Hello everyone! In this video, you will learn about TCP protocol, one of the mandatory protocol used by various applications on Internet.

The transmission control protocol(TCP) is a transport layer protocol, which allows process-to-process communication between two machines such as communication between a browser on the user's machine and the Facebook server. TCP offers many other services to applications, like delivering data in the form of a stream of bytes, providing two-way communication using a connection, and reliability.

The communicating processes on machines are identified by the port numbers, sixteen-bit binary numbers, represented by a decimal number. These port numbers may be well-known, registered, or ephemeral depending on the applications using them. The well-known ports are assigned to popular protocols such as HTTP, FTP, Telnet, etc. while ephemeral port numbers are assigned to the user processes. Well known port number range from 0 to 1023 and ephemeral ports range from **49152 to 65 535. Registered port numbers range from 1024 to 49151 and are reserved for** specific services upon application by a requesting entity such as popular online goals.

TCP has a rich set of features. It numbers bytes in communication and provides flow control between a pair of fast and slow processes, manages errors using a checksum method, and provides congestion control for the smooth delivery of the data.

In TCP the data is sent in the form of segments. Segments carry bytes of data and are numbered with the number of the first data byte contained in the segment.

The TCP segment is a variable length data unit with a header of size 20 to 60 bytes. The TCP header contains a source port address and destination port address representing the sender process and receiver process respectively. The sequence number as mentioned before, represents the number of first byte in the segment. And the acknowledgement represents the number of next byte expected from sender. HLEN filed tells about the size of the header. There are 6 flags in TCP from URG to FIN for connection management and making data transfer suitable for most of the applications. Windows size plays an important role in flow control. Checksum is used for error control and urgent pointer mentions the data byte number to be processed out of order. TCP has many options for the optional functionalities and padding is used in checksum calculation.

Some more details about TCP flags or control fields. The flags SYN, FIN and RST are used in connection management while ACK flag if set, makes the Acknowledgement number field valid. URG flag allows the data to be processed on receiver, out of order against the policies of TCP and makes Urgent pointer valid. PSH flag allows the data to be transmitted in small sizes such as in chat applications.

TCP is a connection-oriented protocol. A connection-oriented transport protocol establishes a virtual path

between the source and destination before data transfer. All of the segments belonging to a message are then sent over this virtual path. The connection is then terminated after the data transfer. A connection-oriented transmission, therefore, requires three phases.
Connection Establishment.
Data transfer.
Connection termination.

Connection setup or establishment in TCP is done using a three-way handshake mechanism. The client transport layer will send a SYN Segment to initiate the connection. The server will then send ACK+ SYN, as a response. Clint will finally establish the connection by sending an ACK segment.

After connection establishment, data is transferred in the form of segments. In this figure, we can see two segments (Sequence No 8001 and 9001) moving from Client to server carrying 2000 bytes of data. Server acknowledges this transfer and sends its own data. Client acknowledges the received data before the connection is terminated.

TCP uses the similar three-way handshake mechanism as of connection setup for connection termination, but with the FIN packets, as shown in the figure.

A half close connection is also used in TCP which can be used in a situation, when one party does not want to send data, but other side is still sending the data. In this situation one side may close the connection by sending FIN packet. Other side will be responding with an acknowledgement of FIN but will not send its own FIN packet, till the transmission continues. After completing the data transfer other side will also send FIN and first side will respond with ACK for closing the connection completely.

Thank You.