

Network Protocols & Standards

OSI Model is not a networking standard.

OSI is a framework into which the various networking standards can be addressed to by which network standard.

- Physical - governs layout of cables & devices such as repeaters & hubs.
- data link - provides mac addresses to uniquely identify network nodes & means of data to sent over physical layer in the form of packets. bridges/switches layer 2 devices.
- network - routing of data across network segments.
- transport - reliable delivery of packets.
- session - sessions between network applications.
- presentation - ^{converts data so that} systems that can use different data formats can exchange information.
- application - allows applications to request network services.

Data link layer - defines size of packet.

- error detection/correction.
- each device on network has an address (media access control address).
(includes a mac address)
local area conn
bluetooth → different mac address.
- sending packets safely over physical media without interference from other nodes attempting to send packets at same time. (CSMA/CD) → listens before sending packet & after sending packet.

Network layers

(27)

logical addressing & routing

network device → physical address → MAC address
other addressing schemes

network layer → task of routing messages from one computer to other.
protocols - IP / TCP

every network device has a physical address → MAC address (fixed)
IP address is assigned to every device connected to internet.
network layer protocol translates logical address to mac address.
because the network layer protocol must use data link layer protocol to actually send packets to device.

Address Resolution
Protocol - ARP

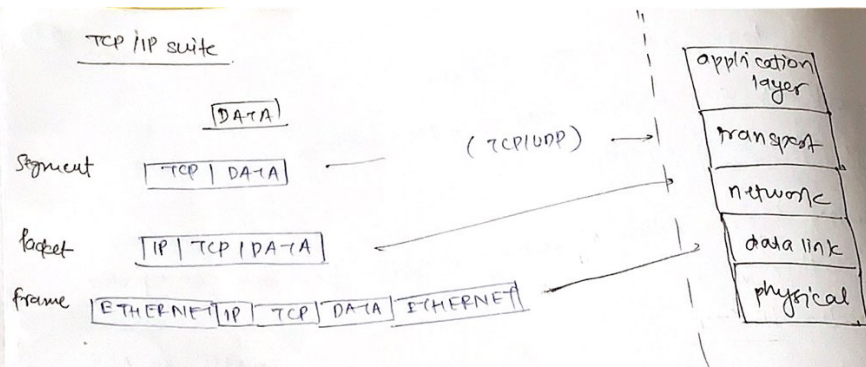
transport layer

- transport layer ensures packets are transported reliably & without errors.
 - connection to the other computer is established, connection is acknowledged.
- break into packets small,
- sometimes speed, reliability > reliability → UDP. a connectionless protocol. It simply sends the packet.

the session layer

~~session~~ sessions

Applications - all programs that are connected to ~~net~~ internet networks, use some protocols to talk ~~with~~ to internet.
DNS, FTP, SMTP



IP

- network layer protocol, delivers packet to network devices.
- uses ARP (address resolution protocol) to convert logical address to MAC address.
- IP contains a network path and a host path.
- Based on the network, routing.

TCP

- transport layer protocol

- connection oriented.

- packet resent if lost.

- used for 1-1 communication.

- well known application layer protocols use TCP

user running web browser uses http to send a request via TCP to the web server.

when web server receives a request, it uses http to

send requested web page back to the browser.

(also FTP, Telnet - terminal emulation)

UDP - connectionless.

- eg: DNS

Web Socket

① http.

(http → unidirectional)
client → request
server → response

So 5 different
game for REST connect

② websocket.

bidirectional comm

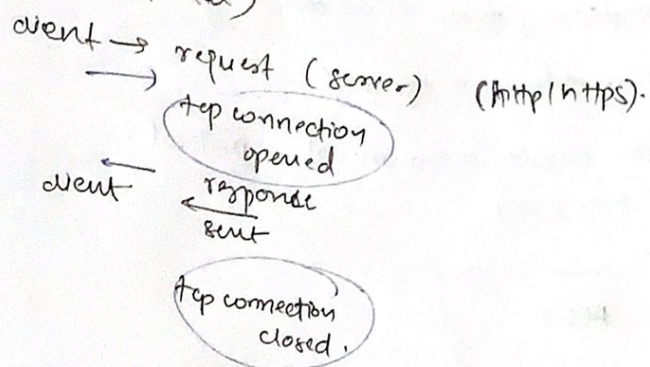
(ws: , wss:)
in the first time

client

Web Socket

① http.

(http → unidirectional)



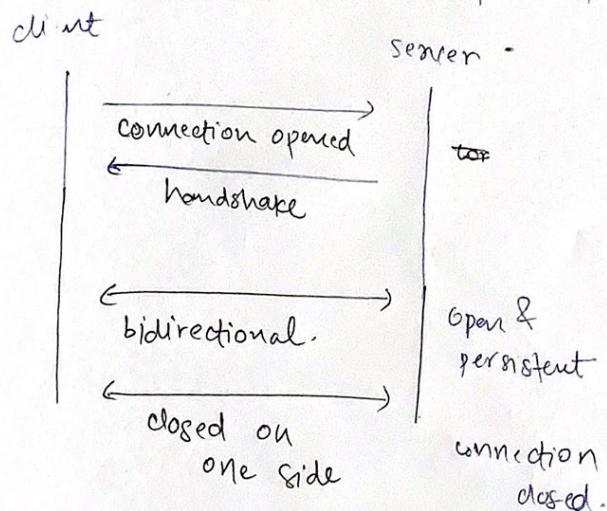
So 5 different requests, 5 different connections opened.
same for REST connection. new request, new connection created.

② websocket.

bidirectional communication.

(ws:, wss:) ~~protocol~~ that how websockets are created.

on the first time the connection is opened, the connection stays open.



(example: coinbase ui uses websocket to update various cryptocurrency values
: socket.io)

so the use of websockets should not be done when real-time updates are not required.

rest \rightarrow stateless query. no context maintained, same as http.

in rest response, there is no choice given ~~to~~ to retrieve specific param

this is an option in graphql.

grpc \rightarrow inter service communication protocol.
(written over 2.0)

API's