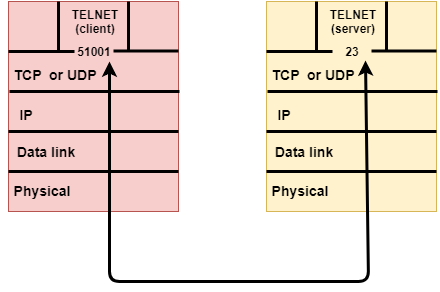
Transport Layer protocols

* The transport layer is represented by two protocols: TCP and UDP.
* The IP protocol in the network layer delivers a datagram from a source host to the destination host.
* Nowadays, the operating system supports multiuser and multiprocessing environments, an executing program is called a process. When a host sends a message to other host means that source process is sending a process to a destination process. The transport layer protocols define some connections to individual ports known as protocol ports.
* An IP protocol is a host-to-host protocol used to deliver a packet from source host to the destination host while transport layer protocols are port-to-port protocols that work on the top of the IP protocols to deliver the packet from the originating port to the IP services, and from IP services to the destination port.
* Each port is defined by a positive integer address, and it is of 16 bits.

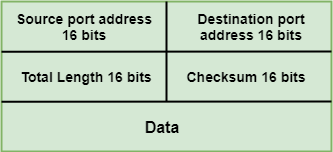


UDP

* UDP stands for **User Datagram Protocol**.
* UDP is a simple protocol and it provides nonsequenced transport functionality.
* UDP is a connectionless protocol.
* This type of protocol is used when reliability and security are less important than speed and size.
* UDP is an end-to-end transport level protocol that adds transport-level addresses, checksum error control, and length information to the data from the upper layer.
* The packet produced by the UDP protocol is known as a user datagram.

User Datagram Format

The user datagram has a 16-byte header which is shown below:



**Where,**

* **Source port address:** It defines the address of the application process that has delivered a message. The source port address is of 16 bits address.
* **Destination port address:** It defines the address of the application process that will receive the message. The destination port address is of a 16-bit address.
* **Total length:** It defines the total length of the user datagram in bytes. It is a 16-bit field.
* **Checksum:** The checksum is a 16-bit field which is used in error detection.

Disadvantages of UDP protocol

* UDP provides basic functions needed for the end-to-end delivery of a transmission.
* It does not provide any sequencing or reordering functions and does not specify the damaged packet when reporting an error.
* UDP can discover that an error has occurred, but it does not specify which packet has been lost as it does not contain an ID or sequencing number of a particular data segment.

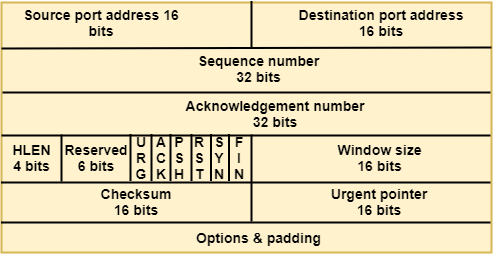
TCP

* TCP stands for Transmission Control Protocol.
* It provides full transport layer services to applications.
* It is a connection-oriented protocol means the connection established between both the ends of the transmission. For creating the connection, TCP generates a virtual circuit between sender and receiver for the duration of a transmission.

Features Of TCP protocol

* **Stream data transfer:** TCP protocol transfers the data in the form of contiguous stream of bytes. TCP group the bytes in the form of TCP segments and then passed it to the IP layer for transmission to the destination. TCP itself segments the data and forward to the IP.
* **Reliability:** TCP assigns a sequence number to each byte transmitted and expects a positive acknowledgement from the receiving TCP. If ACK is not received within a timeout interval, then the data is retransmitted to the destination.  
  The receiving TCP uses the sequence number to reassemble the segments if they arrive out of order or to eliminate the duplicate segments.
* **Flow Control:** When receiving TCP sends an acknowledgement back to the sender indicating the number the bytes it can receive without overflowing its internal buffer. The number of bytes is sent in ACK in the form of the highest sequence number that it can receive without any problem. This mechanism is also referred to as a window mechanism.
* **Multiplexing:** Multiplexing is a process of accepting the data from different applications and forwarding to the different applications on different computers. At the receiving end, the data is forwarded to the correct application. This process is known as demultiplexing. TCP transmits the packet to the correct application by using the logical channels known as ports.
* **Logical Connections:** The combination of sockets, sequence numbers, and window sizes, is called a logical connection. Each connection is identified by the pair of sockets used by sending and receiving processes.
* **Full Duplex:** TCP provides Full Duplex service, i.e., the data flow in both the directions at the same time. To achieve Full Duplex service, each TCP should have sending and receiving buffers so that the segments can flow in both the directions. TCP is a connection-oriented protocol. Suppose the process A wants to send and receive the data from process B. The following steps occur:
  + Establish a connection between two TCPs.
  + Data is exchanged in both the directions.
  + The Connection is terminated.

TCP Segment Format



**Where,**

* **Source port address:** It is used to define the address of the application program in a source computer. It is a 16-bit field.
* **Destination port address:** It is used to define the address of the application program in a destination computer. It is a 16-bit field.
* **Sequence number:** A stream of data is divided into two or more TCP segments. The 32-bit sequence number field represents the position of the data in an original data stream.
* **Acknowledgement number:** A 32-field acknowledgement number acknowledge the data from other communicating devices. If ACK field is set to 1, then it specifies the sequence number that the receiver is expecting to receive.
* **Header Length (HLEN):** It specifies the size of the TCP header in 32-bit words. The minimum size of the header is 5 words, and the maximum size of the header is 15 words. Therefore, the maximum size of the TCP header is 60 bytes, and the minimum size of the TCP header is 20 bytes.
* **Reserved:** It is a six-bit field which is reserved for future use.
* **Control bits:** Each bit of a control field functions individually and independently. A control bit defines the use of a segment or serves as a validity check for other fields.

There are total six types of flags in control field:

* **URG:** The URG field indicates that the data in a segment is urgent.
* **ACK:** When ACK field is set, then it validates the acknowledgement number.
* **PSH:** The PSH field is used to inform the sender that higher throughput is needed so if possible, data must be pushed with higher throughput.
* **RST:** The reset bit is used to reset the TCP connection when there is any confusion occurs in the sequence numbers.
* **SYN:** The SYN field is used to synchronize the sequence numbers in three types of segments: connection request, connection confirmation ( with the ACK bit set ), and confirmation acknowledgement.
* **FIN:** The FIN field is used to inform the receiving TCP module that the sender has finished sending data. It is used in connection termination in three types of segments: termination request, termination confirmation, and acknowledgement of termination confirmation.
  + **Window Size:** The window is a 16-bit field that defines the size of the window.
  + **Checksum:** The checksum is a 16-bit field used in error detection.
  + **Urgent pointer:** If URG flag is set to 1, then this 16-bit field is an offset from the sequence number indicating that it is a last urgent data byte.
  + **Options and padding:** It defines the optional fields that convey the additional information to the receiver.

