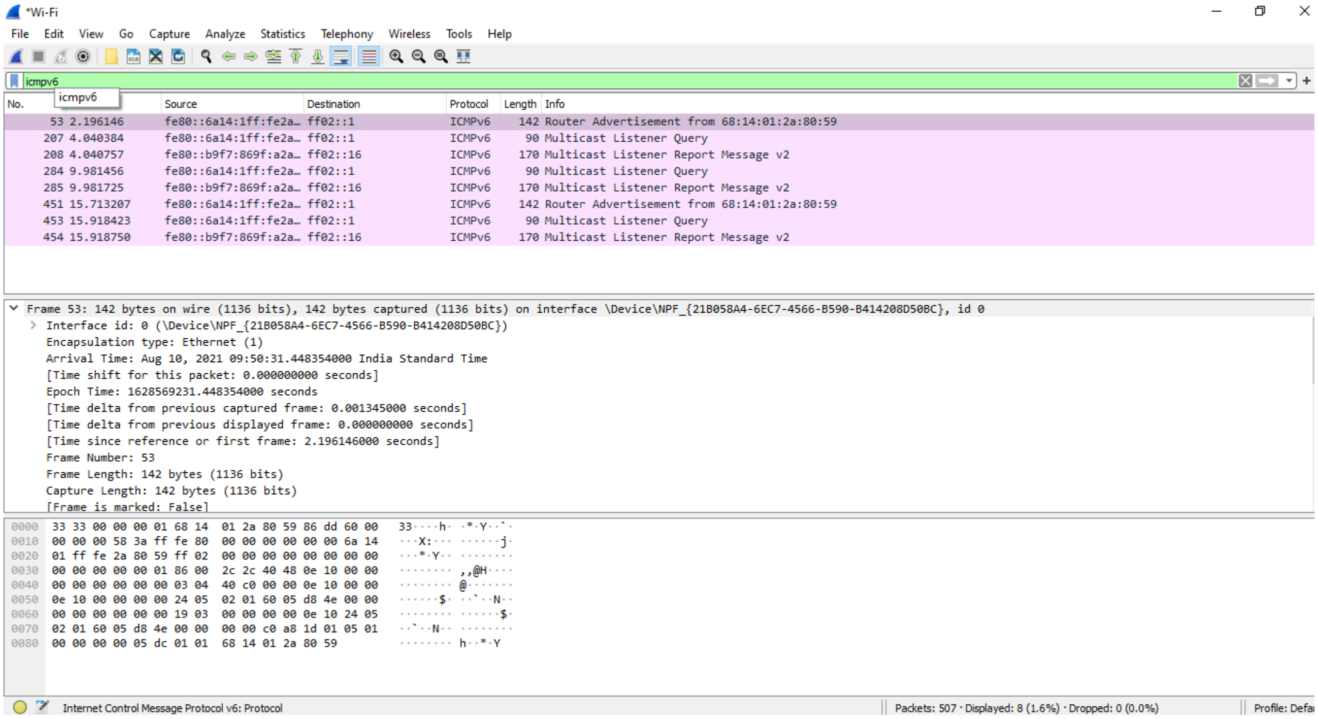

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tls
tls
No.  Time      Source                Destination           Protocol Length Info
   1 0.000000  192.168.29.61         54.182.5.26           TLSv1.2   93 Application Data
   2 0.002710  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  113 Application Data
   4 0.039334  54.182.5.26           192.168.29.61         TLSv1.2   93 Application Data
   5 0.040283  2606:4700::6810:f9f9  2405:201:6005:d84e:… TLSv1.2  113 Application Data
   8 1.261922  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  130 Application Data
   9 1.264418  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  172 Application Data
  10 1.264893  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  130 Application Data
  11 1.265023  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  172 Application Data
  12 1.265344  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  130 Application Data
  13 1.265440  2405:201:6005:d84e:… 2606:4700::6810:f9f9  TLSv1.2  172 Application Data
  16 1.303638  2606:4700::6810:f9f9  2405:201:6005:d84e:… TLSv1.2  487 Application Data

  Frame 1: 93 bytes on wire (744 bits), 93 bytes captured (744 bits) on interface \Device\NPF_{21B058A4-6EC7-4566-B590-B414208D50BC}, id 0
  > Interface id: 0 (\Device\NPF_{21B058A4-6EC7-4566-B590-B414208D50BC})
    Encapsulation type: Ethernet (1)
    Arrival Time: Aug 10, 2021 09:50:29.252208000 India Standard Time
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1628569229.252208000 seconds
    [Time delta from previous captured frame: 0.000000000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 0.000000000 seconds]
    Frame Number: 1
    Frame Length: 93 bytes (744 bits)
    Capture Length: 93 bytes (744 bits)
    [Frame is marked: False]

0000  68 14 01 2a 80 59 30 d1  6b ef ce e3 08 00 45 00   h··*·Y0· k·····E·
0010  00 4f 0e 39 40 00 80 06  d2 ba c0 a8 1d 3d 36 b6   ·O·9@··· ·····=6·
0020  05 1a c4 4f 01 bb 52 50  67 35 6e e3 7e b9 50 18   ···O··RP g5n·~·P·
0030  02 03 a3 33 00 00 17 03  03 00 22 e6 4b 13 07 f7   ···3···· ··"·K···
0040  7f c5 0f 74 53 68 90 af  02 eb 25 13 b1 b1 16 cb   ···tSh·· ··%·····
0050  85 60 da fb 70 56 07 3c  97 81 89 5b 97            ·`··pV·< ···[·

Transport Layer Security: Protocol    Packets: 507 · Displayed: 304 (60.0%) · Dropped: 0 (0.0%)    Profile: Defa
```


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmpv6
icmpv6

```
No.                   Source                 Destination            Protocol  Length Info
     53 2.196146      fe80::6a14:1ff:fe2a… ff02::1                ICMPv6     142 Router Advertisement from 68:14:01:2a:80:59
    207 4.040384      fe80::6a14:1ff:fe2a… ff02::1                ICMPv6      90 Multicast Listener Query
    208 4.040757      fe80::b9f7:869f:a2a… ff02::16               ICMPv6     170 Multicast Listener Report Message v2
    284 9.981456      fe80::6a14:1ff:fe2a… ff02::1                ICMPv6      90 Multicast Listener Query
    285 9.981725      fe80::b9f7:869f:a2a… ff02::16               ICMPv6     170 Multicast Listener Report Message v2
    451 15.713207     fe80::6a14:1ff:fe2a… ff02::1                ICMPv6     142 Router Advertisement from 68:14:01:2a:80:59
    453 15.918423     fe80::6a14:1ff:fe2a… ff02::1                ICMPv6      90 Multicast Listener Query
    454 15.918750     fe80::b9f7:869f:a2a… ff02::16               ICMPv6     170 Multicast Listener Report Message v2
```

```
v Frame 53: 142 bytes on wire (1136 bits), 142 bytes captured (1136 bits) on interface \Device\NPF_{21B058A4-6EC7-4566-B590-B414208D50BC}, id 0
  > Interface id: 0 (\Device\NPF_{21B058A4-6EC7-4566-B590-B414208D50BC})
    Encapsulation type: Ethernet (1)
    Arrival Time: Aug 10, 2021 09:50:31.448354000 India Standard Time
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1628569231.448354000 seconds
    [Time delta from previous captured frame: 0.001345000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 2.196146000 seconds]
    Frame Number: 53
    Frame Length: 142 bytes (1136 bits)
    Capture Length: 142 bytes (1136 bits)
    [Frame is marked: False]
```

```
0000  33 33 00 00 00 01 68 14  01 2a 80 59 86 dd 60 00   33····h· ·*·Y··`·
0010  00 00 00 58 3a ff fe 80  00 00 00 00 00 00 6a 14   ···X:··· ······j·
0020  01 ff fe 2a 80 59 ff 02  00 00 00 00 00 00 00 00   ···*·Y·· ········
0030  00 00 00 00 00 01 86 00  2c 2c 40 48 0e 10 00 00   ········ ,,@H····
0040  00 00 00 00 00 00 03 04  40 c0 00 00 0e 10 00 00   ········ @·······
0050  0e 10 00 00 00 00 24 05  02 01 60 05 d8 4e 00 00   ······$· ··`··N··
0060  00 00 00 00 00 00 19 03  00 00 00 00 0e 10 24 05   ········ ······$·
0070  02 01 60 05 d8 4e 00 00  00 00 c0 a8 1d 01 05 01   ··`··N·· ········
0080  00 00 00 00 05 dc 01 01  68 14 01 2a 80 59         ········ h··*·Y
```

```
Internet Control Message Protocol v6: Protocol        Packets: 507 · Displayed: 8 (1.6%) · Dropped: 0 (0.0%)        Profile: Defa
```

In ***conclusion*** this paper has shed light on how the wire shark application can be used as invaluable tool in academic network protocol research; as well as used as a tool for malicious intent in the scenario of MAC flooding or ARP poisoning; and finally, how Network Engineers and Administrators can utilize Wire Shark to prevent malicious intent as well as increase a computer network's productivity.

References:

- Slater, W. (2012). Using wireshark and other tools to as an aid in cyberwarfare and cybercrime. *Hakin9 On Demand, 01*(07), 09-11. Retrieved from http://www.billslater.com/writing/Hakin9_Magazine_Article_on_Wireshark_and_Cyberwarfare_by_W_F_Slater_III_2012_1017_.pdf
- *Strategies to protect against distributed denial of service attacks(ddos)*. (2008, April 22). Retrieved from http://www.cisco.com/en/US/tech/tk59/technologies_white_paper09186a0080174a5b.shtml
- TigerDirect.com, Inc. Web Development Team. (2013, April 07). *Zonet zfs3016b rack mountable network switch - 16-port, 10/100, oem*. Retrieved from http://www.tigerdirect.com/applications/SearchTools/item-details.asp?EdpNo=1224096
- Mateti, P. (2012). *Wright.edu*. Retrieved from http://www.cs.wright.edu/~pmateti/InternetSecurity/Lectures/Sniffers/index.html
-