**TCP HEADER**

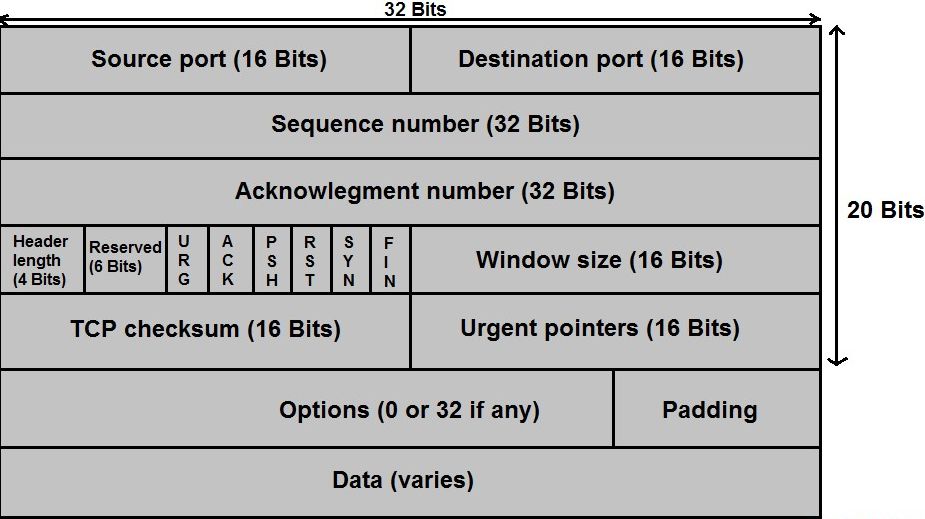
|  |  |
| --- | --- |
| Prepared by: | K Prashanth |
| Reviewed by: | Abhinab |

**Features:**

* TCP is reliable protocol.
* TCP is full duplex it exchanges bi-directional data.
* TCP is connection oriented.
* TCP provides error-checking and recovery mechanism.
* TCP provides end-to-end communication.
* TCP provides flow control, maintains the flow of the data and quality of service.
* Order of data is maintained based on sequence number.

**HEADER FORMAT:**

* TCP header size is (20-60) bytes



**Source port:**

* Source port is a 16-bit field.
* Port of application is sending data.
* Port number is provided by transport layer.

**Destination port:**

* Destination port is a 16-bit field.
* Port of application is receiving data.
* Port number helps each segment to reach correct application.

Note:

* TCP is identified by combination of port numbers & IP addresses of source and destination.
* IP address indicates which system is communicating.
* Port number indicates which end sockets are communicating.

**Sequence number:**

* 32bit field.
* TCP assigns a unique sequence number to each byte of data contained in the TCP segment.
* It contains the sequence of the 1st data byte.

**Acknowledgement number:**

* 32-bit field.
* It contains sequence number of the data byte that receiver expects to receive next from the sender.
* The sequence number of the last received data byte incremental by 1.

**Header length:**

* 4-bit field.
* It contains the length of TCP header.
* It helps to know where the actual data begins.
* Range of header is [20-60] bytes.
* We use scaling factor “4”.

**Reserved bits:**

* 6-bit field.
* Bits are not used, it is used for future purpose.
* It is always set to be zero.

**FLAGS**:

* **URG flag:**
  + It is used to treat certain data on an urgent, when urgent bit is set to 1.
  + Receiver forwards urgent data to the receiving application on the separate channel.
* **ACK flag:**
  + It indicates that acknowledgement number contained in TCP header is valid.
  + If the ACK is set to 0, then the data packet doesn’t contains acknowledgement.
* **PUSH flag:**
  + Push bit is used to push the entire buffer immediately to the receiving application.
  + Push bit is set to 0, then it requests the receiving device to push the data to the receiving application without buffering it.
  + When push bit is set 1, all the segments in the buffer is immediately pushed to the receiving application.
  + No wait is done for filling the entire buffer.
  + These makes the entire buffer to free up immediately
* **RST flag:**
  + It is used to reset TCP connection.
  + Reset bit is set to 0, the receiver to terminate the connection immediately.
  + Its causes both the sides to release the connection and all its resources abnormally.
  + The transfer of data ceases in both the directions.
* **SYN flag:**
* It is used to synchronize the sequence numbers.
* When synchronize bit is set to 1.
* It indicates the receiver that the sequence number contained in the TCP header is the initial sequence.
* Request segment sent for connection establishment during 3-way handshake contains SYN bit set to 1.
* **FIN flag:**
  + FIN bit is used to terminate the TCP connection, when finish bit is set to 1.
* It indicates the receiver that the sender wants to terminate the connection.
* Terminates TCP connection.

**Window size:**

* 16-bit field.
* It contains the size of the data that receiver can accepts.
* It is used for flow control.

**Checksum:**

* It is used for error control.
* Sender adds checksum field to check sum field before sending data.
* Receiver rejects the data that fails the checksum, if checksum is matched data is error free.

**Urgent pointer:**

* Indicates how much data in the correct segment counting from the 1st data byte is urgent.

**Options:**

* These are used for Timestamp, MSS (Maximum segment size) and padding.

Timestamp:

* It provides the age of the TCP segment.
* If the TCP segment is received by the receiver or not is checked by timestamp.

MSS:

* It indicates maximum segment size TCP can receive.

Padding:

* Addition of dummy data to fill up unused space in the transmission unit and make it confirm to the standard size is called padding.

**3 – Way handshake**

* TCP 3-way handshake, also known as a 3-way handshake, is **a protocol for establishing a connection between a server and a client in a TCP/IP network**. A client must initiate a conversation by using the TCP handshake to request a communication session with the server.
* **Step 1 (SYN):**In the first step, the client wants to establish a connection with a server, so it sends a segment with SYN(Synchronize Sequence Number) which informs the server that the client is likely to start communication and with what sequence number it starts segments.
* **Step 2 (SYN + ACK):**Server responds to the client request with SYN-ACK signal bits set. Acknowledgement(ACK) signifies the response of the segment it received and SYN signifies with what sequence number it is likely to start the segments.
* **Step 3 (ACK):**In the final part client acknowledges the response of the server and they both establish a reliable connection with which they will start the actual data transfer.

