Practical 12

Aim: Study and use Wireshark For Various Network Protocol.

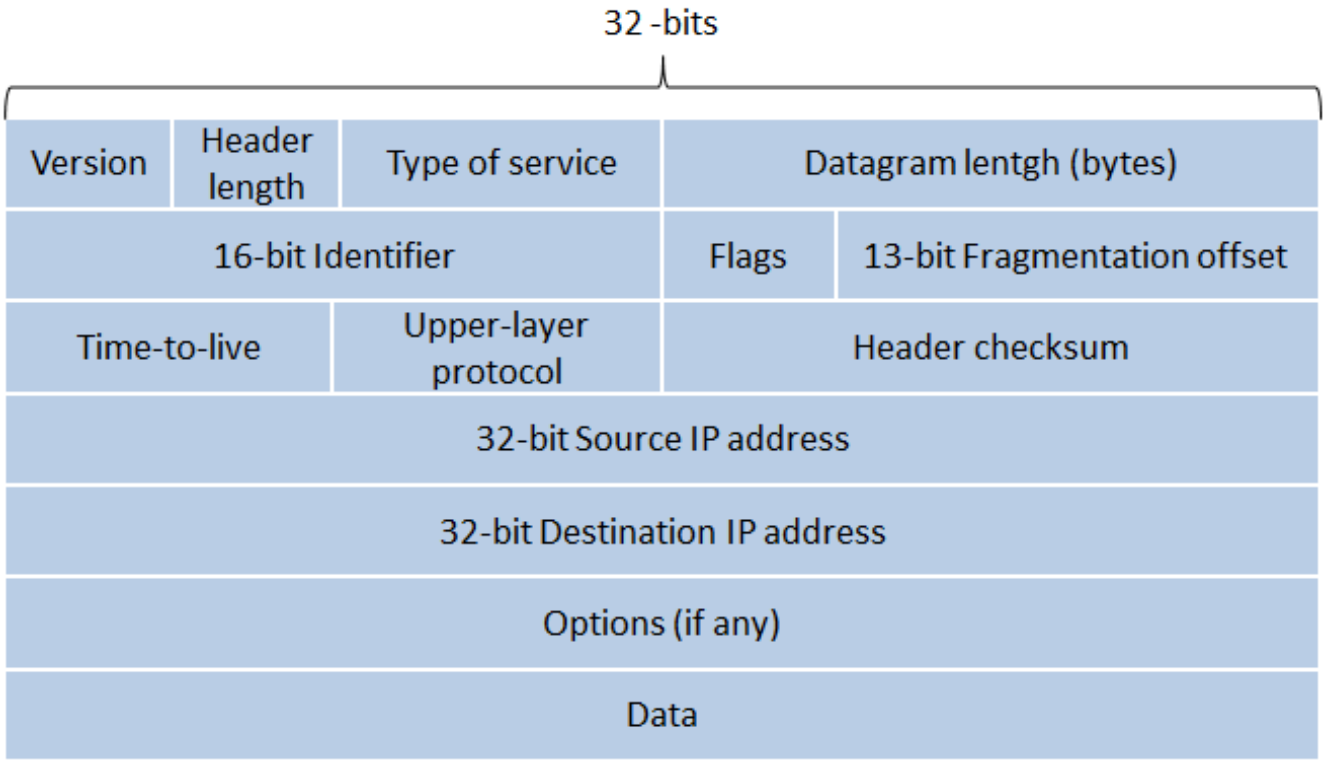
Description:

- Wireshark is a free and open source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education. Originally named Ethereal, the project was renamed Wireshark in May 2006 due to trademark issues.
- Wireshark is cross-platform, using the Qt widget toolkit in current releases to implement its user interface, and using pcap to capture packets; it runs on Linux, macOS, BSD, Solaris, some other Unix-like operating systems, and Microsoft Windows.
- There is also a terminal-based (non-GUI) version called TShark. Wireshark, and the other programs distributed with it such as TShark, are free software, released under the terms of the GNU General Public License.

STUDING TCP/UDP USING WIRESHARK

- **Internet Protocol**

- The Internet Protocol (IP) is the method or protocol by which data is sent from one computer to another on the Internet. Each computer (known as a host) on the Internet has at least one IP address that uniquely identifies from all other computers on the Internet.



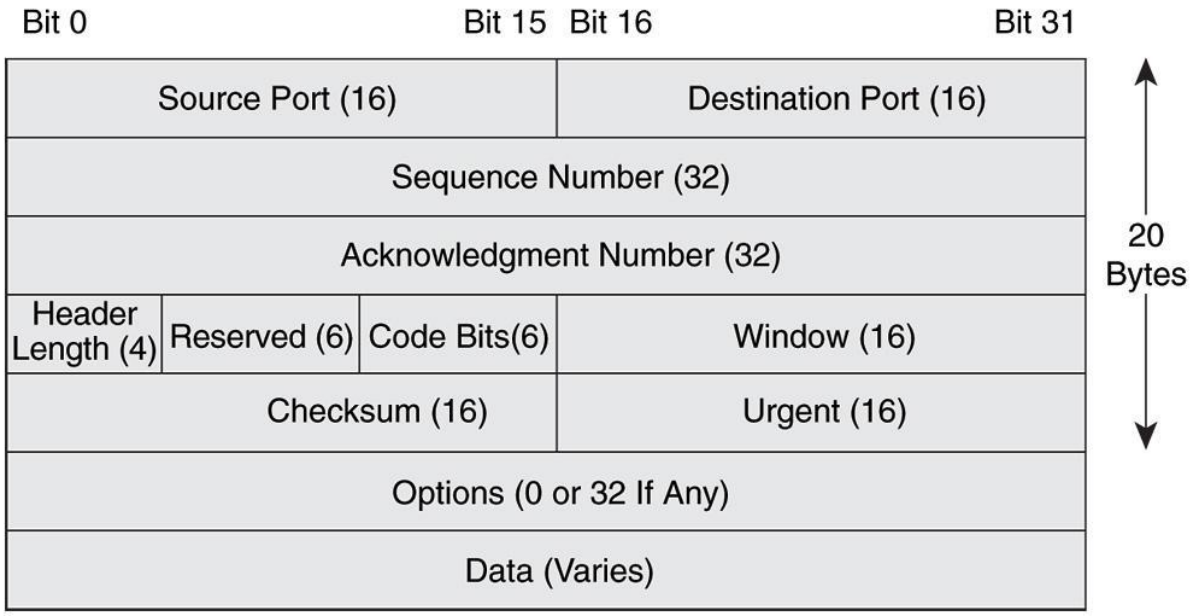
- Packet Analysing

```
⊞ Frame 580: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
⊞ Ethernet II, Src: HewlettP_90:42:d1 (10:60:4b:90:42:d1), Dst: D-LinkIn_6f:b0:3e (cc:b2:55:6f:b0:3e)
⊟ Internet Protocol Version 4, Src: 10.0.62.144 (10.0.62.144), Dst: c-0001.c-msedge.net (13.107.4.50)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 40
    Identification: 0x0727 (1831)
  ⊞ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 128
    Protocol: TCP (6)
  ⊞ Header checksum: 0x997c [correct]
    Source: 10.0.62.144 (10.0.62.144)
    Destination: c-0001.c-msedge.net (13.107.4.50)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
⊞ Transmission Control Protocol, Src Port: menandmice-lpm (1231), Dst Port: http (80), Seq: 266, Ack: 274, L

0000  cc b2 55 6f b0 3e 10 60  4b 90 42 d1 08 00 45 00   ..Uo.>.` K.B...E.
0010  00 28 07 27 40 00 80 06  99 7c 0a 00 3e 90 0d 6b   .(.'@... .|..>..k
0020  04 32 04 cf 00 50 d8 0c  4c 4a 3e a0 60 2a 50 10   .2...P.. LJ>.`*P.
0030  00 00 8d 67 00 00                                 ...g..
```

TCP Frame format:

- The Transmission Control Protocol (TCP) is a core protocol of the Internet protocol suite. It originated in the initial network implementation in which it complemented the Internet Protocol (IP). Therefore, the entire suite is commonly referred to as TCP/IP.



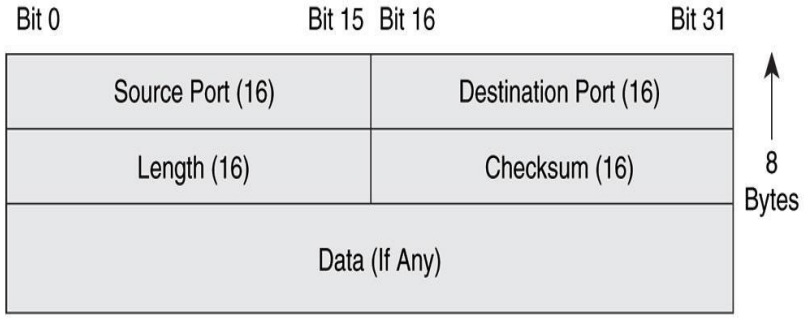
- Packet Analyzing

```
> Frame 99: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0
> Ethernet II, Src: Tp-LinkT_90:a9:40 (30:b5:c2:90:a9:40), Dst: LiteonTe_63:0b:bf (48:d2:24:63:0b:bf)
> Internet Protocol Version 4, Src: 23.52.67.184, Dst: 192.168.0.104
v Transmission Control Protocol, Src Port: 443 (443), Dst Port: 49815 (49815), Seq: 1, Ack: 2, Len: 0
    Source Port: 443
    Destination Port: 49815
    [Stream index: 45]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Acknowledgment number: 2    (relative ack number)
    Header Length: 32 bytes
  > Flags: 0x010 (ACK)
    Window size value: 988
    [Calculated window size: 988]
    [Window size scaling factor: -1 (unknown)]
  > Checksum: 0x0110 [validation disabled]
    Urgent pointer: 0
  > Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), SACK
  > [SEQ/ACK analysis]

0000  48 d2 24 63 0b bf 30 b5  c2 90 a9 40 08 00 45 00   H.$c..0. ...@..E.
0010  00 34 3e 2b 40 00 39 06  e7 9c 17 34 43 b8 c0 a8   .4>+@.9. ...4C...
0020  00 68 01 bb c2 97 01 70  6e 3e 91 e8 cf b3 80 10   .h.....p n>......
0030  03 dc 01 10 00 00 01 01  05 0a 91 e8 cf b2 91 e8   ........ ........
0040  cf b3 4d 7c                                       ..M|
```

UDP frame format:

- UDP uses a simple connectionless transmission model with a minimum of protocol mechanism. It has no handshaking dialogues, and thus exposes any unreliability of the underlying network protocol to the user's program.



No Sequence Or Acknowledgment Fields

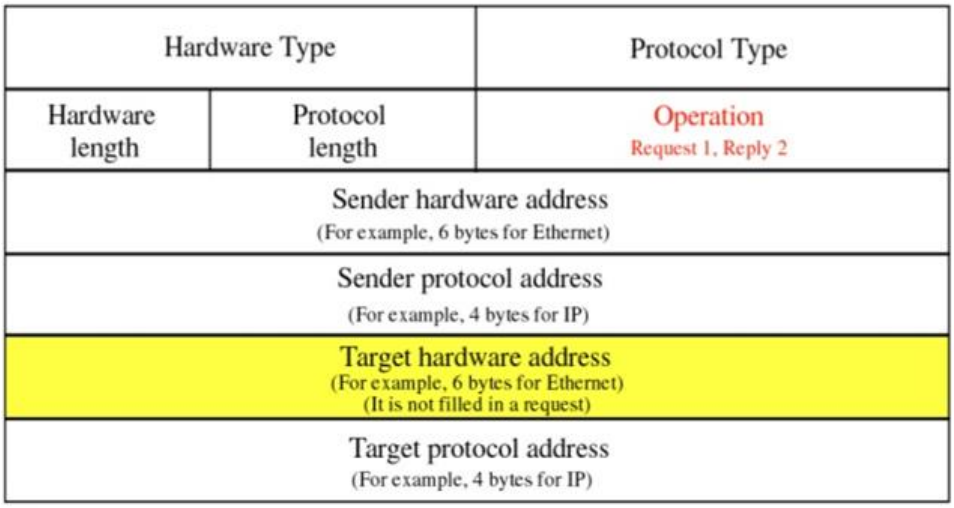
- Packet Analyzing

```
    Protocol: UDP (17)
  + Header checksum: 0x5207 [correct]
    Source: ubuntu-25.local (10.0.62.92)
    Destination: 224.0.0.251 (224.0.0.251)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
+ User Datagram Protocol, Src Port: mdns (5353), Dst Port: mdns (5353)
+ Domain Name System (query)

0020  00 fb 14 e9 14 e9 00 7a  68 eb 00 00 00 00 00 01   .......z h.......
0030  00 00 00 02 00 00 1d 75  62 75 6e 74 75 2d 32 35   .......u buntu-25
0040  20 5b 31 30 3a 36 30 3a  34 62 3a 39 30 3a 34 34    [10:60: 4b:90:44
0050  3a 31 35 5d 0c 5f 77 6f  72 6b 73 74 61 74 69 6f   :15]._wo rkstatio
0060  6e 04 5f 74 63 70 05 6c  6f 63 61 6c 00 00 ff 00   n._tcp.l ocal....
```

Address Resolution Protocol (ARP)

- Address Resolution Protocol (ARP) is one of the major protocol in the TCP/IP suit and the purpose of Address Resolution Protocol (ARP) is to resolve an IPv4 address (32 bit Logical Address) to the physical address (48 bit MAC Address). Network Applications at the Application Layer use IPv4 Address to communicate with another device.



Packet Analyzing