

# Network Protocols

Protocol  $\rightarrow$  agreed set of rules for interaction between parties

## IP Packet (aka) network packet

When a machine sends data to another machine it will be sent in the form of IP packets. This is the fundamental unit of data being sent from one machine to another

IP Packet  $\Rightarrow 2^{16}$  bytes (max size)



## IP Header: (20-60 bytes)

The IP header consists of the

- ① Source IP address
- ② Destination IP address
- ③ IP version  $\Rightarrow$  IPv4 (or) IPv6

Since the IP packets size is  $2^{16}$  bytes = 0.065 mb  
65,000 bytes

If we are sending a file (say 5mb) then it will not fit in one IP packet. So the file will be sent in multiple IP packets.

## Internet Protocol

When the internet protocol is used while sending multiple IP packets there is no guarantee that the packets will be received i.e. some of the packets might get lost and the packets will also not be ordered.

## TCP

TCP  $\Rightarrow$  Transmission Control Protocol. This is built on top of the Internet Protocol. Here the packets are sent in an ordered manner and if some packets get corrupted when they are getting sent TCP will inform so that the packets can be resent.



↓  
contains information about the order  
of the packets

When the browser wants to communicate with a server, the browser will first create a TCP connection with the server. This happens through a handshake. A TCP handshake happens in 3 stages

Stage-1: The browser sends a SYNchronize packet (SYN) to the server (Hey! can we connect)

Stage-2: The server receives the SYN packet and (SYN-ACK) it sends back SYN-ACK packet (okay we can connect)

Stage-3: The browser receives the SYN-ACK and (ACK) sends back ACK (we are now connected)

The server receives the ACK and the TCP socket connection is established

SYN - Synchronize

ACK - Acknowledgement

During this process if one of the machines doesn't respond in a given period then the connection will be timed out.

HTTP: → application layer of TCP/IP

HTTP stands for Hypertext Transfer Protocol. This was built on top of the TCP. HTTP provides higher level <sup>showing only the necessary details</sup> abstraction above TCP. This abstraction is the request-response paradigm. This req-res paradigm allows developers to forget about the IP packets, TCP and just use requests and responses.

In short HTTP provides a framework that helps developers to easily send data between servers and browsers.

This was generally used for transferring hypertext documents such as HTML between the browsers and the servers.

But the data in HTTP is transferred via unencrypted connections.

## HTTPS

HTTPS stands for Hyper Text Transfer Protocol Secure. HTTPS uses SSL (or) TLS to encrypt all the communication between the client and the server. Generally used for banking activities / online shopping.

## TLS (aka) SSL

TLS  $\Rightarrow$  Transport Layer Security

SSL  $\Rightarrow$  Secure Socket Layer.

When a server and a client communicates

using TLS it ensures no third party can secretly listen

eavesdrop or tamper on email, web browsing, messaging etc.

TLS requires the servers and the browsers to provide a valid digital certificate to confirm its identity

## Digital Certificate

A digital certificate is a file that

has a cryptographic key. This key has  
encrypted information about the organization  
Such as the organization's name, location.