Wireshark Tutorial

What is Wireshark?

Wireshark is an open-source packet analyzer, which is used for **education, analysis, software development, communication protocol development, and network troubleshooting**.

It is used to track the packets so that each one is filtered to meet our specific needs. It is commonly called as a **sniffer, network protocol analyzer, and network analyzer**. It is also used by network security engineers to examine security problems.

Wireshark is a free to use application which is used to apprehend the data back and forth. It is often called as a free packet sniffer computer application. It puts the network card into an unselective mode, i.e., to accept all the packets which it receives.

Uses of Wireshark:

Wireshark can be used in the following ways:

1. It is used by network security engineers to examine security problems.

⇧ SCROLL TO TOP

ers to watch all the traffic being passed over the network.

3. It is used by network engineers to troubleshoot network issues.
4. It also helps to troubleshoot latency issues and malicious activities on your network.
5. It can also analyze dropped packets.
6. It helps us to know how all the devices like laptop, mobile phones, desktop, switch, routers, etc., communicate in a local network or the rest of the world.

What is a packet?

A packet is a unit of data which is transmitted over a network between the origin and the destination. Network packets are small, i.e., maximum **1.5 Kilobytes for Ethernet packets and 64 Kilobytes for IP packets**. The data packets in the Wireshark can be viewed online and can be analyzed offline.

History of Wireshark:

In the late 1990's **Gerald Combs**, a computer science graduate of the University of Missouri-Kansas City was working for the small ISP (Internet Service Provider). The protocol at that time did not complete the primary requirements. So, he started writing **ethereal** and released the first version around 1998. The Network integration services owned the Ethernet trademark.

Combos still held the copyright on most of the ethereal source code, and the rest of the source code was re-distributed under the GNU GPL. He did not own the Ethereal trademark, so he changed the name to Wireshark. He used the contents of the ethereal as the basis.

Wireshark has won several industry rewards over the years including eWeek, InfoWorld, PC Magazine and also as a top-rated packet sniffer. Combos continued the work and released the new version of the software. There are around 600 contributed authors for the Wireshark product website.

Functionality of Wireshark:

Wireshark is similar to tcpdump in networking. **Tcpdump** is a common packet analyzer which allows the user to display other packets and TCP/IP packets, being transmitted and received over a network attached to the computer. It has a graphic end and some sorting and filtering functions. Wireshark users can see all the traffic passing through the network.

Wireshark can also monitor the unicast traffic which is not sent to the network's MAC address interface. But, the switch does not pass all the traffic to the port. Hence, the promiscuous mode is not sufficient to see all the traffic. The various network taps or **port mirroring** is used to extend capture at any point.

⇧ SCROLL TO TOP

Port mirroring is a method to monitor network traffic. When it is enabled, the switch sends the copies of all the network packets present at one port to another port.

What is color coding in Wireshark?

The packets in the Wireshark are highlighted with **blue**, **black**, and **green color**. These colors help users to identify the types of traffic. It is also called as **packet colorization**. The kinds of coloring rules in the Wireshark are **temporary rules** and **permanent rules**.

- The temporary rules are there until the program is in active mode or until we quit the program.
- The permanent color rules are available until the Wireshark is in use or the next time you run the Wireshark. The steps to apply color filters will be discussed later in this topic.

Features of Wireshark

- It is multi-platform software, i.e., it can run on Linux, Windows, OS X, FreeBSD, NetBSD, etc.
- It is a standard three-pane packet browser.
- It performs deep inspection of the hundreds of protocols.
- It often involves live analysis, i.e., from the different types of the network like the Ethernet, loopback, etc., we can read live data.
- It has sort and filter options which makes ease to the user to view the data.
- It is also useful in VoIP analysis.
- It can also capture raw USB traffic.
- Various settings, like timers and filters, can be used to filter the output.
- It can only capture packet on the PCAP (an application programming interface used to capture the network) supported networks.
- Wireshark supports a variety of well-documented capture file formats such as the PcapNg and Libpcap. These formats are used for storing the captured data.
- It is the no.1 piece of software for its purpose. It has countless applications ranging from the **tracing down, unauthorized traffic, firewall settings, etc**.

Installation of Wireshark Software

[illegible] to install the Wireshark software on the computer:

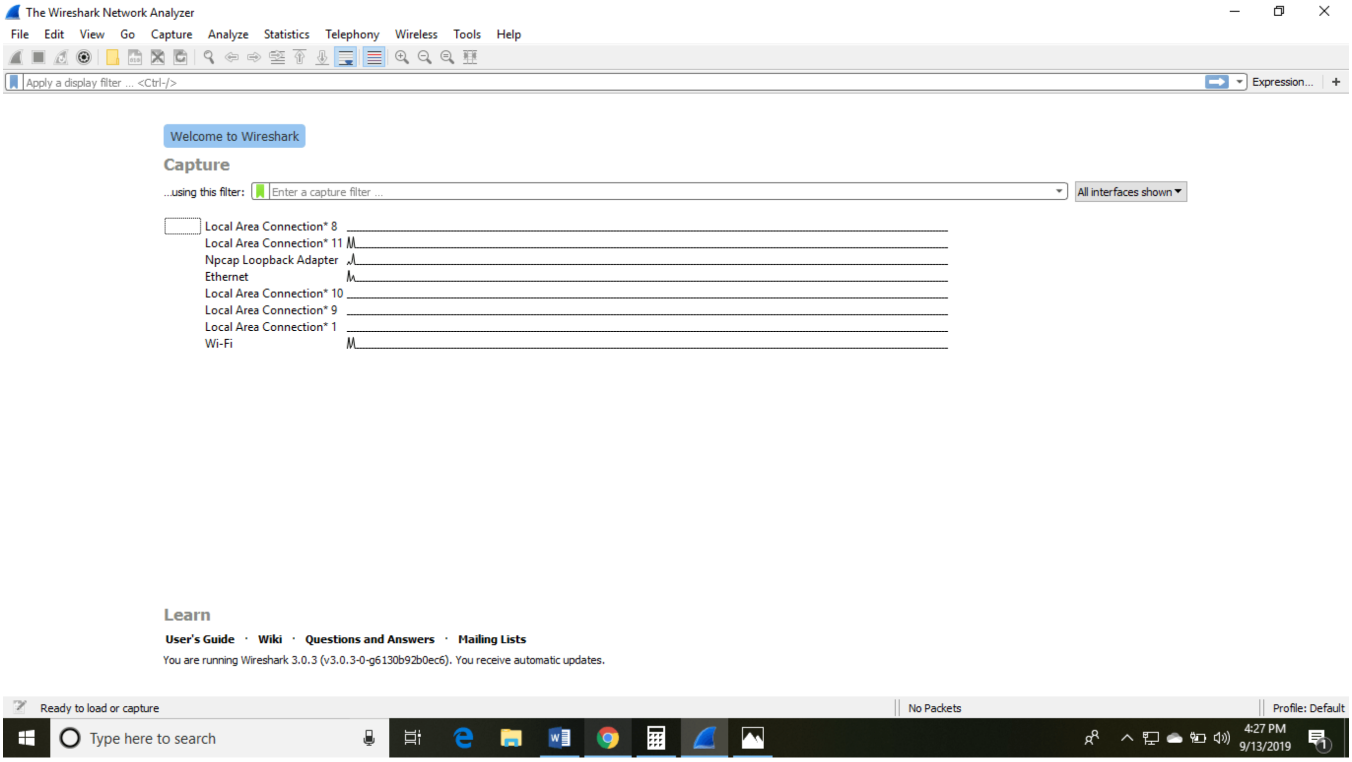
⇧ SCROLL TO TOP

- Open the web browser.
- Search for '**Download Wireshark**.'
- Select the Windows installer according to your system configuration, either 32-bt or 64-bit. Save the program and close the browser.
- Now, open the software, and follow the install instruction by accepting the license.
- The Wireshark is ready for use.

On the network and Internet settings option, we can check the interface connected to our computer.

If you are Linux users, then you will find Wireshark in its package repositories.

By selecting the current interface, we can get the traffic traversing through that interface. The version used here is **3.0.3**. This version will open as:


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
All interfaces shown
Local Area Connection* 8
Local Area Connection* 11
Npcap Loopback Adapter
Ethernet
Local Area Connection* 10
Local Area Connection* 9
Local Area Connection* 1
Wi-Fi
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
Type here to search
4:27 PM
9/13/2019

You are running Wireshark 3.0.3 (v3.0.3-0-g6130b92b0ec6). You receive automatic updates.

Ready to load or capture

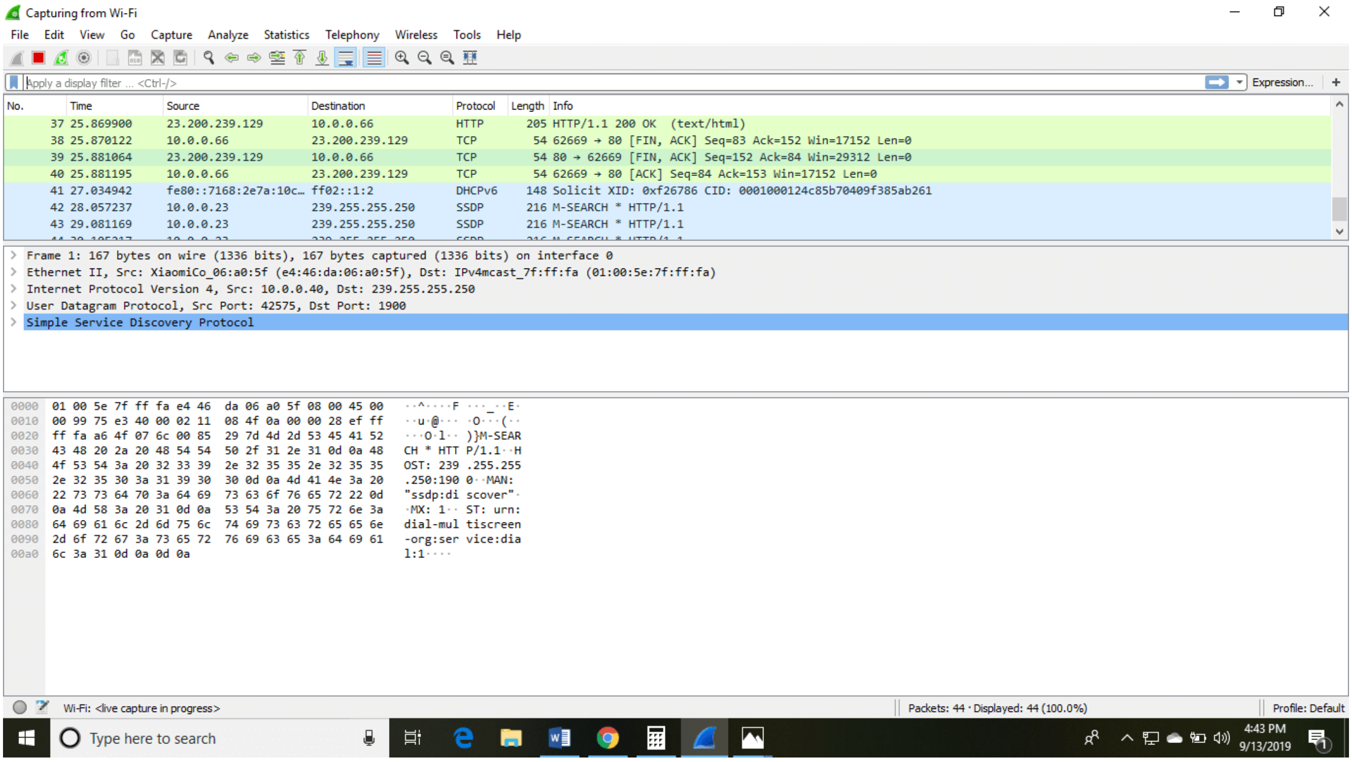
No Packets

Profile: Default

The Wireshark software window is shown above, and all the processes on the network are carried within this screen only.

⇧ SCROLL TO TOP

The options given on the list are the Interface list options. The number of interface options will be present. Selection of any option will determine all the traffic. **For example,** from the above fig. select the Wi-Fi option. After this, a new window opens up, which will show all the current traffic on the network. Below is the image which tells us about the live capture of packets and our Wireshark will look like:



```
No.     Time        Source                 Destination       Protocol  Length  Info
     37 25.869900   23.200.239.129         10.0.0.66         HTTP       205 HTTP/1.1 200 OK  (text/html)
     38 25.870122   10.0.0.66              23.200.239.129    TCP         54 62669 → 80 [FIN, ACK] Seq=83 Ack=152 Win=17152 Len=0
     39 25.881064   23.200.239.129         10.0.0.66         TCP         54 80 → 62669 [FIN, ACK] Seq=152 Ack=84 Win=29312 Len=0
     40 25.881195   10.0.0.66              23.200.239.129    TCP         54 62669 → 80 [ACK] Seq=84 Ack=153 Win=17152 Len=0
     41 27.034942   fe80::7168:2e7a:10c…   ff02::1:2         DHCPv6     148 Solicit XID: 0xf26786 CID: 0001000124c85b70409f385ab261
     42 28.057237   10.0.0.23              239.255.255.250   SSDP       216 M-SEARCH * HTTP/1.1
     43 29.081169   10.0.0.23              239.255.255.250   SSDP       216 M-SEARCH * HTTP/1.1
```

```
> Frame 1: 167 bytes on wire (1336 bits), 167 bytes captured (1336 bits) on interface 0
> Ethernet II, Src: XiaomiCo_06:a0:5f (e4:46:da:06:a0:5f), Dst: IPv4mcast_7f:ff:fa (01:00:5e:7f:ff:fa)
> Internet Protocol Version 4, Src: 10.0.0.40, Dst: 239.255.255.250
> User Datagram Protocol, Src Port: 42575, Dst Port: 1900
> Simple Service Discovery Protocol
```

```
0000  01 00 5e 7f ff fa e4 46  da 06 a0 5f 08 00 45 00   ··^····F ···_··E·
0010  00 99 75 e3 40 00 02 11  08 4f 0a 00 00 28 ef ff   ··u·@··· ·O···(··
0020  ff fa a6 4f 07 6c 00 85  29 7d 4d 2d 53 45 41 52   ···O·l·· )}M-SEAR
0030  43 48 20 2a 20 48 54 54  50 2f 31 2e 31 0d 0a 48   CH * HTT P/1.1··H
0040  4f 53 54 3a 20 32 33 39  2e 32 35 35 2e 32 35 35   OST: 239 .255.255
0050  2e 32 35 30 3a 31 39 30  30 0d 0a 4d 41 4e 3a 20   .250:190 0··MAN:
0060  22 73 73 64 70 3a 64 69  73 63 6f 76 65 72 22 0d   "ssdp:di scover"·
0070  0a 4d 58 3a 20 31 0d 0a  53 54 3a 20 75 72 6e 3a   ·MX: 1·· ST: urn:
0080  64 69 61 6c 2d 6d 75 6c  74 69 73 63 72 65 65 6e   dial-mul tiscreen
0090  2d 6f 72 67 3a 73 65 72  76 69 63 65 3a 64 69 61   -org:ser vice:dia
00a0  6c 3a 31 0d 0a 0d 0a                               l:1····
```

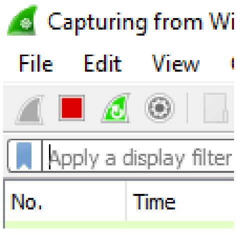
Wi-Fi: <live capture in progress> Packets: 44 · Displayed: 44 (100.0%) Profile: Default

The above arrow shows the packet content written in hexadecimal or the ASCII format. And the information above the packet content, are the details of the packet header.

It will continue listening to all the data packets, and you will get much data. If you want to see a particular data, then you can click on the red button. The traffic will be stationary, and you can note the parameters like time, source, destination, the protocol being used, length, and the Info. To view in-depth detail, you can click on that particular address; a lot of the information will be displayed below that.

⇧ SCROLL TO TOP

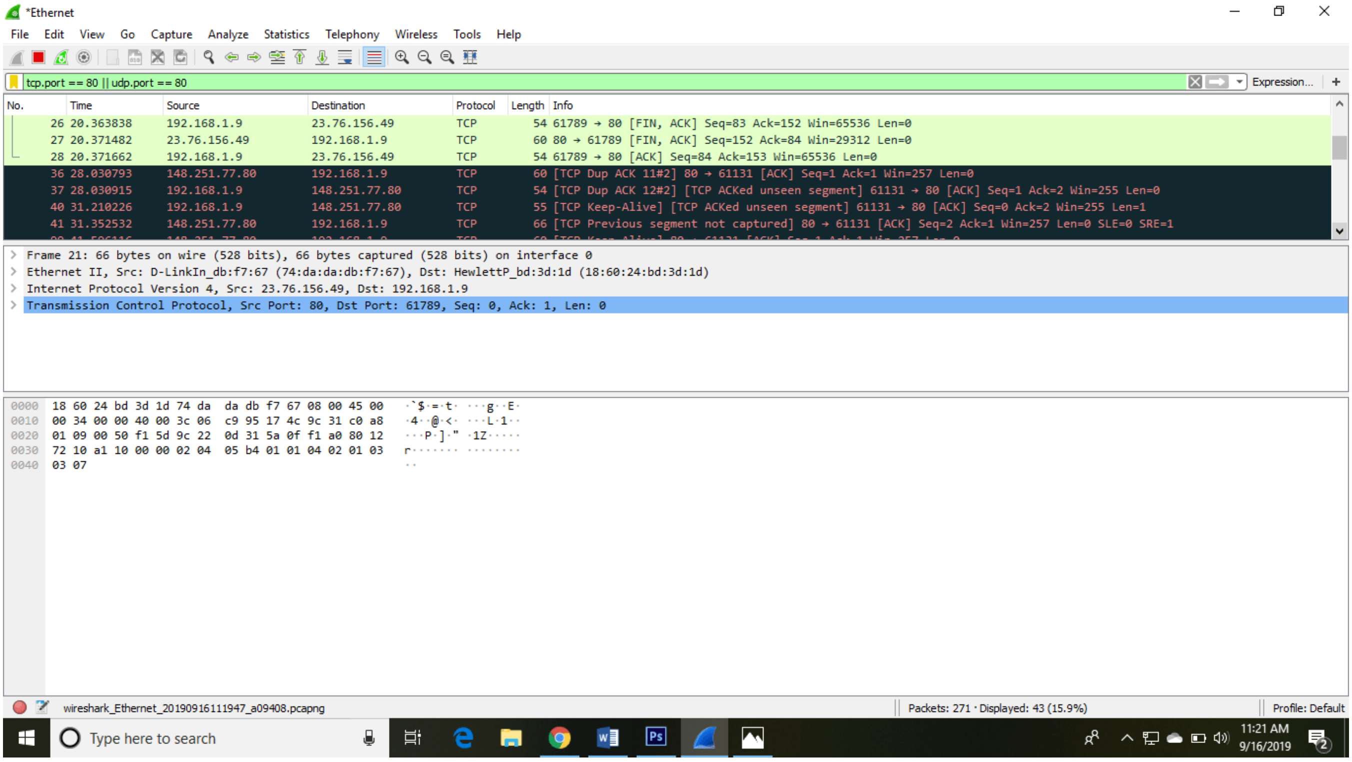
There will be detailed information on HTTP packets, TCP packets, etc. The red button is shown below:



The screen/interface of the Wireshark is divided into five parts:

- First part contains a menu bar and the options displayed below it. This part is at the top of the window. File and the capture menus options are commonly used in Wireshark. The capture menu allows to start the capturing process. And the File menu is used to open and save a capture file.
- The second part is the packet listing window. It determines the packet flow or the captured packets in the traffic. It includes the packet number, time, source, destination, protocol, length, and info. We can sort the packet list by clicking on the column name.
- Next comes the packet header- detailed window. It contains detailed information about the components of the packets. The protocol info can also be expanded or minimized according to the information required.
- The bottom window called the packet contents window, which displays the content in ASCII and hexadecimal format.
- At last, is the filter field which is at the top of the display. The captured packets on the screen can be filtered based on any component according to your requirements. For example, if we want to see only the packets with the HTTP protocol, we can apply filters to that option. All the packets with HTTP as the protocol will only be displayed on the screen, shown below:

⇧ SCROLL TO TOP


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
wireshark_Ethernet_20190916111947_a09408.pcapng
Packets: 271 · Displayed: 43 (15.9%)
Profile: Default
Type here to search
11:21 AM
9/16/2019

```
tcp.port == 80 || udp.port == 80
```

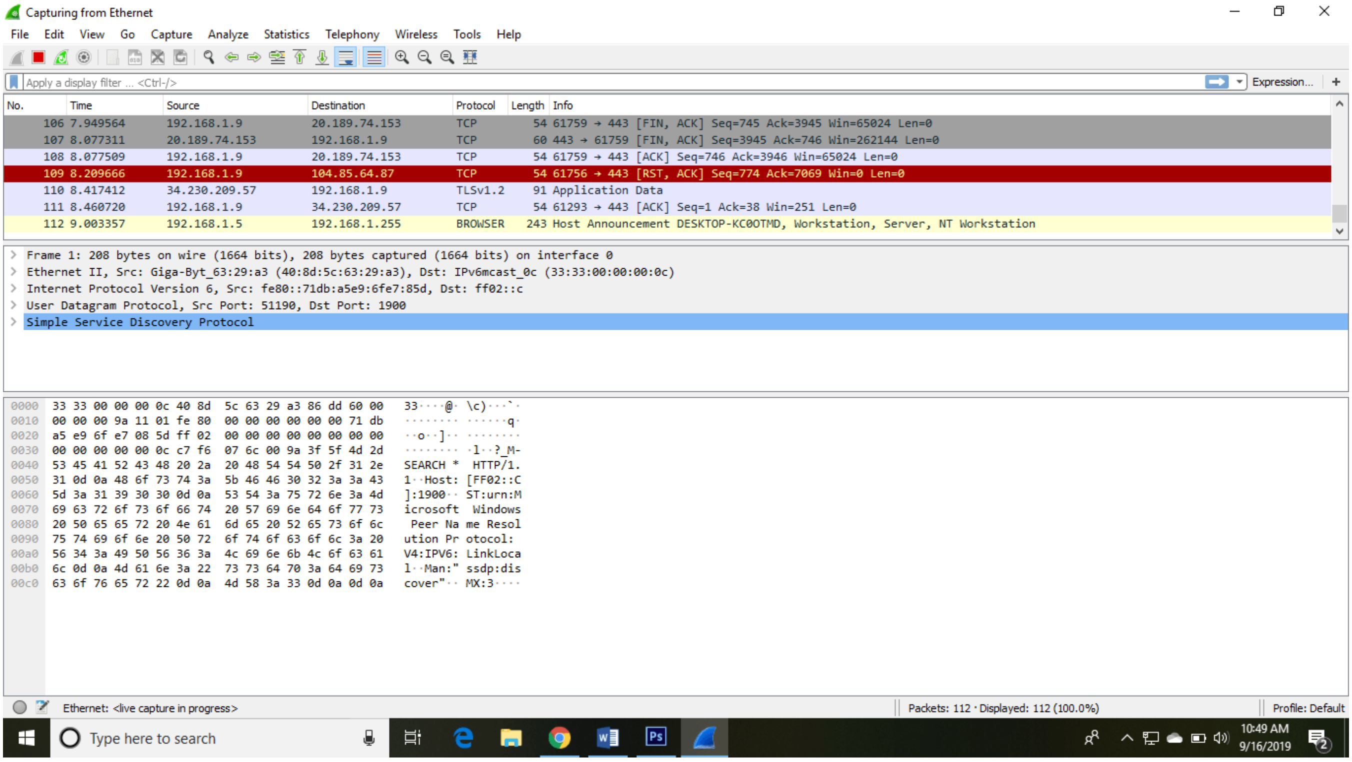
```
No.   Time         Source          Destination     Protocol  Length  Info
   26 20.363838    192.168.1.9     23.76.156.49    TCP          54 61789 → 80 [FIN, ACK] Seq=83 Ack=152 Win=65536 Len=0
   27 20.371482    23.76.156.49    192.168.1.9     TCP          60 80 → 61789 [FIN, ACK] Seq=152 Ack=84 Win=29312 Len=0
   28 20.371662    192.168.1.9     23.76.156.49    TCP          54 61789 → 80 [ACK] Seq=84 Ack=153 Win=65536 Len=0
   36 28.030793    148.251.77.80   192.168.1.9     TCP          60 [TCP Dup ACK 11#2] 80 → 61131 [ACK] Seq=1 Ack=1 Win=257 Len=0
   37 28.030915    192.168.1.9     148.251.77.80   TCP          54 [TCP Dup ACK 12#2] [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
   40 31.210226    192.168.1.9     148.251.77.80   TCP          55 [TCP Keep-Alive] [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=0 Ack=2 Win=255 Len=1
   41 31.352532    148.251.77.80   192.168.1.9     TCP          66 [TCP Previous segment not captured] 80 → 61131 [ACK] Seq=2 Ack=1 Win=257 Len=0 SLE=0 SRE=1
```

```
> Frame 21: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
> Ethernet II, Src: D-LinkIn_db:f7:67 (74:da:da:db:f7:67), Dst: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d)
> Internet Protocol Version 4, Src: 23.76.156.49, Dst: 192.168.1.9
> Transmission Control Protocol, Src Port: 80, Dst Port: 61789, Seq: 0, Ack: 1, Len: 0
```

```
0000  18 60 24 bd 3d 1d 74 da  da db f7 67 08 00 45 00   ·`$·=·t· ···g··E·
0010  00 34 00 00 40 00 3c 06  c9 95 17 4c 9c 31 c0 a8   ·4··@·<· ···L·1··
0020  01 09 00 50 f1 5d 9c 22  0d 31 5a 0f f1 a0 80 12   ···P·]·" ·1Z·····
0030  72 10 a1 10 00 00 02 04  05 b4 01 01 04 02 01 03   r······· ········
0040  03 07                                              ··
```

You can also select the connection to which your computer is connected. For example, in this PC, we have chosen the current network, i.e., the ETHERNET.

After connecting, you can watch the traffic below:


Capturing from Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Ethernet: <live capture in progress>
Packets: 112 · Displayed: 112 (100.0%)
Profile: Default
Type here to search
10:49 AM
9/16/2019

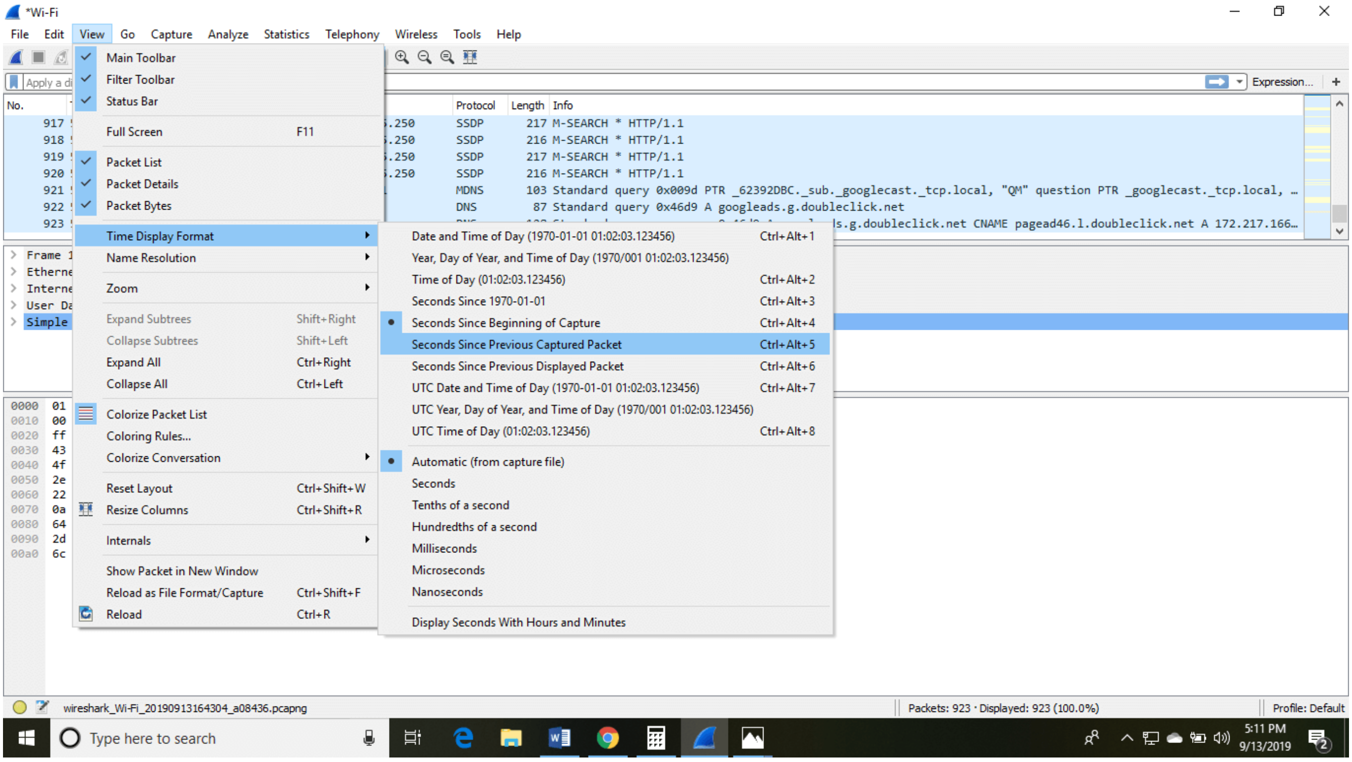
```
No.   Time        Source          Destination     Protocol  Length  Info
  106 7.949564    192.168.1.9     20.189.74.153   TCP          54 61759 → 443 [FIN, ACK] Seq=745 Ack=3945 Win=65024 Len=0
  107 8.077311    20.189.74.153   192.168.1.9     TCP          60 443 → 61759 [FIN, ACK] Seq=3945 Ack=746 Win=262144 Len=0
  108 8.077509    192.168.1.9     20.189.74.153   TCP          54 61759 → 443 [ACK] Seq=746 Ack=3946 Win=65024 Len=0
  109 8.209666    192.168.1.9     104.85.64.87    TCP          54 61756 → 443 [RST, ACK] Seq=774 Ack=7069 Win=0 Len=0
  110 8.417412    34.230.209.57   192.168.1.9     TLSv1.2      91 Application Data
  111 8.460720    192.168.1.9     34.230.209.57   TCP          54 61293 → 443 [ACK] Seq=1 Ack=38 Win=251 Len=0
  112 9.003357    192.168.1.5     192.168.1.255   BROWSER     243 Host Announcement DESKTOP-KC0OTMD, Workstation, Server, NT Workstation
```

```
> Frame 1: 208 bytes on wire (1664 bits), 208 bytes captured (1664 bits) on interface 0
> Ethernet II, Src: Giga-Byt_63:29:a3 (40:8d:5c:63:29:a3), Dst: IPv6mcast_0c (33:33:00:00:00:0c)
> Internet Protocol Version 6, Src: fe80::71db:a5e9:6fe7:85d, Dst: ff02::c
> User Datagram Protocol, Src Port: 51190, Dst Port: 1900
> Simple Service Discovery Protocol
```

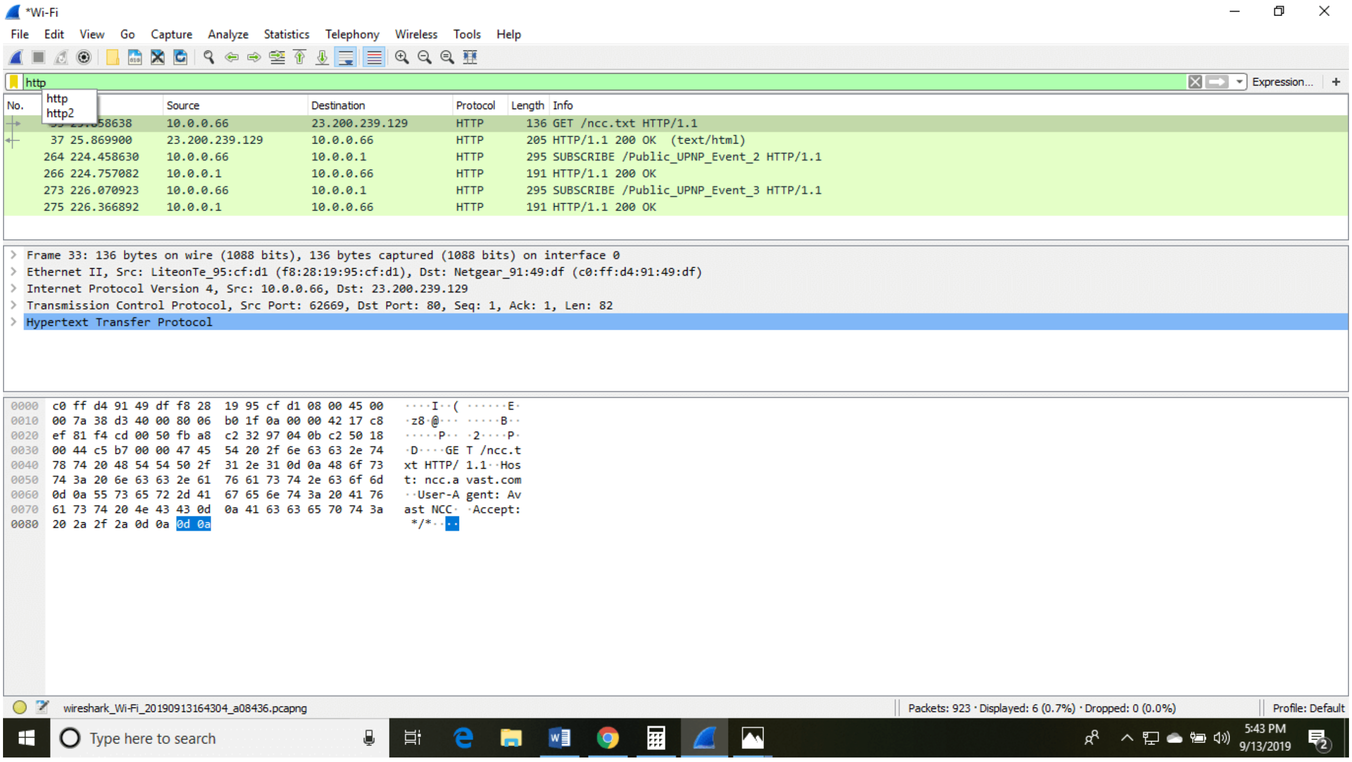
```
0000  33 33 00 00 00 0c 40 8d  5c 63 29 a3 86 dd 60 00   33····@· \c)···`·
0010  00 00 00 9a 11 01 fe 80  00 00 00 00 00 00 71 db   ········ ······q·
0020  a5 e9 6f e7 08 5d ff 02  00 00 00 00 00 00 00 00   ··o··]·· ········
0030  00 00 00 00 00 0c c7 f6  07 6c 00 9a 3f 5f 4d 2d   ········ ·l··?_M-
0040  53 45 41 52 43 48 20 2a  20 48 54 54 50 2f 31 2e   SEARCH *  HTTP/1.
0050  31 0d 0a 48 6f 73 74 3a  5b 46 46 30 32 3a 3a 43   1··Host: [FF02::C
0060  5d 3a 31 39 30 30 0d 0a  53 54 3a 75 72 6e 3a 4d   ]:1900·· ST:urn:M
0070  69 63 72 6f 73 6f 66 74  20 57 69 6e 64 6f 77 73   icrosoft  Windows
0080  20 50 65 65 72 20 4e 61  6d 65 20 52 65 73 6f 6c    Peer Na me Resol
0090  75 74 69 6f 6e 20 50 72  6f 74 6f 63 6f 6c 3a 20   ution Pr otocol:
00a0  56 34 3a 49 50 56 36 3a  4c 69 6e 6b 4c 6f 63 61   V4:IPV6: LinkLoca
00b0  6c 0d 0a 4d 61 6e 3a 22  73 73 64 70 3a 64 69 73   l··Man:" ssdp:dis
00c0  63 6f 76 65 72 22 0d 0a  4d 58 3a 33 0d 0a 0d 0a   cover"·· MX:3····
```

In view option on the menu bar, we can also change the view of the interface. You can change the number of things in the view menu. You can also enable or disable any option according to the requirements.

⇧ SCROLL TO TOP


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a di
Expression...
Main Toolbar
Filter Toolbar
Status Bar
Full Screen F11
Packet List
Packet Details
Packet Bytes
Time Display Format
Name Resolution
Zoom
Expand Subtrees Shift+Right
Collapse Subtrees Shift+Left
Expand All Ctrl+Right
Collapse All Ctrl+Left
Colorize Packet List
Coloring Rules...
Colorize Conversation
Reset Layout Ctrl+Shift+W
Resize Columns Ctrl+Shift+R
Internals
Show Packet in New Window
Reload as File Format/Capture Ctrl+Shift+F
Reload Ctrl+R
Date and Time of Day (1970-01-01 01:02:03.123456) Ctrl+Alt+1
Year, Day of Year, and Time of Day (1970/001 01:02:03.123456)
Time of Day (01:02:03.123456) Ctrl+Alt+2
Seconds Since 1970-01-01 Ctrl+Alt+3
● Seconds Since Beginning of Capture Ctrl+Alt+4
Seconds Since Previous Captured Packet Ctrl+Alt+5
Seconds Since Previous Displayed Packet Ctrl+Alt+6
UTC Date and Time of Day (1970-01-01 01:02:03.123456) Ctrl+Alt+7
UTC Year, Day of Year, and Time of Day (1970/001 01:02:03.123456)
UTC Time of Day (01:02:03.123456) Ctrl+Alt+8
● Automatic (from capture file)
Seconds
Tenths of a second
Hundredths of a second
Milliseconds
Microseconds
Nanoseconds
Display Seconds With Hours and Minutes
No. Protocol Length Info
917 .250 SSDP 217 M-SEARCH * HTTP/1.1
918 .250 SSDP 216 M-SEARCH * HTTP/1.1
919 .250 SSDP 217 M-SEARCH * HTTP/1.1
920 .250 SSDP 216 M-SEARCH * HTTP/1.1
921 MDNS 103 Standard query 0x009d PTR _62392DBC._sub._googlecast._tcp.local, "QM" question PTR _googlecast._tcp.local, …
922 DNS 87 Standard query 0x46d9 A googleads.g.doubleclick.net
923 ls.g.doubleclick.net CNAME pagead46.l.doubleclick.net A 172.217.166…
Frame 1
Etherne
Interne
User Da
Simple
wireshark_Wi-Fi_20190913164304_a08436.pcapng
Packets: 923 · Displayed: 923 (100.0%)
Profile: Default
Type here to search
5:11 PM
9/13/2019

There is a filter block below the menu bar, from where a large amount of data can be filtered. For example, if we apply a filter for HTTP, only the interfaces with the HTTP will be listed.


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
http
http2
Expression...
wireshark_Wi-Fi_20190913164304_a08436.pcapng
Packets: 923 · Displayed: 6 (0.7%) · Dropped: 0 (0.0%)
Profile: Default
Type here to search
5:43 PM
9/13/2019

```
No.                    Source          Destination     Protocol Length Info
    33 25.858638       10.0.0.66       23.200.239.129  HTTP      136 GET /ncc.txt HTTP/1.1
    37 25.869900       23.200.239.129  10.0.0.66       HTTP      205 HTTP/1.1 200 OK  (text/html)
   264 224.458630      10.0.0.66       10.0.0.1        HTTP      295 SUBSCRIBE /Public_UPNP_Event_2 HTTP/1.1
   266 224.757082      10.0.0.1        10.0.0.66       HTTP      191 HTTP/1.1 200 OK
   273 226.070923      10.0.0.66       10.0.0.1        HTTP      295 SUBSCRIBE /Public_UPNP_Event_3 HTTP/1.1
   275 226.366892      10.0.0.1        10.0.0.66       HTTP      191 HTTP/1.1 200 OK
```

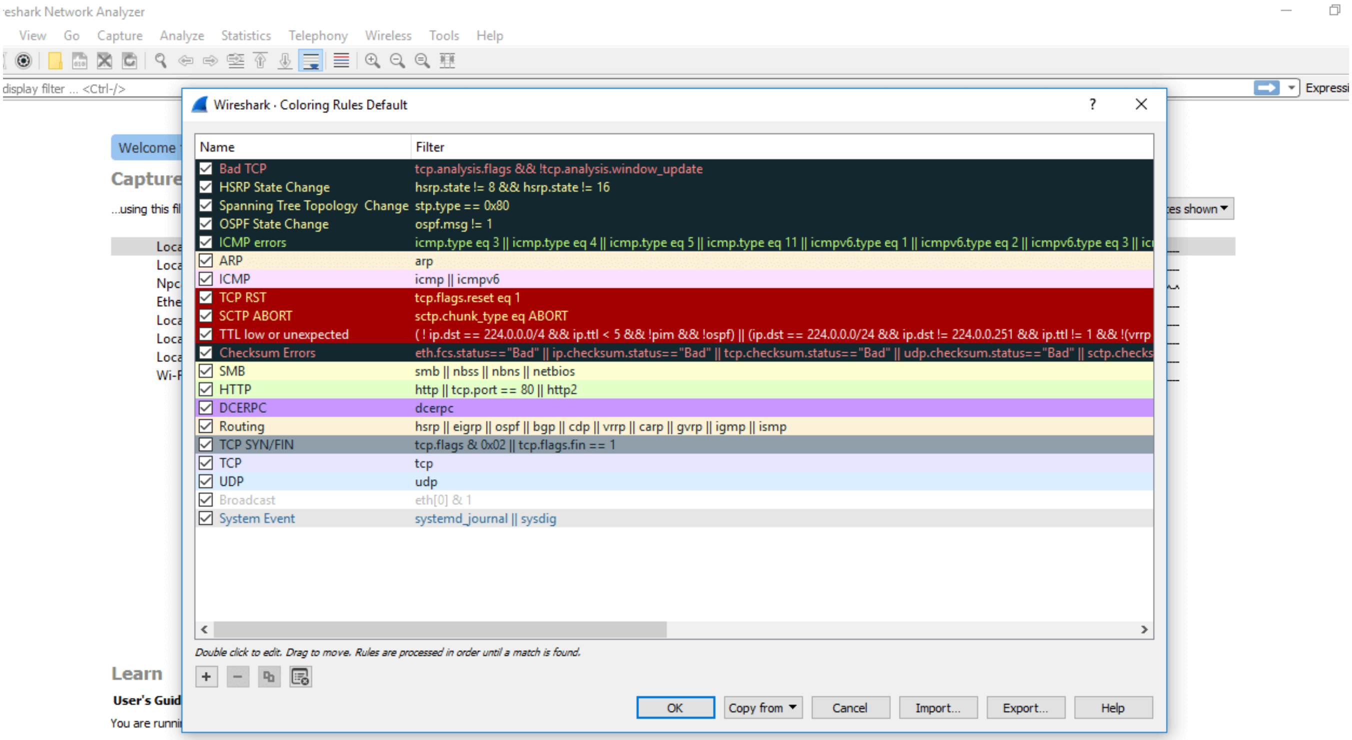
```
> Frame 33: 136 bytes on wire (1088 bits), 136 bytes captured (1088 bits) on interface 0
> Ethernet II, Src: LiteonTe_95:cf:d1 (f8:28:19:95:cf:d1), Dst: Netgear_91:49:df (c0:ff:d4:91:49:df)
> Internet Protocol Version 4, Src: 10.0.0.66, Dst: 23.200.239.129
> Transmission Control Protocol, Src Port: 62669, Dst Port: 80, Seq: 1, Ack: 1, Len: 82
> Hypertext Transfer Protocol
```

```
0000  c0 ff d4 91 49 df f8 28  19 95 cf d1 08 00 45 00   ····I··( ······E·
0010  00 7a 38 d3 40 00 80 06  b0 1f 0a 00 00 42 17 c8   ·z8·@··· ·····B··
0020  ef 81 f4 cd 00 50 fb a8  c2 32 97 04 0b c2 50 18   ·····P·· ·2····P·
0030  00 44 c5 b7 00 00 47 45  54 20 2f 6e 63 63 2e 74   ·D····GE T /ncc.t
0040  78 74 20 48 54 54 50 2f  31 2e 31 0d 0a 48 6f 73   xt HTTP/ 1.1··Hos
0050  74 3a 20 6e 63 63 2e 61  76 61 73 74 2e 63 6f 6d   t: ncc.a vast.com
0060  0d 0a 55 73 65 72 2d 41  67 65 6e 74 3a 20 41 76   ··User-A gent: Av
0070  61 73 74 20 4e 43 43 0d  0a 41 63 63 65 70 74 3a   ast NCC· ·Accept:
0080  20 2a 2f 2a 0d 0a 0d 0a                             */*····
```

If you want to filter according to the source, right-click on the source you want to filter and select 'Apply as Filter' and choose '...and filter.'

Steps for the permanent colorization are: click on the 'View' option on the menu bar and select 'Coloring Rules.' The table will appear like the image shown below:

⇧ SCROLL TO TOP



Name	Filter
☑ Bad TCP	tcp.analysis.flags && !tcp.analysis.window_update
☑ HSRP State Change	hsrp.state != 8 && hsrp.state != 16
☑ Spanning Tree Topology Change	stp.type == 0x80
☑ OSPF State Change	ospf.msg != 1
☑ ICMP errors	icmp.type eq 3 || icmp.type eq 4 || icmp.type eq 5 || icmp.type eq 11 || icmpv6.type eq 1 || icmpv6.type eq 2 || icmpv6.type eq 3 || ic
☑ ARP	arp
☑ ICMP	icmp || icmpv6
☑ TCP RST	tcp.flags.reset eq 1
☑ SCTP ABORT	sctp.chunk_type eq ABORT
☑ TTL low or unexpected	(! ip.dst == 224.0.0.0/4 && ip.ttl < 5 && !pim && !ospf) || (ip.dst == 224.0.0.0/24 && ip.dst != 224.0.0.251 && ip.ttl != 1 && !(vrrp
☑ Checksum Errors	eth.fcs.status=="Bad" || ip.checksum.status=="Bad" || tcp.checksum.status=="Bad" || udp.checksum.status=="Bad" || sctp.checks
☑ SMB	smb || nbss || nbns || netbios
☑ HTTP	http || tcp.port == 80 || http2
☑ DCERPC	dcerpc
☑ Routing	hsrp || eigrp || ospf || bgp || cdp || vrrp || carp || gvrp || igmp || ismp
☑ TCP SYN/FIN	tcp.flags & 0x02 || tcp.flags.fin == 1
☑ TCP	tcp
☑ UDP	udp
☑ Broadcast	eth[0] & 1
☑ System Event	systemd_journal || sysdig

Double click to edit. Drag to move. Rules are processed in order until a match is found.

For the network administrator job, advanced knowledge of Wireshark is considered as the requirements. So, it is essential to understand the concepts of the software. It contains these 20 default coloring rules which can be added or removed according to the requirements.

Select the option '**View**' and then choose '**Colorize Packet List**,' which is used to **toggle the color on and off**.

> Note: If you are not sure about the version of your desktop or the laptop, then you can download the 32-bit Wireshark which will run almost 99% on every type of computers

Now let's start with this basics-

Basic concepts of the Network Traffic

⇧ SCROLL TO TOP

IP Addresses: It was designed for the devices to communicate with each other on a local network or over the Internet. It is used for host or network interface identification. It provides the location of the host and capacity of establishing the path to the host in that network. Internet Protocol is the set of predefined rules or terms under which the communication should be conducted. The types of IP addresses are **IPv4 and IPv6**.

- IPv4 is a **32-bit address** in which each group represents 8 bits ranging from 0 to 255.
- IPv6 is a 128-bit address.

IP addresses are assigned to the host either dynamically or static IP address. Most of the private users have dynamic IP address while business users or servers have a static IP address. Dynamic address changes whenever the device is connected to the Internet.

Computer Ports: The computer ports work in combination with the IP address directing all outgoing and incoming packets to their proper places. There are well-known ports to work with like **FTP** (File Transfer Protocol), which has port no. 21, etc. All the ports have the purpose of directing all packets in the predefined direction.

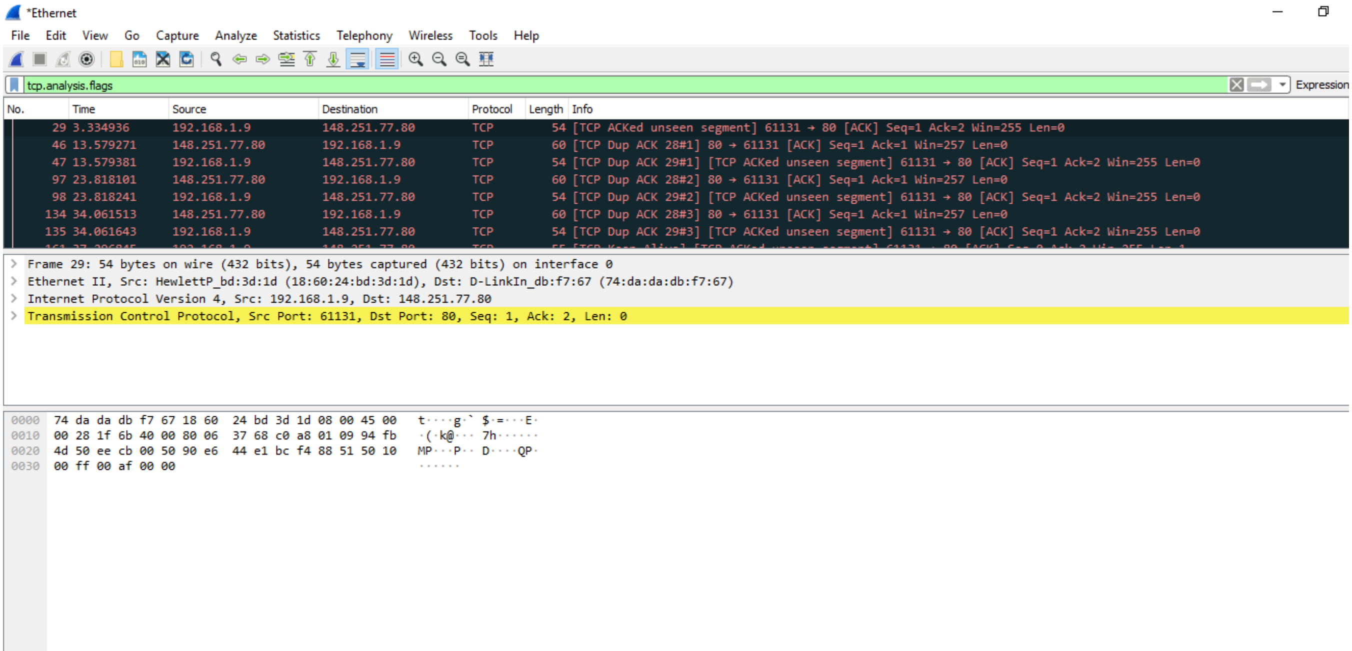
Protocol: The Protocol is a set of predefined rules. They are considered as the standardized way of communication. One of the most used protocol is **TCP/IP**. It stands for **Transmission Control Protocol/ Internet Protocol**.

OSI model: OSI model stands for **Open System Interconnect**. OSI model has seven layers, namely, **Application layer, Presentation layer, Session layer, Transport layer, Network layer, Data link layer, and the physical layer**. OSI model gives a detail representation and explanation of the transmission and reception of data through the layers. OSI model supports both connectionless and connection-oriented communication mode over the network layer. The OSI model was developed by ISO (International Standard Organization).

Most used Filters in Wireshark

Whenever we type any commands in the filter command box, it turns **green** if your command is **correct**. It turns **red** if it is **incorrect** or the Wireshark does not recognize your command.

⇧ SCROLL TO TOP


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression

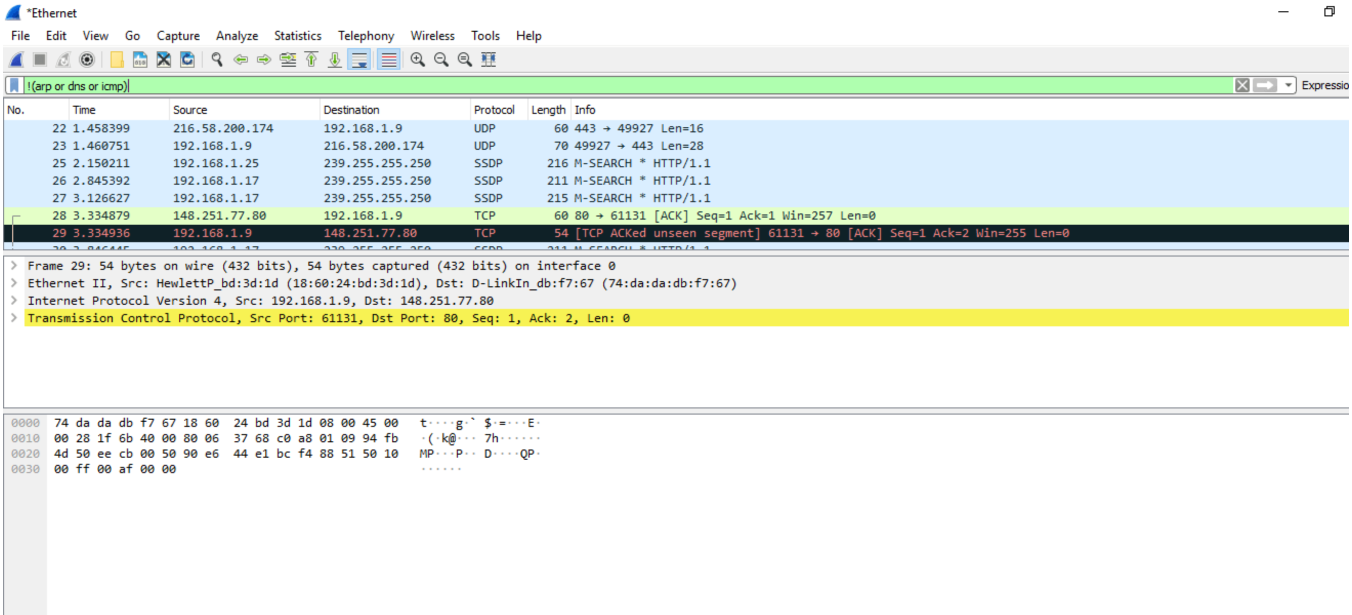
tcp.analysis.flags

```
No.   Time       Source          Destination     Protocol Length Info
   29 3.334936   192.168.1.9     148.251.77.80   TCP      54 [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
   46 13.579271  148.251.77.80   192.168.1.9     TCP      60 [TCP Dup ACK 28#1] 80 → 61131 [ACK] Seq=1 Ack=1 Win=257 Len=0
   47 13.579381  192.168.1.9     148.251.77.80   TCP      54 [TCP Dup ACK 29#1] [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
   97 23.818101  148.251.77.80   192.168.1.9     TCP      60 [TCP Dup ACK 28#2] 80 → 61131 [ACK] Seq=1 Ack=1 Win=257 Len=0
   98 23.818241  192.168.1.9     148.251.77.80   TCP      54 [TCP Dup ACK 29#2] [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
  134 34.061513  148.251.77.80   192.168.1.9     TCP      60 [TCP Dup ACK 28#3] 80 → 61131 [ACK] Seq=1 Ack=1 Win=257 Len=0
  135 34.061643  192.168.1.9     148.251.77.80   TCP      54 [TCP Dup ACK 29#3] [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
```

```
> Frame 29: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Dst: D-LinkIn_db:f7:67 (74:da:da:db:f7:67)
> Internet Protocol Version 4, Src: 192.168.1.9, Dst: 148.251.77.80
> Transmission Control Protocol, Src Port: 61131, Dst Port: 80, Seq: 1, Ack: 2, Len: 0
```

```
0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45 00   t····g·` $·=···E·
0010  00 28 1f 6b 40 00 80 06  37 68 c0 a8 01 09 94 fb   ·(·k@··· 7h······
0020  4d 50 ee cb 00 50 90 e6  44 e1 bc f4 88 51 50 10   MP···P·· D····QP·
0030  00 ff 00 af 00 00                                  ······
```

Fig (5)


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expressio

!(arp or dns or icmp)

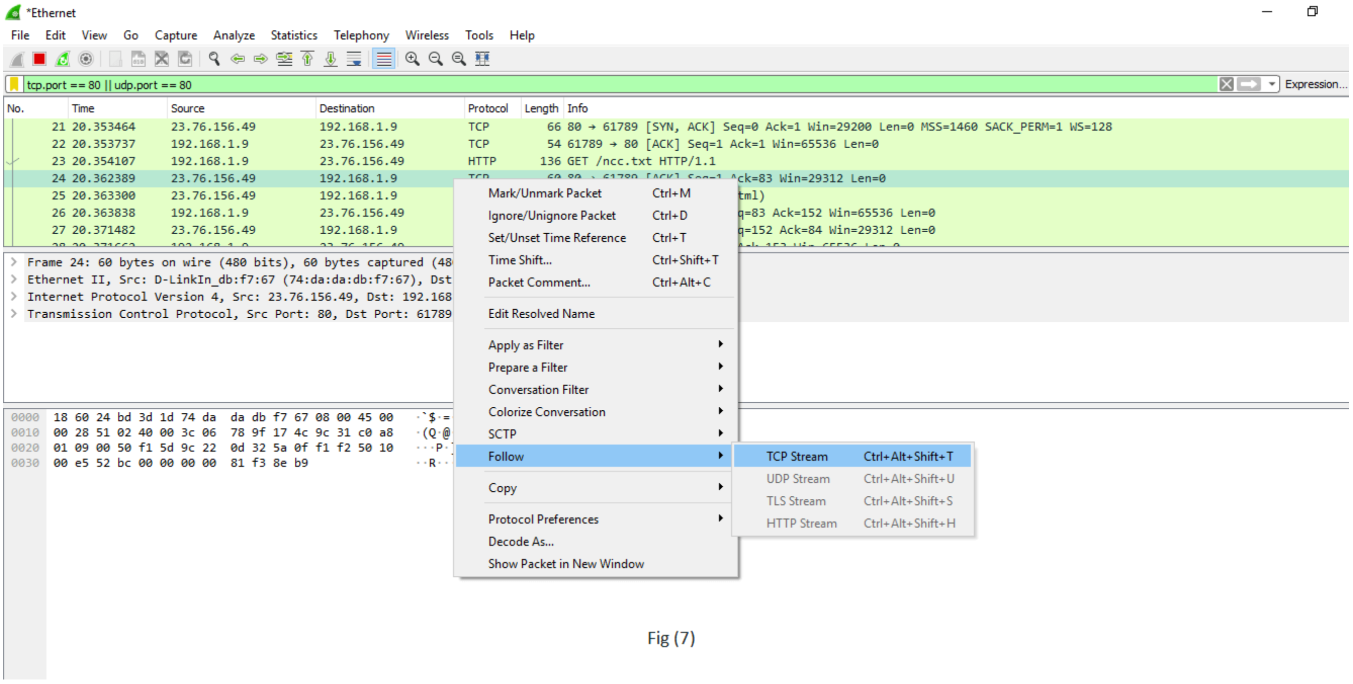
```
No.   Time       Source          Destination      Protocol Length Info
   22 1.458399   216.58.200.174  192.168.1.9      UDP      60 443 → 49927 Len=16
   23 1.460751   192.168.1.9     216.58.200.174   UDP      70 49927 → 443 Len=28
   25 2.150211   192.168.1.25    239.255.255.250  SSDP     216 M-SEARCH * HTTP/1.1
   26 2.845392   192.168.1.17    239.255.255.250  SSDP     211 M-SEARCH * HTTP/1.1
   27 3.126627   192.168.1.17    239.255.255.250  SSDP     215 M-SEARCH * HTTP/1.1
   28 3.334879   148.251.77.80   192.168.1.9      TCP      60 80 → 61131 [ACK] Seq=1 Ack=1 Win=257 Len=0
   29 3.334936   192.168.1.9     148.251.77.80    TCP      54 [TCP ACKed unseen segment] 61131 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
```

```
> Frame 29: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Dst: D-LinkIn_db:f7:67 (74:da:da:db:f7:67)
> Internet Protocol Version 4, Src: 192.168.1.9, Dst: 148.251.77.80
> Transmission Control Protocol, Src Port: 61131, Dst Port: 80, Seq: 1, Ack: 2, Len: 0
```

```
0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45 00   t····g·` $·=···E·
0010  00 28 1f 6b 40 00 80 06  37 68 c0 a8 01 09 94 fb   ·(·k@··· 7h······
0020  4d 50 ee cb 00 50 90 e6  44 e1 bc f4 88 51 50 10   MP···P·· D····QP·
0030  00 ff 00 af 00 00                                  ······
```

Fig (6)

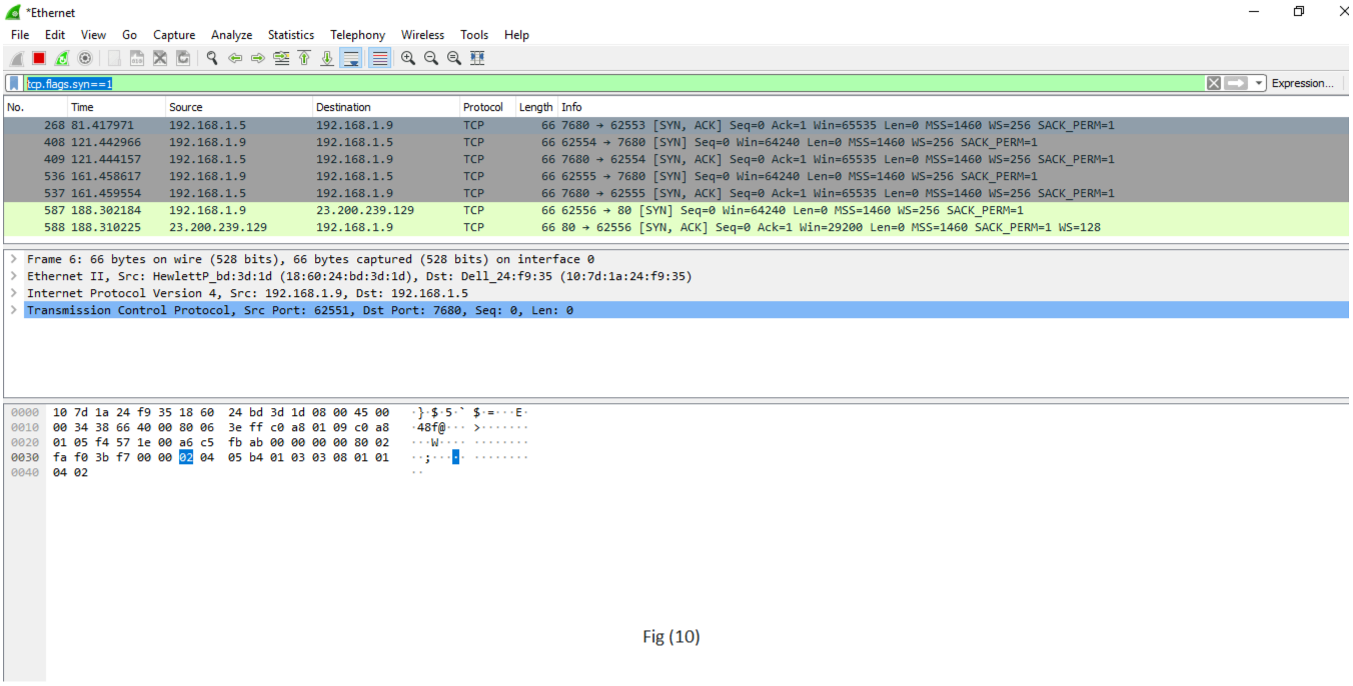
⇧ SCROLL TO TOP



*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
Mark/Unmark Packet Ctrl+M
Ignore/Unignore Packet Ctrl+D
Set/Unset Time Reference Ctrl+T
Time Shift... Ctrl+Shift+T
Packet Comment... Ctrl+Alt+C
Edit Resolved Name
Apply as Filter
Prepare a Filter
Conversation Filter
Colorize Conversation
SCTP
Follow
Copy
Protocol Preferences
Decode As...
Show Packet in New Window
TCP Stream Ctrl+Alt+Shift+T
UDP Stream Ctrl+Alt+Shift+U
TLS Stream Ctrl+Alt+Shift+S
HTTP Stream Ctrl+Alt+Shift+H

```
tcp.port == 80 || udp.port == 80

No.  Time       Source        Destination   Protocol Length Info
  21 20.353464  23.76.156.49  192.168.1.9   TCP        66 80 → 61789 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  22 20.353737  192.168.1.9   23.76.156.49  TCP        54 61789 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
  23 20.354107  192.168.1.9   23.76.156.49  HTTP      136 GET /ncc.txt HTTP/1.1
  24 20.362389  23.76.156.49  192.168.1.9             Ack=83 Win=29312 Len=0
  25 20.363300  23.76.156.49  192.168.1.9             tml)
  26 20.363838  192.168.1.9   23.76.156.49            q=83 Ack=152 Win=65536 Len=0
  27 20.371482  23.76.156.49  192.168.1.9             q=152 Ack=84 Win=29312 Len=0

> Frame 24: 60 bytes on wire (480 bits), 60 bytes captured (48
> Ethernet II, Src: D-LinkIn_db:f7:67 (74:da:da:db:f7:67), Dst
> Internet Protocol Version 4, Src: 23.76.156.49, Dst: 192.168
> Transmission Control Protocol, Src Port: 80, Dst Port: 61789

0000  18 60 24 bd 3d 1d 74 da  da db f7 67 08 00 45 00   ·`$·=
0010  00 28 51 02 40 00 3c 06  78 9f 17 4c 9c 31 c0 a8   ·(Q·@
0020  01 09 00 50 f1 5d 9c 22  0d 32 5a 0f f1 f2 50 10   ···P·
0030  00 e5 52 bc 00 00 00 00  81 f3 8e b9               ··R··
```

Fig (7)



*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...

```
tcp.flags.syn==1

No.  Time        Source          Destination     Protocol Length Info
 268 81.417971   192.168.1.5     192.168.1.9     TCP        66 7680 → 62553 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 408 121.442966  192.168.1.9     192.168.1.5     TCP        66 62554 → 7680 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
 409 121.444157  192.168.1.5     192.168.1.9     TCP        66 7680 → 62554 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 536 161.458617  192.168.1.9     192.168.1.5     TCP        66 62555 → 7680 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
 537 161.459554  192.168.1.5     192.168.1.9     TCP        66 7680 → 62555 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 587 188.302184  192.168.1.9     23.200.239.129  TCP        66 62556 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
 588 188.310225  23.200.239.129  192.168.1.9     TCP        66 80 → 62556 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128

> Frame 6: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Dst: Dell_24:f9:35 (10:7d:1a:24:f9:35)
> Internet Protocol Version 4, Src: 192.168.1.9, Dst: 192.168.1.5
> Transmission Control Protocol, Src Port: 62551, Dst Port: 7680, Seq: 0, Len: 0

0000  10 7d 1a 24 f9 35 18 60  24 bd 3d 1d 08 00 45 00   ·}·$·5·` $·=···E·
0010  00 34 38 66 40 00 80 06  3e ff c0 a8 01 09 c0 a8   ·48f@··· >·······
0020  01 05 f4 57 1e 00 a6 c5  fb ab 00 00 00 00 80 02   ···W···· ········
0030  fa f0 3b f7 00 00 02 04  05 b4 01 03 03 08 01 01   ··;····· ········
0040  04 02                                              ··
```

Fig (10)

Below is the list of filters used in Wireshark:

Filters	Description
ip.addr Example- ip.addr==10.0.10.142 ip.src ip.dst	It is used to specify the IP address as the source or the destination. This example will filter based on this IP address as a source and a destination. If we want for a particular source or destination then, It is used for the source filter. It is used for the destination.

⇧ SCROLL TO TOP

protocol Example- dns or http 'Dns and http' is never used.	This command filters based on the protocol. It requires the packet to be either dns protocol or http protocol and will display the traffic based on this. We would not use the command 'dns and http' because it requires the packet to be both, dns as well as http, which is impossible.
tcp.port Example: tcp.port==443	It sets filter based on the specific port number. It will filter all the packets with this port number.
4. udp.port	It is same as tcp.port. Instead, udp is used.
tcp.analysis.flags example is shown in **fig(5)**.	Wireshark can flag TCP problems. This command will only display the issues that Wireshark identifies. Example, packet loss, tcp segment not captured, etc. are some of the problems. It quickly identifies the problem and is widely used.
6.!() For example, !(arp or dns or icmp) This is shown in **fig (6)**.	It is used to filter the list of protocols or applications, in which we are not interested. It will remove arp, dns, and icmp, and only the remaining will be left or it clean the things that may not be helpful.
Select any packet. Right-click on it and select 'Follow' and then select' TCP stream.' Shown in fig. (7).	It is used if you want to work on a single connection on a TCP conversation. Anything related to the single TCP connection will be displayed on the screen.
tcp contains the filter For example- tcp contains Facebook Or udp contains Facebook	It is used to display the packets which contain such words. In this, Facebook word in any packet in this trace file i.e., finding the devices, which are talking to Facebook. This command is useful if you are looking for a username, word, etc.
http.request For the responses or the response code, you can type http.response.code==200	It will display all the http requests in the trace file. You can see all the servers, the client is involved.
tcp.flags.syn==1 This is shown in fig (10). tcp.flags.reset	This will display all the packets with the sync built-in tcp header set to 1. This will show all the packets with tcp resets.

⇧ SCROLL TO TOP

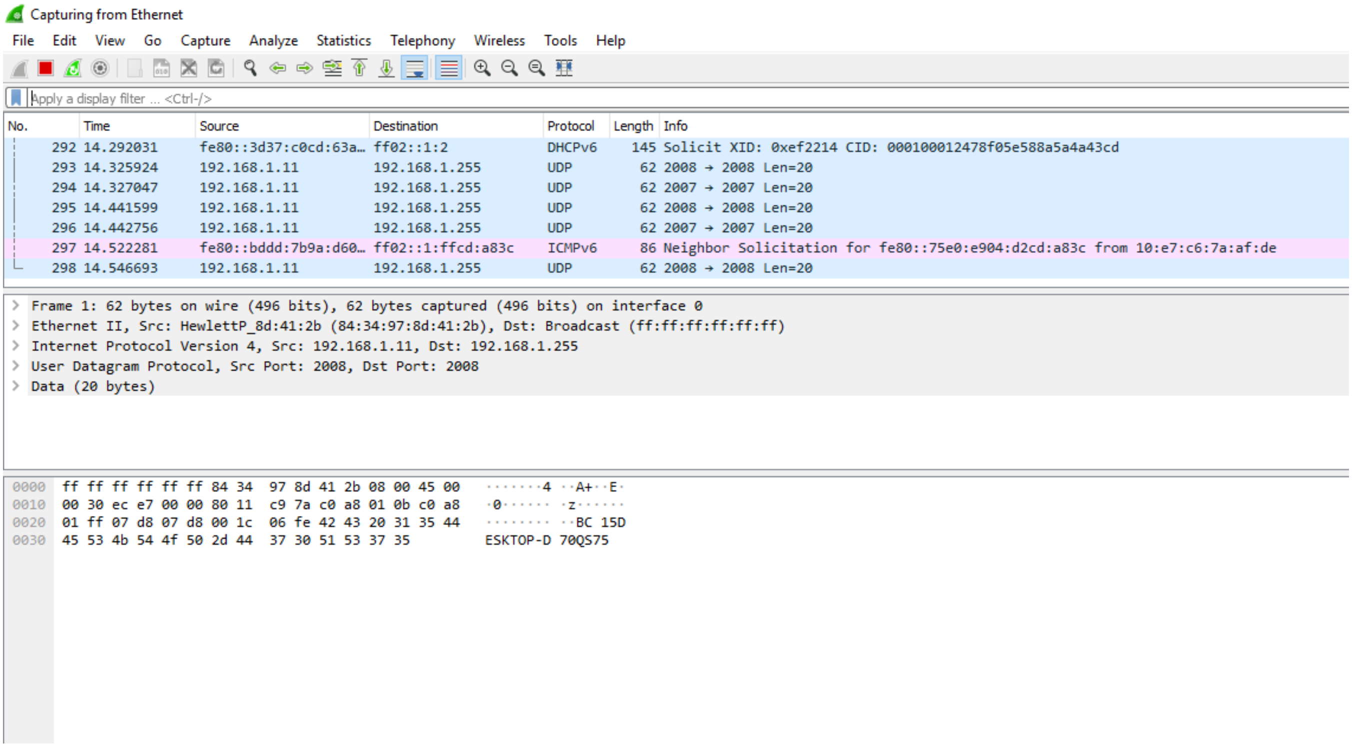
Wireshark packet sniffing

Wireshark is a packet sniffing program that administrators can use to isolate and troubleshoot problems on the network. It can also be used to capture sensitive data like usernames and passwords. It can also be used in wrong way (hacking) to ease drop.

Packet sniffing is defined as the process to capture the packets of data flowing across a computer network. The Packet sniffer is a device or software used for the process of sniffing.

Below are the steps for packet sniffing:

- Open the Wireshark Application.
- Select the current interface. Here in this example, interface is Ethernet that we would be using.
- The network traffic will be shown below, which will be continuous. To stop or watch any particular packet, you can press the red button below the menu bar.



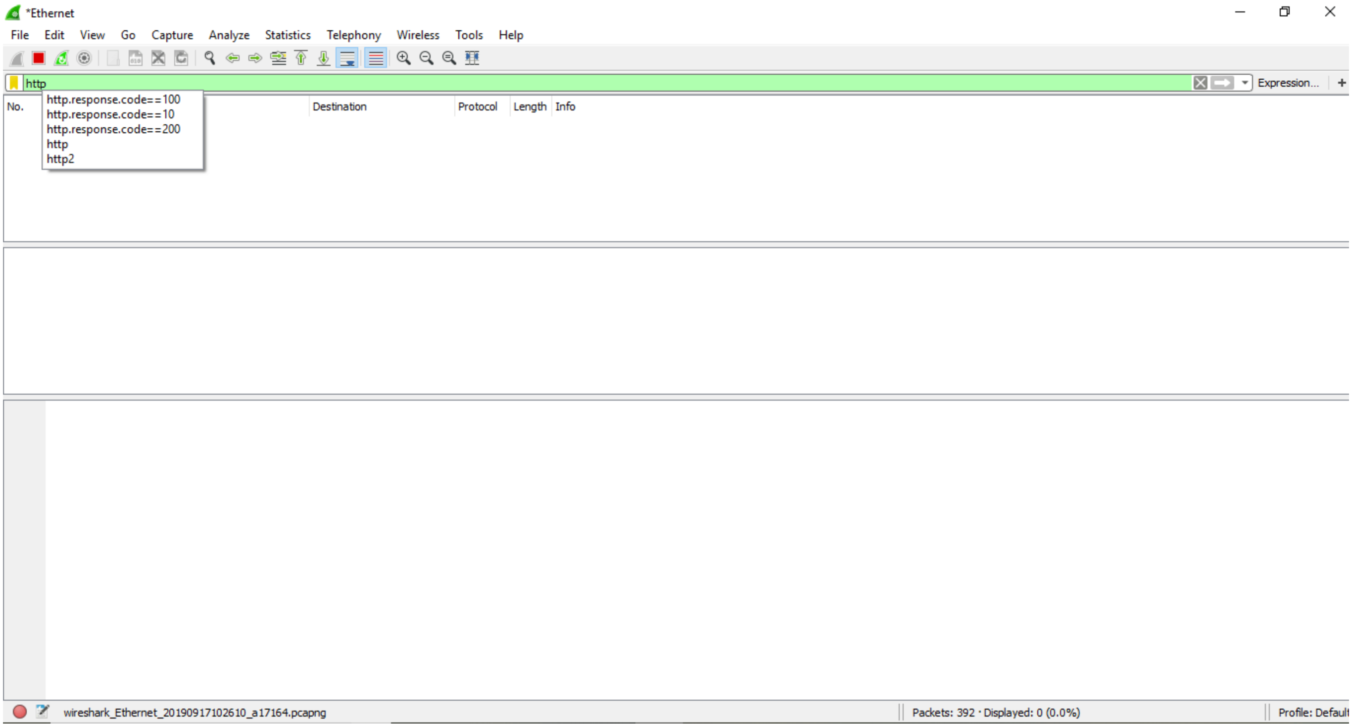
No.	Time	Source	Destination	Protocol	Length	Info
292	14.292031	fe80::3d37:c0cd:63a…	ff02::1:2	DHCPv6	145	Solicit XID: 0xef2214 CID: 000100012478f05e588a5a4a43cd
293	14.325924	192.168.1.11	192.168.1.255	UDP	62	2008 → 2008 Len=20
294	14.327047	192.168.1.11	192.168.1.255	UDP	62	2007 → 2007 Len=20
295	14.441599	192.168.1.11	192.168.1.255	UDP	62	2008 → 2008 Len=20
296	14.442756	192.168.1.11	192.168.1.255	UDP	62	2007 → 2007 Len=20
297	14.522281	fe80::bddd:7b9a:d60…	ff02::1:ffcd:a83c	ICMPv6	86	Neighbor Solicitation for fe80::75e0:e904:d2cd:a83c from 10:e7:c6:7a:af:de
298	14.546693	192.168.1.11	192.168.1.255	UDP	62	2008 → 2008 Len=20

```
> Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0
> Ethernet II, Src: HewlettP_8d:41:2b (84:34:97:8d:41:2b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 192.168.1.11, Dst: 192.168.1.255
> User Datagram Protocol, Src Port: 2008, Dst Port: 2008
> Data (20 bytes)
```

```
0000  ff ff ff ff ff ff 84 34  97 8d 41 2b 08 00 45 00   ·······4 ··A+··E·
0010  00 30 ec e7 00 00 80 11  c9 7a c0 a8 01 0b c0 a8   ·0······ ·z······
0020  01 ff 07 d8 07 d8 00 1c  06 fe 42 43 20 31 35 44   ········ ··BC 15D
0030  45 53 4b 54 4f 50 2d 44  37 30 51 53 37 35         ESKTOP-D 70QS75
```

Apply the filter by the name 'http.' After the filter is applied, the screen will look as:

⇧ SCROLL TO TOP



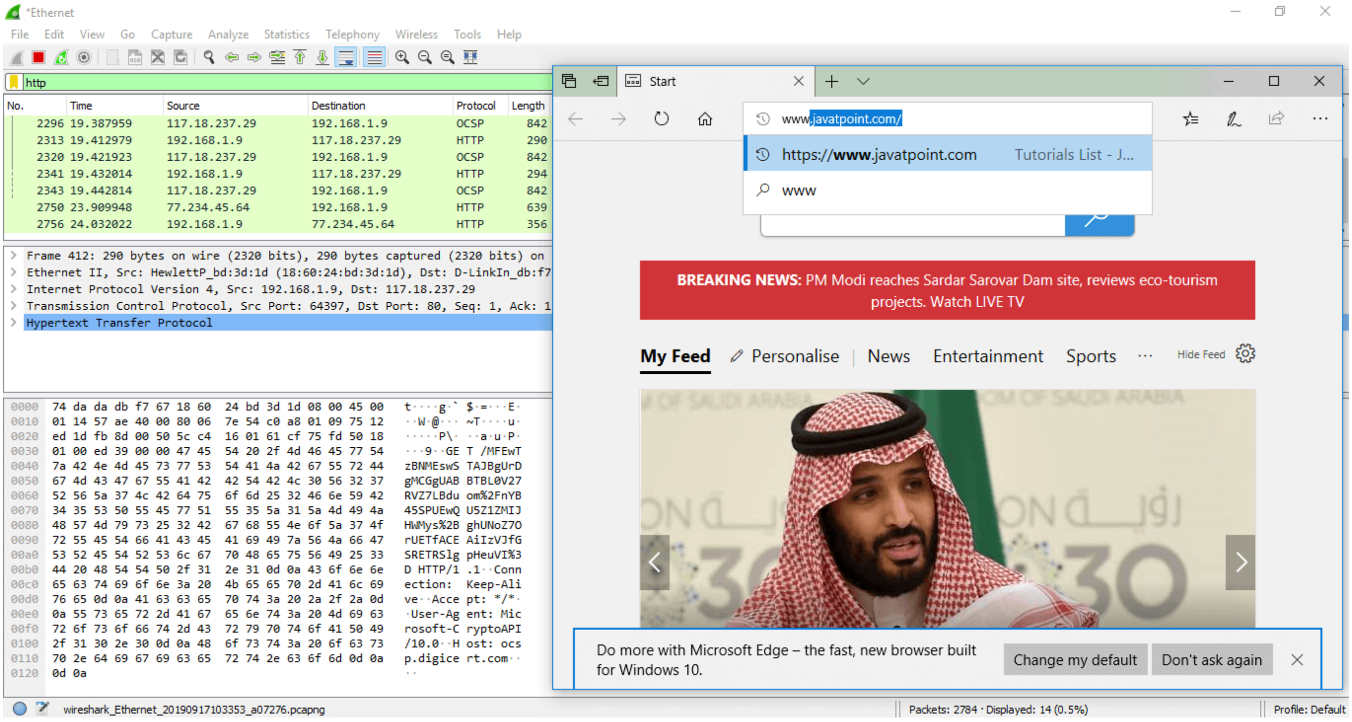
*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
Expression...
No.
http.response.code==100
http.response.code==10
http.response.code==200
http
http2
Destination
Protocol
Length
Info
wireshark_Ethernet_20190917102610_a17164.pcapng
Packets: 392 · Displayed: 0 (0.0%)
Profile: Default

The above screen is blank, i.e.; there is no network traffic as of now.

Open the browser. In this example, we have opened the 'Internet Explorer.' You can choose any browser.

As soon as we open the browser, and type any address of the website, the traffic will start showing, and exchange of the packets will also start. The image for this is shown below:

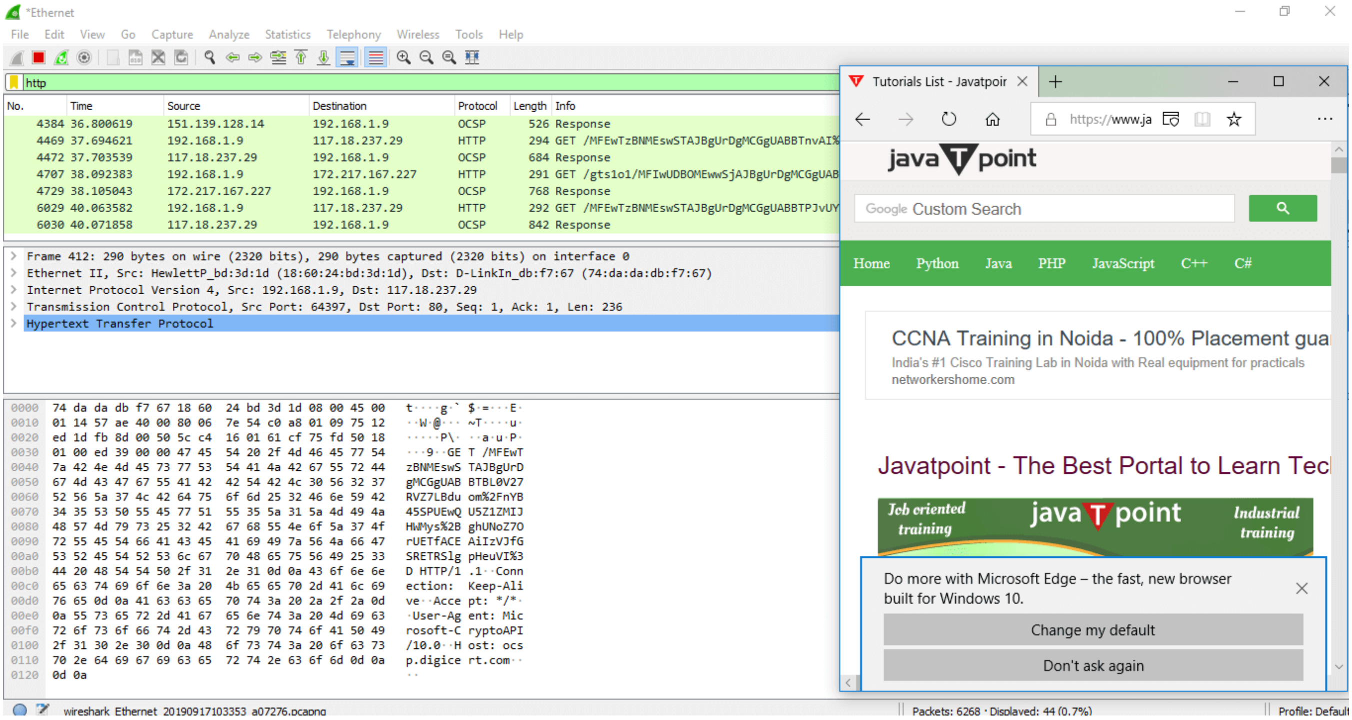
⇧ SCROLL TO TOP



```
No.   Time        Source          Destination     Protocol Length
 2296 19.387959   117.18.237.29   192.168.1.9     OCSP     842
 2313 19.412979   192.168.1.9     117.18.237.29   HTTP     290
 2320 19.421923   117.18.237.29   192.168.1.9     OCSP     842
 2341 19.432014   192.168.1.9     117.18.237.29   HTTP     294
 2343 19.442814   117.18.237.29   192.168.1.9     OCSP     842
 2750 23.909948   77.234.45.64    192.168.1.9     HTTP     639
 2756 24.032022   192.168.1.9     77.234.45.64    HTTP     356

> Frame 412: 290 bytes on wire (2320 bits), 290 bytes captured (2320 bits) on
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Dst: D-LinkIn_db:f7
> Internet Protocol Version 4, Src: 192.168.1.9, Dst: 117.18.237.29
> Transmission Control Protocol, Src Port: 64397, Dst Port: 80, Seq: 1, Ack: 1
> Hypertext Transfer Protocol

0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45 00   t····g·` $·=···E·
0010  01 14 57 ae 40 00 80 06  7e 54 c0 a8 01 09 75 12   ··W·@··· ~T····u·
0020  ed 1d fb 8d 00 50 5c c4  16 01 61 cf 75 fd 50 18   ·····P\· ··a·u·P·
0030  01 00 ed 39 00 00 47 45  54 20 2f 4d 46 45 77 54   ···9··GE T /MFEwT
0040  7a 42 4e 4d 45 73 77 53  54 41 4a 42 67 55 72 44   zBNMEswS TAJBgUrD
0050  67 4d 43 47 67 55 41 42  42 54 42 4c 30 56 32 37   gMCGgUAB BTBL0V27
0060  52 56 5a 37 4c 42 64 75  6f 6d 25 32 46 6e 59 42   RVZ7LBdu om%2FnYB
0070  34 35 53 50 55 45 77 51  55 35 5a 31 5a 4d 49 4a   45SPUEwQ U5Z1ZMIJ
0080  48 57 4d 79 73 25 32 42  67 68 55 4e 6f 5a 37 4f   HWMys%2B ghUNoZ7O
0090  72 55 45 54 66 41 43 45  41 69 49 7a 56 4a 66 47   rUETfACE AiIzVJfG
00a0  53 52 45 54 52 53 6c 67  70 48 65 75 56 49 25 33   SRETRSlg pHeuVI%3
00b0  44 20 48 54 54 50 2f 31  2e 31 0d 0a 43 6f 6e 6e   D HTTP/1 .1··Conn
00c0  65 63 74 69 6f 6e 3a 20  4b 65 65 70 2d 41 6c 69   ection:  Keep-Ali
00d0  76 65 0d 0a 41 63 63 65  70 74 3a 20 2a 2f 2a 0d   ve··Acce pt: */*·
00e0  0a 55 73 65 72 2d 41 67  65 6e 74 3a 20 4d 69 63   ·User-Ag ent: Mic
00f0  72 6f 73 6f 66 74 2d 43  72 79 70 74 6f 41 50 49   rosoft-C ryptoAPI
0100  2f 31 30 2e 30 0d 0a 48  6f 73 74 3a 20 6f 63 73   /10.0··H ost: ocs
0110  70 2e 64 69 67 69 63 65  72 74 2e 63 6f 6d 0d 0a   p.digice rt.com··
0120  0d 0a                                              ··

wireshark_Ethernet_20190917103353_a07276.pcapng    Packets: 2784 · Displayed: 14 (0.5%)
```

BREAKING NEWS: PM Modi reaches Sardar Sarovar Dam site, reviews eco-tourism projects. Watch LIVE TV

Do more with Microsoft Edge – the fast, new browser built for Windows 10.



```
No.   Time        Source           Destination      Protocol Length Info
 4384 36.800619   151.139.128.14   192.168.1.9      OCSP      526 Response
 4469 37.694621   192.168.1.9      117.18.237.29    HTTP      294 GET /MFEwTzBNMEswSTAJBgUrDgMCGgUABBTnvAI%
 4472 37.703539   117.18.237.29    192.168.1.9      OCSP      684 Response
 4707 38.092383   192.168.1.9      172.217.167.227  HTTP      291 GET /gts1o1/MFIwUDBOMEwwSjAJBgUrDgMCGgUAB
 4729 38.105043   172.217.167.227  192.168.1.9      OCSP      768 Response
 6029 40.063582   192.168.1.9      117.18.237.29    HTTP      292 GET /MFEwTzBNMEswSTAJBgUrDgMCGgUABBTPJvUY
 6030 40.071858   117.18.237.29    192.168.1.9      OCSP      842 Response

> Frame 412: 290 bytes on wire (2320 bits), 290 bytes captured (2320 bits) on interface 0
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Dst: D-LinkIn_db:f7:67 (74:da:da:db:f7:67)
> Internet Protocol Version 4, Src: 192.168.1.9, Dst: 117.18.237.29
> Transmission Control Protocol, Src Port: 64397, Dst Port: 80, Seq: 1, Ack: 1, Len: 236
> Hypertext Transfer Protocol

0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45 00   t····g·` $·=···E·
0010  01 14 57 ae 40 00 80 06  7e 54 c0 a8 01 09 75 12   ··W·@··· ~T····u·
0020  ed 1d fb 8d 00 50 5c c4  16 01 61 cf 75 fd 50 18   ·····P\· ··a·u·P·
0030  01 00 ed 39 00 00 47 45  54 20 2f 4d 46 45 77 54   ···9··GE T /MFEwT
0040  7a 42 4e 4d 45 73 77 53  54 41 4a 42 67 55 72 44   zBNMEswS TAJBgUrD
0050  67 4d 43 47 67 55 41 42  42 54 42 4c 30 56 32 37   gMCGgUAB BTBL0V27
0060  52 56 5a 37 4c 42 64 75  6f 6d 25 32 46 6e 59 42   RVZ7LBdu om%2FnYB
0070  34 35 53 50 55 45 77 51  55 35 5a 31 5a 4d 49 4a   45SPUEwQ U5Z1ZMIJ
0080  48 57 4d 79 73 25 32 42  67 68 55 4e 6f 5a 37 4f   HWMys%2B ghUNoZ7O
0090  72 55 45 54 66 41 43 45  41 69 49 7a 56 4a 66 47   rUETfACE AiIzVJfG
00a0  53 52 45 54 52 53 6c 67  70 48 65 75 56 49 25 33   SRETRSlg pHeuVI%3
00b0  44 20 48 54 54 50 2f 31  2e 31 0d 0a 43 6f 6e 6e   D HTTP/1 .1··Conn
00c0  65 63 74 69 6f 6e 3a 20  4b 65 65 70 2d 41 6c 69   ection:  Keep-Ali
00d0  76 65 0d 0a 41 63 63 65  70 74 3a 20 2a 2f 2a 0d   ve··Acce pt: */*·
00e0  0a 55 73 65 72 2d 41 67  65 6e 74 3a 20 4d 69 63   ·User-Ag ent: Mic
00f0  72 6f 73 6f 66 74 2d 43  72 79 70 74 6f 41 50 49   rosoft-C ryptoAPI
0100  2f 31 30 2e 30 0d 0a 48  6f 73 74 3a 20 6f 63 73   /10.0··H ost: ocs
0110  70 2e 64 69 67 69 63 65  72 74 2e 63 6f 6d 0d 0a   p.digice rt.com··
0120  0d 0a                                              ··

wireshark_Ethernet_20190917103353_a07276.pcapng    Packets: 6268 · Displayed: 44 (0.7%)
```

CCNA Training in Noida - 100% Placement gua
India's #1 Cisco Training Lab in Noida with Real equipment for practicals
networkershome.com

Javatpoint - The Best Portal to Learn Tec

Do more with Microsoft Edge – the fast, new browser built for Windows 10.

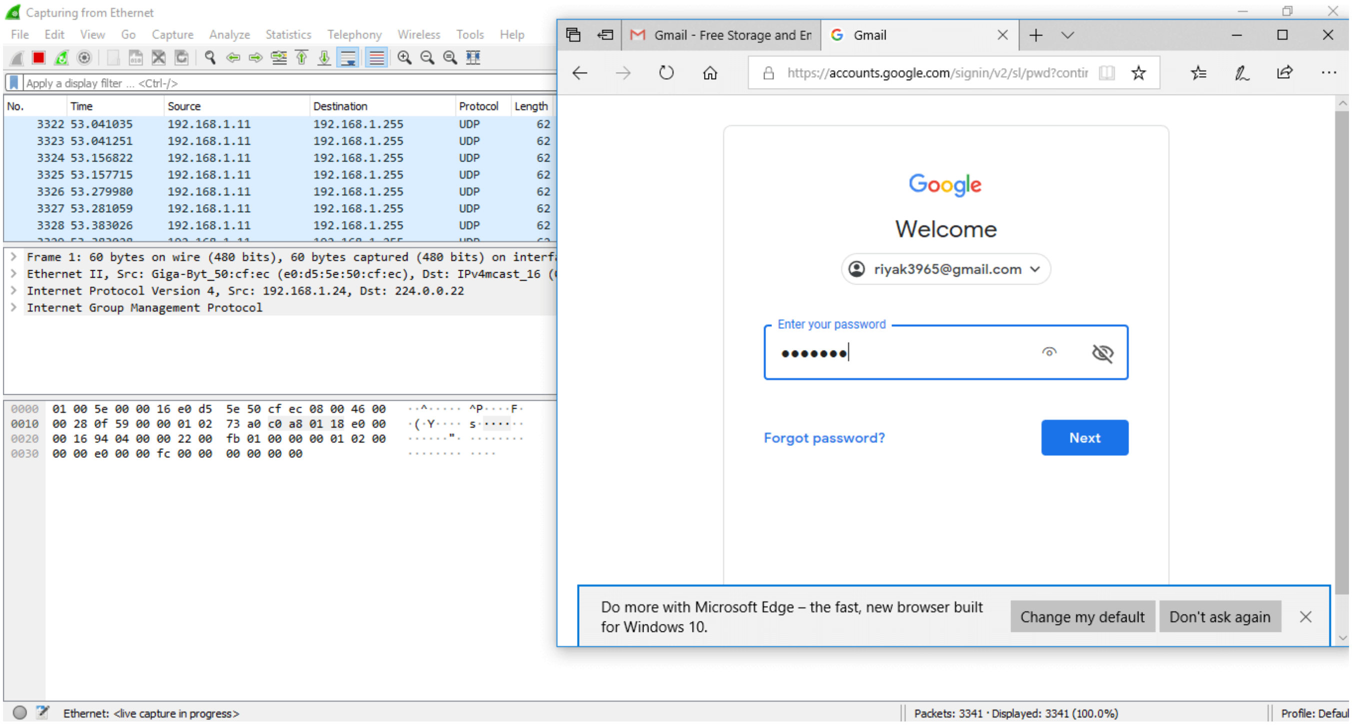
The above process explained is called as **packet sniffing**.

Username and password sniffing

It is the process used to know the passwords and username for the particular website. Let's take an example of gmail.com. Below are the steps:

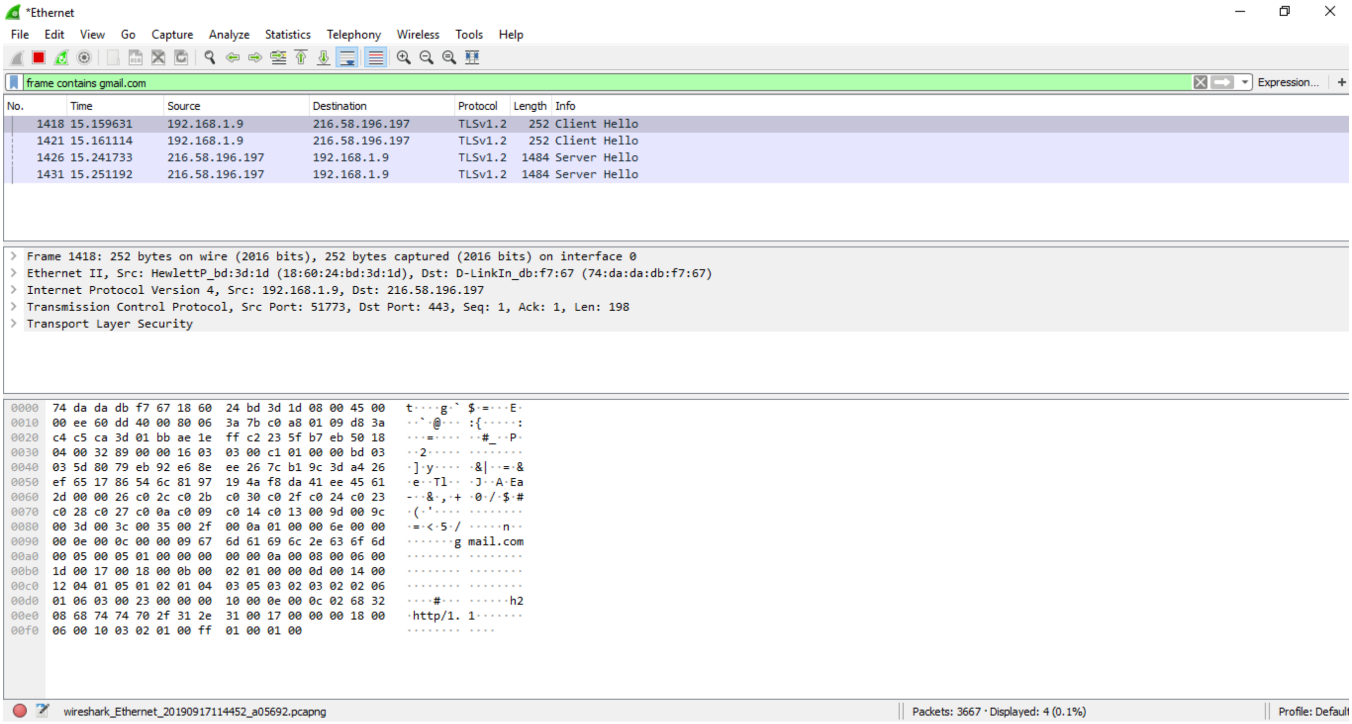
- Open the Wireshark and select the suitable interface.
- Open the browser and enter the web address. Here, we have entered gmail.com, which is highly secured. Enter your email address and the password. The image is shown below:

⇧ SCROLL TO TOP


Capturing from Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Gmail - Free Storage and En
Gmail
https://accounts.google.com/signin/v2/sl/pwd?contir
Google
Welcome
riyak3965@gmail.com
Enter your password
•••••••
Forgot password?
Next
Do more with Microsoft Edge – the fast, new browser built for Windows 10.
Change my default
Don't ask again
Ethernet: <live capture in progress>
Packets: 3341 · Displayed: 3341 (100.0%)
Profile: Defaul

```
No.   Time        Source         Destination     Protocol  Length
 3322 53.041035   192.168.1.11   192.168.1.255   UDP       62
 3323 53.041251   192.168.1.11   192.168.1.255   UDP       62
 3324 53.156822   192.168.1.11   192.168.1.255   UDP       62
 3325 53.157715   192.168.1.11   192.168.1.255   UDP       62
 3326 53.279980   192.168.1.11   192.168.1.255   UDP       62
 3327 53.281059   192.168.1.11   192.168.1.255   UDP       62
 3328 53.383026   192.168.1.11   192.168.1.255   UDP       62

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interf
> Ethernet II, Src: Giga-Byt_50:cf:ec (e0:d5:5e:50:cf:ec), Dst: IPv4mcast_16 (
> Internet Protocol Version 4, Src: 192.168.1.24, Dst: 224.0.0.22
> Internet Group Management Protocol

0000  01 00 5e 00 00 16 e0 d5  5e 50 cf ec 08 00 46 00   ··^····· ^P····F·
0010  00 28 0f 59 00 00 01 02  73 a0 c0 a8 01 18 e0 00   ·(·Y···· s·······
0020  00 16 94 04 00 00 22 00  fb 01 00 00 00 01 02 00   ······"· ········
0030  00 00 e0 00 00 fc 00 00  00 00 00 00               ········ ····
```

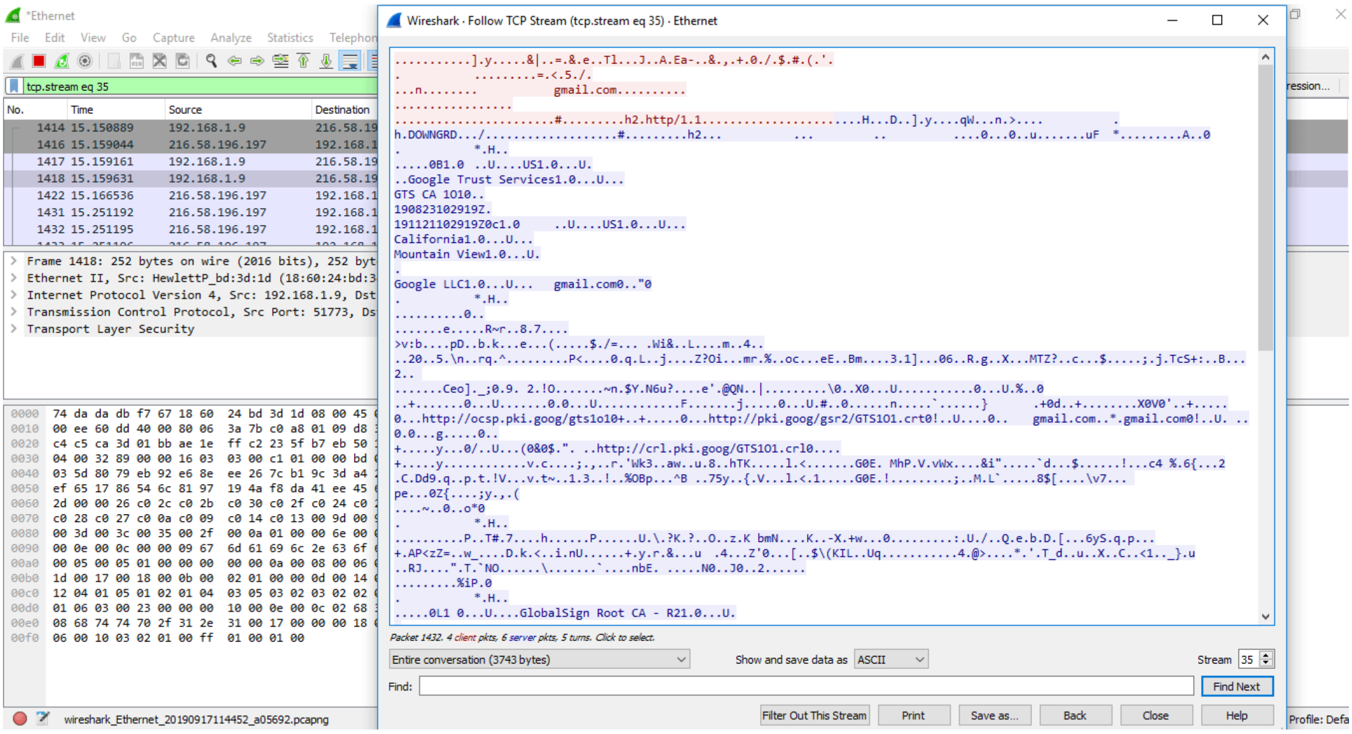
- Now, go to the Wireshark and on the filters block, enter 'frame contains gmail.com.' Then you can see some traffic.


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
wireshark_Ethernet_20190917114452_a05692.pcapng
Packets: 3667 · Displayed: 4 (0.1%)
Profile: Defaul

```
frame contains gmail.com

No.   Time        Source          Destination     Protocol  Length Info
 1418 15.159631   192.168.1.9     216.58.196.197  TLSv1.2    252 Client Hello
 1421 15.161114   192.168.1.9     216.58.196.197  TLSv1.2    252 Client Hello
 1426 15.241733   216.58.196.197  192.168.1.9     TLSv1.2   1484 Server Hello
 1431 15.251192   216.58.196.197  192.168.1.9     TLSv1.2   1484 Server Hello

> Frame 1418: 252 bytes on wire (2016 bits), 252 bytes captured (2016 bits) on interface 0
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Dst: D-LinkIn_db:f7:67 (74:da:da:db:f7:67)
> Internet Protocol Version 4, Src: 192.168.1.9, Dst: 216.58.196.197
> Transmission Control Protocol, Src Port: 51773, Dst Port: 443, Seq: 1, Ack: 1, Len: 198
> Transport Layer Security

0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45 00   t····g·` $·=···E·
0010  00 ee 60 dd 40 00 80 06  3a 7b c0 a8 01 09 d8 3a   ··`·@··· :{·····:
0020  c4 c5 ca 3d 01 bb ae 1e  ff c2 23 5f b7 eb 50 18   ···=···· ··#_··P·
0030  04 00 32 89 00 00 16 03  03 00 c1 01 00 00 bd 03   ··2····· ········
0040  03 5d 80 79 eb 92 e6 8e  ee 26 7c b1 9c 3d a4 26   ·]·y···· ·&|··=·&
0050  ef 65 17 86 54 6c 81 97  19 4a f8 da 41 ee 45 61   ·e··Tl·· ·J··A·Ea
0060  2d 00 00 26 c0 2c c0 2b  c0 30 c0 2f c0 24 c0 23   -··&·,·+ ·0·/·$·#
0070  c0 28 c0 27 c0 0a c0 09  c0 14 c0 13 00 9d 00 9c   ·(·'···· ········
0080  00 3d 00 3c 00 35 00 2f  00 0a 01 00 00 6e 00 00   ·=·<·5·/ ·····n··
0090  00 0e 00 0c 00 00 09 67  6d 61 69 6c 2e 63 6f 6d   ·······g mail.com
00a0  00 05 00 05 01 00 00 00  00 00 0a 00 08 00 06 00   ········ ········
00b0  1d 00 17 00 18 00 0b 00  02 01 00 00 0d 00 14 00   ········ ········
00c0  12 04 01 05 01 02 01 04  03 05 03 02 03 02 02 06   ········ ········
00d0  01 06 03 00 23 00 00 00  10 00 0e 00 0c 02 68 32   ····#··· ······h2
00e0  08 68 74 74 70 2f 31 2e  31 00 17 00 00 00 18 00   ·http/1. 1·······
00f0  06 00 10 03 02 01 00 ff  01 00 01 00               ········ ····
```

- Right-click on the particular network and select 'Follow', and then 'TCP Stream.' You can see that all the data is secured in the encrypted form.

⇧ SCROLL TO TOP



tcp.stream eq 35

```
No.   Time       Source          Destination
 1414 15.150889  192.168.1.9     216.58.19
 1416 15.159044  216.58.196.197  192.168.1
 1417 15.159161  192.168.1.9     216.58.19
 1418 15.159631  192.168.1.9     216.58.19
 1422 15.166536  216.58.196.197  192.168.1
 1431 15.251192  216.58.196.197  192.168.1
 1432 15.251195  216.58.196.197  192.168.1
```

```
> Frame 1418: 252 bytes on wire (2016 bits), 252 byt
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3
> Internet Protocol Version 4, Src: 192.168.1.9, Dst
> Transmission Control Protocol, Src Port: 51773, Ds
> Transport Layer Security
```

```
0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45
0010  00 ee 60 dd 40 00 80 06  3a 7b c0 a8 01 09 d8
0020  c4 c5 ca 3d 01 bb ae 1e  ff c2 23 5f b7 eb 50
0030  04 00 32 89 00 00 16 03  03 00 c1 01 00 00 bd
0040  03 5d 80 79 eb 92 e6 8e  ee 26 7c b1 9c 3d a4
0050  ef 65 17 86 54 6c 81 97  19 4a f8 da 41 ee 45
0060  2d 00 00 26 c0 2c c0 2b  c0 30 c0 2f c0 24 c0
0070  c0 28 c0 27 c0 0a c0 09  c0 14 c0 13 00 9d 00
0080  00 3d 00 3c 00 35 00 2f  00 0a 01 00 00 6e 00
0090  00 0e 00 0c 00 00 09 67  6d 61 69 6c 2e 63 6f
00a0  00 05 00 05 01 00 00 00  00 00 0a 00 08 00 06
00b0  1d 00 17 00 18 00 0b 00  02 01 00 00 0d 00 14
00c0  12 04 01 05 01 02 01 04  03 05 03 02 03 02 02
00d0  01 06 03 00 23 00 00 00  10 00 0e 00 0c 02 68
00e0  08 68 74 74 70 2f 31 2e  31 00 17 00 00 00 18
00f0  06 00 10 03 02 01 00 ff  01 00 01 00
```

wireshark_Ethernet_20190917114452_a05692.pcapng

Wireshark · Follow TCP Stream (tcp.stream eq 35) · Ethernet

```
..........].y.....&|..=.&.e..Tl...J..A.Ea-..&.,.+.0./.$.#.(.'.
.          .........=.<.5./.
...n........          gmail.com.........
.................
.......................#.........h2.http/1.1......................H...D..].y....qW...n.>....          .
h.DOWNGRD.../..................#.........h2...          ...        ..          ....0...0..u.......uF  *.........A..0
.          *.H..
.....0B1.0  ..U....US1.0...U.
..Google Trust Services1.0...U...
GTS CA 1O1O..
190823102919Z.
191121102919Z0c1.0     ..U....US1.0...U...
California1.0...U...
Mountain View1.0...U.
.
Google LLC1.0...U...   gmail.com0.."0
.          *.H..
..........0..
.......e.....R~r..8.7....
>v:b....pD..b.k...e...(.....$./=... .Wi&..L....m..4..
..20..5.\n..rq.^.........P<....0.q.L..j....Z?Oi...mr.%..oc...eE..Bm....3.1]...06..R.g..X...MTZ?..c...$.....;.j.TcS+:..B...
2..
.......Ceo]._;0.9. 2.!O.......~n.$Y.N6u?....e'.@QN..|.........\0..X0...U...........0...U.%..0
..+.......0...U.......0.0...U............F.......j.....0...U.#..0......n.....`......}      .+0d..+........X0V0'..+.....
0...http://ocsp.pki.goog/gts1o10+..+.....0...http://pki.goog/gsr2/GTS1O1.crt0!..U....0..   gmail.com..*.gmail.com0!..U. ..
0.0...g.....0..
+.....y...0/..U...(0&0$.". ..http://crl.pki.goog/GTS1O1.crl0....
+.....y............v.c....;.,..r.'Wk3..aw..u.8..hTK.....l.<.......G0E. MhP.V.vWx....&i".....`d...$......!...c4 %.6{...2
.C.Dd9.q..p.t.!V...v.t~..1.3..!..%OBp...^B ..75y..{.V...l.<.1.....G0E.!.........;..M.L`.....8$[....\v7...
pe...0Z{....;y.,.(
....~..0..o*0
.          *.H..
.........P..T#.7....h......P......U.\.?K.?..O..z.K bmN....K..-X.+w...0.........:.U./..Q.e.b.D.[...6yS.q.p...
+.AP<zZ=..w_....D.k.<..i.nU......+.y.r.&...u  .4...Z'0...[..$\(KIL..Uq..........4.@>....*.'.T_d..u..X..C..<1.._}.u
..RJ....".T.`NO......\.......`....nbE. .....N0..J0..2......
.........%iP.0
.          *.H..
.....0L1 0...U....GlobalSign Root CA - R21.0...U.
```

Packet 1432. 4 client pkts, 6 server pkts, 5 turns. Click to select.

Entire conversation (3743 bytes)

Show and save data as: ASCII

Stream: 35

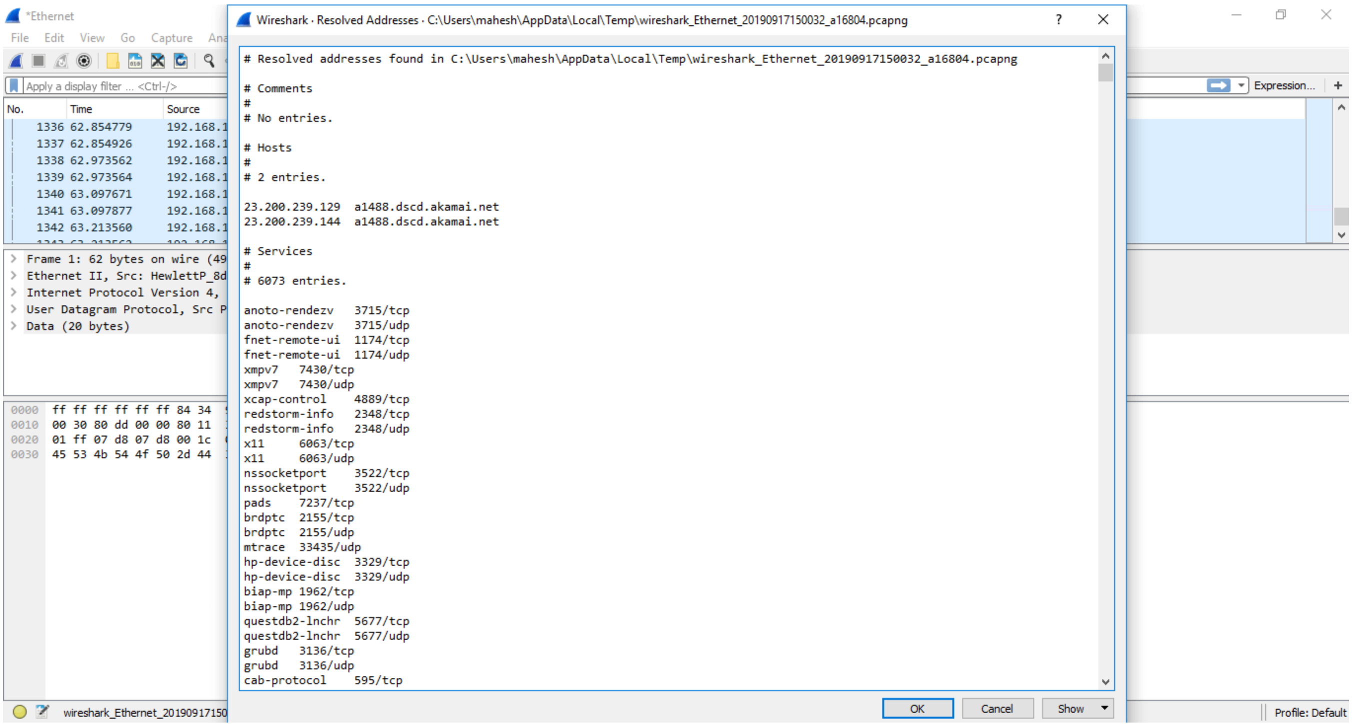
Find:

In the arrow shown above, the 'show and save data as' has many choices. These options are- **ASCII, C Arrays, EBCDIC (Extended Binary Coded Decimal Interchange Code)**, etc. EBCDIC is used in mainframe and mid-range IBM computer operating systems.

Wireshark Statistics

The Wireshark provides a wide domain of statistics. They are listed below:

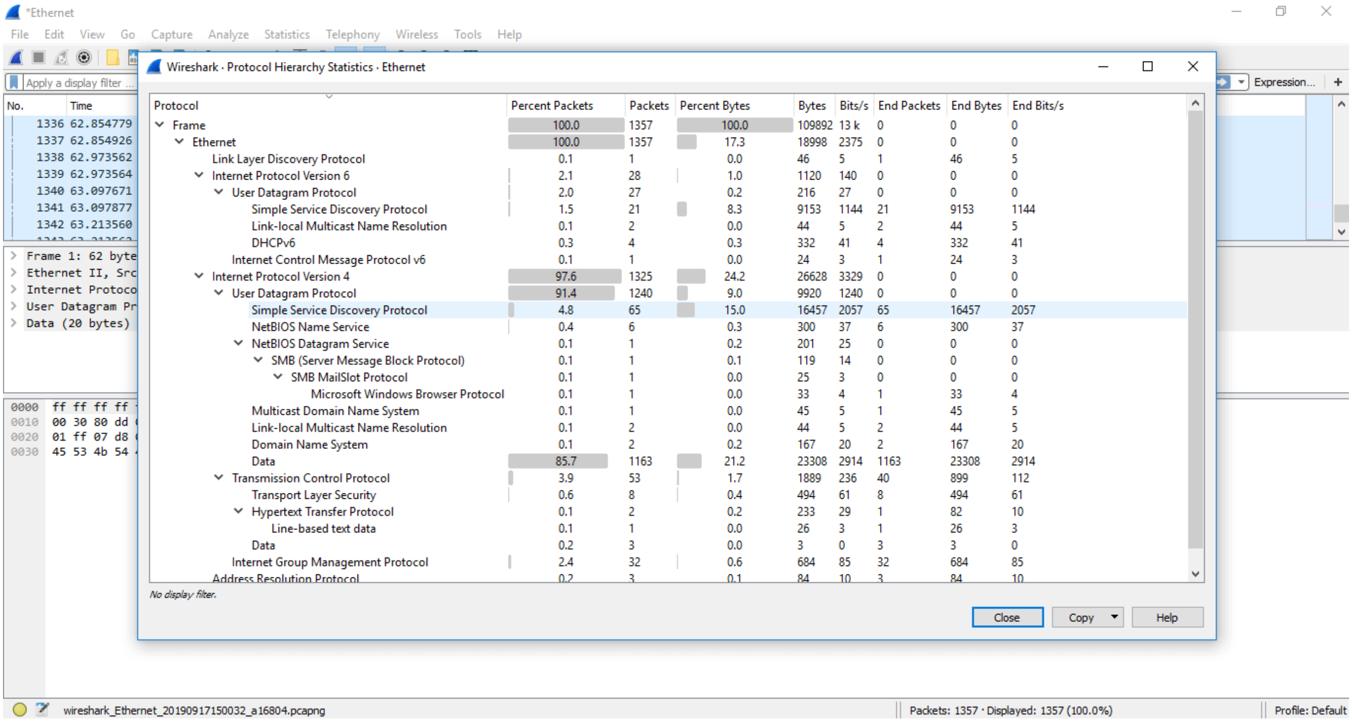
⇧ SCROLL TO TOP



Wireshark · Resolved Addresses · C:\Users\mahesh\AppData\Local\Temp\wireshark_Ethernet_20190917150032_a16804.pcapng

```
# Resolved addresses found in C:\Users\mahesh\AppData\Local\Temp\wireshark_Ethernet_20190917150032_a16804.pcapng

# Comments
#
# No entries.

# Hosts
#
# 2 entries.

23.200.239.129  a1488.dscd.akamai.net
23.200.239.144  a1488.dscd.akamai.net

# Services
#
# 6073 entries.

anoto-rendezv   3715/tcp
anoto-rendezv   3715/udp
fnet-remote-ui  1174/tcp
fnet-remote-ui  1174/udp
xmpv7   7430/tcp
xmpv7   7430/udp
xcap-control    4889/tcp
redstorm-info   2348/tcp
redstorm-info   2348/udp
x11     6063/tcp
x11     6063/udp
nssocketport    3522/tcp
nssocketport    3522/udp
pads    7237/tcp
brdptc  2155/tcp
brdptc  2155/udp
mtrace  33435/udp
hp-device-disc  3329/tcp
hp-device-disc  3329/udp
biap-mp 1962/tcp
biap-mp 1962/udp
questdb2-lnchr  5677/tcp
questdb2-lnchr  5677/udp
grubd   3136/tcp
grubd   3136/udp
cab-protocol    595/tcp
```

Fig (b)



Wireshark · Protocol Hierarchy Statistics · Ethernet

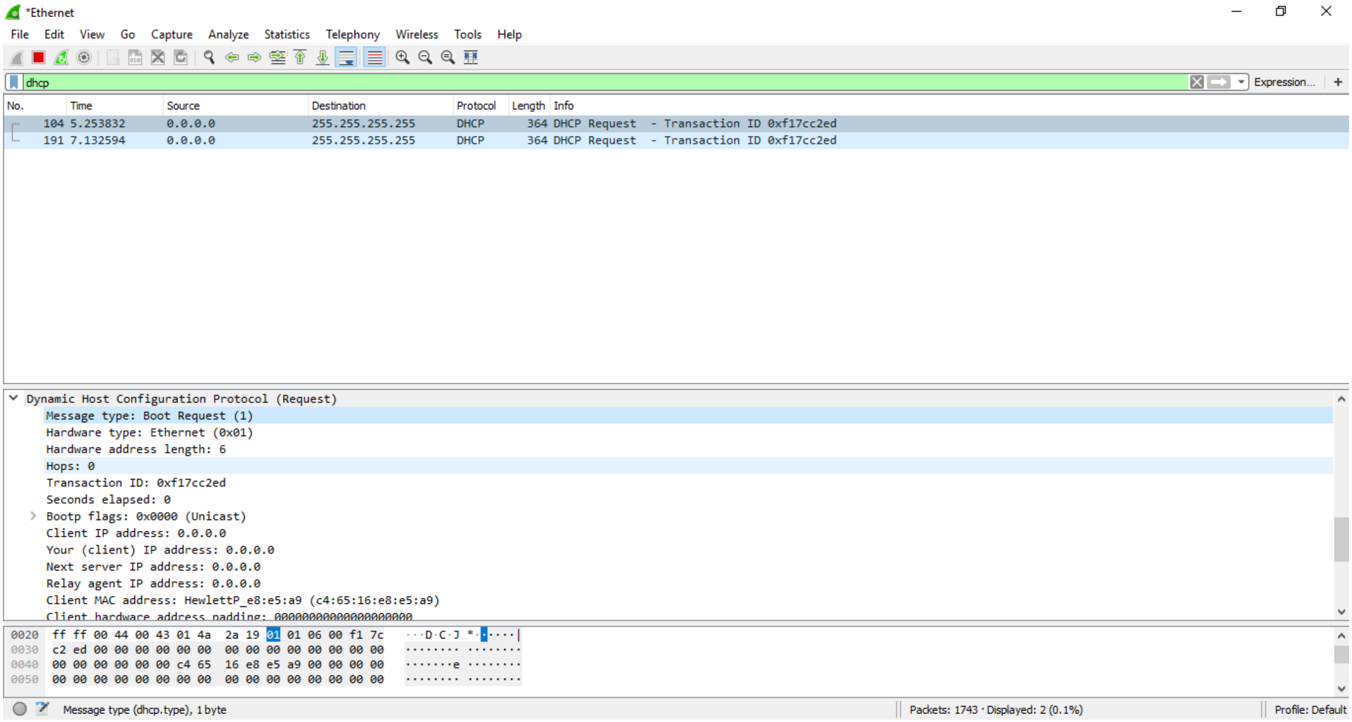
Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	1357	100.0	109892	13 k	0	0	0
Ethernet	100.0	1357	17.3	18998	2375	0	0	0
Link Layer Discovery Protocol	0.1	1	0.0	46	5	1	46	5
Internet Protocol Version 6	2.1	28	1.0	1120	140	0	0	0
User Datagram Protocol	2.0	27	0.2	216	27	0	0	0
Simple Service Discovery Protocol	1.5	21	8.3	9153	1144	21	9153	1144
Link-local Multicast Name Resolution	0.1	2	0.0	44	5	2	44	5
DHCPv6	0.3	4	0.3	332	41	4	332	41
Internet Control Message Protocol v6	0.1	1	0.0	24	3	1	24	3
Internet Protocol Version 4	97.6	1325	24.2	26628	3329	0	0	0
User Datagram Protocol	91.4	1240	9.0	9920	1240	0	0	0
Simple Service Discovery Protocol	4.8	65	15.0	16457	2057	65	16457	2057
NetBIOS Name Service	0.4	6	0.3	300	37	6	300	37
NetBIOS Datagram Service	0.1	1	0.2	201	25	0	0	0
SMB (Server Message Block Protocol)	0.1	1	0.1	119	14	0	0	0
SMB MailSlot Protocol	0.1	1	0.0	25	3	0	0	0
Microsoft Windows Browser Protocol	0.1	1	0.0	33	4	1	33	4
Multicast Domain Name System	0.1	1	0.0	45	5	1	45	5
Link-local Multicast Name Resolution	0.1	2	0.0	44	5	2	44	5
Domain Name System	0.1	2	0.2	167	20	2	167	20
Data	85.7	1163	21.2	23308	2914	1163	23308	2914
Transmission Control Protocol	3.9	53	1.7	1889	236	40	899	112
Transport Layer Security	0.6	8	0.4	494	61	8	494	61
Hypertext Transfer Protocol	0.1	2	0.2	233	29	1	82	10
Line-based text data	0.1	1	0.0	26	3	1	26	3
Data	0.2	3	0.0	3	0	3	3	0
Internet Group Management Protocol	2.4	32	0.6	684	85	32	684	85
Address Resolution Protocol	0.2	3	0.1	84	10	3	84	10

No display filter.

Close
Copy
Help
wireshark_Ethernet_20190917150032_a16804.pcapng
Packets: 1357 · Displayed: 1357 (100.0%)
Profile: Default

Fig (c)

⇧ SCROLL TO TOP


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
+

dhcp

```
No.      Time        Source       Destination        Protocol  Length Info
    104 5.253832    0.0.0.0      255.255.255.255    DHCP         364 DHCP Request  - Transaction ID 0xf17cc2ed
    191 7.132594    0.0.0.0      255.255.255.255    DHCP         364 DHCP Request  - Transaction ID 0xf17cc2ed
```

```
Dynamic Host Configuration Protocol (Request)
    Message type: Boot Request (1)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0xf17cc2ed
    Seconds elapsed: 0
  > Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 0.0.0.0
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: HewlettP_e8:e5:a9 (c4:65:16:e8:e5:a9)
    Client hardware address padding: 00000000000000000000
```

```
0020  ff ff 00 44 00 43 01 4a  2a 19 01 01 06 00 f1 7c   ···D·C·J *·······
0030  c2 ed 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 c4 65  16 e8 e5 a9 00 00 00 00   ·······e ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
```

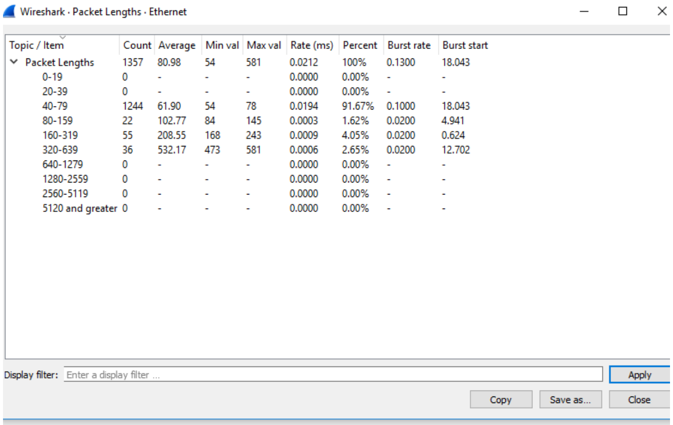
Message type (dhcp.type), 1 byte | Packets: 1743 · Displayed: 2 (0.1%) | Profile: Default

Fig (d)

Below is the list of statistics of Wireshark along with the description:

Capture file properties	It includes file, time, capture, interfaces (current interface in use), and Statistics (measurements).
Resolved addresses	This option includes all the types of the Top IP addresses and DNS that were resolved in your packet capture. It gives the idea of the different accessed resources during the packet capture process. It is shown in fig (b).
Protocol hierarchy	It is named as the tree of all the protocols listed in the capture process. The image is shown above in fig (c).
Conversations	Each row of the list gives the statistical value of a particular conversation.
Endpoints	It is defined as a logical endpoint of the separate protocol traffic of the specified protocol layer. For example0 IP address will send and receive all types of the packet to the particular IP addresses.

⇧ SCROLL TO TOP

<table>
<tr>
<td><b>Packet lengths</b></td>
<td>It simply displays the characteristics of different packets lengths determined in the network.


<table>
<tr><th>Topic / Item</th><th>Count</th><th>Average</th><th>Min val</th><th>Max val</th><th>Rate (ms)</th><th>Percent</th><th>Burst rate</th><th>Burst start</th></tr>
<tr><td>Packet Lengths</td><td>1357</td><td>80.98</td><td>54</td><td>581</td><td>0.0212</td><td>100%</td><td>0.1300</td><td>18.043</td></tr>
<tr><td>0-19</td><td>0</td><td>-</td><td>-</td><td>-</td><td>0.0000</td><td>0.00%</td><td>-</td><td>-</td></tr>
<tr><td>20-39</td><td>0</td><td>-</td><td>-</td><td>-</td><td>0.0000</td><td>0.00%</td><td>-</td><td>-</td></tr>
<tr><td>40-79</td><td>1244</td><td>61.90</td><td>54</td><td>78</td><td>0.0194</td><td>91.67%</td><td>0.1000</td><td>18.043</td></tr>
<tr><td>80-159</td><td>22</td><td>102.77</td><td>84</td><td>145</td><td>0.0003</td><td>1.62%</td><td>0.0200</td><td>4.941</td></tr>
<tr><td>160-319</td><td>55</td><td>208.55</td><td>168</td><td>243</td><td>0.0009</td><td>4.05%</td><td>0.0200</td><td>0.624</td></tr>
<tr><td>320-639</td><td>36</td><td>532.17</td><td>473</td><td>581</td><td>0.0006</td><td>2.65%</td><td>0.0200</td><td>12.702</td></tr>
<tr><td>640-1279</td><td>0</td><td>-</td><td>-</td><td>-</td><td>0.0000</td><td>0.00%</td><td>-</td><td>-</td></tr>
<tr><td>1280-2559</td><td>0</td><td>-</td><td>-</td><td>-</td><td>0.0000</td><td>0.00%</td><td>-</td><td>-</td></tr>
<tr><td>2560-5119</td><td>0</td><td>-</td><td>-</td><td>-</td><td>0.0000</td><td>0.00%</td><td>-</td><td>-</td></tr>
<tr><td>5120 and greater</td><td>0</td><td>-</td><td>-</td><td>-</td><td>0.0000</td><td>0.00%</td><td>-</td><td>-</td></tr>
</table>
Display filter: Enter a display filter ...

</td>
</tr>
<tr>
<td><b>I/O Graphs</b></td>
<td>It is the term used to display the graph of the captured packets. You can also apply filters during this process.
The process is explained below in detail.</td>
</tr>
<tr>
<td><b>Service Response Time</b></td>
<td>It is the type of information which is available for many protocols. It is defined as the time it takes between the request and the response time. The protocol for which this service is available are:
AFP (Apple Filing Protocol)
CAMEL
DCE-RPC
DIAMETER
FC (Fiber Channel)
GTP (GPRS Tunneling Protocol)
H.225 RAS
LDAP (Lightweight Directory Access Protocol)
MEGACO
MGCP (Media Gateway Control Protocol)
NCP (NetWare Core Protocol)
ONC-RPC
RADIUS
SCSI
SMB (Server Message Block Protocol)
SMB2 (Server Message Block Protocol version 2)</td>
</tr>
<tr>
<td><b>DHCP (BOOTP) Statistics</b></td>
<td>It is implemented as the option of BOOTP. DHCP is client/server protocol, dynamically used to assign IP addresses to a DHCP client. If DHCP does not work, then some computer system uses APIPA (Automatic Private IP Address) to assign the IP addresses.</td>
</tr>
</table>

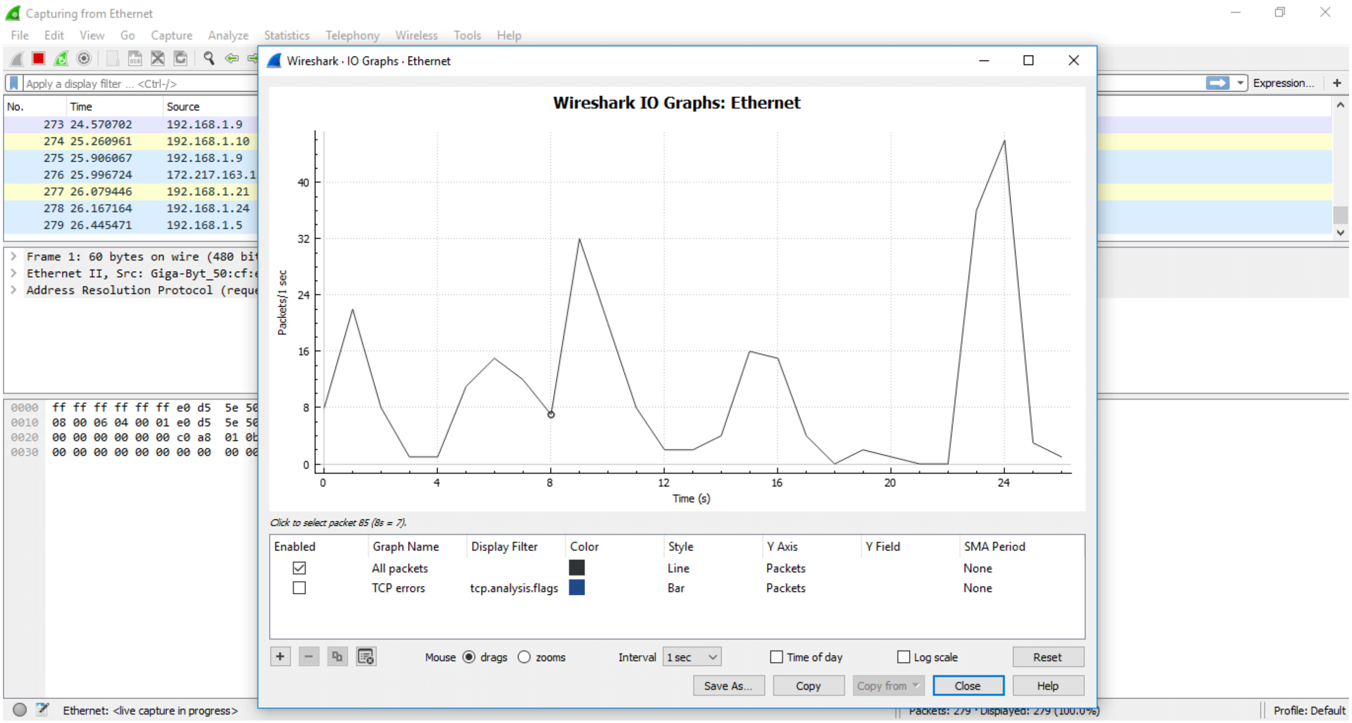
⇧ SCROLL TO TOP

ONC-RPC Programs	It stands for Open Network Computing- Remote Procedure Call. It can use TCP and UDP as its transport protocol. ONC-RPC cannot be applied directly to filter in a capture process, but you can use TCP or UDP to filter on that one. It is shown in fig (d).
29West	It is defined as ULLM technology. It stands for Ultra-Low Latency Messaging.
ANCP	It stands for **Access Node Control Protocol**. It is an L2CP (Layer 2 Control Protocol) and a TCP based one. It has its adjacency layer which decides the messages exchange by the ANCP endpoints with the use of 'Capabilities.'
BACnet	It was designed specially to meet the communication needs of control systems and building automation. It is used for applications such as fire detecting systems, light control, etc. It provides the structure to exchange information despite the particular building service it performs.
Collectd	It is used to monitor the traffic on the specific TCP port.
DNS	It stands for Domain Name Server, which gives a detailed analysis of the DNS traffic. It provides the list of the codes returned in DNS. You can also view the errors through the traffic.
Flow-graph	It is a method to check connections between the client and the server. It is an efficient way to verify the connections between two endpoints. It also assists us with troubleshooting capabilities.
HART-IP	It gives the detail for the response, request, publishes, and error packets. It stands for Highway Addressable Remote Transducer over IP stats.
HPFEEDS	It determines the 'payload size per channel and Opcodes.'
HTTP	It has four options: ○ Packet counter (request types and response codes) ○ Requests (based on URL and the host) ○ Load distribution (based on server address and host) ○ Request sequences (sequences the HTTP's capture request as a tree)
HTTP2	It is the HTTP version 2.
Sametime	It is used to analyze the slow network traffic when the server and client have the netime.

⇧ SCROLL TO TOP

TCP Stream Graphs	It is explained below in detail:
UDP Multicast Streams	Through this command, stream parameters and burst parameters can be set. It includes OSPF, IGMP, and video streams.
F5	It includes the virtual server distribution and the tmm distribution. It specifies the tcpdump commands.
IPv4 Statistics **IPv6 Statistics**	These options determine all addresses, destination and ports, IP protocol types, and the source and destination address.

I/O GRAPHS



Capturing from Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Wireshark · IO Graphs · Ethernet
Wireshark IO Graphs: Ethernet
Packets/1 sec
0
8
16
24
32
40
0
4
8
12
16
20
24
Time (s)

```
No.  Time         Source
 273 24.570702    192.168.1.9
 274 25.260961    192.168.1.10
 275 25.906067    192.168.1.9
 276 25.996724    172.217.163.1
 277 26.079446    192.168.1.21
 278 26.167164    192.168.1.24
 279 26.445471    192.168.1.5
```

```
> Frame 1: 60 bytes on wire (480 bi
> Ethernet II, Src: Giga-Byt_50:cf:
> Address Resolution Protocol (reque
```

```
0000  ff ff ff ff ff ff e0 d5  5e 50
0010  08 00 06 04 00 01 e0 d5  5e 50
0020  00 00 00 00 00 00 c0 a8  01 0b
0030  00 00 00 00 00 00 00 00  00 00
```

Click to select packet 85 (8s = 7).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		■	Line	Packets		None
□	TCP errors	tcp.analysis.flags	■	Bar	Packets		None

+ − Mouse ● drags ○ zooms Interval 1 sec □ Time of day □ Log scale Reset

Save As... Copy Copy from Close Help

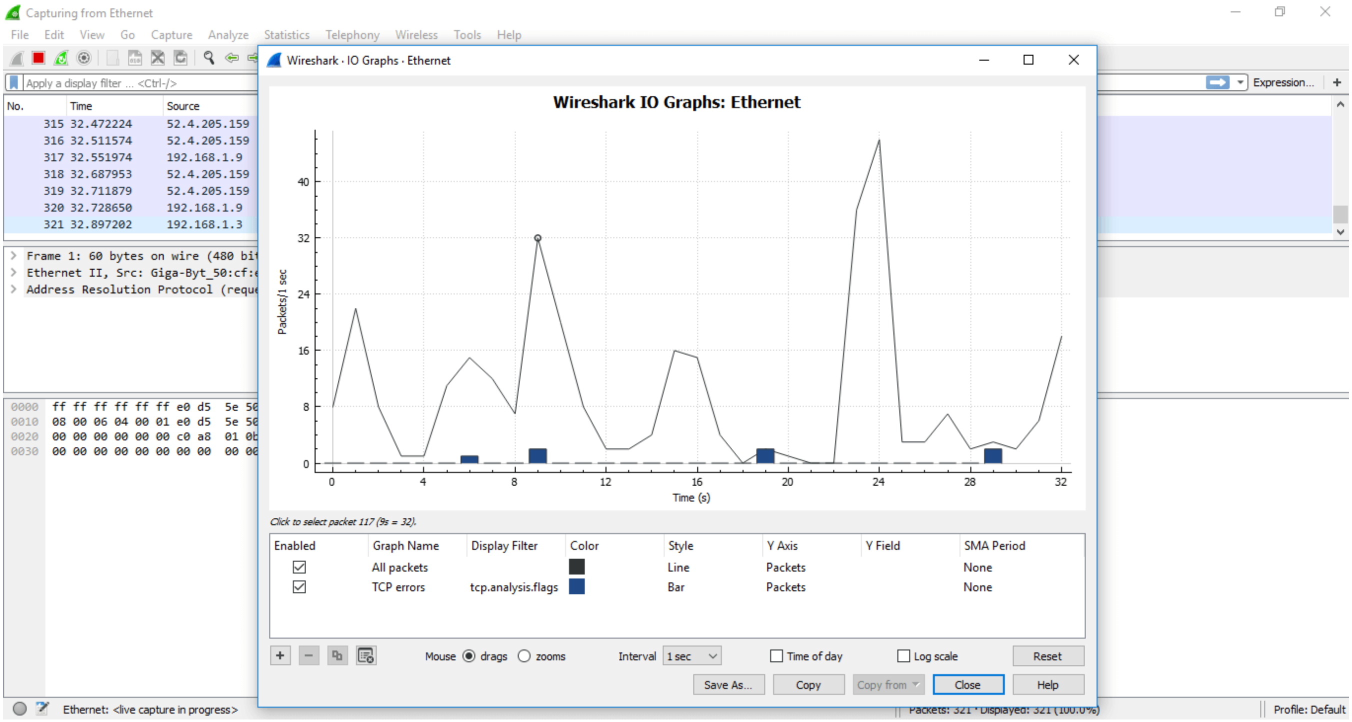
Ethernet: <live capture in progress> Packets: 279 · Displayed: 279 (100.0%) Profile: Default

It shows the graph for the network traffic. The graph will look similar but changes as per the traffic involved. There is a table below the figure, which has some filters. Using the '+' sign, you can add more filters and use '-sign you can remove the existing filters. You can also change the color. For every particular filter, you can add a colored layer, which increases the visibility of the graph.

The tick option under the 'Enabled,' displays the layer according to your requirements.

For example, we have applied the filter 'TCP errors' and the changes can be viewed easily. The image is shown below:

⇧ SCROLL TO TOP


Capturing from Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Wireshark · IO Graphs · Ethernet
Wireshark IO Graphs: Ethernet
Packets/1 sec
0
8
16
24
32
40
0
4
8
12
16
20
24
28
32
Time (s)
Ethernet: <live capture in progress>
Packets: 321 · Displayed: 321 (100.0%)
Profile: Default

```
No.   Time         Source
  315 32.472224    52.4.205.159
  316 32.511574    52.4.205.159
  317 32.551974    192.168.1.9
  318 32.687953    52.4.205.159
  319 32.711879    52.4.205.159
  320 32.728650    192.168.1.9
  321 32.897202    192.168.1.3

> Frame 1: 60 bytes on wire (480 bi
> Ethernet II, Src: Giga-Byt_50:cf:
> Address Resolution Protocol (reque

0000  ff ff ff ff ff ff e0 d5  5e 50
0010  08 00 06 04 00 01 e0 d5  5e 50
0020  00 00 00 00 00 00 c0 a8  01 0b
0030  00 00 00 00 00 00 00 00  00 00
```

Click to select packet 117 (9s = 32).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		■	Line	Packets		None
☑	TCP errors	tcp.analysis.flags	■	Bar	Packets		None

Mouse ● drags ○ zooms   Interval 1 sec   □ Time of day   □ Log scale

If you click on the particular point on the graph, you can watch the corresponding packet will be shown on the screen of the network traffic. You can also apply a filter on the particular port.

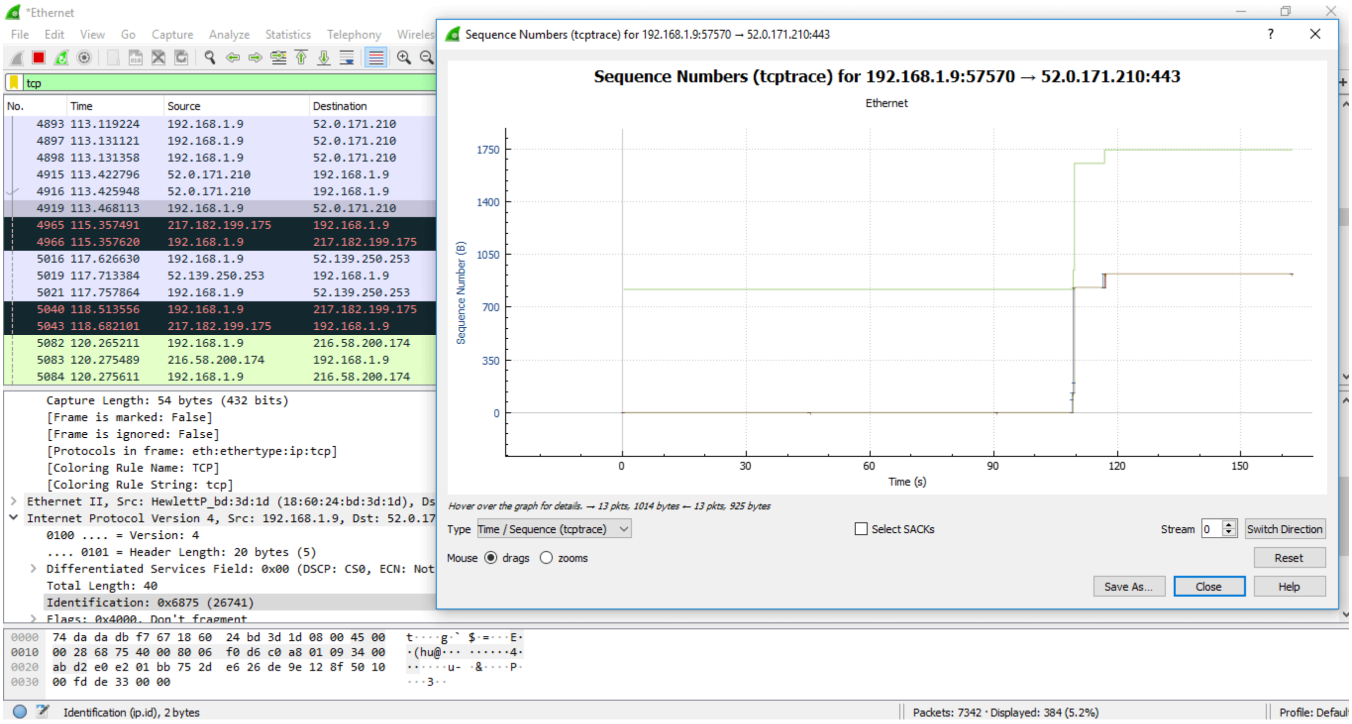
Another category of the graph comes under the option '**TCP Stream graphs**.'

It gives the visualization of the TCP sequence number with time.

Below are the steps to understand the **TCP Stream graphs**:

- Open the Wireshark. Click on the interface to watch the network traffic.
- Apply the filter as 'tcp.'
- Click on the option 'Statistics 'on the menu bar and select '**TCP Stream graphs**' and select 'Time sequence (tcptrace). You can also choose other options in the 'TCP Stream graphs' category depending on your requirements. Now the screen will look as:

⇧ SCROLL TO TOP


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireles
tcp
Sequence Numbers (tcptrace) for 192.168.1.9:57570 → 52.0.171.210:443
Sequence Numbers (tcptrace) for 192.168.1.9:57570 → 52.0.171.210:443
Ethernet
Sequence Number (B)
1750
1400
1050
700
350
0
0
30
60
90
120
150
Time (s)
Save As... Close Help
Reset
Switch Direction

```
No.   Time        Source           Destination
 4893 113.119224  192.168.1.9      52.0.171.210
 4897 113.131121  192.168.1.9      52.0.171.210
 4898 113.131358  192.168.1.9      52.0.171.210
 4915 113.422796  52.0.171.210     192.168.1.9
 4916 113.425948  52.0.171.210     192.168.1.9
 4919 113.468113  192.168.1.9      52.0.171.210
 4965 115.357491  217.182.199.175  192.168.1.9
 4966 115.357620  192.168.1.9      217.182.199.175
 5016 117.626630  192.168.1.9      52.139.250.253
 5019 117.713384  52.139.250.253   192.168.1.9
 5021 117.757864  192.168.1.9      52.139.250.253
 5040 118.513556  192.168.1.9      217.182.199.175
 5043 118.682101  217.182.199.175  192.168.1.9
 5082 120.265211  192.168.1.9      216.58.200.174
 5083 120.275489  216.58.200.174   192.168.1.9
 5084 120.275611  192.168.1.9      216.58.200.174
```

```
    Capture Length: 54 bytes (432 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:tcp]
    [Coloring Rule Name: TCP]
    [Coloring Rule String: tcp]
> Ethernet II, Src: HewlettP_bd:3d:1d (18:60:24:bd:3d:1d), Ds
v Internet Protocol Version 4, Src: 192.168.1.9, Dst: 52.0.17
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not
    Total Length: 40
    Identification: 0x6875 (26741)
  > Flags: 0x4000, Don't fragment
```

```
0000  74 da da db f7 67 18 60  24 bd 3d 1d 08 00 45 00   t····g·` $·=···E·
0010  00 28 68 75 40 00 80 06  f0 d6 c0 a8 01 09 34 00   ·(hu@··· ······4·
0020  ab d2 e0 e2 01 bb 75 2d  e6 26 de 9e 12 8f 50 10   ······u- ·&····P·
0030  00 fd de 33 00 00                                  ···3··
```

Hover over the graph for details. → 13 pkts, 1014 bytes ← 13 pkts, 925 bytes

Type Time / Sequence (tcptrace) □ Select SACKs Stream 0

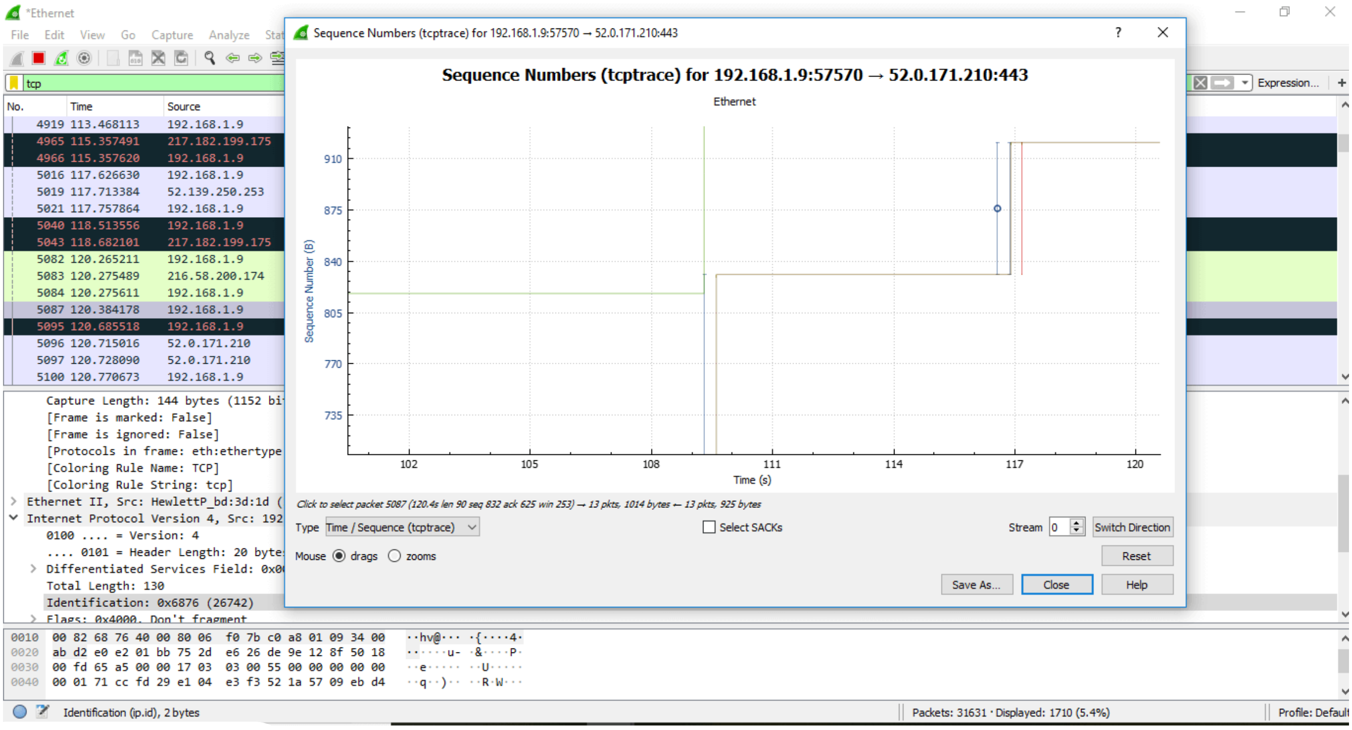
Mouse ● drags ○ zooms

Identification (ip.id), 2 bytes | Packets: 7342 · Displayed: 384 (5.2%) | Profile: Defaul

Now, as you zoom on the graph, you will notice the points in detail. The lines shown are the packets. The length along the Y-axis shows how big the packet is. You can also see the green line going up and then comes at the same level. This means that the data has been ACK (Acknowledged). Here going up means that more data is being sent.

The data is being sent and then ACK, this is the proper use of the TCP. The flat line here signifies that nothing is happening.

The green line above is called '**received window**.' The gap between the received window and the packet, defines how much space is in the received buffer.


*Ethernet
File Edit View Go Capture Analyze Stat
tcp
Expression...
Sequence Numbers (tcptrace) for 192.168.1.9:57570 → 52.0.171.210:443
Sequence Numbers (tcptrace) for 192.168.1.9:57570 → 52.0.171.210:443
Ethernet
Sequence Number (B)
910
875
840
805
770
735
102
105
108
111
114
117
120
Time (s)
Save As... Close Help
Reset
Switch Direction

```
No.   Time        Source
 4919 113.468113  192.168.1.9
 4965 115.357491  217.182.199.175
 4966 115.357620  192.168.1.9
 5016 117.626630  192.168.1.9
 5019 117.713384  52.139.250.253
 5021 117.757864  192.168.1.9
 5040 118.513556  192.168.1.9
 5043 118.682101  217.182.199.175
 5082 120.265211  192.168.1.9
 5083 120.275489  216.58.200.174
 5084 120.275611  192.168.1.9
 5087 120.384178  192.168.1.9
 5095 120.685518  192.168.1.9
 5096 120.715016  52.0.171.210
 5097 120.728090  52.0.171.210
 5100 120.770673  192.168.1.9
```

```
    Capture Length: 144 bytes (1152 bi
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype
    [Coloring Rule Name: TCP]
    [Coloring Rule String: tcp]
> Ethernet II, Src: HewlettP_bd:3d:1d (
v Internet Protocol Version 4, Src: 192
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 byte
  > Differentiated Services Field: 0x0
    Total Length: 130
    Identification: 0x6876 (26742)
  > Flags: 0x4000, Don't fragment
```

```
0010  00 82 68 76 40 00 80 06  f0 7b c0 a8 01 09 34 00   ··hv@··· ·{····4·
0020  ab d2 e0 e2 01 bb 75 2d  e6 26 de 9e 12 8f 50 18   ······u- ·&····P·
0030  00 fd 65 a5 00 00 17 03  03 00 55 00 00 00 00 00   ··e····· ··U·····
0040  00 01 71 cc fd 29 e1 04  e3 f3 52 1a 57 09 eb d4   ··q··)·· ··R·W···
```

Click to select packet 5087 (120.4s len 90 seq 832 ack 625 win 253) → 13 pkts, 1014 bytes ← 13 pkts, 925 bytes

Type Time / Sequence (tcptrace) □ Select SACKs Stream 0

Mouse ● drags ○ zooms

Identification (ip.id), 2 bytes | Packets: 31631 · Displayed: 1710 (5.4%) | Profile: Defaul

⇧ SCROLL TO TOP

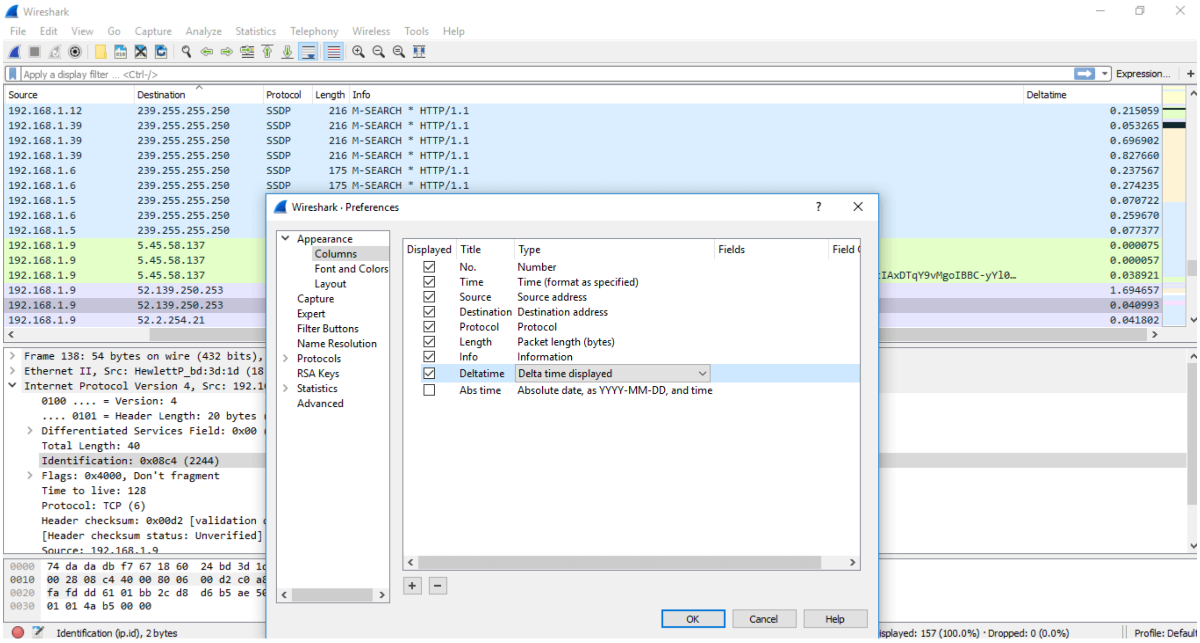
FACTS ABOUT WIRESHARK/ IMPORTANT STEPS/ MOST USED

Below are the facts or points implemented in real life:

Adding a delta column: To add any column, below are the steps:

- On any of the column menu, right-click and choose 'Column Preferences' and then select 'Column.'
- Click on the '+' sign, and add the column by name like delta-time and under the 'Type' category, select the delta time or delta time displayed.

The screen will then look as:



```
Source         Destination      Protocol Length Info                                  Deltatime
192.168.1.12   239.255.255.250  SSDP     216 M-SEARCH * HTTP/1.1                      0.215059
192.168.1.39   239.255.255.250  SSDP     216 M-SEARCH * HTTP/1.1                      0.053265
192.168.1.39   239.255.255.250  SSDP     216 M-SEARCH * HTTP/1.1                      0.696902
192.168.1.39   239.255.255.250  SSDP     216 M-SEARCH * HTTP/1.1                      0.827660
192.168.1.6    239.255.255.250  SSDP     175 M-SEARCH * HTTP/1.1                      0.237567
192.168.1.6    239.255.255.250  SSDP     175 M-SEARCH * HTTP/1.1                      0.274235
192.168.1.5    239.255.255.250                                                        0.070722
192.168.1.6    239.255.255.250                                                        0.259670
192.168.1.5    239.255.255.250                                                        0.077377
192.168.1.9    5.45.58.137                                                            0.000075
192.168.1.9    5.45.58.137                                                            0.000057
192.168.1.9    5.45.58.137                  :IAxDTqY9vMgoIBBC-yYl0…                   0.038921
192.168.1.9    52.139.250.253                                                         1.694657
192.168.1.9    52.139.250.253                                                         0.040993
192.168.1.9    52.2.254.21                                                            0.041802

> Frame 138: 54 bytes on wire (432 bits),
> Ethernet II, Src: HewlettP_bd:3d:1d (18
v Internet Protocol Version 4, Src: 192.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes
  > Differentiated Services Field: 0x00
    Total Length: 40
    Identification: 0x08c4 (2244)
  > Flags: 0x4000, Don't fragment
    Time to live: 128
    Protocol: TCP (6)
    Header checksum: 0x00d2 [validation
    [Header checksum status: Unverified]
    Source: 192.168.1.9

0000  74 da da db f7 67 18 60  24 bd 3d 1c
0010  00 28 08 c4 40 00 80 06  00 d2 c0 a8
0020  fa fd dd 61 01 bb 2c d8  d6 b5 ae 50
0030  01 01 4a b5 00 00

Identification (ip.id), 2 bytes
isplayed: 157 (100.0%) · Dropped: 0 (0.0%)
```

Displayed	Title	Type	Fields	Field (
☑	No.	Number		
☑	Time	Time (format as specified)		
☑	Source	Source address		
☑	Destination	Destination address		
☑	Protocol	Protocol		
☑	Length	Packet length (bytes)		
☑	Info	Information		
☑	Deltatime	Delta time displayed		
☐	Abs time	Absolute date, as YYYY-MM-DD, and time		

Below the captured packets, the data you see in the **square brackets** is the information that is not available in the packet itself. It is something that Wireshark displays for your benefit. If you want to add anything from this screen to the column area, you can right-click and select 'Apply as column.' That option will be added to the capture screen.

The most important is:

3 Way-Handshake

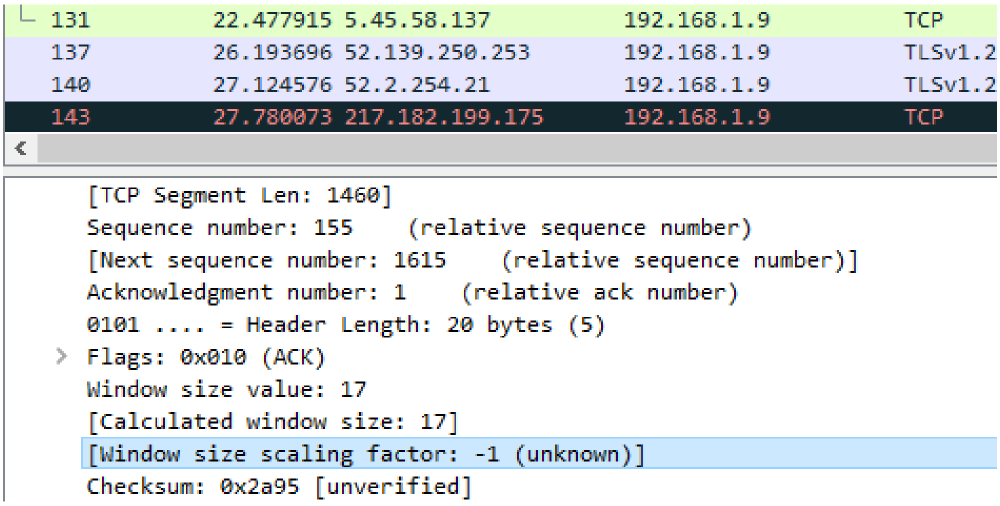
- When you are capturing your data, analyze the problem, you will get the three-way handshake.
- It contains good options like the TCP options.
- From this, you can determine the shift time and figure out if you have captured packets on the client-side or the server-side. There is a little delay between SYN SYN- ACK packet at server-side while there is a more delay between the SYN

⇧ SCROLL TO TOP

and SYN-ACK at the client-side. There is a delay at the server-side only between the SYN-ACK and ACK. The SYN has to reach to the client. After the three-way handshake, the data has to reach the server.

Y

- ou can also notice the difference in the TCP options between the SYN and SYN-ACK packets. The window scaling factor is also essential, as shown below:

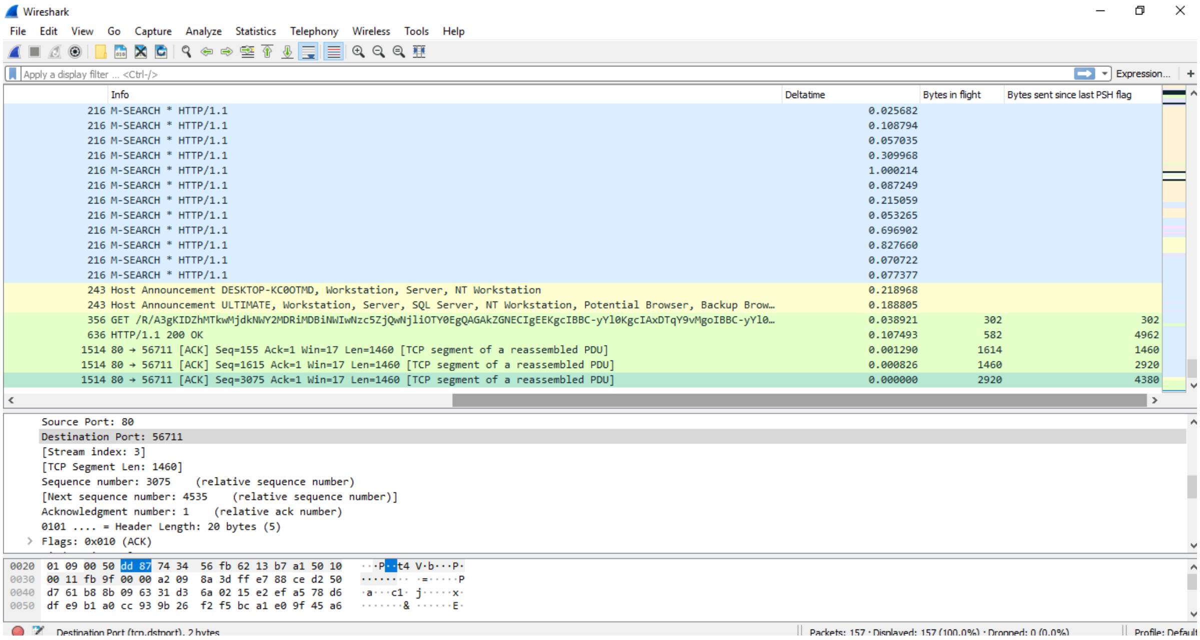


```
└ 131    22.477915 5.45.58.137        192.168.1.9    TCP
  137    26.193696 52.139.250.253     192.168.1.9    TLSv1.2
  140    27.124576 52.2.254.21        192.168.1.9    TLSv1.2
  143    27.780073 217.182.199.175    192.168.1.9    TCP
```

```
    [TCP Segment Len: 1460]
    Sequence number: 155    (relative sequence number)
    [Next sequence number: 1615    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 17
    [Calculated window size: 17]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0x2a95 [unverified]
```

Without three-way handshake, you cannot view the window scaling factor.

- One sequence number means 1 byte of data. It also has an importance of the TCP Stream Graphs which is already explained above.
- Under the TCP options, capture window, you can see the information about the 'PSH byte' and 'Bytes in flight.' Right-click on that and choose 'Apply as Column.' You can see both the columns and data according to it. The image for this is shown below:



Wireshark
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

```
     Info                                                                                       Deltatime   Bytes in flight   Bytes sent since last PSH flag
 216 M-SEARCH * HTTP/1.1                                                                        0.025682
 216 M-SEARCH * HTTP/1.1                                                                        0.108794
 216 M-SEARCH * HTTP/1.1                                                                        0.057035
 216 M-SEARCH * HTTP/1.1                                                                        0.309968
 216 M-SEARCH * HTTP/1.1                                                                        1.000214
 216 M-SEARCH * HTTP/1.1                                                                        0.087249
 216 M-SEARCH * HTTP/1.1                                                                        0.215059
 216 M-SEARCH * HTTP/1.1                                                                        0.053265
 216 M-SEARCH * HTTP/1.1                                                                        0.696902
 216 M-SEARCH * HTTP/1.1                                                                        0.827660
 216 M-SEARCH * HTTP/1.1                                                                        0.070722
 216 M-SEARCH * HTTP/1.1                                                                        0.077377
 243 Host Announcement DESKTOP-KC0OTMD, Workstation, Server, NT Workstation                     0.218968
 243 Host Announcement ULTIMATE, Workstation, Server, SQL Server, NT Workstation, Potential Browser, Backup Brow…   0.188805
 356 GET /R/A3gKIDZhMTkwMjdkNWY2MDRiMDBiNWIwNzc5ZjQwNjliOTY0EgQAGAkZGNECIgEEKgcIBBC-yYl0KgcIAxDTqY9vMgoIBBC-yYl0…   0.038921   302   302
 636 HTTP/1.1 200 OK                                                                            0.107493    582   4962
1514 80 → 56711 [ACK] Seq=155 Ack=1 Win=17 Len=1460 [TCP segment of a reassembled PDU]          0.001290   1614   1460
1514 80 → 56711 [ACK] Seq=1615 Ack=1 Win=17 Len=1460 [TCP segment of a reassembled PDU]         0.000826   1460   2920
1514 80 → 56711 [ACK] Seq=3075 Ack=1 Win=17 Len=1460 [TCP segment of a reassembled PDU]         0.000000   2920   4380
```

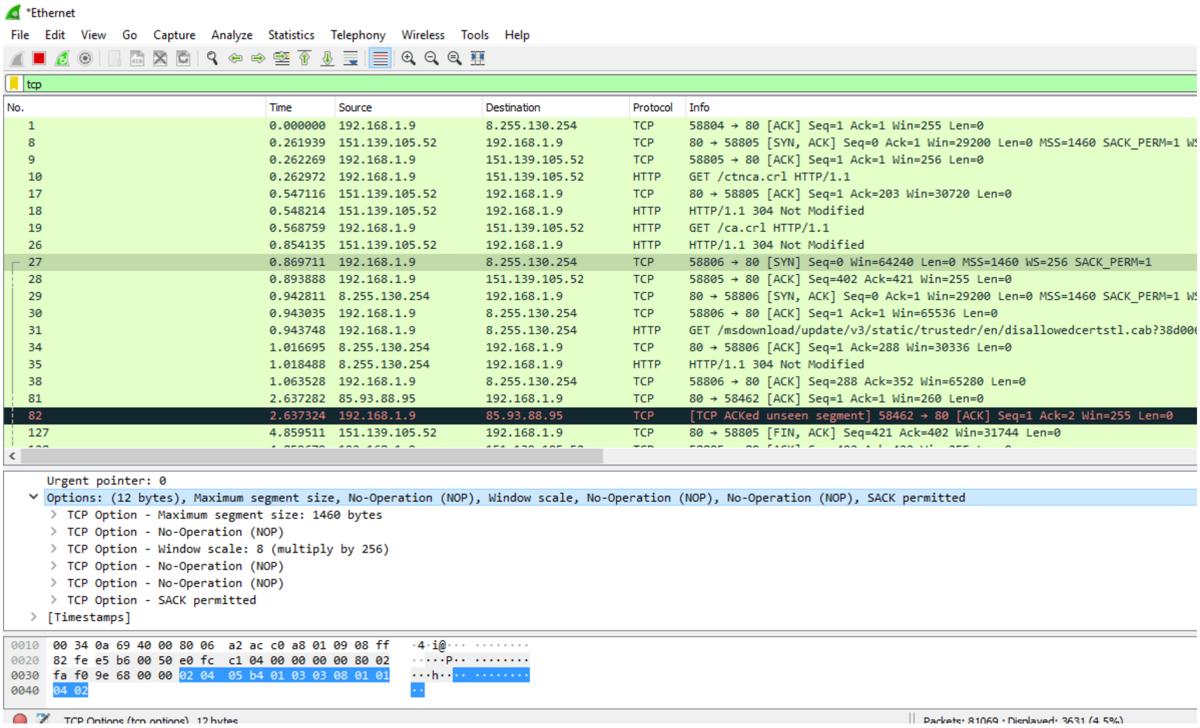
```
    Source Port: 80
    Destination Port: 56711
    [Stream index: 3]
    [TCP Segment Len: 1460]
    Sequence number: 3075    (relative sequence number)
    [Next sequence number: 4535    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
```

```
0020  01 09 00 50 dd 87 74 34  56 fb 62 13 b7 a1 50 10   ···P··t4 V·b···P·
0030  00 11 fb 9f 00 00 a2 09  8a 3d ff e7 88 ce d2 50   ········ ·=·····P
0040  d7 61 b8 8b 09 63 31 d3  6a 02 15 e2 ef a5 78 d6   ·a···c1· j·····x·
0050  df e9 b1 a0 cc 93 9b 26  f2 f5 bc a1 e0 9f 45 a6   ·······& ······E·
```

Destination Port (tcp.dstport), 2 bytes   Packets: 157 · Displayed: 157 (100.0%) · Dropped: 0 (0.0%)   Profile: Default

- In TCP Header, three-way handshake MSS (Maximum Header Size) means that the maximum amount of data it can receive of TCP payload. The image is shown ›w:

⇧ SCROLL TO TOP


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.9	8.255.130.254	TCP	58804 → 80 [ACK] Seq=1 Ack=1 Win=255 Len=0
8	0.261939	151.139.105.52	192.168.1.9	TCP	80 → 58805 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS
9	0.262269	192.168.1.9	151.139.105.52	TCP	58805 → 80 [ACK] Seq=1 Ack=1 Win=256 Len=0
10	0.262972	192.168.1.9	151.139.105.52	HTTP	GET /ctnca.crl HTTP/1.1
17	0.547116	151.139.105.52	192.168.1.9	TCP	80 → 58805 [ACK] Seq=1 Ack=203 Win=30720 Len=0
18	0.548214	151.139.105.52	192.168.1.9	HTTP	HTTP/1.1 304 Not Modified
19	0.568759	192.168.1.9	151.139.105.52	HTTP	GET /ca.crl HTTP/1.1
26	0.854135	151.139.105.52	192.168.1.9	HTTP	HTTP/1.1 304 Not Modified
27	0.869711	192.168.1.9	8.255.130.254	TCP	58806 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
28	0.893888	192.168.1.9	151.139.105.52	TCP	58805 → 80 [ACK] Seq=402 Ack=421 Win=255 Len=0
29	0.942811	8.255.130.254	192.168.1.9	TCP	80 → 58806 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS
30	0.943035	192.168.1.9	8.255.130.254	TCP	58806 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
31	0.943748	192.168.1.9	8.255.130.254	HTTP	GET /msdownload/update/v3/static/trustedr/en/disallowedcertstl.cab?38d00
34	1.016695	8.255.130.254	192.168.1.9	TCP	80 → 58806 [ACK] Seq=1 Ack=288 Win=30336 Len=0
35	1.018488	8.255.130.254	192.168.1.9	HTTP	HTTP/1.1 304 Not Modified
38	1.063528	192.168.1.9	8.255.130.254	TCP	58806 → 80 [ACK] Seq=288 Ack=352 Win=65280 Len=0
81	2.637282	85.93.88.95	192.168.1.9	TCP	80 → 58462 [ACK] Seq=1 Ack=1 Win=260 Len=0
82	2.637324	192.168.1.9	85.93.88.95	TCP	[TCP ACKed unseen segment] 58462 → 80 [ACK] Seq=1 Ack=2 Win=255 Len=0
127	4.859511	151.139.105.52	192.168.1.9	TCP	80 → 58805 [FIN, ACK] Seq=421 Ack=402 Win=31744 Len=0

```
    Urgent pointer: 0
  ⌄ Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SACK permitted
    > TCP Option - Maximum segment size: 1460 bytes
    > TCP Option - No-Operation (NOP)
    > TCP Option - Window scale: 8 (multiply by 256)
    > TCP Option - No-Operation (NOP)
    > TCP Option - No-Operation (NOP)
    > TCP Option - SACK permitted
  > [Timestamps]
```

```
0010  00 34 0a 69 40 00 80 06  a2 ac c0 a8 01 09 08 ff   ·4·i@··· ········
0020  82 fe e5 b6 00 50 e0 fc  c1 04 00 00 00 00 80 02   ·····P·· ········
0030  fa f0 9e 68 00 00 02 04  05 b4 01 03 03 08 01 01   ···h···· ········
0040  04 02                                              ··
```

TCP Options (tcp.options), 12 bytes
Packets: 81069 · Displayed: 3631 (4.5%)

- MSS 1460 implies that this is per packet amount of data. This size varies from packet to packet. Something like a router, firewall, etc. will do MSS clamping because it knows what is going forward. It checks the value greater than 8000 bytes and brings down it to an appropriate level so that it can go across without fragmentation or being dropped.
- The data with the 0 is the ax coming back in the capture window. You can notice that the data and ACK are different at each point. If we are on the acknowledgment side, we know that we have to send the ACK after two packets. A sender can send X amount of packets depending on its congestion window. A sender can send packets at once also. After the packets will go at the receiver and then the acknowledgment comes back. The sender can send all packets before the ACK reaches it. If the buffer has less space left, then the sender has to send the packets according to space. The ACK arrives on time, and if there is a delay in the ACK, syncing will be delayed. So above it's, just a perspective example explained.

Some Facts about Wireshark:

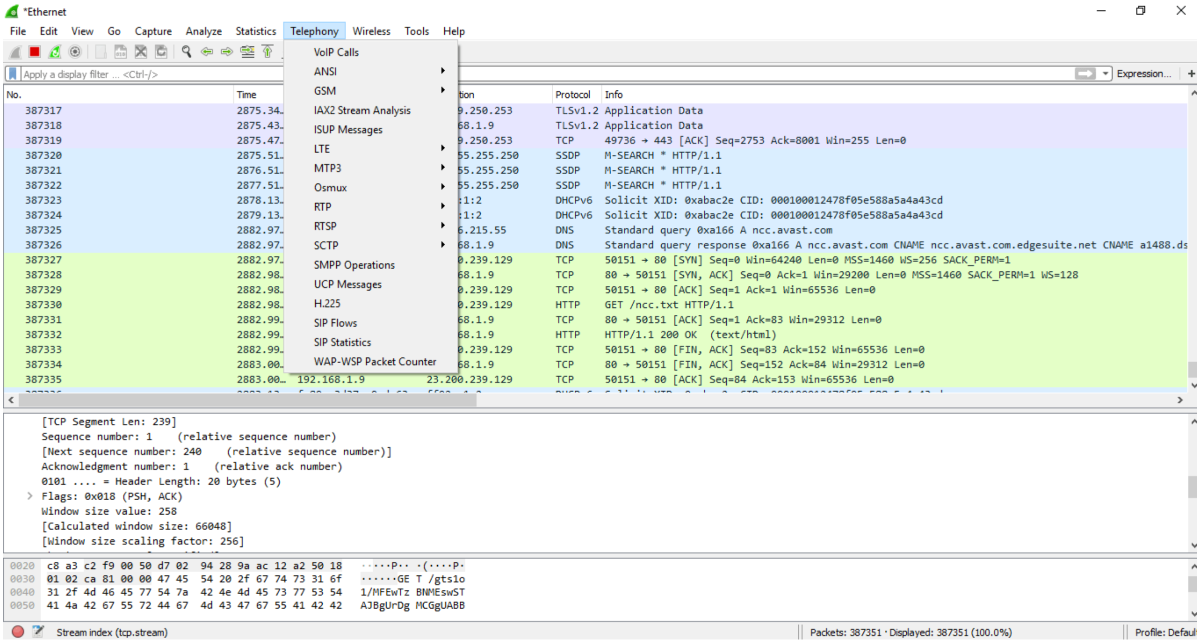
- We do recommend not to disable the default settings of the TCP and Wireshark unless you know what you?re doing.
- If there are the blank page and slow loading, then it is unusable.
- It is good to capture packets from both ends.
- Lean on your provider when you have the data.
- It is a LIVE CAPTURE software used widely.
- an also capture packets from a set of captured one's.
- ere are many protocols dissectors.

⇧ SCROLL TO TOP

- The list of commonly used Endpoints or IP endpoints is: Bluetooth (MAC 48-bit addresses), Ethernet, fiber channel, USB, UDP, FDDI, IPv4, IPv6, JXTA, NCP, TCP, etc.
- Name resolutions are used to convert numerical values into the human-readable format. There are two ways- network services resolution and resolve from Wireshark configuration files. It is only possible when capturing is not in progress. It can be resolved after the packet is added to the list. To rebuild the list with correct resolved names you can use **View-> Reload**.
- In ARP, Wireshark asks the OS to convert the Ethernet address to the IP address.
- Since it is a live capture process, so it is important to set the correct time and zone on your computer.

TELEPHONY

The Telephony is the option on the menu bar. The image is shown below:



```
No.      Time          tion            Protocol  Info
387317   2875.34…      9.250.253       TLSv1.2   Application Data
387318   2875.43…      68.1.9          TLSv1.2   Application Data
387319   2875.47…      9.250.253       TCP       49736 → 443 [ACK] Seq=2753 Ack=8001 Win=255 Len=0
387320   2875.51…      55.255.250      SSDP      M-SEARCH * HTTP/1.1
387321   2876.51…      55.255.250      SSDP      M-SEARCH * HTTP/1.1
387322   2877.51…      55.255.250      SSDP      M-SEARCH * HTTP/1.1
387323   2878.13…      :1:2            DHCPv6    Solicit XID: 0xabac2e CID: 000100012478f05e588a5a4a43cd
387324   2879.13…      :1:2            DHCPv6    Solicit XID: 0xabac2e CID: 000100012478f05e588a5a4a43cd
387325   2882.97…      6.215.55        DNS       Standard query 0xa166 A ncc.avast.com
387326   2882.97…      68.1.9          DNS       Standard query response 0xa166 A ncc.avast.com CNAME ncc.avast.com.edgesuite.net CNAME a1488.ds
387327   2882.97…      0.239.129       TCP       50151 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
387328   2882.98…      68.1.9          TCP       80 → 50151 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
387329   2882.98…      0.239.129       TCP       50151 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
387330   2882.98…      0.239.129       HTTP      GET /ncc.txt HTTP/1.1
387331   2882.99…      68.1.9          TCP       80 → 50151 [ACK] Seq=1 Ack=83 Win=29312 Len=0
387332   2882.99…      68.1.9          HTTP      HTTP/1.1 200 OK  (text/html)
387333   2882.99…      0.239.129       TCP       50151 → 80 [FIN, ACK] Seq=83 Ack=152 Win=65536 Len=0
387334   2883.00…      68.1.9          TCP       80 → 50151 [FIN, ACK] Seq=152 Ack=84 Win=29312 Len=0
387335   2883.00…  192.168.1.9  23.200.239.129  TCP  50151 → 80 [ACK] Seq=84 Ack=153 Win=65536 Len=0
```

```
    [TCP Segment Len: 239]
    Sequence number: 1    (relative sequence number)
    [Next sequence number: 240    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x018 (PSH, ACK)
    Window size value: 258
    [Calculated window size: 66048]
    [Window size scaling factor: 256]
```

```
0020  c8 a3 c2 f9 00 50 d7 02  94 28 9a ac 12 a2 50 18   ·····P·· ·(····P·
0030  01 02 ca 81 00 00 47 45  54 20 2f 67 74 73 31 6f   ······GE T /gts1o
0040  31 2f 4d 46 45 77 54 7a  42 4e 4d 45 73 77 53 54   1/MFEwTz BNMEswST
0050  41 4a 42 67 55 72 44 67  4d 43 47 67 55 41 42 42   AJBgUrDg MCGgUABB
```

Stream index (tcp.stream) | Packets: 387351 · Displayed: 387351 (100.0%) | Profile: Defaul

The options are explained below:

VoIP calls	It stands for Voice over Internet Protocol. It gives the list of all the detected VoIP calls in the captured traffic. It shows the **start time, stop time, initial speaker, protocol, duration, packet, state**.
ANSI	It stands for American National Standards Institute. ANSI standards are developed by organizations who are authorized by it.

⇧ SCROLL TO TOP

GSM	It stands for Global System for Mobile. It has various options. It has multiple options, which are used to view the messages count over the traffic. For this, you have to connect your phone to the computer through the USB-TTL converter, verify the layer. After you have to load layer 1 Firmware into the osmocon. Run mobile and specify the interface for sending GSM TAP to listen to the interface through Wireshark.
IAX2 Stream Analysis	It shows the graph with the forward and the reverse streams.
ISUP Messages	It stands for **ISDN User Parts**. It is used to establish and release calls between telephone exchanges. It shows the messages by count and direction.
LTE	It stands for **Long Term Evolution**. It uses RRC (Radio Resource Control) protocol, which controls MAC and RLC layers in the LTE interface. It shows the statistics of the captured LTE MAC and LTE RLC traffic.
MTP3	It provides messaging routing between signaling points in the SS7 network. It shows its statistics and summary. It stands for **Message Transfer Part**.
Osmux	It is a multiplex protocol, which reduces the bandwidth by substituting the voice and signaling traffic. If it is not detected then Wireshark display this information of Osmux on UDP packets or flow.
RTP	It is called as RTP streams. It starts with the sequence number, packet number, and further stats are created based on the jitter, packet size, arrival time, and delay. It stands for **Real-time Transport Protocol**.
RTSP	It stands for Real-Time Streaming Protocol. It provides information about the packet counter of response packets and requests packets.
SCTP	It stands for Stream Control Transmission Protocol. It is designed to transmit PSTN signaling messages over IP networks. It is only applicable for broader applications.
SMPP Operations	It stands for Short Messages Peer to Peer. It determines the response, request, and operations of SMPP.

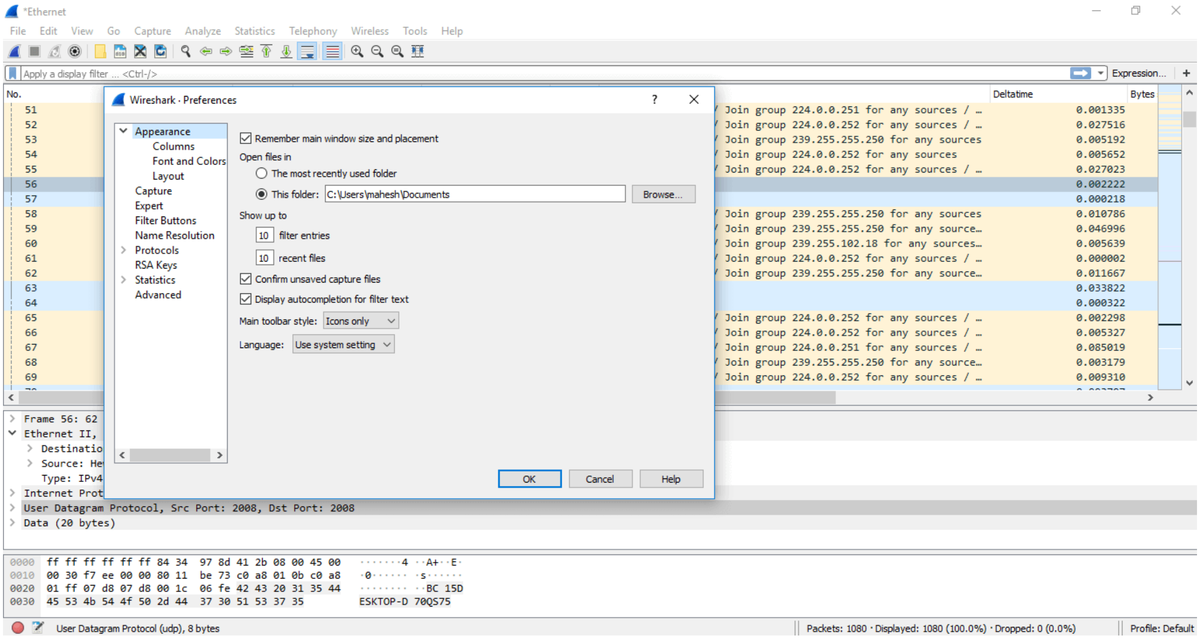
⇧ SCROLL TO TOP

UCP Messages	It is used to determine whether the captured packet is UCP or Nacks.
H.225	It is a streamed packetization and signaling protocol used for packet-based multimedia communication systems.
SIP Flows	It stands for Session Initiation Protocol. There is no need for any regular connection or multiples lines. Instead, it is installed on your current internet connection. It works with VoIP.
SIP Statistics	It gives information about the request methods and all of the SIP requests over a connection.
WAP-WSP Packet Counter	WSP stands for **Wireless Session Protocol**. It indicates the packets counts for all the Extended post methods, status codes, and PDU types. WAP uses short messages as a carrier.

WIRESHARK DECRYPTION

The decryption process is used for the data to be in a readable format. Below are the steps for the decryption process.

- Open the Wireshark and then select the particular interface as explained above.
- Go to the 'Edit' option and select the 'Preferences' option.
- A dialogue will appear as shown below:



☑ Remember main window size and placement

Open files in

○ The most recently used folder

● This folder: C:\Users\mahesh\Documents

Show up to

10 filter entries

10 recent files

☑ Confirm unsaved capture files

☑ Display autocompletion for filter text

Main toolbar style: Icons only

Language: Use system setting

```
51   Join group 224.0.0.251 for any sources / …      0.001335
52   Join group 224.0.0.252 for any sources / …      0.027516
53   Join group 239.255.255.250 for any sources      0.005192
54   Join group 224.0.0.252 for any sources          0.005652
55   Join group 224.0.0.252 for any sources / …      0.027023
56                                                   0.002222
57                                                   0.000218
58   Join group 239.255.255.250 for any sources      0.010786
59   Join group 239.255.255.250 for any source…      0.046996
60   Join group 239.255.102.18 for any sources…      0.005639
61   Join group 224.0.0.252 for any sources / …      0.000002
62   Join group 239.255.255.250 for any source…      0.011667
63                                                   0.033822
64                                                   0.000322
65   Join group 224.0.0.252 for any sources / …      0.002298
66   Join group 224.0.0.252 for any sources / …      0.005327
67   Join group 224.0.0.251 for any sources / …      0.085019
68   Join group 239.255.255.250 for any source…      0.003179
69   Join group 224.0.0.252 for any sources / …      0.009310
```

```
> Frame 56: 62
v Ethernet II,
  > Destinatio
  > Source: He
    Type: IPv4
> Internet Prot
> User Datagram Protocol, Src Port: 2008, Dst Port: 2008
> Data (20 bytes)
```

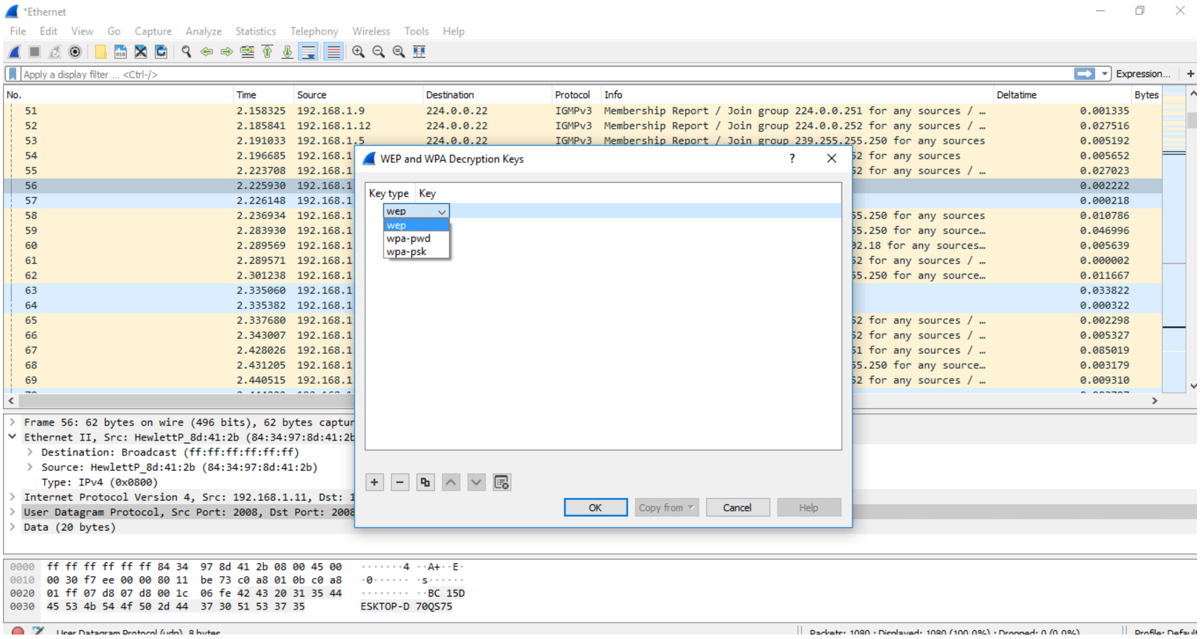
```
0000  ff ff ff ff ff ff 84 34  97 8d 41 2b 08 00 45 00   ·······4 ··A+··E·
0010  00 30 f7 ee 00 00 80 11  be 73 c0 a8 01 0b c0 a8   ·0······ ·s······
0020  01 ff 07 d8 07 d8 00 1c  06 fe 42 43 20 31 35 44   ········ ··BC 15D
0030  45 53 4b 54 4f 50 2d 44  37 30 51 53 37 35         ESKTOP-D 70QS75
```

User Datagram Protocol (udp), 8 bytes | Packets: 1080 · Displayed: 1080 (100.0%) · Dropped: 0 (0.0%) | Profile: Default

- Select the 'Protocol' option in the left column.
- From the drop-down list, select the 'IEEE 802.11' option. Check the box of [illegible]yption and click on the Edit option under it.

⇧ SCROLL TO TOP

- A box will appear. Click on the option shown below:


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+
WEP and WPA Decryption Keys
?
Key type
Key
wep
wep
wpa-pwd
wpa-psk
OK
Copy from
Cancel
Help
User Datagram Protocol (udp), 8 bytes
Packets: 1080 · Displayed: 1080 (100.0%) · Dropped: 0 (0.0%)
Profile: Default

```
No.   Time      Source        Destination  Protocol Info                                                         Deltatime  Bytes
51    2.158325  192.168.1.9   224.0.0.22   IGMPv3   Membership Report / Join group 224.0.0.251 for any sources / …   0.001335
52    2.185841  192.168.1.12  224.0.0.22   IGMPv3   Membership Report / Join group 224.0.0.252 for any sources / …   0.027516
53    2.191033  192.168.1.5   224.0.0.22   IGMPv3   Membership Report / Join group 239.255.255.250 for any sources   0.005192
54    2.196685  192.168.1                                                    52 for any sources                  0.005652
55    2.223708  192.168.1                                                    52 for any sources / …              0.027023
56    2.225930  192.168.1                                                                                        0.002222
57    2.226148  192.168.1                                                                                        0.000218
58    2.236934  192.168.1                                                    55.250 for any sources              0.010786
59    2.283930  192.168.1                                                    55.250 for any source…              0.046996
60    2.289569  192.168.1                                                    02.18 for any sources…              0.005639
61    2.289571  192.168.1                                                    52 for any sources / …              0.000002
62    2.301238  192.168.1                                                    55.250 for any source…              0.011667
63    2.335060  192.168.1                                                                                        0.033822
64    2.335382  192.168.1                                                                                        0.000322
65    2.337680  192.168.1                                                    52 for any sources / …              0.002298
66    2.343007  192.168.1                                                    52 for any sources / …              0.005327
67    2.428026  192.168.1                                                    51 for any sources / …              0.085019
68    2.431205  192.168.1                                                    55.250 for any source…              0.003179
69    2.440515  192.168.1                                                    52 for any sources / …              0.009310
```

```
> Frame 56: 62 bytes on wire (496 bits), 62 bytes captur
v Ethernet II, Src: HewlettP_8d:41:2b (84:34:97:8d:41:2b
  > Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  > Source: HewlettP_8d:41:2b (84:34:97:8d:41:2b)
    Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 192.168.1.11, Dst: 1
> User Datagram Protocol, Src Port: 2008, Dst Port: 2008
> Data (20 bytes)
```

```
0000  ff ff ff ff ff ff 84 34  97 8d 41 2b 08 00 45 00   ·······4 ··A+··E·
0010  00 30 f7 ee 00 00 80 11  be 73 c0 a8 01 0b c0 a8   ·0······ ·s······
0020  01 ff 07 d8 07 d8 00 1c  06 fe 42 43 20 31 35 44   ········ ··BC 15D
0030  45 53 4b 54 4f 50 2d 44  37 30 51 53 37 35         ESKTOP-D 70QS75
```

- Select the option **wpa-pwd** and set the password accordingly.
- The data will be decrypted.
- But the above decryption process is only possible if there is a proper handshake.

⇧ SCROLL TO TOP