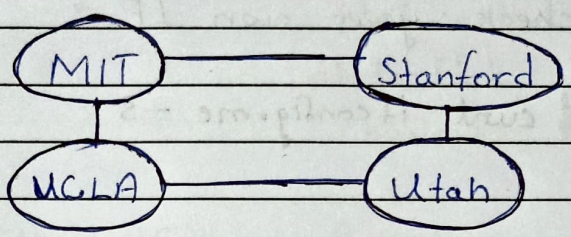
18/1/22

Computer Networking

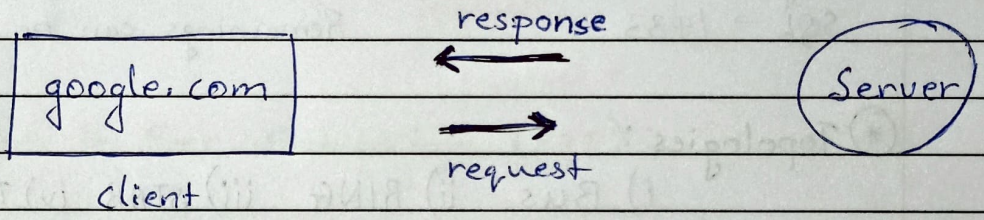
(*) How did it started ?



ARPANET
Using TCP/IP
(Transmission Control Protocol)

The regulations are controlled by "Internet Society"

(*) Brief Example :



(*) Protocols : Rules defined by Internet society

1) TCP : (Transmission Control Protocol)
It will ensure that the data will reach it's destination and it will not be corrupted in the way.

2) UDP : (User Datagram Protocol)
All data may not reach the recepient
Eg. - Video Conferencing.

3) HTTP : (Hyper Text Transfer Protocol)
Used by Web Servers - Client requests.

__/__/____

(*) IP Address : X . X . X . X

[0-255]

Command to check your own IP →

$ curl ifconfig.me -s

(*) Ports :

$2^{16} \approx 65,000$

HTTP → 80

MongoDB → 27017

SQL → 1433

0 - 1023 → Reserved Ports

1024 - 49152 → Applications

Remainings can be used.

(*) Topologies :

i) Bus ii) RING iii) STAR iv) TREE v) MESH

Structure of the Network

(*) OSI - Model → (Open Systems Inter connection Model)

Application	
Presentation	
Session	
Transport	- Segments
Network	- Packets
Data Link	- Frames
Physical	

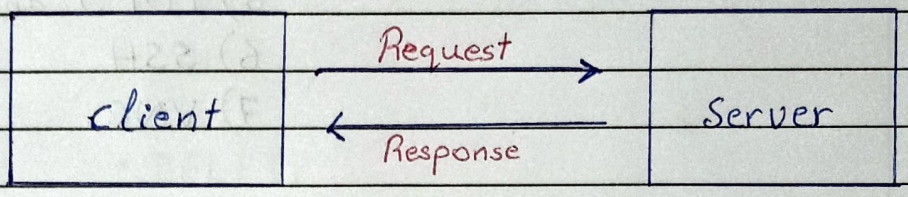
__/__/____

TCP/IP Model →

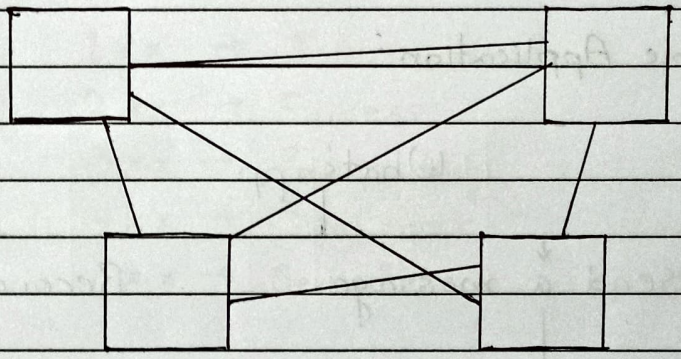
Application
Transport
Network
Data Link
Physical

[Application Layer]

Client - Server Architecture :



Peer to Peer Architecture (P2P) :



Anyone can be Server / Client.

Networking Devices :

- Repeater
- Hub
- Bridge
- Switch
- Router
- Gateway
- Brouter

__/__/____

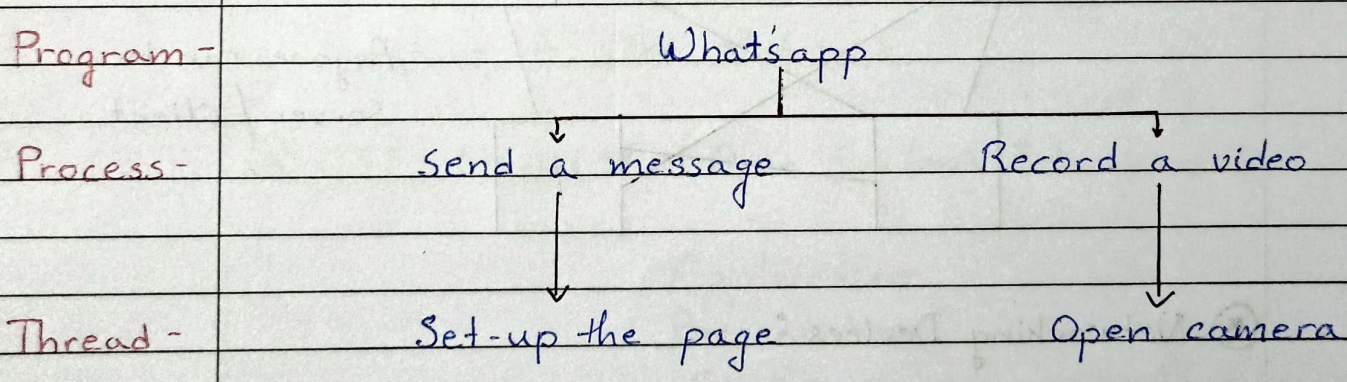
(*) Protocols :

Web Protocols:

- TCP / IP :
  1) HTTP
  2) DHCP
  3) FTP
  4) SMTP
  5) POP3 and IMAP
  6) SSH
  7) VNC

- Telnet : Port - 23
- UDP : Stateless Connections.

Sample Application :



(*) Socket : Interface between processes and internet

(*) Ports :

- Ephemeral Ports - Applications assigns itself random port numbers.

__/__/____

⊛ HTTP: It is a client - server protocol which tells us how we request data to the server and how server sends the response to clients.

- HTTP uses TCP
- HTTP is Stateless

HTTP Methods -

- GET
- POST
- PUT
- DELETE

Status Codes:

1xx → Informational
2xx → Success
3xx → Redirecting
4xx → Client error
5xx → Server error

Cookies: Unique String stored in my browser

⊛ How Email works ?

- For sending - SMTP
- For Recieving - POP3

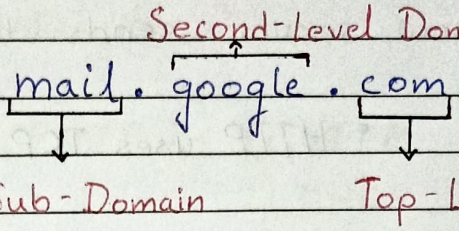
Command to check IP of SMTP servers -

```
$ nslookup -type=mx gmail.com
```

__/__/____

⊛ DNS: (Domain Name System)

For eg.

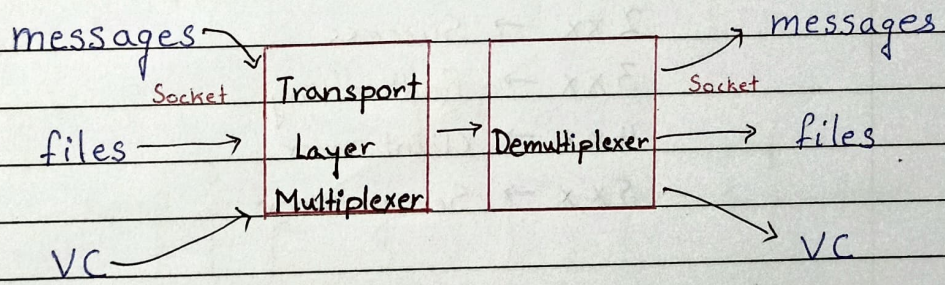


Root DNS Servers

.com .io .org .edu .in

These are maintained by ICANN

[Transport Layer]



- Data travels in packets
- Transport layer will attach these socket port numbers
- Transport layer also takes care of congestion control.
- Congestion control algorithms are built in TCP.

⊛ **Checksum:** A small-sized block of data derived from another block of data for the purpose of detecting errors that may have been introduced during it's transmission or storage.

__/__/____

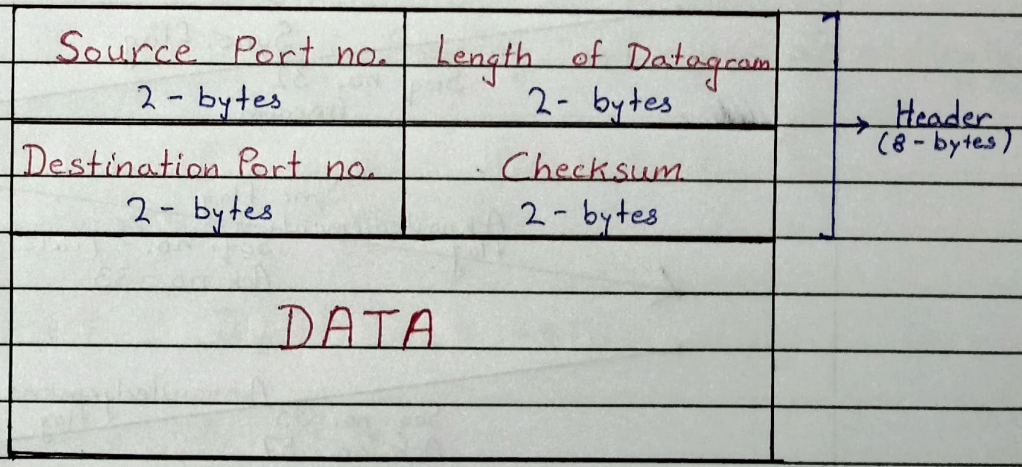
⊛ Timers : (Retransmission Timer)

To retransmit lost segments / data packets, the timer starts and stops when the acknowledgement is recieved.

⊛ UDP : (User Datagram Protocol)

- Data may or may not be delivered.
- Data may change
- Data may not be in order.
- Connectionless Protocol.
- It uses checksums.

UDP Packet →



- Total packet size is 2^{16}

Command to track packets

$ sudo tcpdump -c 5

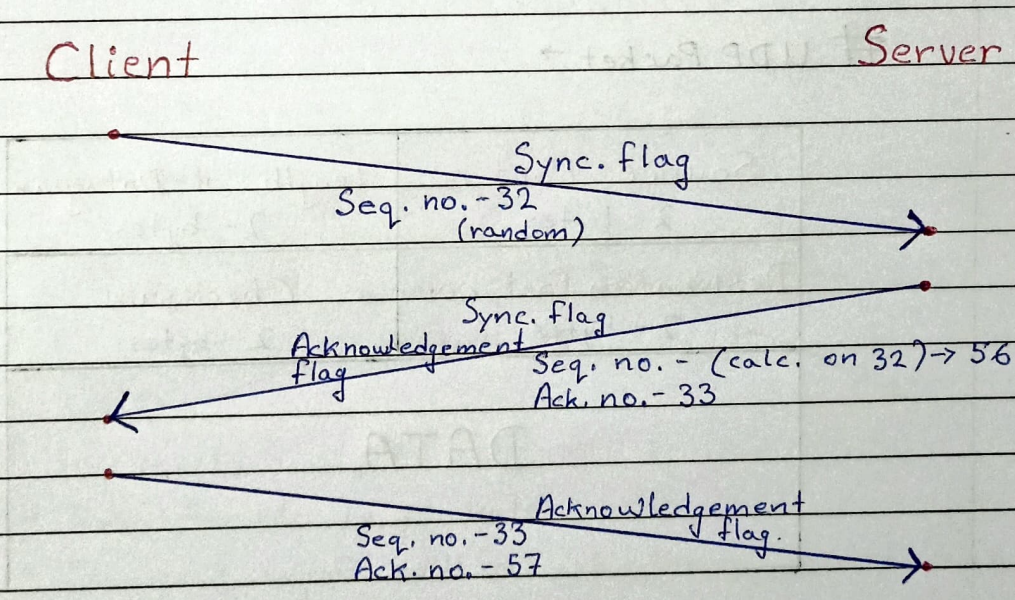
__/__/____

(*) TCP: (Transmission Control Protocol)

- Transport layer protocol.
- It segments the data, received from Application layer.
- Congestion control
- Takes care of → when data does not arrive
  → maintains the order of data.

- Connection Oriented
- Full-Duplex

3-Way Handshake:



__/__/____

[Network Layer] Here, we work with "routers".

⊛ Hop - by - Hop Forwarding :

Hoping routers to routers until it reaches the correct router.
Checks Forwarding and Routing tables for this task.

Control Plane :

Used to build these routing tables.

- Router → Nodes
- Links → Edges

i) Static Routing - (Manual)
ii) Dynamic Routing - (Evolves accordingly)

⊛ IP - (Internet Protocol)

IPv4 → 32 bits, 4-words

5.6.9.14

00000101
8-bits

- Classes of IP Address →

A	0.0.0.0 - 127.255.255.255
B	128.0.0.0 - 191.255.255.255
C	192.0.0.0 - 223.255.255.255
D	224.0.0.0 - 239.255.255.255
E	240.0.0.0 - 255.255.255.255

__/__/____

- IETF assigns the IP Addresses.

(*) Packets: Header is of 20 bytes

- It consists of IPv, length, Identification number, flags, protocols, checksums, addresses, TTL, etc.

#IPv6 → 128 bits

Cons - • Not Backward Compatible
• ISPs need to shift, which is a lot of work.

a.a.a.a.a.a.a.a
Hexadecimal (16 bit)

(*) Middle box :

Firewall - Global Internet
- Your trusted network.

- Filters out IP Packets based on various rules
  - Address
  - Modify Packets
  - Port no.s
  - Flags
  - Protocols

- Two types - i) Stateless Firewall
ii) Stateful Firewall (more Efficient)

__/__/____

NAT : (Network Address Translation)

- Slows down the consumption of IP Addresses.

[Data-Link Layer]

⊛ DHCP : (Dynamic Host Configuration Protocol)

- New device connects DHCP server which assigns IP address to the device.

In data link layer, devices communicate with "data-link layer address" (MAC Address)

Here, we work with frames.

- ARP Cache helps to bring the address.

(Address Resolution Protocol)

⊛ The physical layer is associated with Hardware and Electronics.

This is how Internet works.