Transport Layer (L4)

Transport Layer acts like an intermediary between the application layer and the IP layer. For example, it needs to forward the data in packets to the correct application. This process is known as De-multiplexing. To do this, each application is assigned a port number.

* User Datagram Protocol (UDP) does exactly this, nothing more.

However, we might have a few problems if a large file transfer is being done. The data is divided into segments of size ~1500 Bytes.

Prob # 1) Segments may be lost

#2) Due to change in path, the segments arrive out of order

* Transmission Control Protocol (TCP)

TCP solves the following problems :-

1. Reliable transfer by retransmitting lost segments
2. Re-order segments
3. Congestion Control
4. Flow Control

TCP also allows bi-directional transfer of data.

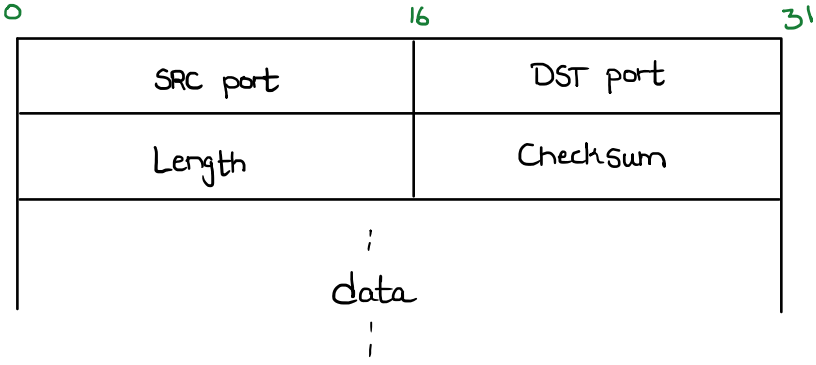
* L4 routers use "drop-tail" mechanism, that is, if the buffer is full then newly received packets are dropped. We say that the router is congested when this occurs. TCP solves this by reducing the input rate, freeing up the queue.

* Flow Control is just congestion control for the destination. As bi-directional communication is possible, DST tells SRC to reduce the data rate.

* Header Structure

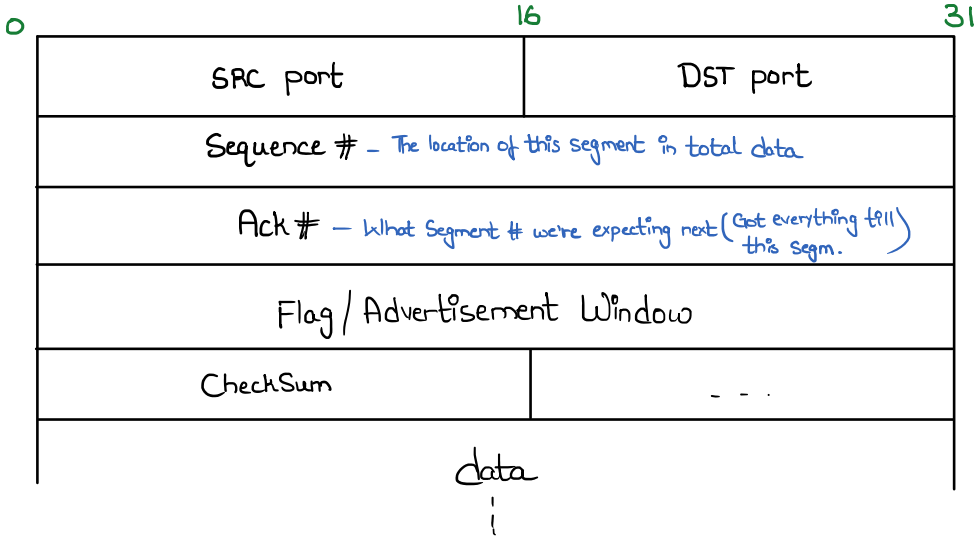
1. UDP header

8 Bytes long



2. TCP Header

20 Bytes long!



- Flags :- SYN / FIN / RESET / PUSH / URG / ACK

SYN: Start of connection
FIN: End of connection
RESET: Reset connection (Prone to attacks)
ACK: ACK is present in header

Represented by either a 0 or 1.

For example, a RESET flag is rep. as 001000.

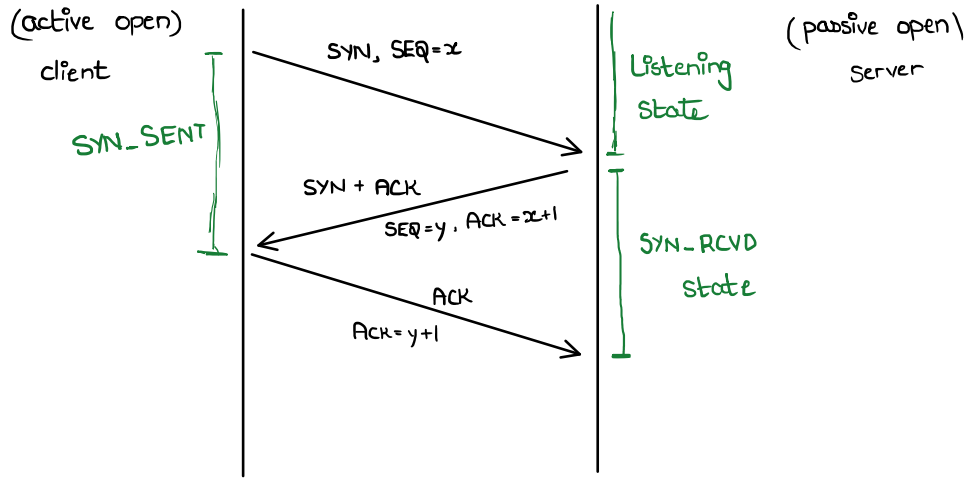
* The Protocol Field at IP layer tells us whether TCP/UDP is used.

6 - TCP 17 - UDP

As stated earlier, TCP allows for bi directional communication between the communicating nodes. It also allows for ACK and SEQ to be sent at the same time. We shall now look at how connection is established by the TCP protocol.

* Three - Way - Handshake

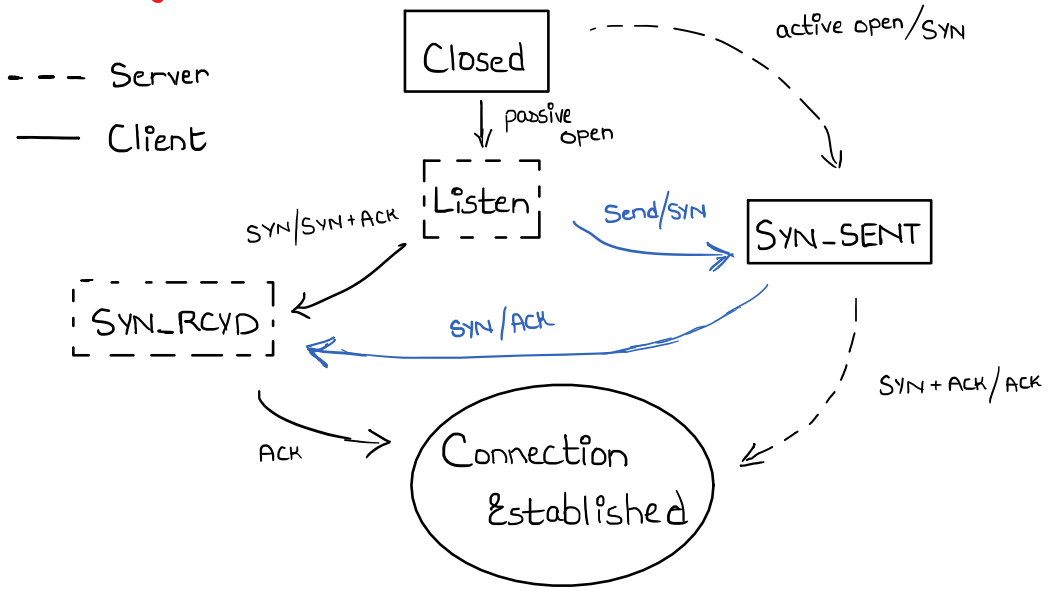
This is how connection between a server (passive open) and a client (active open) is established. Three packets are shared, which is why its called as the three-way-handshake.



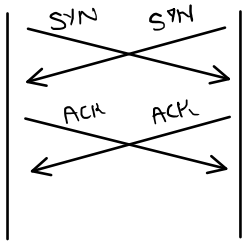
An ACK # of x means that everything from start to (x-1) has been received, and it is expecting the x^{th} sequence. Similarly, we take that SYN, FIN packets convey 1 Byte although they donot have any data. Significance?

The values x,y are picked randomly for indicating the start of a sequence. This decreases chances of a packet from a previous data transfer arriving late and becoming a part of current data exchange.

* State Diagram



The blue lines depict a case where the server actively initiates the connection by sending a SYN after getting information to do so at higher layer.

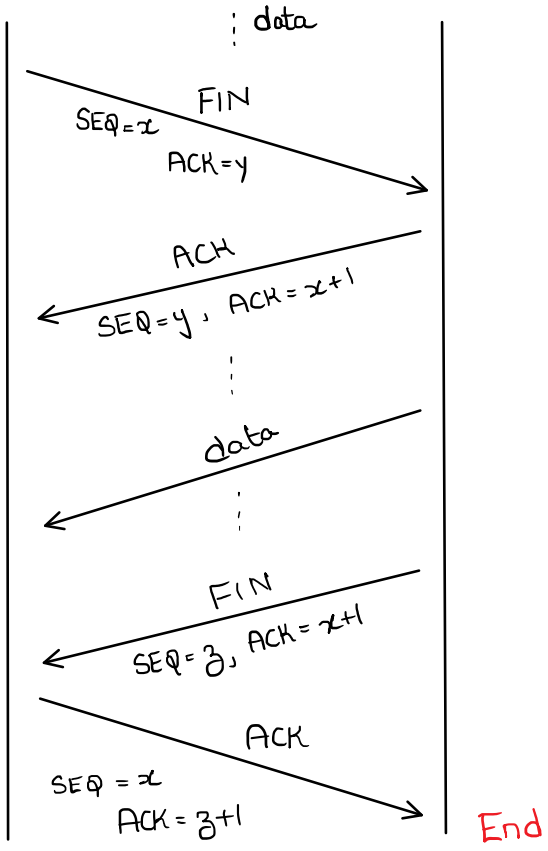


* Connection Termination

The server stores state information during the connection, and closing the connection frees up this memory. We discuss the ways in which this is acheived.

1 Half Close

This is done when one of the nodes has no more data left to transfer, but is still receiving data. That is, one end stops sending but not receiving.



2. Three Way Handshake

Similar to how a connection is established.