CN important

* Jitter > 30 ms → causes distortion

* local host → 127.0.0.1   Broadcast ⌐ f → binary 4 (11)
   or → ffff:ffff:ffff:ffff
   → 255.255.255.25!

*

Reciever MAC	Sender MAC	Riciever IP	Sender IP	PORT NO	DATA	FCS

SEGMENT

FRAME
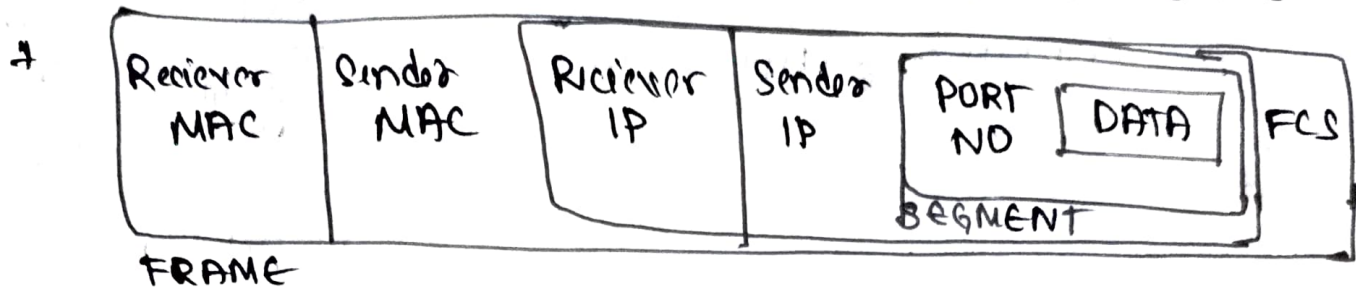
* Internet → Global network of network (Book store)

   WWW → Collection of information accessed using internet
   (Books in the book store)

* PORT NO

   * 16 bits (Range 0-65000) ie 2B

   * 0 - 1023 → Reserved by Internet authority (For
                  protocols etc like HTTP → Port No 80)

   * 1024 - 49152 → Reserved by application's (eg vs code,
                  mongo DB)

   * Rest can be used by others

   * HTTP → uses Port NO 80   HTTPS → 443
     SMTP → PORT NO 25

* Network devices

   • Repeater → Repeats weakened signal (2 port)   PHY LAYER
   • Hub → Multiport repeater + Cannot filter data   PHY LAYER
   • Bridge → Repeater + Data filteration using MAC → DLL device
              (2 port)
   • Switch → Multiport bridge with buffer (DLL) + error checking
   • Router → Connects LAN and WAN (IP add) (NL device)
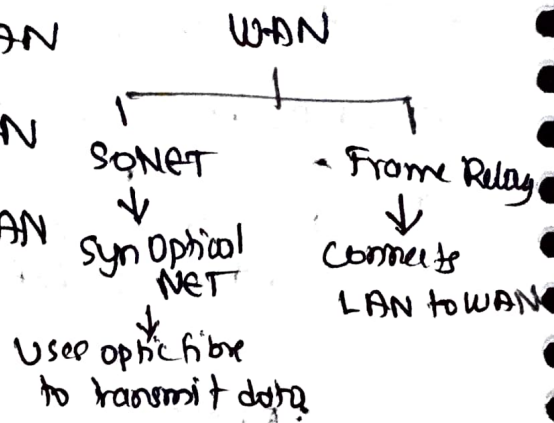   • Gateway → Connects two networks which use diff protocol

* Mesh topology → $n(n-1)$ channels for n-devices

$\frac{n(n-1)}{2}$ duplex links

* Speed → LAN > MAN > WAN

propagation delay → LAN < MAN < WAN

Congestion → LAN < MAN < WAN

WAN
├── SONET → Syn Optical NET → Uses optic fibre to transmit data
└── Frame Relay → Connects LAN to WAN

\# <u>OSI Model</u> → Open System Interconnection

(Inside our local computer)

1) Application layer → Message sent to presentation layer

2) Presentation layer → APPL $\xrightarrow{msg}$ Machine Readable code → Encryption + Abstraction ↓ Compression

   (Translation)

3) Session layer → Session b/w communicating system is created.
   → Before session is created, it does authentication

4) Transport layer → SERVICE to SERVICE Delivery
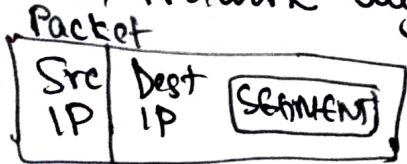   → Responsible to assign port no
   → Data divided into segments (Port No + Seg No + Data)
   → TCP & UDP

   | PORT NO | Data | Segment |

5) Network Layer → END to END delivery
   → Forms IP packet
   → IP → Internet Protocol add / Logical address
   → Not Permanent address

   Packet
   | Src IP | Dest IP | SEGMENT |
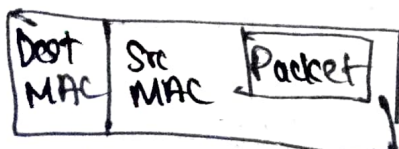
6) Data Link Layer → HOP to HOP delivery
   → Responsible for interfacing with physical layer
   → MAC → Media Access Control → 12 digit alpha numeric
       → Physical Address
   → NIC → Network Interface Card
   → Forms frame

   | Dest MAC | Src MAC | Packet |
   Frame

BY SHIVKARAMAN

1) **Physical Layer** → Anything that carries bits (0,1)

Wired → Ethernet → Carries bits in form of electric pulse

Wireless → WIFI → In form of radio waves

- **Network device in detail**

  1) **Switch**
     → Operates in DLL
     → Maintains MAC add table
     → MAC add table is populated by looking at src MAC of any recieved frame.
     → No of ports → 24 to 96 or more
     → Functions —— Learning, Flooding, Forwarding

Store & Forward	Cut through	Fragment Free	Filtering
- Error check	- Does min checking	- Combination	- Prevent forwarding packet to src
- Slowest	- sto Fastest	- Checks only first 64 B	

   → Most switches operate in Store & forward

  2) **Router**
     → Facilitates comm b/w networks
     → Operates in NL
     → Maintains Routing Table → Path to all networks & ARP table
     → Routers assumption → It knows all networks

     Functions {
     → Populating Router Table
     → Populating ARP table
     }

     — Directed Connected Router
       Router knows the network to which it is directly connected
     — Static Route
       By Admin
     — Dynamic routing - Router talks to each other informing the routes it knows

Routing Table	Network		
Method	IP address	Interface	Next hop
DC/. static	10.xx.xx.xx	Left/Right	
	22.22.22.2		

- **ARP** - Address Resolution Protocol

  Used to find unknown MAC address using known IP address

  - Has ARP Table / ARP cache
  - ARP packet - 28 B

Protocols Used in each layer

1) Application Layer

- HTTP, SMTP, DNS, SSH, FTP
  - HTTP ↓ (TCP/IP)
  - client & server ↓
  - SMTP → Send mail
  - SSH ↓ Login to terminal

- ### HTTP
  - Stateless Protocol ie it doesnot have client info/state
  - Methods - GET, POST, PUT, DELETE
  - Status Codes -
    - 1XX - Informational — Request is being processed by server
    - 2XX - Success
    - 3XX - Redirection Purpose — client needs to redirect to new url
    - 4XX - Client error — Unauthorised, Not found etc
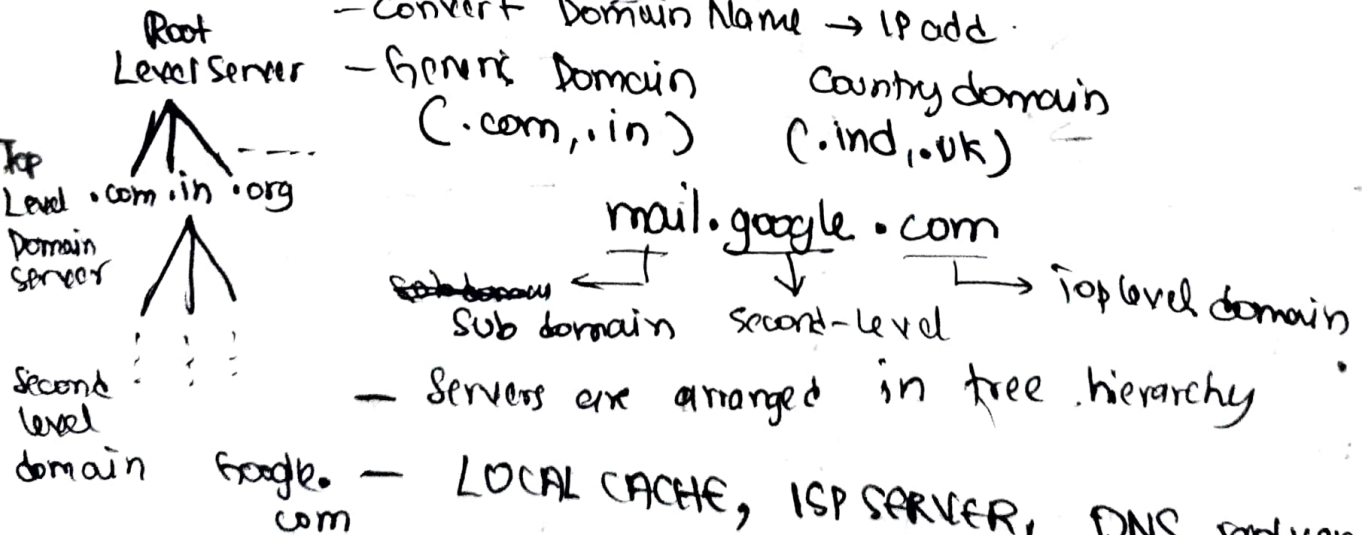   - 5XX - Server Error — Timeout, gateway error etc

- ### Cookies
  - HTTP is stateless. So we use cookies to remember things about user.
  - Eg: Without cookies, every time you visit Amazon, you have to login coz for HTTP, you or a new client every time
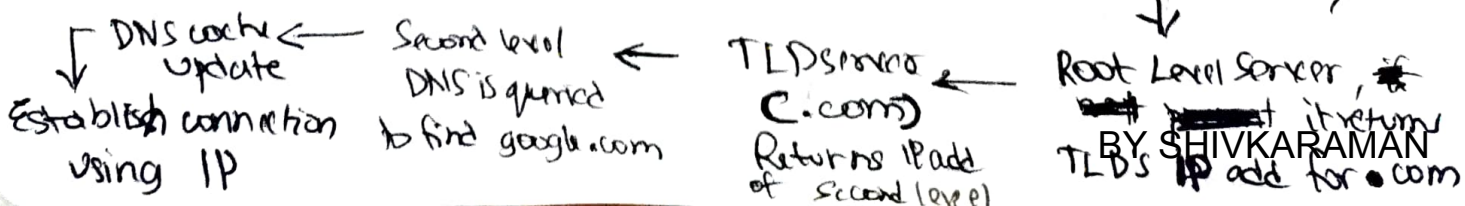 - Every cookie has expiry date specified by website

- ### DNS (Domain Name System)
  - Convert Domain Name → IP add.

  - Root Level Server
  - Generic Domain (.com, .in)   Country domain (.ind, .uk)

  - Top Level Domain Server: .com .in .org

  - Domain Server

  - mail.google.com
    - Sub domain ← | Second-level ↓ | → Top level domain

  - Second level domain

  - google.com

  - Servers are arranged in tree hierarchy

  - LOCAL CACHE, ISP SERVER, DNS resolver,
    ↓
    Root Level Server,

  - DNS cache ← Second level DNS is queried to find google.com ← TLD servers (.com) Returns IP add of second level ← Root Level Server, it returns TLD's IP add for .com
    update ↓
    Establish connection using IP

2) Presentation Layer → Translation , Encryption, Compression
(→Machine code)

3) Session Layer → Session creation

4) Transport Layer → TCP, UDP
          CRC, Checksum
          ↑
     → Other Responsibility − Flow & Error control
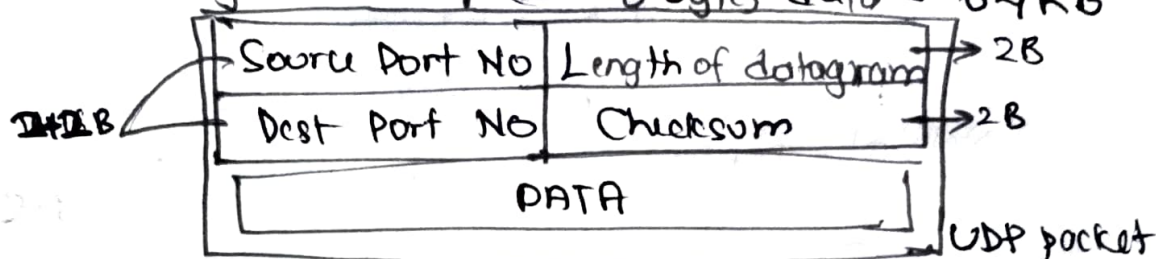   − Congestion Control

- UDP (User Datagram Protocol)
                              ↓
    − Connectionless Protocol   Leaky, Token bucket
    − Uses Checksum for error control
    − 8 byte header + $(2^{16} - 8)$ bytes data = 64 KB

Source Port No	Length of datagram	→ 2B
Dest Port No	Checksum	→ 2B
DATA		

8+8 B

                                      ⌋ UDP packet
                                  − Not reliable

- TCP (Transmission Control Protocol)
                                 − Doesnot maintain
                                     Same order
    − Connection oriented
      − 3 way handshake
                                       Features

Client   Server
                       − Full duplex
SYN
Seq = 32   − Reliable

SYN & ACK   say 56   − Error checking
Seq = (maths on   using
32)   Checksum
ACK = 32+1 = 33

ACK   − Maintains
Seq = 33   same order
ACK = 57
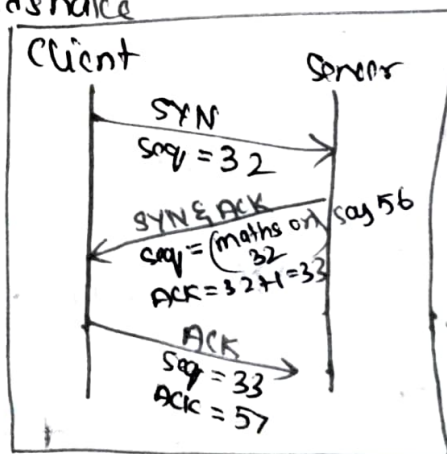
5) Network Layer

192.168.10.11,
    ‿‿‿‿‿‿‿‿   ‿‿
       ↓   ↓
Network Add   Device address
(Subnet ID)   (HOST ID)

10.0.0.0 −	10.255.255.255
17.16.0.0 −	17.16.255.255
192.168.0.0 −	192.168.255.255

IPV4 → 32 bits (4B) → 4 words
IPV6 → 128 bits → 8 words each 16 bit hexadecimal | Private IP

12.0.0.0 / 31
   ⌞→ First 31 bits belongs to subnet part

BY SHIVKARAMAN

8) MAC Layer / Data Link Layer

- ARP
- MAC address → 12 digit alpha numeric string
  → 48 bits
  → 6 groups of 8 bits each represented in hexa decimal digits separated by :
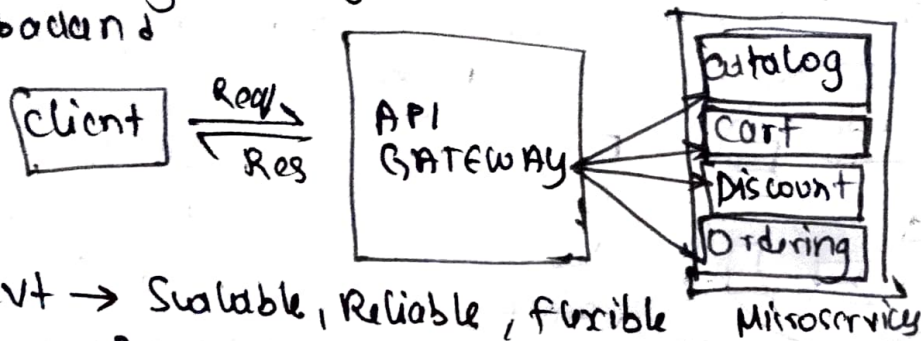
• Networking Commands

1) **Ping** : Sends echo packet, to destination to check if it is reachable and ready to establish connection. Uses ICMP

   ping target-ip-add

2) **Netstat** - TCP/IP command to provide statistics & info of n/w
   - -a → all collection & ports | -r → routing table
   - -n → address & port no

3) **Ip config** — IP add, Subnet mask and default gateway

4) **hostname** - used in application layer

5) **Tracert** - Trace route - path from src to dest
   - displays. IP add of each router along the path

7) **nslookup** - name server lookup - nslookup domain-name
   - used to resolve domain name to IP and vice-versa

8) **route** - view & modify IP routing table

9) **arp** - " " ARP table

10) **path ping** - ping + tracert
    - gives round trip time, no of hops, packet loss %

    IP address assignment

    static   Dynamic   DHCP
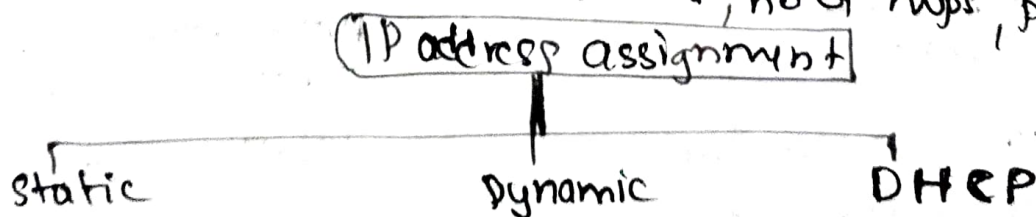
API Gateway

- Server acting as bridge b/w client & multiple microserves in backend


```
Client  ⇌ Req/Res  API GATEWAY → Catalog
                                → Cart
                                → Discount
                                → Ordering
                                  Microservice
```
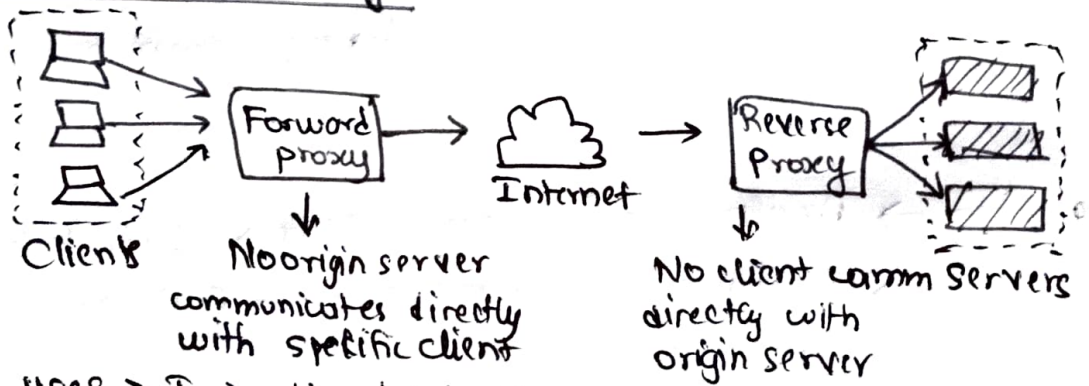
- Advt → Scalable, Reliable, flexible Performance optimization, Secure,
- Disadvt → Heavy Loading, overhead
  Sol → Use horizontal scaling, caching etc

Reverse and Forward proxy


```
Clients → Forward Proxy → Internet → Reverse Proxy → Servers
```

No origin server communicates directly with specific client

Uses → To impliment restriction
     → To block access to certain content
     → To block certain users
     → To hide identity

No client comm servers directly with origin server

Uses → Load balancing
     → Caching
     → Security
     → Hides backend servers.

# Load Balancing

- Distributing load accross different server
- Algos - Round robin, hashing Chash (IP)), etc

Scaling

- Horizontal / Scaling out — Increasing resources by adding more servers, Database etc. Used for unpredictable & rapidly growing workload
  Uses → Cloud Computing
- Vertical / Scaling in — Increasing capacity of node such as adding ram. Used for stable and predictable workloads.
  Uses → Small to medium sized system

• Virtual IP address (VIP)

→ Backup mechanism for default gateway

→ VIP add and MAC add

Routers

default gateway

LAN
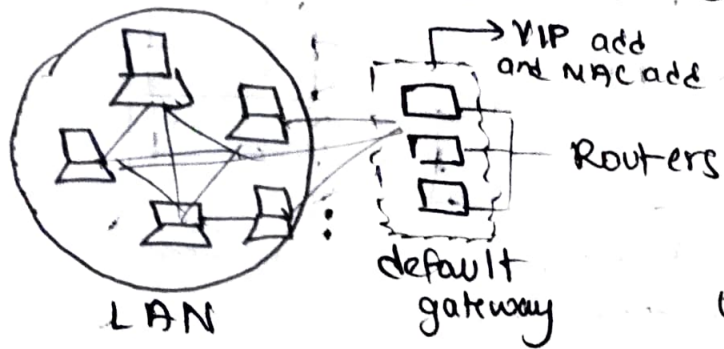
Router with highest priority is assigned VIP address

when a sys requests for default gateway, it requests for VIP add

• Container networking → Containers talk to each other just like applications on different device communicate over n/w

Eg: Docker Networking

- 1G → Analog signal, slow. Max → 2.4 Kbps, Voice only

  2G → Digital signal, more efficient BW and improved networking capacity. Introduced encryption to enhance security
  → Introduced SMS & MMS, mobile data, browsing, email)

  3G → browsing, email, video downloading, video streaming   max → 64 kbps
  → 2G + New Tech $\xrightarrow{uses}$ UMTS (Universal Mobile Telecommunication System)
  → Max → 144 Kbps - 2 Mbps

  4G → High speed, security, high capacity
  → Key Tech → MIMO (Multiple Input Multiple Output)
     and OFDM (Orthogonal Frequency Division Multiplexing)
  → Max → 100 Mbps - 1 Gbps

  5G → Max → 35 Gbps

• VPN - Virtual Private Network

Users internet traffic is routed through an encrypted tunnel
to VPN server. VPN server acts as intermediatory.

forwarding users internet traffic to destination website / server.
website server sends it back through VPN server to the user.

- ~~Router~~ Network address Translater (NAT)

Multiple devices on private ~~IP address~~ network to share single public IP address. NAT gateway act as an intermediatory between Private & public nw.

Private N — NAT Gateway — Public IP → Internet
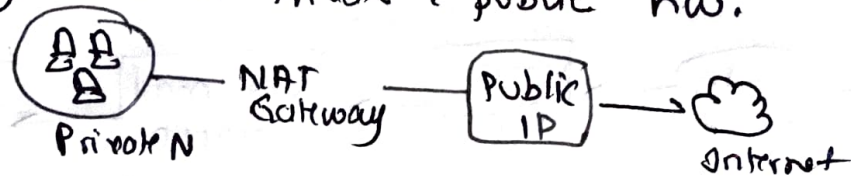
- Bluetooth
  - Short range radiowaves
  - Piconet (master and slaves) (master decides which device/slave can transfer data)
    - N/w of upto 8 bluetooth devices connected to single central device/master
  - Each device can comm with several piconets simuchaniously
  - Scatternet → Nw of multiple piconets.
    - used when >8 devices

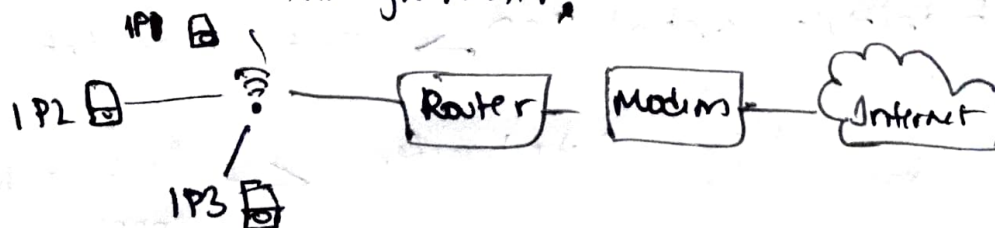Working → device discovery, pairing, connection establishment

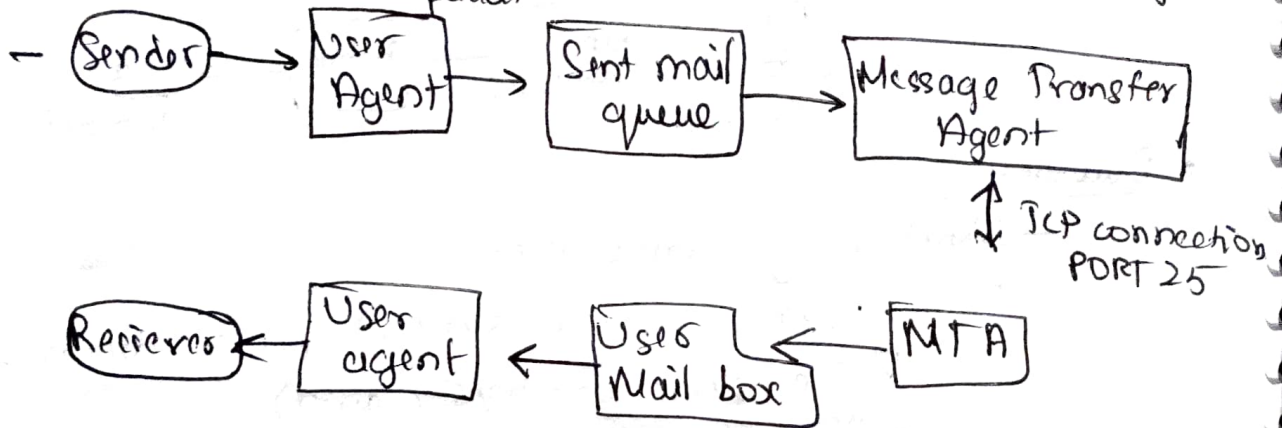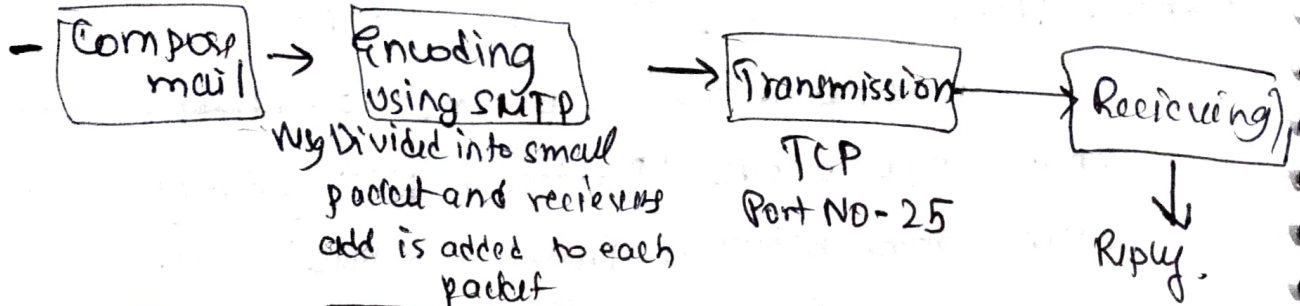| Inquiry Scanning | | piconet |
Data trans, Disconnection
Link Controller Protocol

- Hotspot

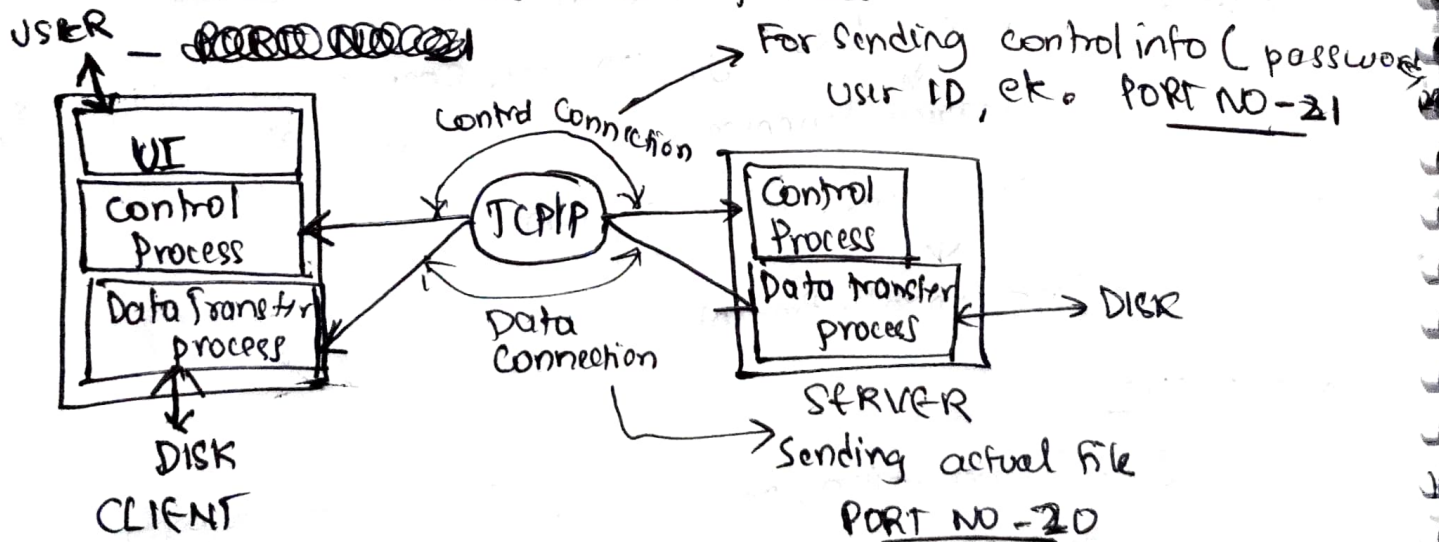Internet connection, Router, Wifi, Device connect to wifi, Data trans through router.

IP1 — IP2 — ((•)) — Router — Modem — Internet
IP3

BY SHIVKARAMAN

Email

SMTP (Simple Mail Transfer Protocol)

- Compose mail → Encoding using SMTP → Transmission → Recieving

  Msg Divided into small packet and recievers add is added to each packet

  TCP Port No - 25

  ↓ Reply.

- Sender → User Agent → Sent mail queue → Message Transfer Agent

  ↕ TCP connection PORT 25

  Reciever ← User agent ← User Mail box ← MTA

FTP

- Application Layer Prot
- Uses TCP, & transfers between local & remote systems
- 2TCP connections used in parallel

USER — ~~PORTED NODE~~

For Sending control info (password, user ID, etc. PORT NO - 21

Control Connection

```
CLIENT                                    SERVER
[ UI                    ]   TCP/IP   [ Control Process    ]
[ Control Process       ] <------->  [ Data transfer      ] → DISK
[ Data Transfer process ]            [   process          ]
         ↕                Data
        DISK            Connection
```

→ Sending actual file
PORT NO - 20

FTP — Client initiates control TCP → Client control info sent → Server recieves

| Control connection is active while Data transfer | Data transfer ← Initiates data connection TCP ← to client

— Max size — 2GB, No encryption, Multiple recievers not supported

BY SHIVKARAMAN