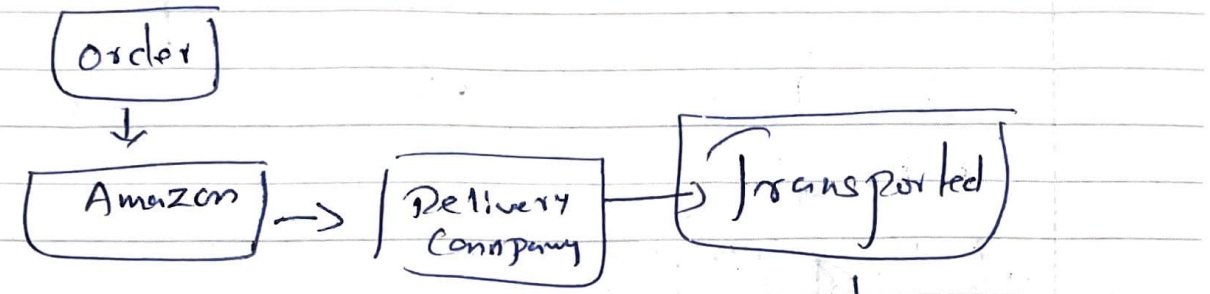
Structure of Network.

Example: ~~Food order~~ Online Ordering



Map or understanding (Internet) with above example.

OSI Model

Open Systems Interconnection

7 Layers.

① Application : Implemented in Software (applications)

In our system ↓ data

② Presentation : Converts / Translates from Human data to machine language.
→ Data is Compressed.

③ Session - Authentication, Authorization takes place.

④ Transport - protocol & data transfer.
- Segmentation + Flow Control + error Control
src & dest (port no.)
sequence no.
↳ checksum (verification & data transfer)

⑤ Network : Router lives here, it is the point where data moves from One Computer to another Computer (Has data packets & transfers based on ip)

Logical point

⑥ Data link : Allows to Communicate to dest Computer based on physical Address.
- Add's physical addres to the data frame (MAC addr) & sends it to physical layer.

⑦ Physical : Receives the signal at receivers end.

The

ME

Application Layer

MONTU

Whatups message : sent

"Hey Montu"

Whatups message : Receive

"Hey Montu"

"Hey montu" → Machine understantable language : Data Compressed.

Presentation Layer

checks for Autentication Authorization of user, data

Session Layer

Set the protocol to the data transfer.

1) Segmentation : src / dest. port
   Sequence num
2) Flow Control
3) Error code

Transport Layer

A mid point b/w src / dest where transfer of data is made using IP

Network Layer

Add's the physical address to dataframe. & sent's to below layer

Data Link

Checks the physical layer & transfers to dest

Hey Montu

Physical Layer

Another Model: TCP/IP Model.

The Internet protocol suite.

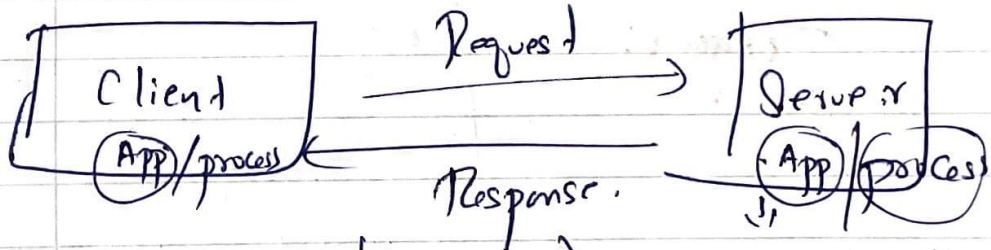
5 layers.

More practical.

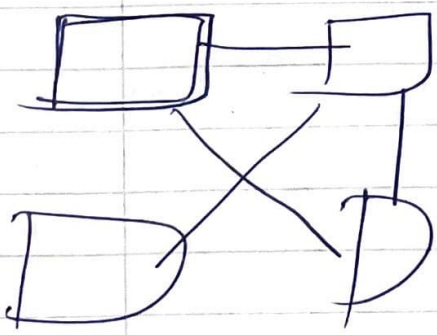
- Application
- Transport
- N/w
- Data link
- physical

Application Layer:
* User Interaction.
* For eg. Browser, whatsup etc.
* Lies in our devices
* Has some protocol
* Client - Server Architecture.

① Client - Server Arch



② P2P arch. (peer to peer)



-> Decentralized
-> Every single Computer is Server / Client.

Protocols

web protocols:

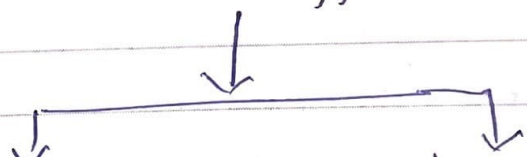
TCP / IP :

* Http : How Html page transfer
* DHCP : Allocates free ip addr to connected ppl (Dynamic host Ctl protocol)
* FTP : files transfer (file transfer prn)
* SMTP : Emails transfer (Simple mail transfer)
* POP3 & IMAC : Receive mail
* SSH : Connecting to another system. terminal.
* VNC : Graphical ctl (Virtual Nfw Ctrl)

* Telnet : Port 23
  * Connect to the user using Telnet.
  * Not encoded & encrypted.

* UDP :
  Stateless Connection.

Program : whatsApp.



Process — Send a Message — Record a video

Thread : (Lighter version of process) — Set up page — Open Camera.

(Single job)

Sockets: Interface b/w process & Internet

Ports: **Ephemeral ports**

We know that port number determines the "app". But what if user had opened multiple tab's & window? How will know which port to communicate?

↓

That's where **Ephemeral ports** comes into picture.

↓

Created for Instance & gets removed after Instance is closed!

↓

Exists on Client side.

HTTP: Application layer protocol

Client —HTTP req→ Server

Server —HTTP response→ Client

HTTP uses **TCP** ~~inside App layer~~

↳ stateless protocol

App layer

↳ Transfer protocal.

↳ Transport layer.

Method: Tells us what to do.

① **GET:** Request the data.

② **POST:** upate provide the Input data.

③ **PUT** : puts data at spectic location

④ **DELETE** : Delete data

STATUS CODES / Error Code.

200 - Sucessful
400 - Bad Request
500 - Internal Server error.

- 1xx → Information.
  2xx → Success
  3xx → Redirecting
  4xx → Client error
  5xx → Server error

NETWORKING:

12/10/

Scenario:

Hey, am using my browser to open an Amazon shopping site, log's in and after some days when I try to open amazon site, am still logged in

Provided http is stateless, how am still logged in?

ANS: Cookies — He is the One.

Cookies:

> Unique String -> stored in client browser.

> First time it will saved in browser

↓

Next time, when the request is made to the Server, A Cookies will be sent.

↓

> Servers: will check and Auto log's in

In Inspect page (check for)

→ Set Cookie } In N/W Section.

→ Cookie

* 3rd party Cookies:

Cookies for Websites which we aren't visiting but Our Cookies from Website we visited is used for unvisited website.

How Email works?

SMTP : Simple Mail transfer Protocol (for Sending)

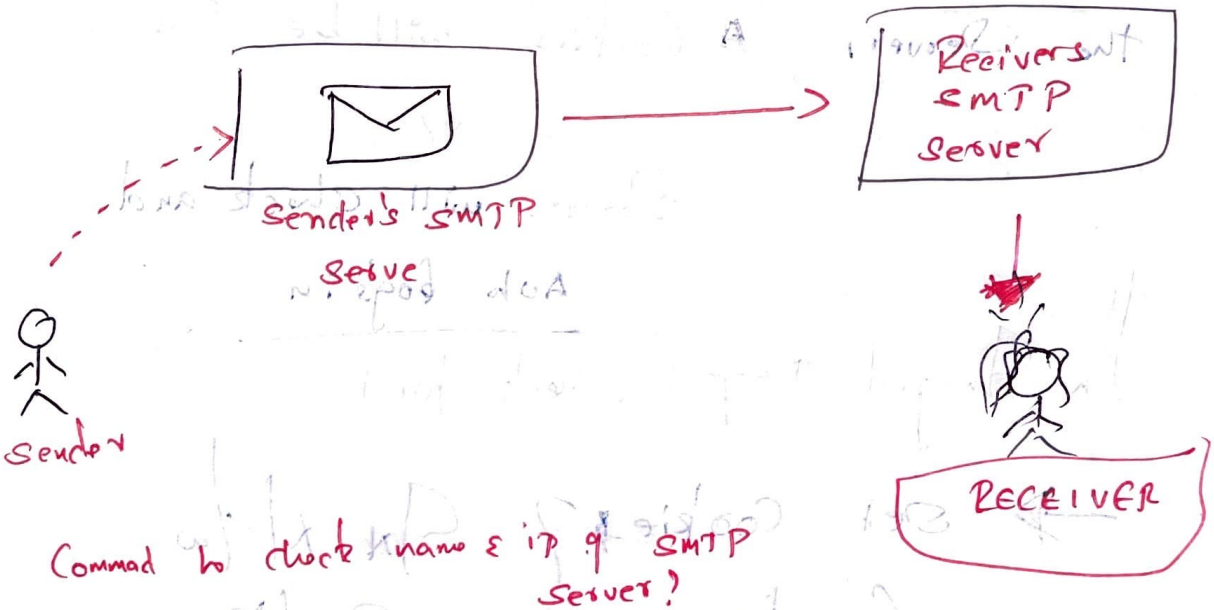
Pop3 : Receiving mail.

APPLICATION LAYER

Application Layer how is mail transferred

xyz@yahoo.com.

abc@gmail.com

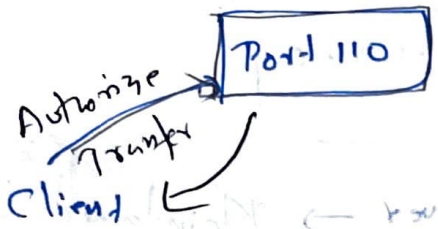


Command to check name & IP of SMTP server?

```
nslookpp -type=mx gmail.com
```

mail exchange

POP: Post Office protocol.



IMAP: Internet Message Access Protocol

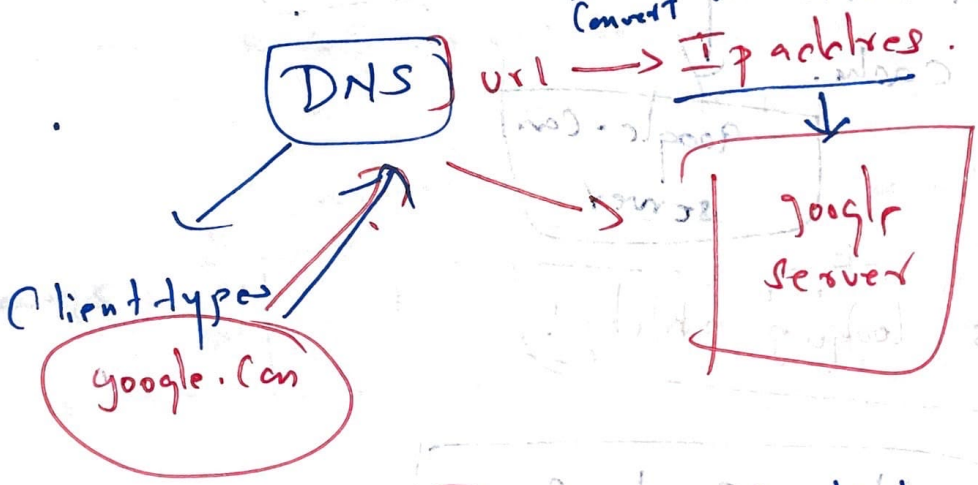
↳ (For Receiving Mails

-> Allows us to view message in multiple Server.

-> Ever device are in sync with Server.

DNS - Domain Name System.

DNS mapped IP addr



DNS -> Directory Database which has Domain name (url). & ip's

Divided into different Class.

Second level Domain

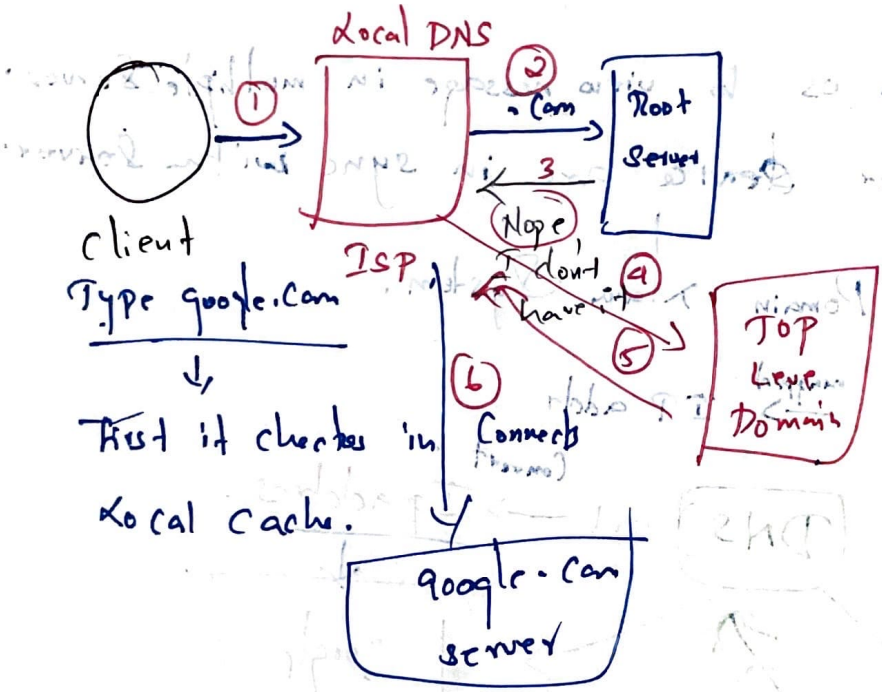
mail . google . com .

Sub domain ↳ Top level domain.

Root DNS Servers : 400 first point of Contact.

Checkout : root-servers.org

Database for Root DNS Server → Maintained by ICANN

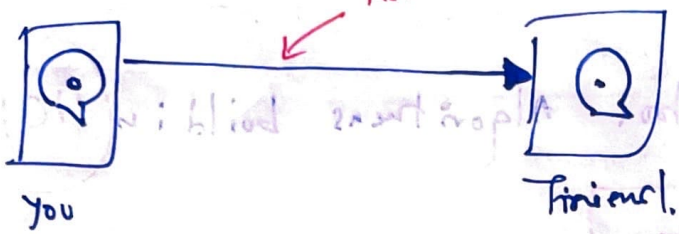


DNS lookup ability!

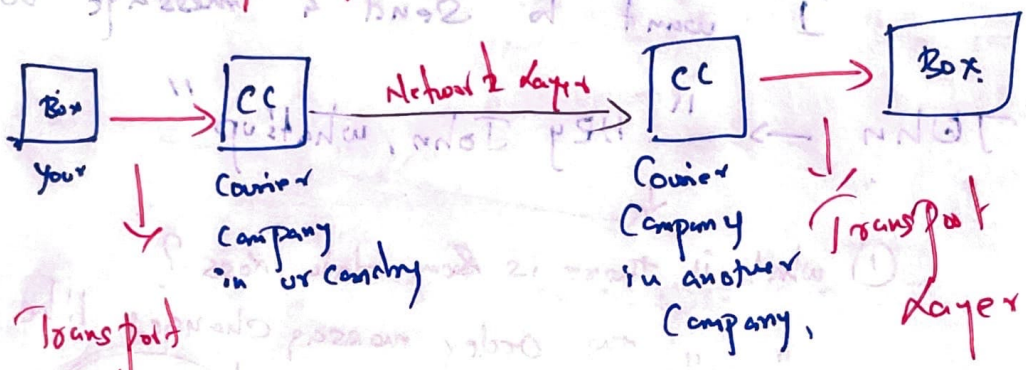
dig google.com

That's all folks for Application Layer.

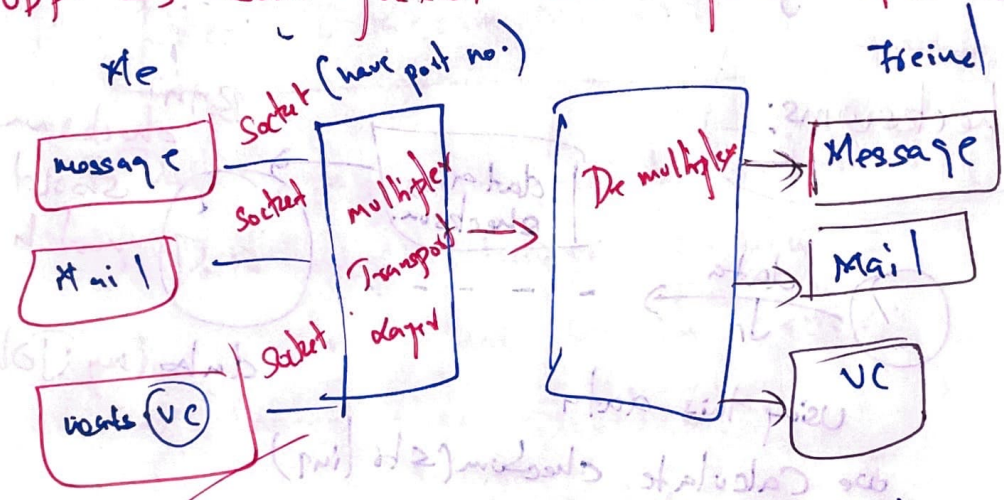
② Transport Layer : Take the info from network to Device/app



Take an example of Courier Companies.



TCP: Ensure 100% of data is transferred whereas UDP → Some packet loss might be present. } Transport layer.



Multiplex: Allows multiple i/p to multiple o/p

Data travels in packets → Transport Layer will attach these Socket port numbers.

* Transport layer takes care of **Congestion Control**

* Congestion Control Algorithms build in **TCP**

Small Back story

I want to send a message to John → " Hey John, what's up "

① what if there is some data loss ?

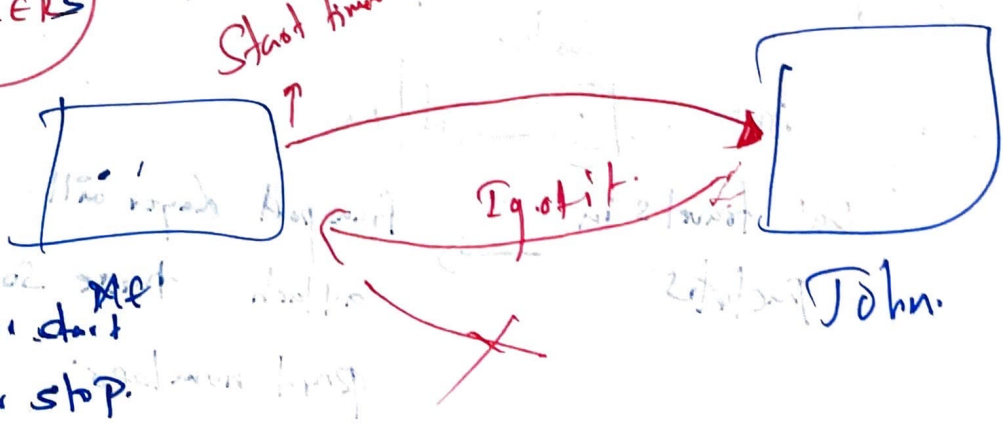
" " the order mossag changes like what's up John Hey.

That's where you need to know Checksum.

Checksums:

using this data we Calculate checksum(string)

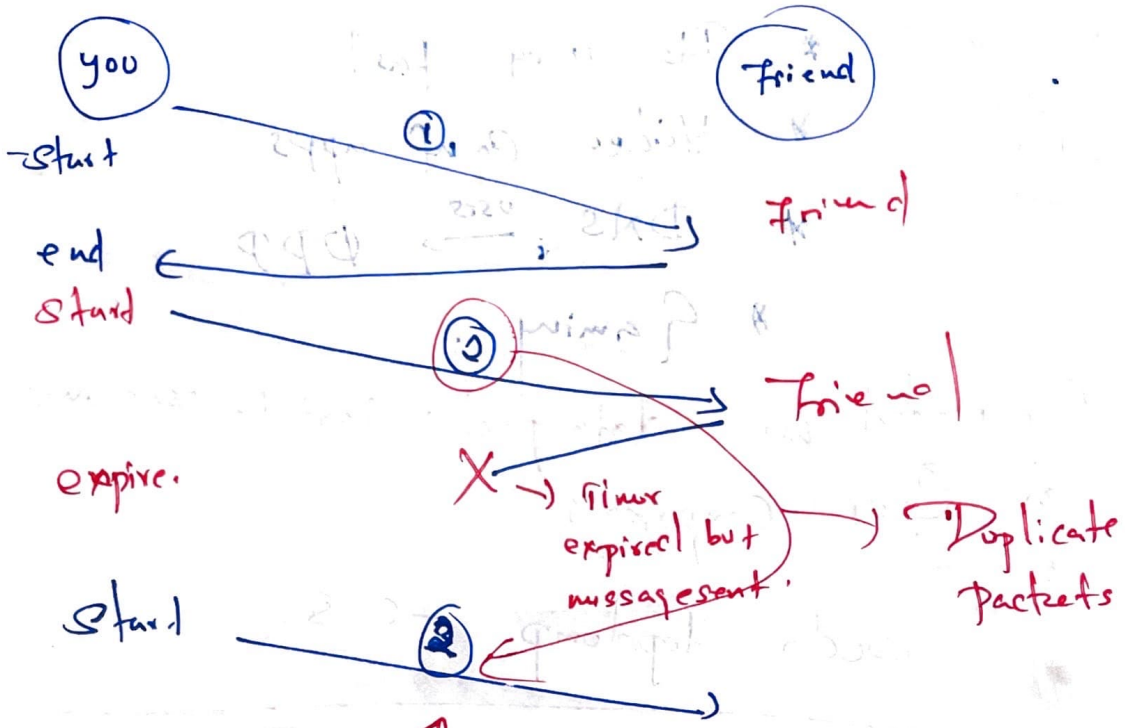
TIMERS



Timer start

Timer stop.

Time expire.



How Solve this problem.

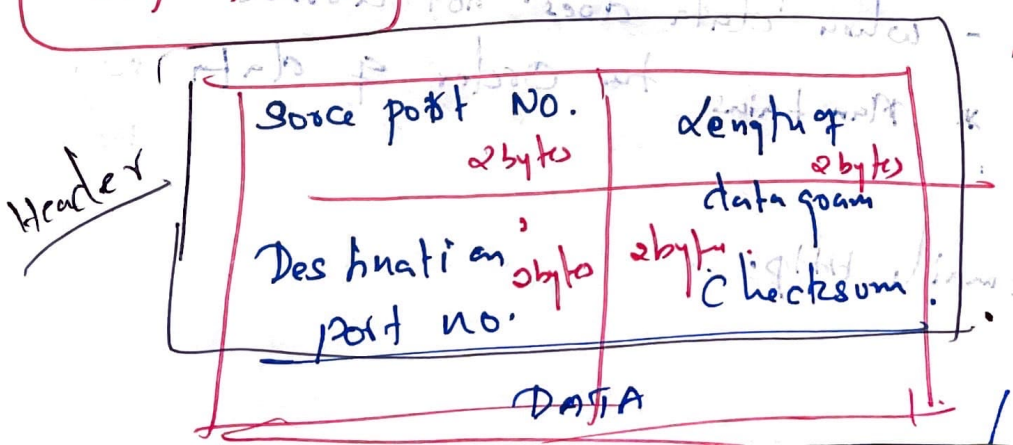
Sequence Number:

UDP: User Datagram protocol.

Transport layer protocol.

* Data may (or) may not be Delivered
* Data may change in b/w
* " " not be in order
* Connectionless protocol.
* UDP uses Checksums.

UDP Packet



for eg.
Total size = 2^{16}
$2^{16} - 8$
Data size
65336 Bytes

Use Cases :

or

UDP

* Its very fast
* Video Conf apps
* DNS uses → UDP
* Gaming

Command for ⇐ data packets that is received on Our Computer :

sudo tcpdump -c 5

TCP : Transmission Control Protocol :

Transport layer protocol

* Application layer sends lots of Raw data → TCP segments ! this data → divide into "data", "add" headers
  → It may also collect the data
  network layer.
* Congestion Control.

Takes Care of
- when data does not arrive
- * Maintains the order of data (seq. no)

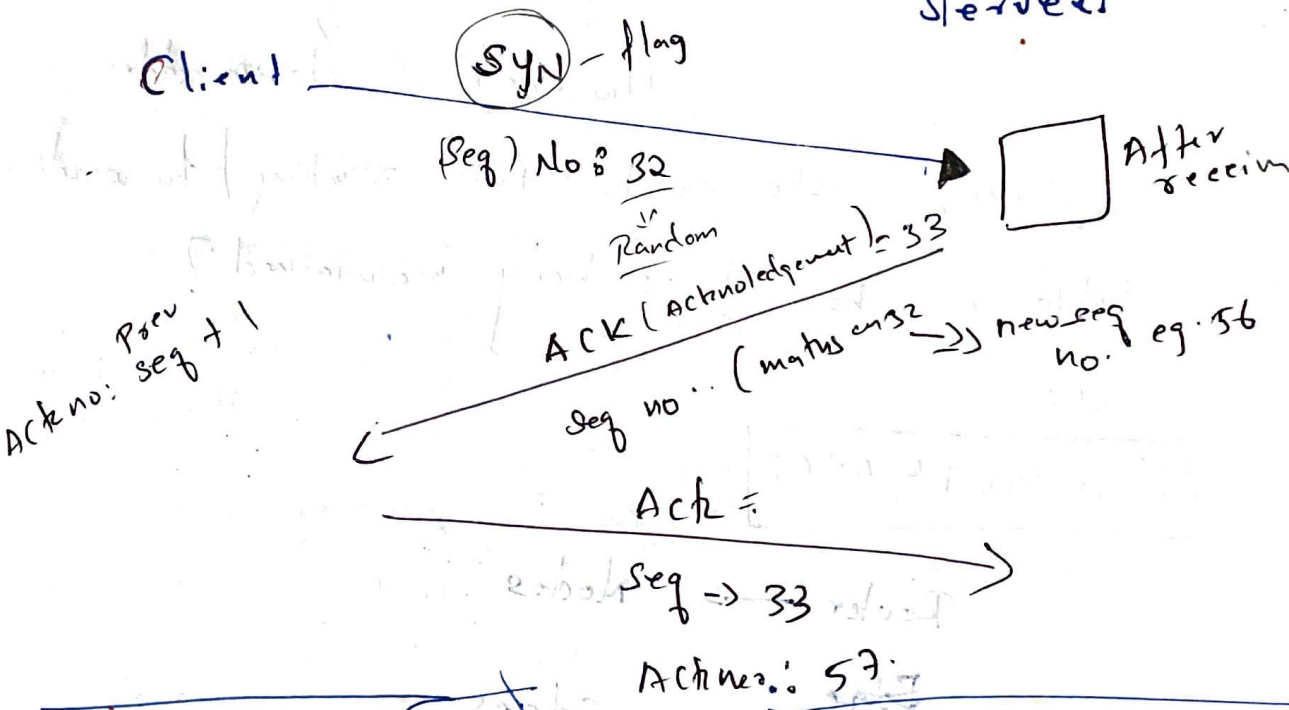
Uses → email, http.

Features :

* Connection Oriented
* Error Control
* Congestion Control
* Full Duplex. (Bidirectional)

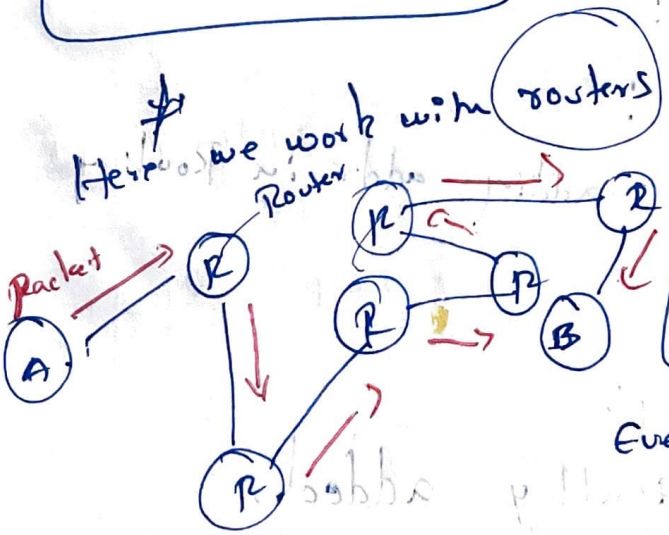
↳ Both A & B can send Both files simultaneous

3-way Handshake:



Network layer

* Here we work with routers



In transport layer
↓
segments

N/w layer → packets

Data link → frames

Every Routers has N/w address.

from (A) → Packet is send with N/w address

↓

Router Now Router will check the forwarding Table & then forwards the packet from A to next Router.

Hop-by Hop

Routing table: It has the paths for src → dest.

forwarding table: It has only one path

IP address: 192.168.2.30

↓ N/w addr ↓ device addr.

BTW, who creates this routing / forwarding Table, how it is being maintained?

↓

CONTROL PLANE:

Router → Nodes

~~Edge~~ Links → Edges.

2 types of Routing:

① Static Routing:

Manually adding addr in Routing Table.

② Dynamic Routing:

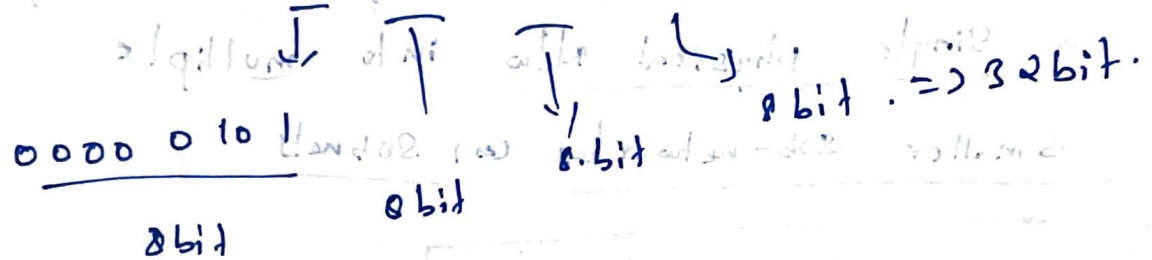
- Dynamically added.

Internet protocol (IP)

IPV4 : 32 bits, 4 words

IPV6 : 128 bits

For eg.



192 . 168 . 2 . 30 → device ID

↓
Subnet ID
eg. airtel TATA

3 types of address

Class 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

SUBNET MASKING:

For eg 255.255.0.0

This we can decide.

It is the technique for logically partioning a single physical n/w into multiple smaller sub-networks called subnets

Variable length subnet masking

For eg: 192.0.1.0 /24

24 → 24 bit remains same

& remaining is 8 bit (can be variable)

start

192.0.1.0 +0 192.0.1.255

same same

Reserved Address

127.0.0.0 /8

Ex: Localhost : 127.0.0.1

Loopback address

Packets: Header is of 20 bytes.

↓

Contains → IPv, length, Identification, flags, Protocol, checksum, address, (TTL)

TTL: Time To Live

Eg: if ttl = 60 → After 60 hops if it doesn't Reach dest then it can drop.

DATA LINK LAYER

IPV6:

IPV4: $2^{32} \simeq 4.3$ billon. (They thought it was sufficient)

IPv6: $2^{32 \times 4} \simeq 2^{128} = 3.4 \times 10^{38}$

↓

unique Id.

↓

Cons: * Not Backward Compatible.

* So much effort required, ISP's need to shift, Lot of Hardware req.

a:a:a:a:a:a:a:a

↓

Hexadecimal (16 bit)

Ex: ABEF:FOO1:3210:9182:0:0:1:3

MIDDLE BOXES

Extra device also interacts with IP addr.

Ex: Firewall: → Global Internet
→ your trusted N/w

Filter out IP packets based on various rules

Based on
- → Address
- → Modify packets
- → Port no's (Rules can be set)
- → Flags
- → protocols.

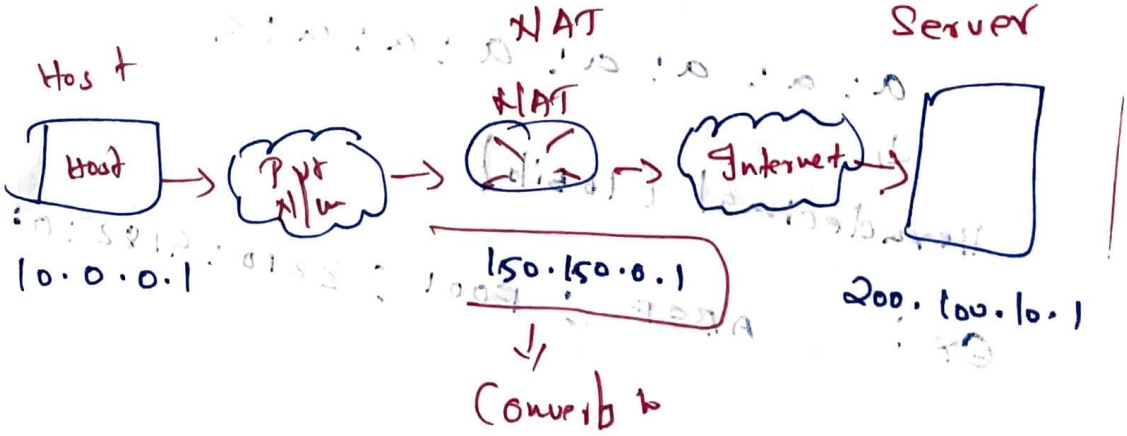
2 types

Stateless vs Stateful firewall.

↓ is more efficient.

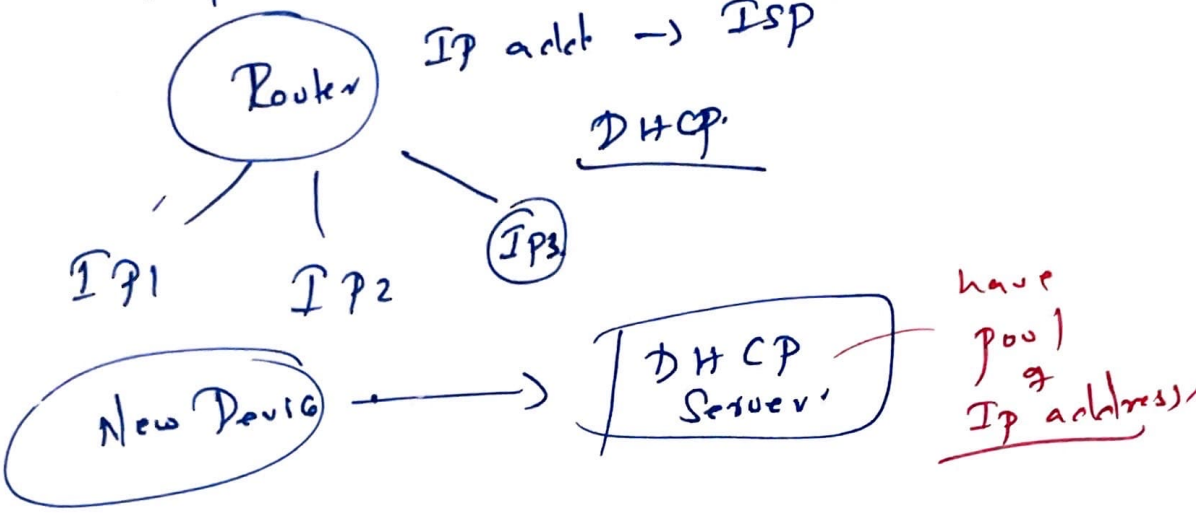
Have this in N/w found in system

② NAT → N/w address Translation. (Middle Box)

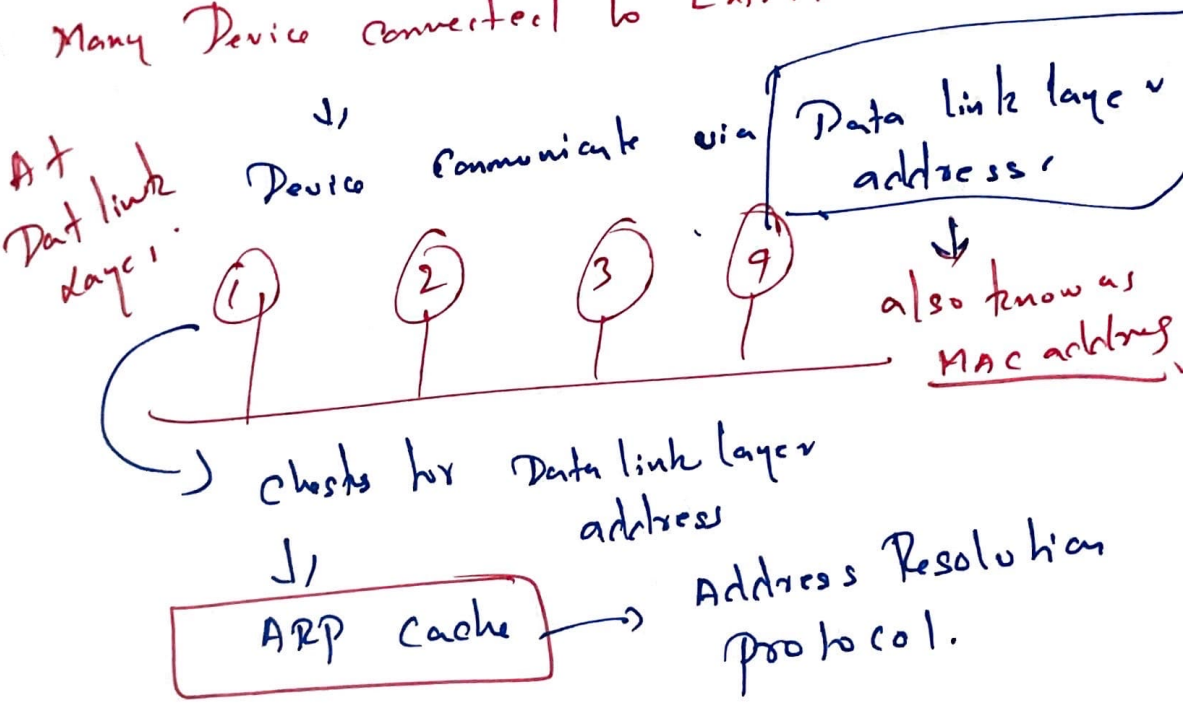


DATA LINK LAYER:

Responsible to send the packets over a physical layer.



Many Device connected to LAN:



Frame: Contains

① Datat link Lay Addr. of Sender.

② IP addr. of destination.