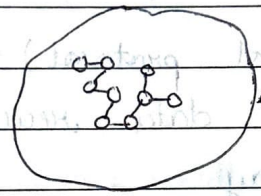
15/5/23

Computer Networking

- CN - Computers connected together
- COMPUTER - Commonly oriented machine particularly used for training education & research

- Internet
  ↓
  Collection of computer network

(A) → (B) ⇒



← Region / country

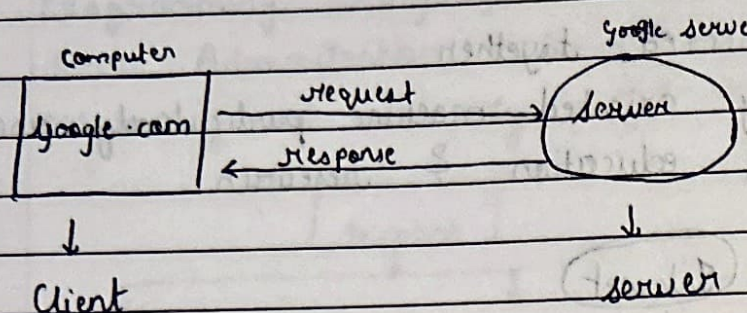
* How did it start ?

- During cold war, there was a battle betⁿ USA & Soviet Union, who will launch first satellite.

- Russia won and launched Sputnik in 1957
- USA created ARPA (Advance research project agency) for scientific discoveries.
- ARPA had facilities (buildings) wanted a way to communicate with each other as buildings were far
- So they created ARPA net. They were 4 places where computers were kept

①	②	③	④
MIT	Stanford	UCLA	University of UTAH

- They used TCP / IP

* 

*] Protocols : Rules defined by the internet society

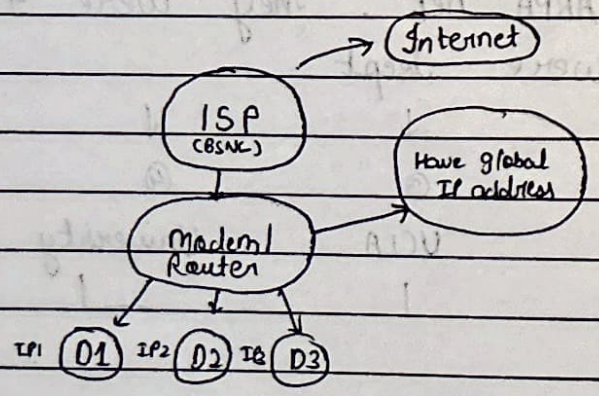
* ① TCP (Transmission control protocol) :
→ It ensures that the data reaches its destination & not corrupted on the way

② UDP (User datagram protocol) :
→ 100% of data is not reaching Eg: Video conferencing

③ HTTP (Hyper text transfer protocol) :
→ It defines the format of the data being transferred on web servers. Used by web browsers

• IP address of self :- ($ curl if config.me -s)

* How data is transferred ?



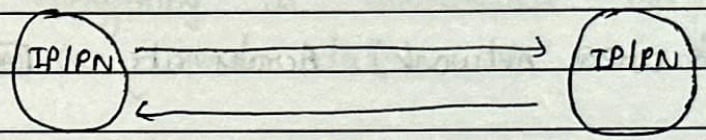
• Every device connected to router will have same IP address

• Modem gives IP addresses to the devices as well known as local IP address

• Modem assign IP addresses using DHCP protocol (Dynamic Host Config protocol)

- Internet service provider (ISP) gives you a Modem / Router which has a global IP address
- Devices connected to modem will have same IP address
- D1 will make a request to modem and modem will make a request to google.com and internet will return back the request
- Modem will decide the device with the request (D1) using NAT (Network access translater)
- IP address decides, which device to send the data and port numbers decide, which application should it send data

Friends chatting



- IP address will determine where your computers are located
- PN will determine which app you are using to communicate

Ports → 16 bit numbers

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

HTTP = 80 PN

MongoDB = 27017

Ports from 0 - 1023 are reserved ports

If we created one app and want to host it on port 80, it is not possible as it is reserved for HTTP

PN 1024 - 49152 → reserved for app (MongoDB, SQL-S (1433))

Remaining we can use

Camlin Page
Date / /

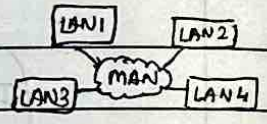
1 mbps = 1 mega bits transferred per second (1000000 bits/s)
1 gbps = 10^9 bits/s
1 kbps = 1000 bits/s

* How communication betn 2 computers happen?

→ Guided way (Eg: wired connection) path defined
→ Unguided way (Eg: Wifi / Bluetooth) no single path

* Connections:

1] LAN (Local area network): Small house / office
Ethernets, wifi



2] MAN (metropolitan area network): Across city

3] WAN (Wide area network): Across countries
optical fibre cables

↳ SONET (Synchronous optical networking): Carries data using OFC hence it can cover larger distances
optical fibre cables

↳ Frame relay: A way to connect to LAN to WAN

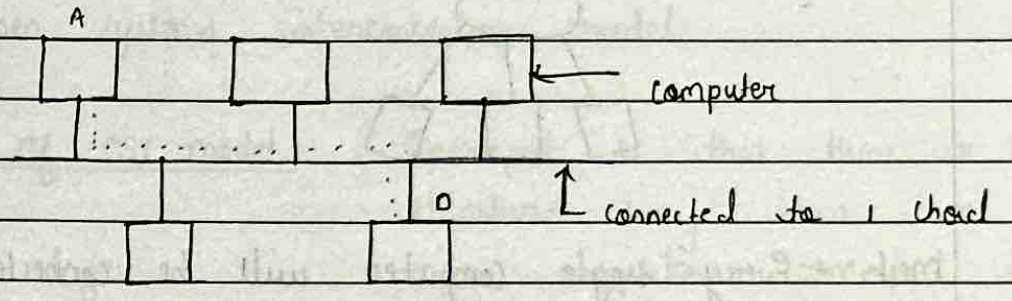
* MODEM / Router

- Device used to convert digital signals to analog sig & vice versa
- Digital data in the computer can be converted to electrical signal
- Router is a device which routes the data packet based on their IP addresses

- ISP - Companies that provide access to the internet
  - They are connected to larger service providers (Tier-1) i.e TATA, (Tier-2) i.e Airtel

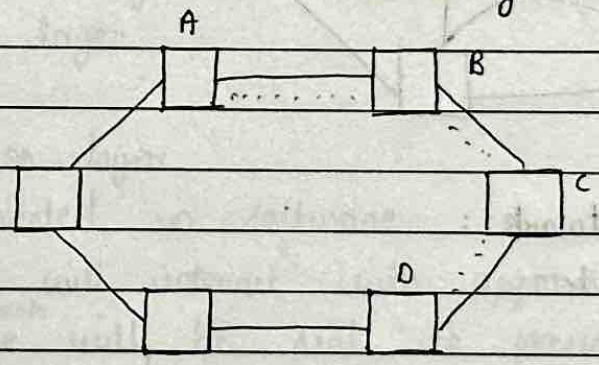
* Topologies (How computers are connected?)

1.] BUS - Connected to 1 chord / backbone



- Since everything is transmitted via 1 cable, only 1 person can send data at a time.

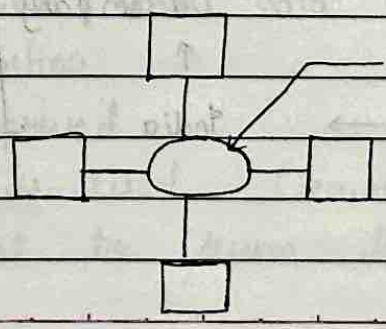
2.] Ring - Connected in a ring)



- Data also goes to B & C in order to reach D

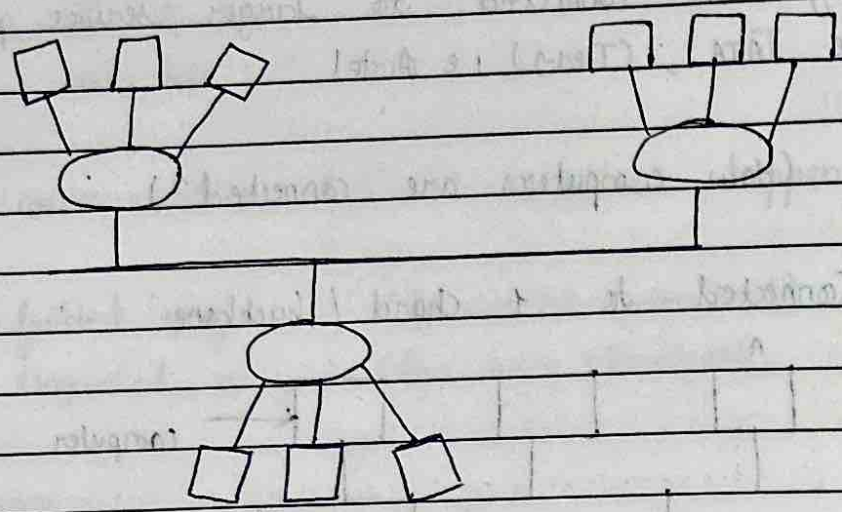
- Every system communicates with one another

3.] Star - All computers are connected to one device

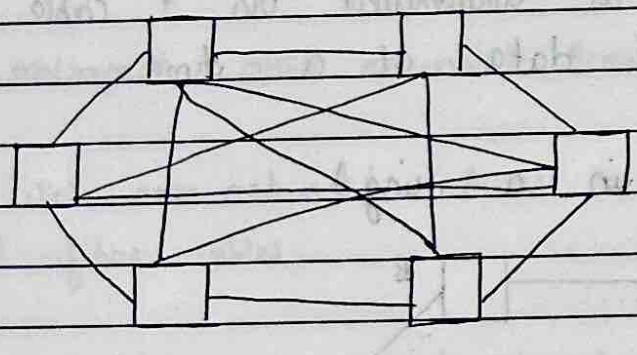


- If central device fails, network will go down

4.] Tree - Combination of Star & Bus)



5.] Mesh - Every single computer will be connected to every single computer



- Expensive
- Not scalable

* Structure of network :

Eg of Amazon order

Order → Received → Application layer

↓ ↑

Amazon → Amazon

↓ ↑

Delivery company(is) → Del. company

↓ ↑

Transported → India

- Suppose, you request a video from YT
- YT will prepare the data from its server
- The data will arrive in the server of your country
- Received to you

*** **OSI model:** (overview)

- OSI - Open systems interconnection model

7 layers of OSI model: Developed so that there is a standard way how two or more computers connected to each other

1] Application layer
2] Presentation layer
3] Session layer
4] Transport layer
5] Network layer → Router lives here
6] Data link layer
7] Physical layer

① Application layer:
- Implemented in software
- Users will interact with application (emails, browser)
- Message/data will be sent to presentation layer

② Presentation layer:
- It will get the data from app^n layer & that data will be in form of characters, words etc
- PL will convert this data into binary format
- Provides abstraction
- Data is compressed
- SSL protocol is used (Secure Socket layer)
- Data is sent to session layer

Camlin Page
Date / /

③ Session layer
- Helps in setting up and managing the connections & it enables the sending & receiving of data followed by termination of the connected sessions
- Authentication (username & password)
- Just creates a session (Eg: Flipkart & computer)

④ Transport layer
- Data is received from session layer
- Transportation is done in 3 ways
a.] Segmentation :- Data received from session layer will be divided into small data units known as segments
- Every segment will contain the source and destination port number and sequence number
- Sequence number helps to reassemble the segments in correct order

b.] Flow control :- Controls amount of data being transferred

c.] Error control :- Some data packets got lost or corrupted

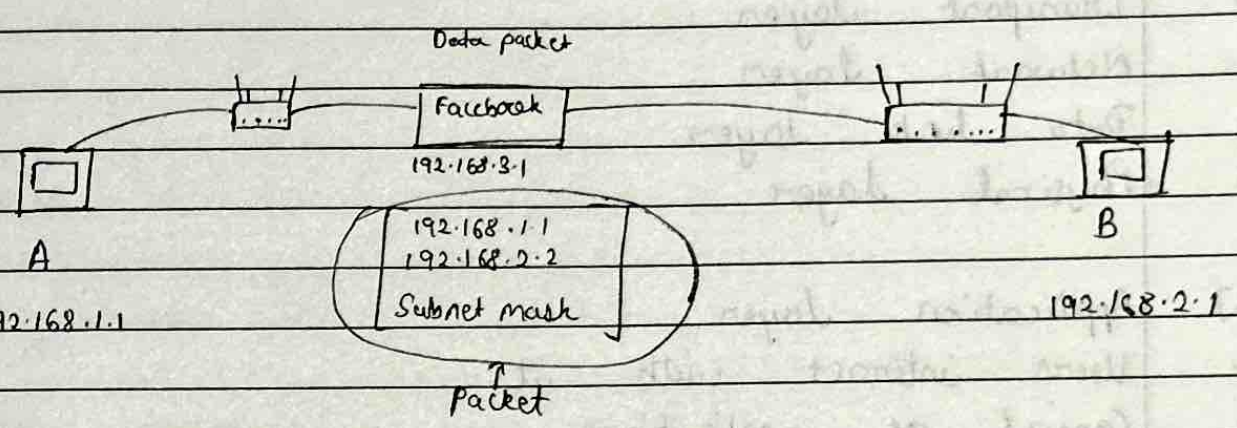
⑤ Network layer
- Works for transmission of received data segments from one computer to another that is located in different network

- Network layer assigns the sender and receiver's IP address to every segment and it forms IP packet

- Performs routing (Best path to transfer data)

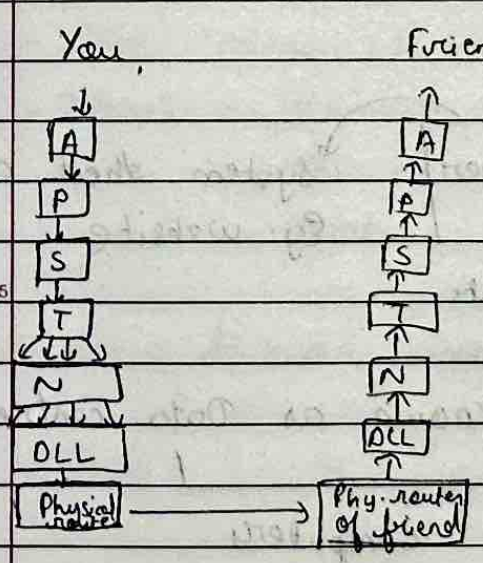
(6) Data link layer :

- Allows you to directly communicate with computers and hosts
- DLL will recieve the data and packets from network layer and data packets contain the IP addresses of both sender and receiver



(7) Physical layer :

- It's a hardware medium
- Have hardware medium like wires etc
- Transmits bits from electrical signal



* Another model : TCP/IP model

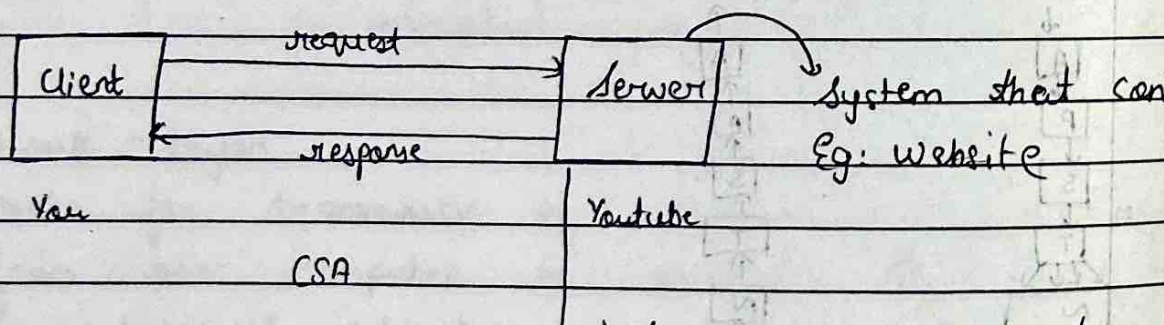
- Similar to OSI model
- Layers are reduced

① Application layer
② Transport layer
③ Network layer
④ Data link layer
⑤ Physical layer

1.] Application layer
- Users interact with it
- Consist of applications
- Eg. whatsapp, Browsers
- Lies on devices
  - → Protocols
  - → Client - Server architecture

* How application talks

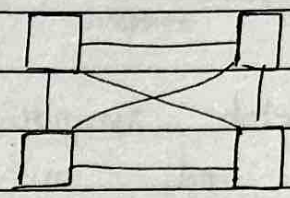
1.



- Collection of huge number of computers
- Have static IP addresses
- Good internet connection (High upload speed)

2. P2P (Peer to Peer)

- In this, application running on various devices get connected to each other
- There is no 1 dedicated data center / server.



* Protocols:

a.) Web protocols:

1. TCP/IP protocols:
   - HTTP "
   - DHCP "
   - FTP (File transfer) - not used
   - SMTP (Simple mail transfer protocol) - Used to send emails
   - POP3 & IMAC - Used to receive emails
   - SSH protocol - Used to log in others computer
   - VNC (Virtual network computing) - For graphical control)
   - Telnet - Terminal emulation that enables user to connect to a remote host / device using telnet client - Port number 23

- UDP: Stateless session (data) maybe lost

Program : whatsapp

- Send a message → Set up page
- Record video → Open camera } Process Thread

* SOCKETS :

- Used to send messages from one system to another
- It's the interface between process and internet

* IP address :

- Tells which device we are working with

* PORTS :

- Tells us which application we are working with

① Ephemeral ports :

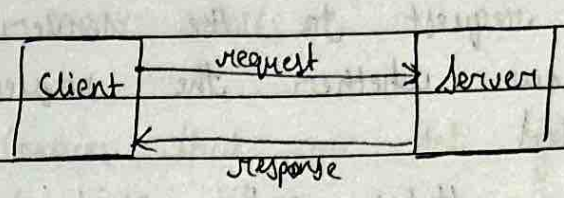
→ When the data needs to be sent to the desired application Eg: Google Chrome and if there are multiple tabs open, so how we will determine which tab to send data to.

→ Eg: Port 80 reserved for HTTP, the application will internally assign random port numbers

→ EP can exist on client side but on server side we have to know the PN

A.1 HTTP (Hypertext transfer protocol)

→ It is a client server protocol and it tells us



how you request data from the server and it also tells us how the server will send data to the client

→ when a client makes a request is known as HTTP request and when server sends data is known as server response

→ HTTP uses TCP (Transmission control protocol)

↳ stateless protocol ↳ Transport layer

↳ Application layer ↳ Thinks that new person logging in

* HTTP methods:

① GET

- You are requesting some data Eg: Give me webpage

② POST

- Client gives something to the server Eg: username, password

③ PUT

- Puts data at specific location

④ DELETE

- Delete data from server

Camlin Page
Date / /

* STATUS CODES :

- When you send request to the server you need a way to know whether the request was successful or failed.
- For this purpose, status codes exists
- 200 means request was successful
- 404 means you couldn't find it
- 400 means its a bad request
- 500 means there is an internal server error

* ERROR CODES :

- 1 - 100 → Informational category
- 2xx → Success
- 3xx → Redirecting
- 4xx → Client error
- 5xx → Server error

* COOKIES :

- It is a unique string
- It is a file which is stored in client's browser
- When you visit a site for the very first time, it will set a cookie
- After that whenever you make a new request, in that request's header a cookie will be sent.
- Then the server will know this request is coming from Sahil and he sent a cookie. Then the server will check its database and it will find the state

- Cookie also has expiration date

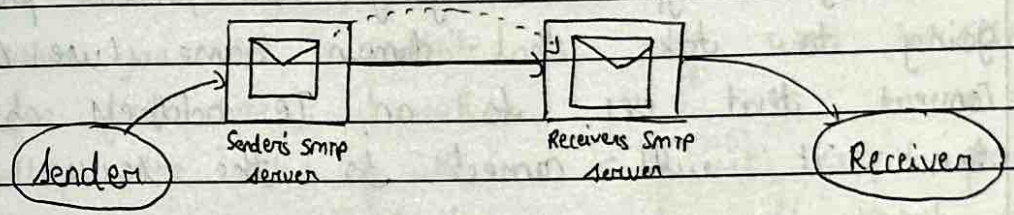
• THIRD PARTY COOKIES:

↳ Cookies that are set for the URL'S that you do not visit

* HOW EMAIL WORKS:

↳ SMTP (Simple mail transfer protocol)

↳ Mail transfer from Gmail to Yahoo



• To find the name and IP address of the SMTP server

~ nslookup -type=mx gmail.com

↓

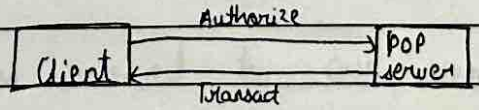
mail exchange - SMTP servers

↳ Downloading emails

↓

POP (Post Office Protocol)

- Client connects to the POP server using TCP (Port 110)
- Then client asks the server for new emails



↳ IMAP (Internet message access protocol)

- Allows us to view emails on multiple devices
- Emails will be kept on server
- Local copies will be available on various devices and we can sync all the folders

* DNS (Domain Name System):

- When you enter google.com, it will use DNS to find IP address of Google's server

- When you type www.google.com, HTTP protocol is going to take that domain name (use DNS) and convert that URL to an IP address and after this, it will connect to the server

- DNS is database service

- Data cannot be stored in a single database, hence data is divided into classes

Eg: mail . google . com

sub-domain ← second-lvl domain → top-lvl domain

① Root DNS servers → WB: root-servers.org

= .io = .org = .com } Top-level domain

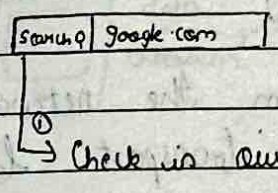
↓ ↓ ↓

student.io commclassroom.org google.com } second-level domain

a · com - commercial organization
b · edu - educational institutions
c · org - non-profit organizations
d · us/.uk - country specific

↳ Database is managed by ICANN (Internation cooperation for assigned names and numbers) - Top level domains

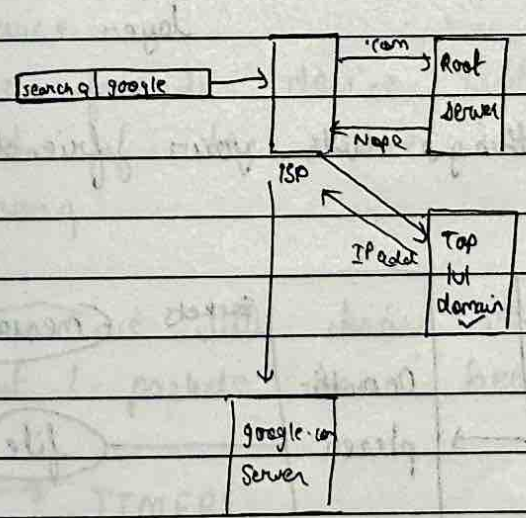
* What happens when we put Google.com



- When you visit a website for first time, it will store the value of its IP address in your local cache / local DB

- It's its not found in local cache, it will look in local DNS server of user i.e ISP

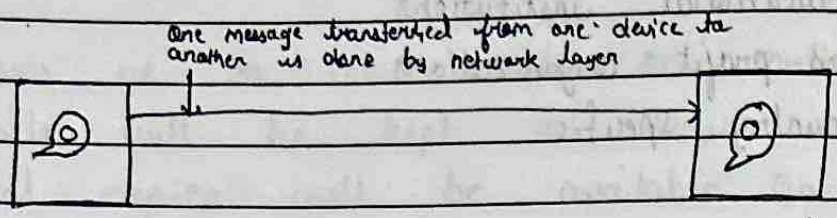
- If it doesn't find in DNS it will search in root server



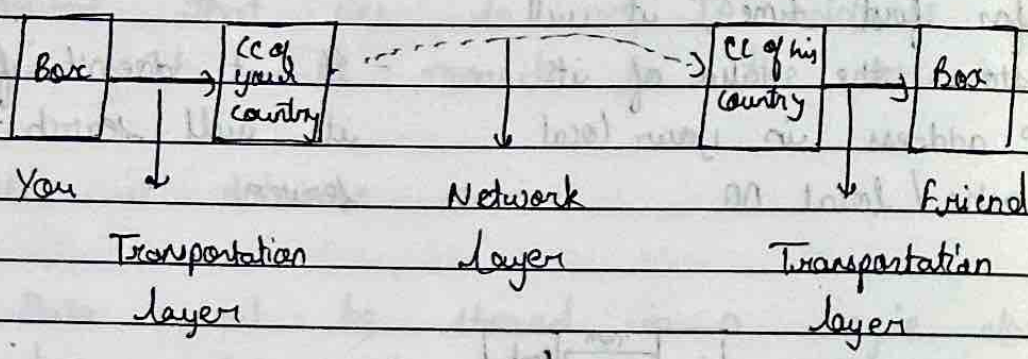
- We cannot buy the domain, we can only rent it

• Messages received by DNS service ~ dig google.com

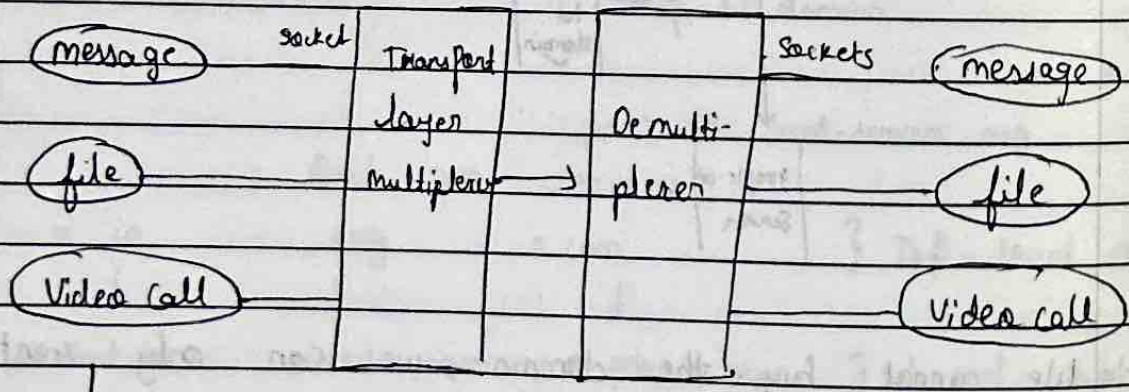
2] Transport layer



- Transport layer lies inside devices
- Role of transport layer is to take the information whatever friend is sending from network to the application
- Transportation of the data from the network to the applications is done by transport layer



* You are doing 3 things with your friend at same time

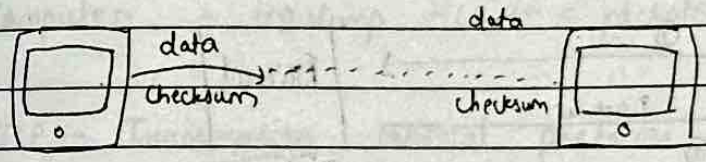


→ Data travel in packets

- Transport layer will attach these socket port numbers.
- TL also takes care of congestion control i.e traffic
- Congestion control algorithms built in TCP.

* Data may be corrupted in between while sending data from Network layer - NL. TCP takes care of it using checksums

* CHECKSUMS:

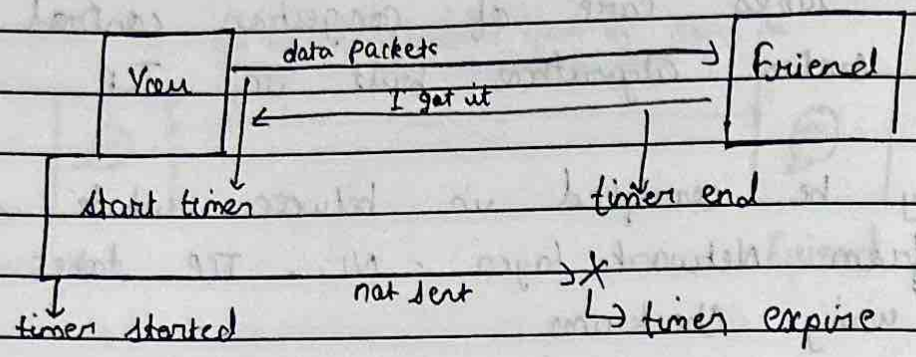


- Our device will calculate the data and will have a checksum
- When we send the data, checksum will be attached to it
- The ~~check~~ data you calculated on your device i.e checksum will be attached to your friends device
- The friends device will also calculate the checksum and if the value is different, something is gone wrong

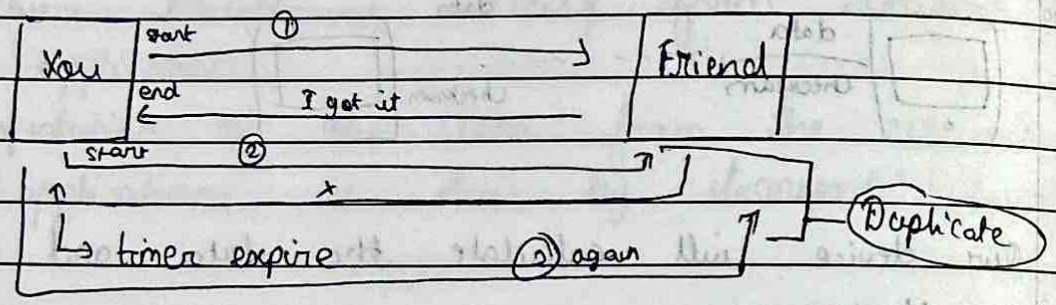
• How we will know that the data packets are lost / packets have been received by friend.

↳ TIMERS

* TIMERS :



- If friend didn't tell that I received data packet



Solve this using sequence number

* UDP (User - datagram protocol)
  - ↳ Data may not be delivered
  - ↳ Data may change
  - ↳ Data may not be in order

- Connectionless
- UDP uses checksums but it won't do anything about it

- UDP Packet :-

Header
- Source Port number (16)
- Destination " " (16)
- Length of data gram (16)
- Checksum (16)

- Data

- Total size = 2^16

- UDP is faster

- Use cases of UDP:

  - Video conferencing
  - DNS
  - Gaming

To see data packets coming in and out of your computer - tcpdump -cs ← s packets

** TCP - Transmission control protocol

- Transport layer protocol
- Application layer sends a lot of raw data
  TCP segments this data → divide in chunks, add headers, checksums etc
- It may also collect the data from Network layer
- Congestion control

- Takes care of:
  ↳ when data doesn't arrive
  ↳ Maintains order of data

- Features:
  ↳ Connection oriented (i.e connection needs to be established and then you can send data
  ↳ Error & Congestion control
  ↳ Full Duplex A ⇄ B

* How connection gets established betn 2 devices using TCP

3 way handshake :-

Client → Server

SYN (synchronization flag)
sequence no : 32 ← random

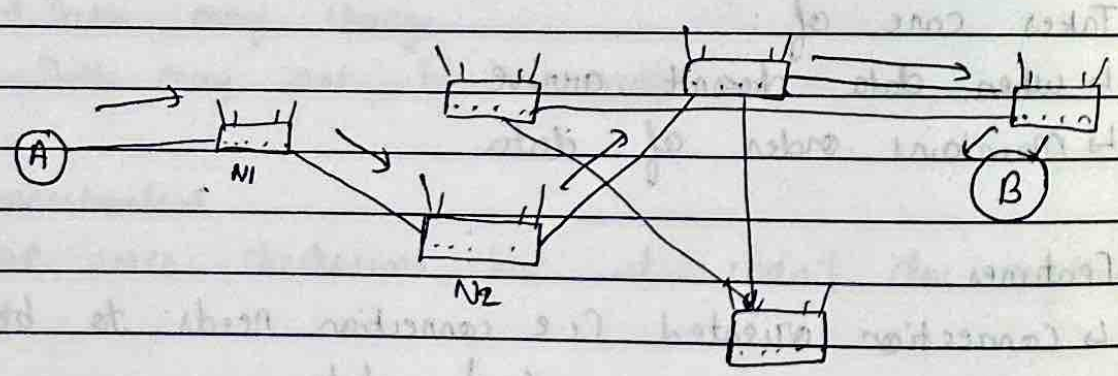
SYN
ACK (acknowledgement)
seq no : (maths on 32) → 60) 56
ACK no :- Prev. seq no +1

ACK
SN => 33
ACKNO : 57

* Network layer :

Transport layer $\xrightarrow{\text{data in}}$ segments
Network layer $\xrightarrow{\text{data in}}$ Packets
Data link $\xrightarrow{\text{data in}}$ Frames

- Here we work with routers



- when we send our data packet from A to N1 it will check its forwarding table (i.e I have receive this packet and this needs to be sent to destination which is around east so it will send the packet to N2

- This is known as hop-by-hop forwarding

$\underbrace{192.168.2}_{\text{Network address (Subnet ID)}}.\underbrace{30}_{\text{Device address (Host ID)}}$

- You are doing all this and what if someone adds a new router in their locality and who creates this routing tables. How router knows where to forward data packet. Who creates these table

→ CONTROL PLANE

Routers → Nodes
Links → Edges

① Static Routing:
- Adding addresses manually
- Manually add all these routes

② Dynamic routing:
- Evolved process

* Internet Protocol (IP):

- It is protocol that lies in the network layer

IP address version 4 (IPv4) → 32 bit no with 4-words
IPv6 → 128 bits

8 bits 8 bits 8 bits 8 bits
5. 6. 9. 14

00000101

- Hopping happens through ISP
- ISP has block of IP addresses

- Class of IP addresses

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:

- It will mask the network part of IP address and it will leave for us to use the host part Eg: Class C: 255.255.0.0 ← we can add here

- VARIABLE LENS SUBNETS:

- You can set your own length of the subnet network

- RESERVED ADDRESSES

$\overbrace{127}^{\text{reserved}}.\underbrace{0.0.0}_{\text{Anything}}/8$

Eg: localhost : 127.0.0.1
↳ loopback addresses

Camlin Page
Date / /

* PACKETS :

- Header is of 20 bytes → IPV, length, Identifications, flags, protocols, checksum, addresses, TTL → Time to live
  ↓
  It doesn't stay in network forever and after certain hops, it will be dropped

* IPV6 :

IPV4 : $2^{32} \approx 4.3$ billion IP address (unique)

IPV6 : $2^{32 \times 4} \approx 2^{128} = 3.4 \times 10^{38}$

↳ Cons :- Not Backward compatible
- ISP would have to shift lot of hardware work

* MIDDLE BOXES

↳ Device apart from your computer & router that also work with packets

Firewall

Global internet — Your trusted network

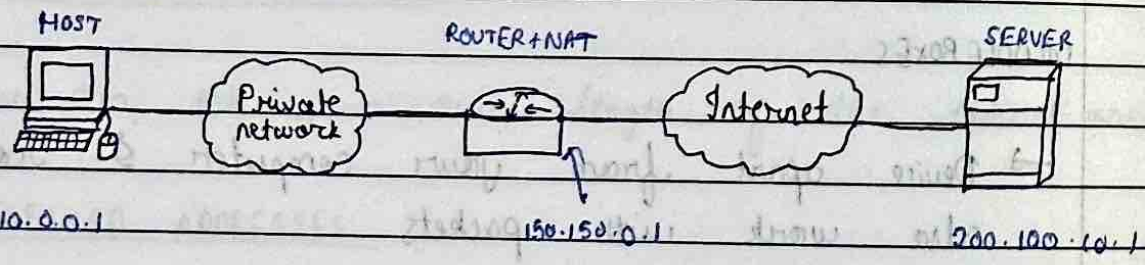
- Firewall provide filters i.e all the IP packets coming to the network can be filtered out based on the address

- Address
- Modify packets
- Port numbers
- Flags
- Protocols

Stateless firewall vs Stateful firewall
↳ more efficient

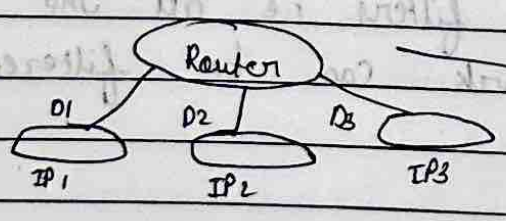
* NETWORK ADDRESS TRANSLATION (NAT) :

↳ It is a method of mapping an IP address space into another by modifying network address information in IP header of packets while they are in transit



* DATA LINK LAYER :

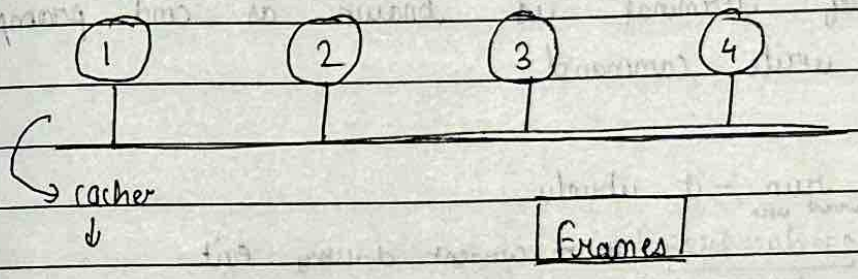
- The data packets that we receive from network layer, data link layer is responsible into share these packets over a physical link



→ IP address → ISP
Subnet
↓
DHCP

New device → DHCP server (To get IP address)
↓
Pool of IP address

- Many devices connected in LAN:
  ↳ Have data link layer address (MAC address)



ARP cache
If it (a) doesn't find anything in cache to connect to D4, it will ask D2 & D3 (Address resolution protocol)

↳ DLLA of sender & IP address of destination