

Anket Sanjay Kotkar

Roll no.: 180101037

Question 1

From studying the traces, it can be inferred that following layers are present in the packets: TLS protocol (session layer), TCP (transport layer), IP4 (networks layer), Ethernet (link layer). The detailed explanation about them is as follows:

Session Layer

TLS: This is the cryptographic protocol which encrypts the data to provide communications security over a computer network. TLS is the betterment of older SSL layer. basic unit of TLS is called record. The identity of the communicating parties is authenticated using public-key cryptography. Some fields of the layer are as follows:

- **Content type:** This denotes the type of action which took place. There are standard values and description assigned to it. Those are 0-19 for Unassigned (Requires coordination), 20 for change_cipher_spec, 21 for alert, 22 for handshake, 23 for application_data, 24 for heartbeat, 25 for tls12_cid, 26-63 for Unassigned. Value observed is 22.
- **Version:** It denotes the version of TLS we are currently using for communication. Value observed 1.2
- **Length:** It is length of data which this layer transmits for the particular packet. Value observed: 173.
- **Handshake Protocol - Handshake type:** This protocol is used to negotiate the secure attributes of a session. Handshake messages are supplied to the TLS Record Layer, where they are encapsulated within one or more TLSPlaintext structures, which are processed and transmitted as specified by the current active session state. Value observed Client Hello(1).
- **Handshake Protocol - Random Number:** To generate the session keys used for secure connection, client encrypts a random number with server's public key and sends the result to server.

```
▼ Transport Layer Security
  ▼ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 173
  ▼ Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 169
    Version: TLS 1.2 (0x0303)
    ▼ Random: 5f731f6d28339bc7021af8ec17c549e8e894b2f09f637930...
      GMT Unix Time: Sep 29, 2020 17:20:05.000000000 India Standard Time
      Random Bytes: 28339bc7021af8ec17c549e8e894b2f09f637930f35afaab...
    Session ID Length: 0
    Cipher Suites Length: 42
    > Cipher Suites (21 suites)
    Compression Methods Length: 1
    > Compression Methods (1 method)
    Extensions Length: 86
    > Extension: server_name (len=21)
    > Extension: supported_groups (len=8)
    > Extension: ec_point_formats (len=2)
    > Extension: signature_algorithms (len=26)
    > Extension: session_ticket (len=0)
    > Extension: extended_master_secret (len=0)
    > Extension: renegotiation_info (len=1)
```

Transport Layer

TCP: TCP (Transmission Control Protocol) is a standard that defines how to establish and maintain a network conversation through which application programs can exchange data. TCP works with the Internet Protocol (IP), which defines how computers send packets of data to each other. It is a connection-oriented protocol,

which means a connection is established and maintained until the application programs at each end have finished exchanging messages. Some fields of the protocol are as follow:

- **Source Port:** It is port thorough which the netowrk request/responce is sent.
 - **Destination port:** It is port thorough which the netowrk request/responce is recieved.
- In my case, the connection is between 443 port of whatsapp server and 53314 port of my ip address. •
- Stream Index:**the stream index is an internal Wireshark mapping to: [IP address A, TCP port A, IP address B, TCP port B] All the packets for the same tcp.stream value should have the same values for these fields (though the src/dest will be switched for A- \leftrightarrow B and B- \leftrightarrow A packets).
- **Sequence Number:** The sequence number is the byte number of the first byte of data in the TCP packet sent (also called a TCP segment).
 - **Acknowledgement Number:** The acknowledgement number is the sequence number of the next byte the receiver expects to receive.
 - **Window Size value and scaling factor:** The TCP window size is used by the receiver to tell the sender how much data to transmit before expecting an acknowledgment.Maximum allowed value is 65536 bytes. To increase this beyond the limit, scaling factor is used.
 - **Flags:** TCP flags are used within TCP packet transfers to indicate a particular connection state.

```
Transmission Control Protocol, Src Port: https (443), Dst Port: 5:
  Source Port: https (443)
  Destination Port: 53314 (53314)
  [Stream index: 2]
  [TCP Segment Len: 0]
  Sequence number: 2636      (relative sequence number)
  Sequence number (raw): 2516148480
  [Next sequence number: 2636      (relative sequence number)]
  Acknowledgment number: 1250      (relative ack number)
  Acknowledgment number (raw): 2838828229
  0101 .... = Header Length: 20 bytes (5)
  ▾ Flags: 0x010 (ACK)
    000. .... = Reserved: Not set
    ...0 .... = Nonce: Not set
    .... 0... = Congestion Window Reduced (CWR): Not set
    .... .0.. = ECN-Echo: Not set
    .... ..0. = Urgent: Not set
    .... ....1 = Acknowledgment: Set
    .... .... 0... = Push: Not set
    .... .... .0.. = Reset: Not set
    .... .... ..0. = Syn: Not set
    .... .... ...0 = Fin: Not set
    [TCP Flags: .....A....]
  Window size value: 122
  [Calculated window size: 31232]
  [Window size scaling factor: 256]
  Checksum: 0xf915 [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
  > [SEQ/ACK analysis]
```

Network Layer

Internet Protocol 4:It is one of the core protocols of standards-based internetworking methods in the Internet and other packet-switched networks. IPv4 uses a 32-bit address.It uses a logical addressing system and performs routing, which is the forwarding of packets from a source host to the next router that is one hop closer to the intended destination host on another network.

- **Version:** Version of internet protocol used.
- **Header Length:** It consists of 14 fields, with 14th field being optional. So it is variable in length from 20 byte to 60 byte. It has the values of various fields correpsonding to the version of ip using , type of service, total length, etc.
- **Total length:** It is the total size of IP4 datagram in bytes.
- **Identification:** Identification (ID) field is a 16-bit value that is unique for every datagram for a given source address, destination address, and protocol, such that it does not repeat within the maximum datagram lifetime (MDL).
- **Source:** The IP address of the host from which datagram is sent.

- **Destination:** The IP address of the host to which datagram is sent.
- **Time to live:** Time-to-live (TTL) is a value in an Internet Protocol (IP) packet that tells a network router whether or not the packet has been in the network too long and should be discarded.

```

Internet Protocol Version 4, Src: mmx-ds.cdn.whatsapp.net (31.13.79.53), Dst: 192.168.0.10
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1432
    Identification: 0xd56d (54637)
  > Flags: 0x4000, Don't fragment
    0... .. = Reserved bit: Not set
    .1... .. = Don't fragment: Set
    ..0. .... = More fragments: Not set
  Fragment offset: 0
  Time to live: 90
  Protocol: TCP (6)
  Header checksum: 0x16a0 [validation disabled]
  [Header checksum status: Unverified]
  Source: mmx-ds.cdn.whatsapp.net (31.13.79.53)
  Destination: 192.168.0.104 (192.168.0.104)

```

Link Layer

Ethernet II: Ethernet is a family of computer networking technologies. Systems communicating over Ethernet divide a stream of data into shorter pieces called frames. Each frame contains source and destination addresses, and error-checking data so that damaged frames can be detected and discarded; most often, higher-layer protocols trigger retransmission of lost frames. As per the OSI model, Ethernet provides services up to and including the data link layer.

- **Destination address:** The address to which a frame or packet of data is sent over a network. The destination address can be one of the following: The physical address, such as the MAC address of an Ethernet frame or The logical address, such as the IP address of an IP packet.
- **Source address:** The address from which a frame or packet of data is sent over a network. The destination address can be one of the following: The physical address, such as the MAC address of an Ethernet frame or The logical address, such as the IP address of an IP packet.
- **Type:**

It is used to indicate which protocol is encapsulated in the payload of the frame and is used at the receiving end by the data link layer to determine how the payload is processed.

- **LG bit and IG bit :** LG bit (sometimes also referred to as UL bit) and the IG bit are located in the most significant byte of each MAC address, where the IG bit is the least significant bit in this byte and the LG bit is the second least significant bit in this byte.

```

Ethernet II, Src: D-LinkIn_c2:2d:7e (bc:0f:9a:c2:2d:7e), Dst: HonHaiPr_ea:64:7b (74:40:bb:ea:64:7b)
  > Destination: HonHaiPr_ea:64:7b (74:40:bb:ea:64:7b)
    Address: HonHaiPr_ea:64:7b (74:40:bb:ea:64:7b)
    .... ..0. .... = LG bit: Globally unique address (factory default)
    .... ..0. .... = IG bit: Individual address (unicast)
  > Source: D-LinkIn_c2:2d:7e (bc:0f:9a:c2:2d:7e)
    Address: D-LinkIn_c2:2d:7e (bc:0f:9a:c2:2d:7e)
    .... ..0. .... = LG bit: Globally unique address (factory default)
    .... ..0. .... = IG bit: Individual address (unicast)
  Type: IPv4 (0x0800)

```

Question 2

The important functionalities of whatsapp (application assigned to me):

- Sending image - protocol used TCP
- Sending Video - protocol used TCP
- Sending Files (pdf, word, etc) - protocol used TCP
- Voice messages - protocol used UDP

(e) Send Messages - protocol used TCP

Observed protocols and their functionalities can be described as follow:

- (a) **TCP:** It does handshake which is the transfer of the data when a connection request between a server and the client is established. It sends the acknowledgement signals in respond to the received signal by the server. This protocol is used while sending messages, images, videos or files. This is because TCP ensures the correct transmission of the data without error tolerance.
- (b) **UDP:** It is the transportaion layer protocol used for purposes like voice messages as in this type of data transfer. tolerance for some amount of data loss is alllowed.
- (c) **DNS:** DNS is general directory service used for resolving the server name by mapping between IP address and sevrer names.

Question 3

Let us discuss the sequence of messages exchanged between host and the server for following 2 functionalities.

- **Sending files :**

- (a) **TCP handshaking:** There is a three-way-handshake between the source, the destination. We have already seen In the previous question that port 53314 is being used in my laptop and port 443 is being used by whatsapp. As the SYN packet which helps in synchronizing the sequence numbers is sent, the handshaking process advents. Then destination responds to this packet by sending an ACK (acknowledgement) and another SYN packet which asks the source to synchronize the packet number with its sequence number. In the end, the source sends a final ACK packet and with this, we say that the handshaking process is successful.

```
6 0.001273 192.168.0.104 mmx-ds.cdn.whatsapp... TCP 66 59019 → https(443) [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
7 0.096621 mmx-ds.cdn.whatsapp... 192.168.0.104 TCP 66 https(443) → 59019 [SYN, ACK] Seq=0 Ack=1 Win=27840 Len=0 MSS=1392 SACK_PERM=1 WS=256
8 0.000083 192.168.0.104 mmx-ds.cdn.whatsapp... TCP 54 59019 → https(443) [ACK] Seq=1 Ack=1 Win=132096 Len=0
```

- (b) **TLS handshaking:** TLS handshaking proceeds once the TCP handshaking is done. TLS handshaking makes sure that communication is secure and going properly. My laptop sends the Client Hello message and the server in respoce responds with Server Hello which can be seen by the screenshot attached.

```
9 0.001034 192.168.0.104 mmx-ds.cdn.whatsapp... TLSv1.3 571 Client Hello
10 0.101327 mmx-ds.cdn.whatsapp... 192.168.0.104 TCP 54 https(443) → 59019 [ACK] Seq=1 Ack=518 Win=28928 Len=0
11 0.000565 mmx-ds.cdn.whatsapp... 192.168.0.104 TLSv1.3 266 Server Hello, Change Cipher Spec, Application Data
```

- (c) **Sending file:** TCP has a limit how many bytes can be sent in a single data packate. So usually the file sent is packed in different packets partially and sent. Following screen shot shows one of the such instances when the file is getting sent.

```
0.003974 192.168.0.104 mmx-ds.cdn.whatsapp... TCP 54 63617 → https(443) [ACK] Seq=1057 Ack=2553 Win=131840 Len=0
0.197846 mmx-ds.cdn.whatsapp... 192.168.0.104 TLSv1.3 433 Application Data
0.042105 192.168.0.104 mmx-ds.cdn.whatsapp... TLSv1.2 587 Application Data
0.005721 mmx-ds.cdn.whatsapp... 192.168.0.104 TCP 54 https(443) → 53314 [ACK] Seq=1401 Ack=653 Win=548 Len=0
0.010458 192.168.0.104 mmx-ds.cdn.whatsapp... TCP 54 63617 → https(443) [ACK] Seq=1057 Ack=2932 Win=131328 Len=0
0.203952 mmx-ds.cdn.whatsapp... 192.168.0.104 TLSv1.2 99 Application Data
```

Wireshark · Packet 72 · 1 mb video sent.pcapng

> Frame 72: 587 bytes on wire (4696 bits), 587 bytes captured (4696 bits) on interface \Device\NPF_{7BCC5183-50CB-44E5-9473-213E7BB39417}, id 0
> Ethernet II, Src: HonHaiPr_ea:64:7b (74:40:bb:ea:64:7b), Dst: D-LinkIn_c2:2d:7e (bc:0f:9a:c2:2d:7e)
> Internet Protocol Version 4, Src: 192.168.0.104 (192.168.0.104), Dst: mmx-ds.cdn.whatsapp.net (31.13.79.53)
> Transmission Control Protocol, Src Port: 53314 (53314), Dst Port: https (443), Seq: 120, Ack: 1401, Len: 533
✓ Transport Layer Security
 ✓ TLSv1.2 Record Layer: Application Data Protocol: http-over-tls
 Content Type: Application Data (23)
 Version: TLS 1.2 (0x0303)
 Length: 528
 Encrypted Application Data: cbd0b7578d39ca8ebbc7a57dceba0a618d5280613a64e47...

- (d) **Keeping the whatsapp idle:** During this time host sends the server a TCP keep-alive request and receives and a ACK response to it. That is shown as in following scren shot.

```
7.595326 192.168.0.101 239.255.255.250 SSDP 167 M-SEARCH * HTTP/1.1
0.213481 auto.au.download.wi... 192.168.0.104 TCP 54 [TCP Keep-Alive] http(80) → 64441 [ACK] Seq=351 Ack=283 Win=30720 Len=0
0.000042 192.168.0.104 auto.au.download.wi... TCP 54 [TCP Keep-Alive ACK] 64441 → http(80) [ACK] Seq=283 Ack=352 Win=131584 Len=0
0.117391 192.168.0.101 239.255.255.250 SSDP 167 M-SEARCH * HTTP/1.1
```

- (e) **Closing the application:** While closing the application, a request is sent from my laptop to the server with FIN flag to close the session. In response, the server responds with a similar entry. Then ACK flag request is sent which successfully closes the session/connection.

0.000212	192.168.0.104	whatsapp-cdn-shv-02..	TCP	54	64372 → https(443) [FIN, ACK] Seq=629 Ack=8232 Win=516 Len=0
0.005393	whatsapp-cdn-shv-02..	192.168.0.104	TCP	54	https(443) → 64372 [ACK] Seq=8232 Ack=629 Win=544 Len=0
0.000000	whatsapp-cdn-shv-02..	192.168.0.104	TCP	54	https(443) → 64372 [FIN, ACK] Seq=8232 Ack=630 Win=544 Len=0
0.000053	192.168.0.104	whatsapp-cdn-shv-02..	TCP	54	64372 → https(443) [ACK] Seq=630 Ack=8233 Win=516 Len=0

Important:

The basic sequence of connections is same in this case as the above.

• Sending text message:

- (a) **TCP handshaking:** There is a three-way-handshake between the source, the destination. We have already seen In the previous question that port 53314 is being used in my laptop and port 443 is being used by whatsapp. As the SYN packet which helps in synchronizing the sequence numbers is sent, the handshaking process advents. Then destination responds to this packet by sending an ACK (acknowledgement) and another SYN packet which asks the source to synchronize the packet number with its sequence number. In the end, the source sends a final ACK packet and with this, we say that the handshaking process is successful.

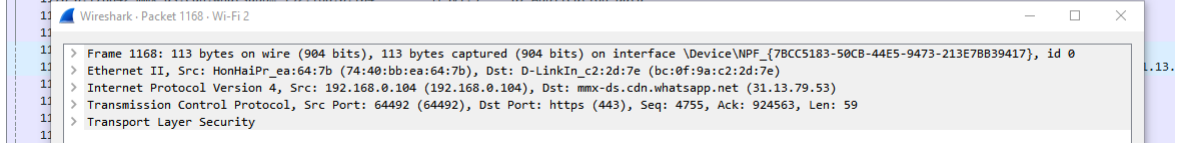
16	0.008276	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	66	64490 → https(443) [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
17	0.005041	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	66	https(443) → 64490 [SYN, ACK] Seq=0 Ack=1 Win=27840 Len=0 MSS=1392 SACK_PERM=1 WS=256
18	0.000059	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	64490 → https(443) [ACK] Seq=1 Ack=1 Win=132096 Len=0

- (b) **TLS handshaking:** TLS handshaking proceeds once the TCP handshaking is done. TLS handshaking makes sure that communication is secure and going properly. My laptop sends the Client Hello message and the server in respoce responds with Server Hello which can be seen by the screenshot attached.

19	0.003608	192.168.0.104	mmx-ds.cdn.whatsapp..	TLSv1.2	232	Client Hello
20	0.005318	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	54	https(443) → 64490 [ACK] Seq=1 Ack=179 Win=28928 Len=0
21	0.000826	mmx-ds.cdn.whatsapp..	192.168.0.104	TLSv1.2	1446	Server Hello

- (c) **Sending file:** TCP has a limit how many bytes can be sent in a single data packate. So usually the file sent is packed in different packets partially and sent. Following screen shot shows one of the such instances when the file is getting sent.

1166	0.000027	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	64492 → https(443) [ACK] Seq=4755 Ack=924563 Win=132096 Len=0
1167	1.006788	192.168.0.103	224.0.0.251	MDNS	103	Standard query 0x00df PTR _C32E753._sub._googlecast._tcp.local, "QM" question PTR _googlecast._tcp.local
1168	0.405386	192.168.0.104	mmx-ds.cdn.whatsapp..	TLSv1.3	113	Application Data
1169	0.045231	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	54	https(443) → 64492 [ACK] Seq=924563 Ack=4814 Win=32256 Len=0
1170	0.178842	mmx-ds.cdn.whatsapp..	192.168.0.104	TLSv1.3	87	Application Data



- (d) **Keeping the whatsapp idle:** During this time host sends the server a TCP keep-alive request and receives and a ACK response to it. That is shown as in following screen shot.

1343	0.765575	192.168.0.104	192.168.0.255	NBNS	92	Name query NB WORKGROUP<1c>
1344	0.595841	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	66	[TCP Keep-Alive] https(443) → 64491 [ACK] Seq=61939 Ack=4567 Win=39936 Len=0 TSval=3786402741 TSecr=12263214
1345	0.000046	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	[TCP Keep-Alive ACK] 64491 → https(443) [ACK] Seq=4567 Ack=61940 Win=131840 Len=0
1346	0.254061	192.168.0.104	192.168.0.1	DNS	74	Standard query 0xb0d3 A kms.iitg.ac.in

- (e) **Closing the application:** While closing the application, a request is sent from my laptop to the server with FIN flag to close the session. In response, the server responds with a similar entry. Then ACK flag request is sent which successfully closes the session/connection.

1362	0.000188	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	64492 → https(443) [FIN, ACK] Seq=5617 Ack=925874 Win=130816 Len=0
1363	0.002021	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	64491 → https(443) [FIN, ACK] Seq=4567 Ack=61940 Win=131840 Len=0
1364	0.002803	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	54	https(443) → 64492 [ACK] Seq=925874 Ack=5617 Win=35584 Len=0
1365	0.001094	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	54	https(443) → 64492 [FIN, ACK] Seq=925874 Ack=5618 Win=35584 Len=0
1366	0.000055	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	64492 → https(443) [ACK] Seq=5618 Ack=925875 Win=130816 Len=0
1367	0.001885	mmx-ds.cdn.whatsapp..	192.168.0.104	TCP	54	https(443) → 64491 [FIN, ACK] Seq=61940 Ack=4568 Win=39936 Len=0
1368	0.000074	192.168.0.104	mmx-ds.cdn.whatsapp..	TCP	54	64491 → https(443) [ACK] Seq=4568 Ack=61941 Win=131840 Len=0

Question 4

(a) • At 4:30 PM:

Field	value
Throughput	15k B/sec
RTT	0.016429 sec
Avg Packate Size	621 B
Dropped Number of packates	0
Number of TCP packates	75

Field	value
Throughput	42 B/sec
Avg Packate Size	157 B
Dropped Number of packates	0
Number of UDP packates	4

(b) • At 6:24 PM:

Field	value
Throughput	26k B/sec
RTT	0.010426 sec
Avg Packate Size	840 B
Dropped Number of packates	0
Number of TCP packates	1487

Field	value
Throughput	143 B/sec
Avg Packate Size	151 B
Dropped Number of packates	0
Number of UDP packates	35

(c) • At 9:39 PM:

Field	value
Throughput	21k B/sec
RTT	0.002377 sec
Avg Packate Size	829 B
Dropped Number of packates	0
Number of TCP packates	1301

Field	value
Throughput	140 B/sec
Avg Packate Size	124 B
Dropped Number of packates	0
Number of UDP packates	68

Question 5

Even when I tried at different time of the day and at the same time too, I didn't found any change in the destination IP address.