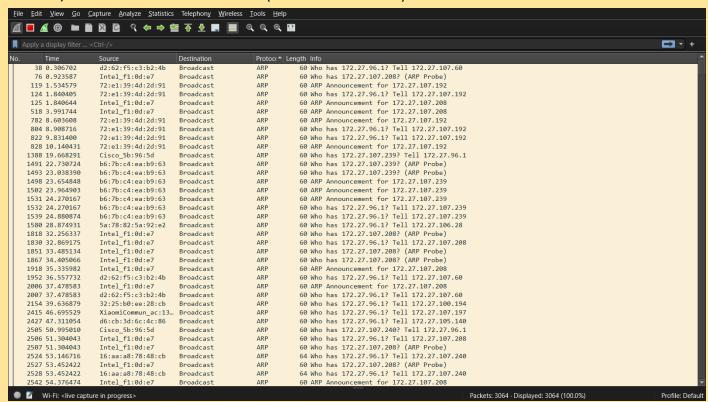
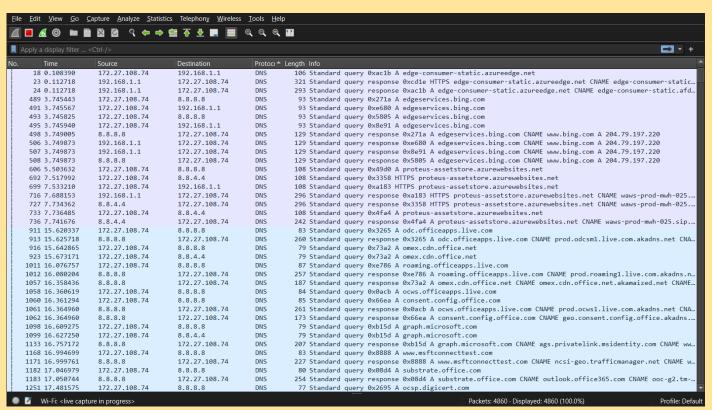
(a

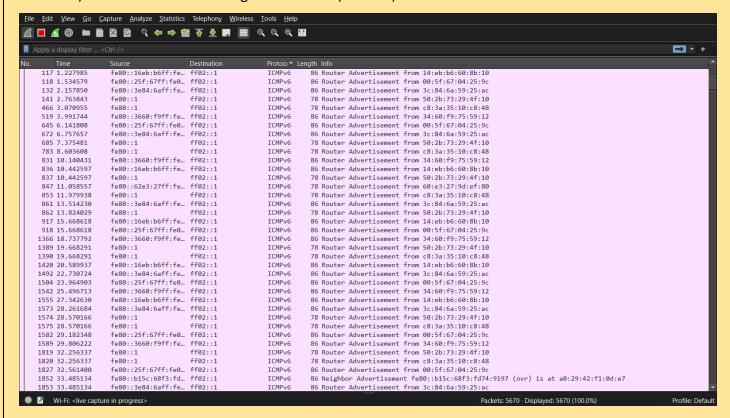
i) Address Resolution Protocol (ARP Announcement):



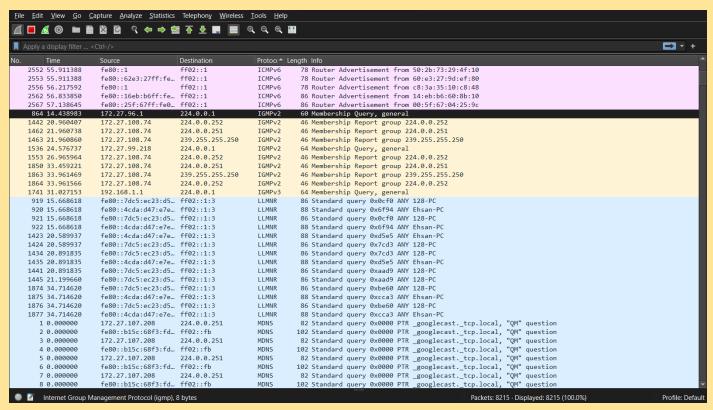
ii) Domain Name System (response)(DNS):



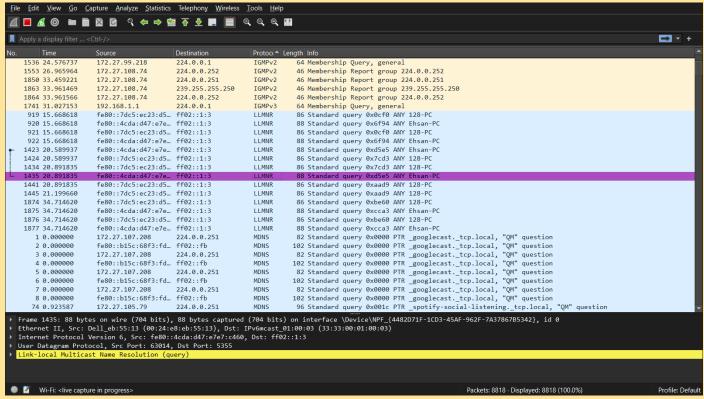
iii) Internet Control Message Protocol v6 (ICMPv6):



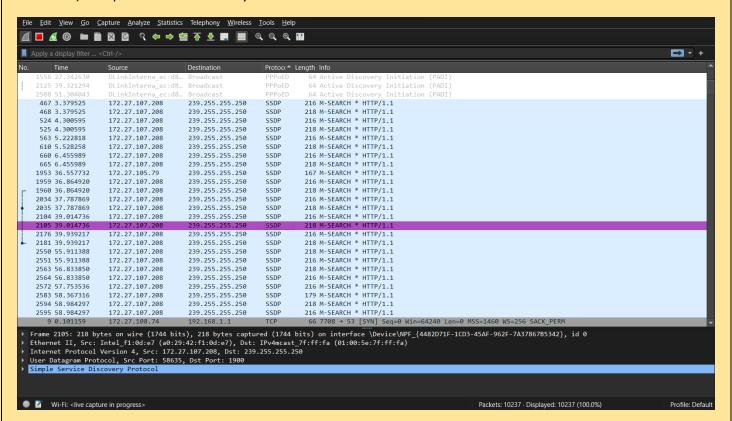
iv) Internet Group Management Protocol:



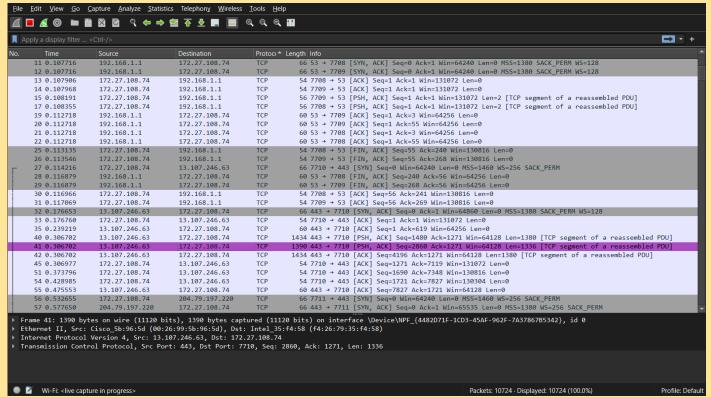
v) Link-local Multicast Name Resolution:



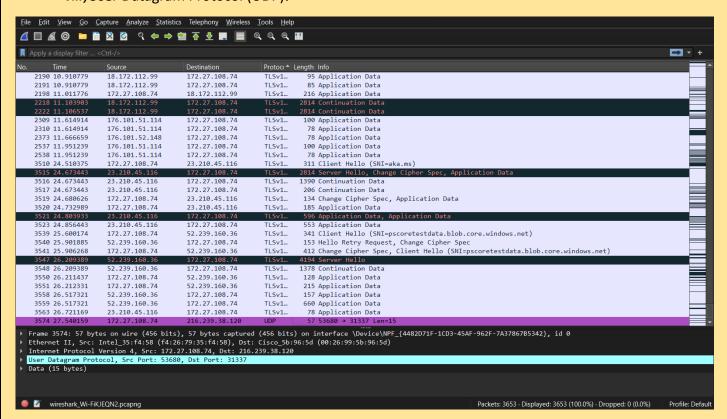
vi) Simple Service Discovery Protocol:

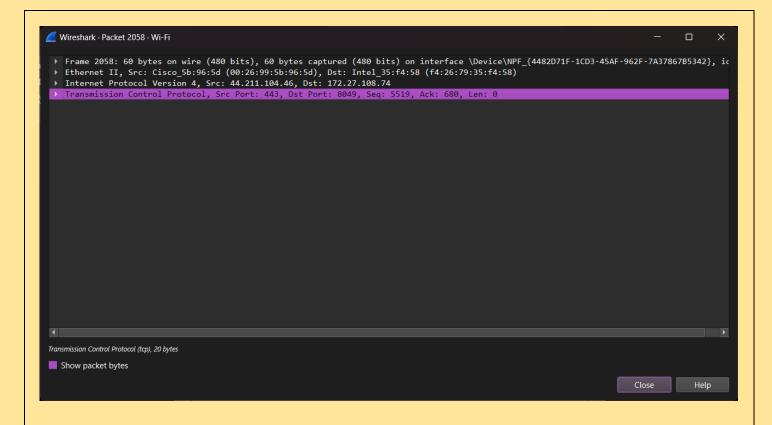


vii) Transmission Control Protocol (TCP):



viii)User Datagram Protocol (UDP):





i) Physical Layer: Bit

ii) Link Layer: Ethernet 2

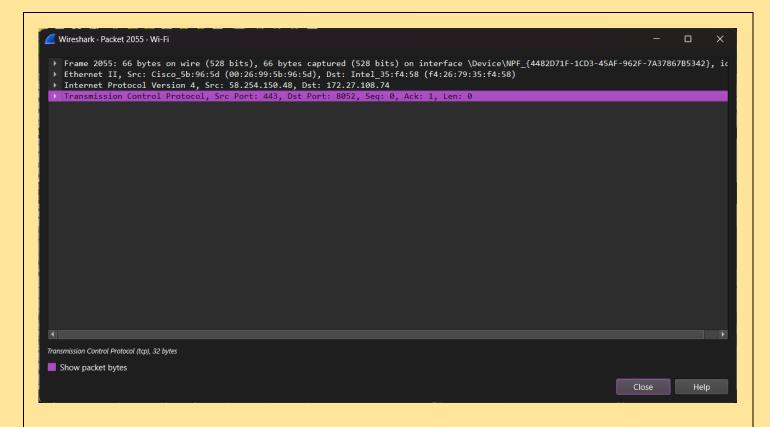
iii) Network Layer: Internet Protocol Version 4 (IPV4)

iv) Transport Layer: Transmission Control Protocol (TCP)

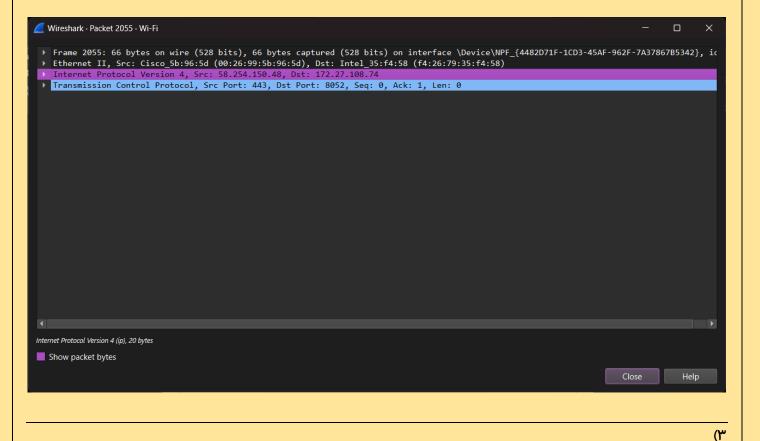
b)

کل بیت ها بوسیله لایه فیزیکی انتقال پیدا می کنند و ابتدای بیت ها مربوط به link layer است ،بعد network layer ،بعد transport layer ،بعد transport layer

c) 32 bytes يا 256 bits



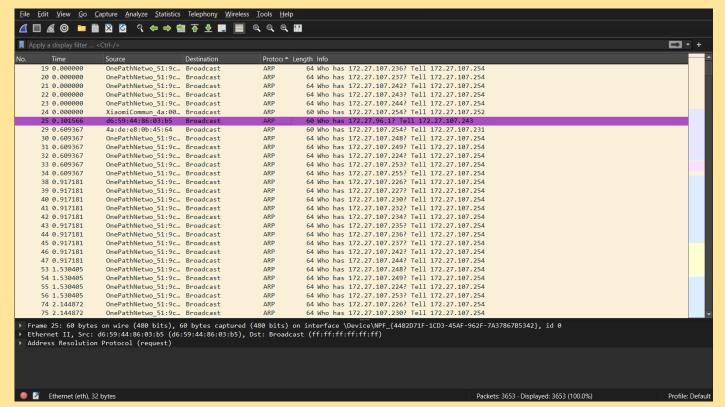
d) 20 bytes يا 160 bits



a) بله –

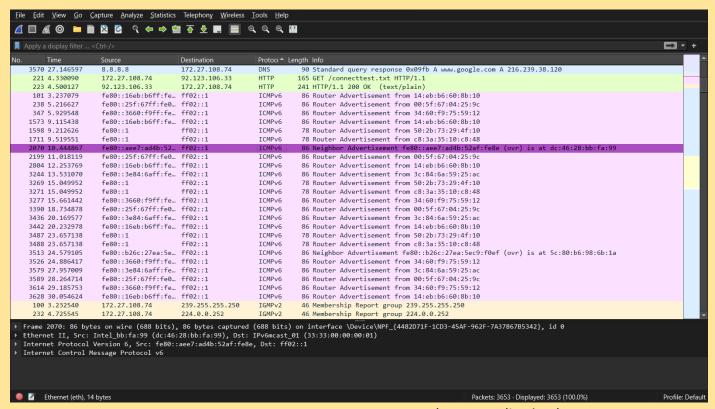
(b

i) ARP:



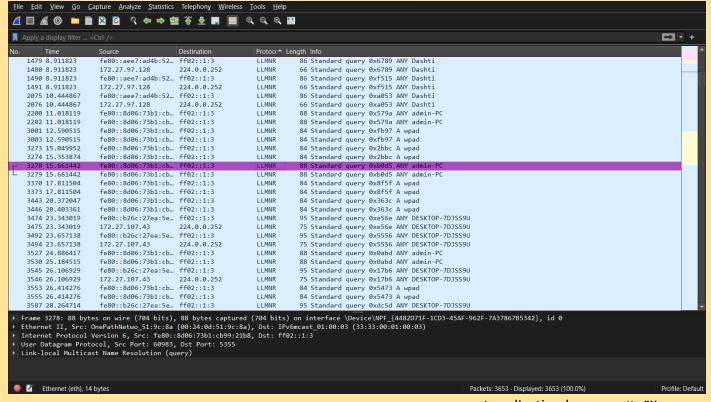
فاقد لایه های transport layer ، application layer و network layer است.

ii) ICMPv6, IGMPv2:



فاقد لایه های application layer و transport layer است.

iii) Link-local Multicast Name Resolution, Multicast Domain Name System:



فاقد لایه ی application layer است.

(۴

a) TCP:

```
■ Wireshark · Packet 1029 · Wi-Fi

  Internet Protocol Version 4, Src: 216.239.38.120, Dst: 172.27.108.74
  Transmission Control Protocol, Src Port: 80, Dst Port: 7347, Seq: 10444, Ack: 103, Len: 1380
      [Stream index: 24]
      [Conversation completeness: Complete, WITH_DATA (31)]
      [TCP Segment Len: 1380]
      Sequence Number: 10444
                                    (relative sequence number)
      Sequence Number (raw): 591080490
     [Next Sequence Number: 11824 (relative sequence number)]
Acknowledgment Number: 103 (relative ack number)
Acknowledgment number (raw): 44098867
     0101 .... = Header Length: 20 bytes (5)
     Flags: 0x010 (ACK)
     Window: 256
      [Calculated window size: 65536]
      [Window size scaling factor: 256]
     Urgent Pointer: 0
      [Timestamps]
      [SEQ/ACK analysis]
      TCP payload (1380 bytes)
      TCP segment data (1380 bytes)
```

b) UDP:

```
Wireshark · Packet 959 · Wi-Fi

Frame 959: 57 bytes on wire (456 bits), 57 bytes captured (456 bits) on interface \Device\NPF_{4482D71F-1CD3-45AF-962F-7A37867B5342}, id

Ethernet II, Src: Intel_35:f4:58 (f4:26:79:35:f4:58), Dst: Cisco_5b:96:5d (00:26:99:5b:96:5d)

Internet Protocol Version 4, Src: 172.27.108.74, Dst: 216.239.38.120

User Datagram Protocol, Src Port: 55454, Dst Port: 31337

Source Port: 55454

Destination Port: 31337

Length: 23

Checksum: 0x17f6 [correct] (matches partial checksum, not 0xf8b3, likely caused by "UDP checksum offload")

[Checksum Status: Good]

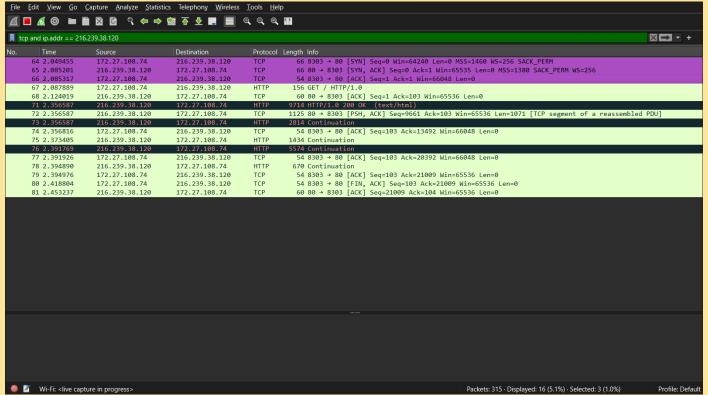
[Stream index: 30]

[Timestamps]

UDP payload (15 bytes)
```

(۵

a) Handshake:

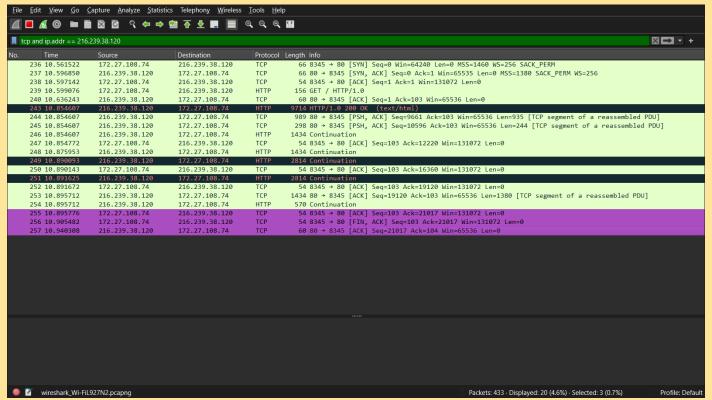


- i) Client → Server: Seq number: 0, Ack number: 0, Window size: 64240, tcp.flags.syn = 1, tcp.flags.ack = 0, tcp.flags.fin = 0;
- ii) Server→Client: Seq number: 0, Ack number: 1, Window size: 65535, tcp.flags.syn = 1,
 tcp.flags.ack = 1, tcp.flags.fin = 0;
- iii) Client → Server: Seq number: 1, Ack number: 1, Window size: 512 tcp.flags.syn = 0,

tcp.flags.ack = 1, tcp.flags.fin = 0;

ابتدا کلاینت به سرور درخواست سینک شدن می فرستد با فعال کردن tcp.flags.syn بعد سرور درخواست کلاینت را با قبول کردن شماره سکونس می پزیرد و بیت های مریوط به tcp.flags.ack و tcp.flags.syn را یک می کنه و برای کلاینت ارسال می کند و کلاینت هم با قبول کردن سکونس نامبر قبلی درخاستش را پاسخ می دهد.

b) Connection Closer:



i) Client \rightarrow Server: Seq number = 20104, Ack number = 103, Window size = 256,

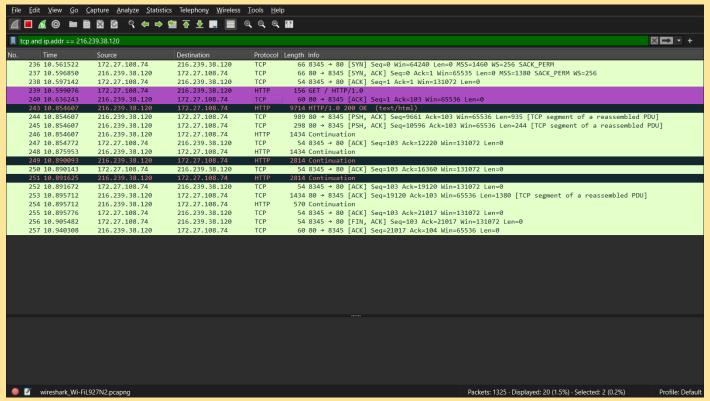
tcp.flags.syn = 0, tcp.flags.ack = 1, tcp.flags.fin = 1;

- ii) Server→Client: Seq number: 103, Ack number: 21068, Window size: 508, Tcp.flags.syn = 0,
 Tcp.flags.ack = 1, tcp.flags.fin = 0;
- iii) Server→Client: Seq number: 103, Ack number: 21068, Window size: 508, Tcp.flags.syn = 0,Tcp.flags.ack = 1, tcp.flags.fin = 1;
- iv) Client →Server: Seq number: 21068, Ack number: 104, Window size: 256, Tcp.flags.syn = 0,
 Tcp.flags.ack = 1, tcp.flags.fin = 0;

ابتدا کلاینت درخواست fin را به سرور می دهد و منتظر پاسخ سرور می ایستد بعد سرو fin و ack قبلی را در جواب کلاینت پشت

سر هم می فرستد و کلاینت هم با ack کردن درخواست سرور ارتباط را می بندد.

c) Connection Duration:

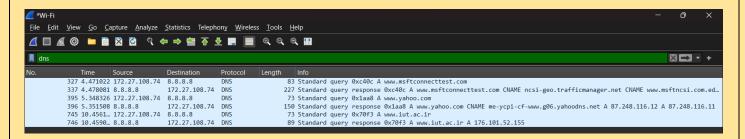


- i) Client → Server: Seq number: 1, Ack number: 1, Window size: 512, Tcp.flags.syn = 0, Tcp.flags.ack
 = 1, tcp.flags.fin = 0;
- ii) Server→Client: Seq number: 1, Ack number: 103, Window size:256, Tcp.flags.syn = 0,Tcp.flags.ack = 1, tcp.flags.fin = 0;

کلاینت درخواست خودش را با سکونس نامبری به سمت سرور ارسال می کند و سرور با اک نامبر سکونس بعدی و سکونس نامبر

خودش به درخواست کلاینت پاسخ می دهد.

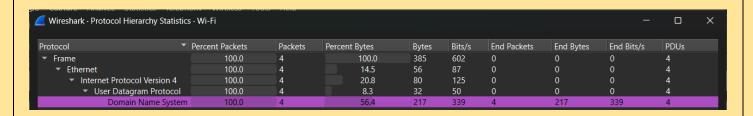




Response time www.yahoo.com: [Time: 0.003182000 seconds];

Response time <u>www.iut.ac.ir</u>: [Time: 0.002912000 seconds]

Protocol Hierarchy:



(Y

