CS349 ASSIGNMENT 2

Network Protocol Analysis Using Wireshark

Rashi Singh
170101052

Q1 : PROTOCOLS USED IN DIFF LAYERS

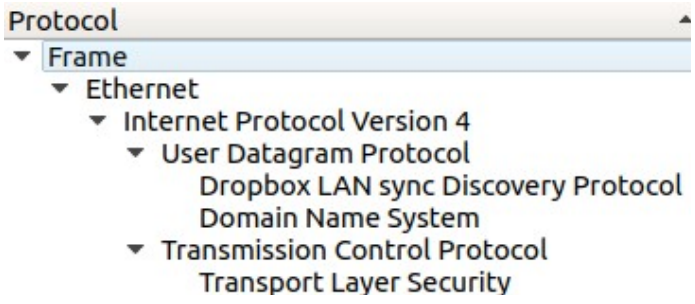

Figure 1: Protocol Hierarchy used by DropBox

Network Layer :

IPv4 : Internet Protocol Version 4 :

IPv4 is a network-layer protocol used in **packet-switched** layer networks, such as Ethernet. It provides a logical connection between network devices by providing **identification** for each device and **routing** data among them over the underlying network. IP uses best effort delivery, i.e. it does not guarantee that packets would be delivered to the destined host, but it will do its best to reach the destination. Internet Protocol version 4 uses **32-bit logical** address.

IPv4 - Packet Structure :

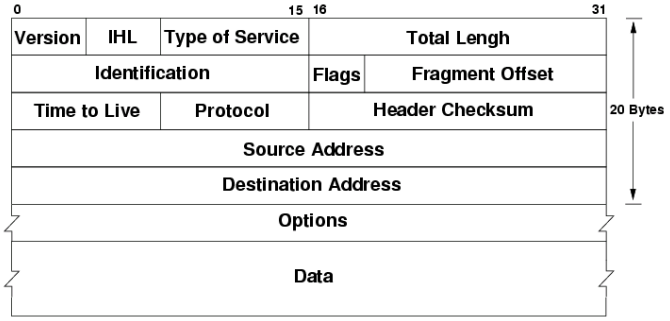

Figure 2: IPv4 Header Structure

```
Internet Protocol Version 4, Src: 162.125.19.131, Dst: 10.16.0.46
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 52
    Identification: 0xf753 (63315)
  ▸ Flags: 0x0000
    ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 62
    Protocol: TCP (6)
    Header checksum: 0xc532 [validation disabled]
    [Header checksum status: Unverified]
    Source: 162.125.19.131
    Destination: 10.16.0.46
```

Figure 3: WireShark IPv4 layer packet details

VERSION	- (4 bit) Version of the Internet Protocol used
HEADER LENGTH	- (4 bit) Length of the IP header in 32 bit increments. **Min length of IP header is 20 bytes, so with 32 bit increments, min value of IHL is 5. **Max value of IHL is 15 so with 32 bit increments, max length of IP header is 60 bytes
TYPE OF SERVICE	- Provides an indication of the abstract parameters of the quality of service desired.
TOTAL LENGTH	- (16 bit) Length of the datagram (in bytes), including internet header and data.
IDENTIFICATION	- An identifying value assigned by the sender to help assemble the fragments of a datagram
Flags	- (3 bit) Various Control Flags
FRAGMENT OFFSET	- (13 bit) Indicating where in the datagram this fragment belongs.
TIME TO LIVE	- (8 bit) The max time the datagram is allowed to remain in the internet system.
PROTOCOL	- Next level protocol used in the data portion of the internet datagram.
HEADER CHECKSUM	- A checksum on the header only.
SOURCE/DEST ADDRESS	- (32 bit) Addr of the source/destination respectively

Transport Layer :

TCP : Transport Control Protocol :

TCP is a **connection oriented** protocol which offers **end-to-end packet delivery**. TCP ensures reliability by sequencing bytes with a forwarding **acknowledgement** number that indicates to the destination, the next byte the source expect to receive.It retransmits the bytes not acknowledged with in specified time period.
Dropbox mainly used TCP to send application data mainly due its **reliability** feature

TCP - Packet Structure :

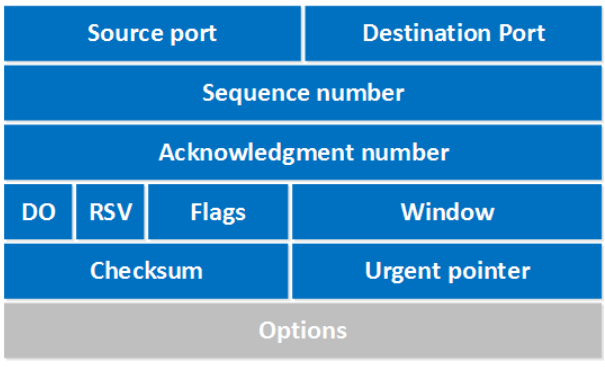

Figure 4: TCP header structure

```
Transmission Control Protocol, Src Port: 443, Dst Port: 38288, Seq: 102, Ack: 95, Len: 0
    Source Port: 443
    Destination Port: 38288
    [Stream index: 18]
    [TCP Segment Len: 0]
    Sequence number: 102    (relative sequence number)
    [Next sequence number: 102    (relative sequence number)]
    Acknowledgment number: 95    (relative ack number)
    1000 .... = Header Length: 32 bytes (8)
  ▸ Flags: 0x010 (ACK)
    Window size value: 513
    [Calculated window size: 513]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0x1c9d [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  ▸ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
  ▾ [SEQ/ACK analysis]
      [This is an ACK to the segment in frame: 75]
      [The RTT to ACK the segment was: 0.000225906 seconds]
```

Figure 5: WireShark TCP packet details

SOURCE/DEST PORT : (16 bit) Fields to identify the end points of the connection.

SEQUENCE # : (32 bit) Number assigned to the first byte of data in the current message
ACKNOWLEDGEMENT # : (32 bit) Value of the next sequence # that the sender of the segment is expecting to receive
DATA OFFSET : Specifies how many 32-bit words are contained in the TCP header.
RESERVED : (6 bit) Must be zero. This is for future use.
FLAGS : (6 bit) URG, ACK, PSH, RST, SYN, FIN
WINDOW : (16 bit) Specifies the size of the sender's receive window
CHECKSUM : (16 bit) Indicates whether the header was damaged in transit.
URGENT : pointer (16 bit) Points to the first urgent data byte in the packet.
OPTIONS : (variable length) Specifies various TCP options.
DATA : (variable length) Contains upper-layer information.

UDP : User Datagram Protocol :

```
▾ User Datagram Protocol, Src Port: 17500, Dst Port: 17500
    Source Port: 17500
    Destination Port: 17500
    Length: 201
    Checksum: 0xfbcb [unverified]
    [Checksum Status: Unverified]
    [Stream index: 3]
  ▸ [Timestamps]
▸ Dropbox LAN sync Discovery Protocol
```

Figure 6: WireShark UDP packet details

- UDP is **connectionless** and **unreliable** protocol. It doesn't require making a connection with the host to exchange data. Since UDP is unreliable protocol, there is no mechanism for ensuring that data sent is received.
- UDP is used by the application that typically transmit **small** amount of **data** at one time. UDP datagram consists of the following : Src/Dest Port, Length,Checksum (same as TCP header)

DB-LSP-DISC : DropBox LAN Sync Discovery Protocol :

```
▾ Dropbox LAN sync Discovery Protocol
  ▾ JavaScript Object Notation
    ▾ Object
      ▸ Member Key: version
      ▸ Member Key: port
      ▸ Member Key: host_int
      ▸ Member Key: displayname
      ▸ Member Key: namespaces
```

Figure 7: WireShark DB-LSP-DISC packet details

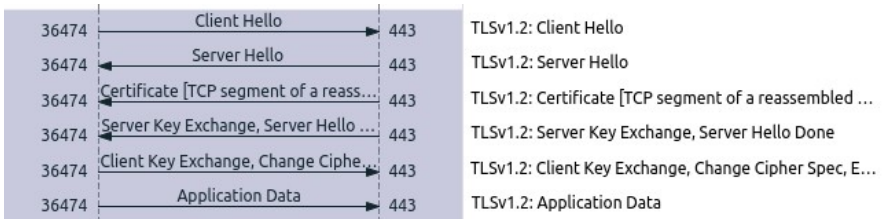

Figure 8: TLS packet exchange

B/w Application, Transport Layer :

TLSv1.2 : Transport Layer Security

It is a cryptographic protocol, developed from the generalized version of SSL(Secure Socket Layer)(now deprecated). It provides 3 essential services to the applications running above it:

Verification of validity of identity : AUTHENTICATION a)
Detection of msg tampering, forgery : DATA INTEGRITY b)
A mechanism to obfuscate what is sent from one host to another : ENCRYPTION c)

***TLS requires a reliable transport. Hence, it uses TCP*

PACKETS for TLSv1.2

The common parameters present in diff. kinds of TLS packets are :

a) VERSION : 16 byte version
b) LENGTH : 16 byte record length, formatted in network order
c) CONTENT TYPE : This signifies the types of TLS packets that are recognized. As in Figure 8: TLS packet exchange, some of the common types are :
   1) handshake
   2) application_data
   3) change_cipher_spec
   4) alert, heartbeat etc...

```
▾ Transport Layer Security
  ▾ TLSv1.2 Record Layer: Application Data Protocol: http-over-tls
      Content Type: Application Data (23)
      Version: TLS 1.2 (0x0303)
      Length: 324
```

Figure 9: Common parameters in TLS packets

```
▾ Transport Layer Security
  ▾ TLSv1.2 Record Layer: Handshake Protocol: Client Key Exchange
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 37
    ▾ Handshake Protocol: Client Key Exchange
        Handshake Type: Client Key Exchange (16)
        Length: 33
      ▾ EC Diffie-Hellman Client Params
          Pubkey Length: 32
          Pubkey: ace9eac27150502e14355b3748e9125b2abbe167ca83b778…
  ▾ TLSv1.2 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
      Content Type: Change Cipher Spec (20)
      Version: TLS 1.2 (0x0303)
      Length: 1
      Change Cipher Spec Message
  ▾ TLSv1.2 Record Layer: Handshake Protocol: Encrypted Handshake Message
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 40
      Handshake Protocol: Encrypted Handshake Message
```