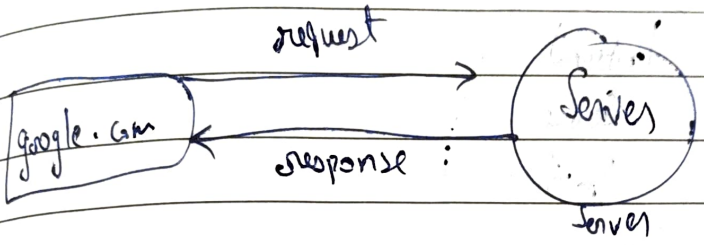
Computer Networking Course

Internet
→ Bunch of computers connected with each other globally Ex



Protocols
↳ Rules made by Internet Society

TCP (Transmission Control Protocol)
↳ when we don't want data to get corrupted on its way (all the data is req)

UDP (User Datagram Protocol)
↳ not necessary to receive all the data
Ex: Video conferencing

HTTP (Hyper Text Transfer Protocol)
↳ Used by WWW ...
→ server client data transfer

Packets
→ when you req data, you get back data in individual packets.

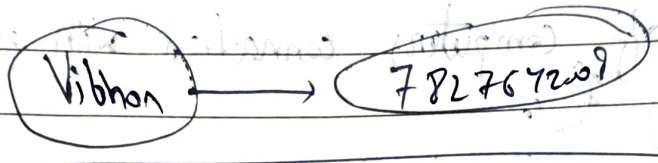
Workmate

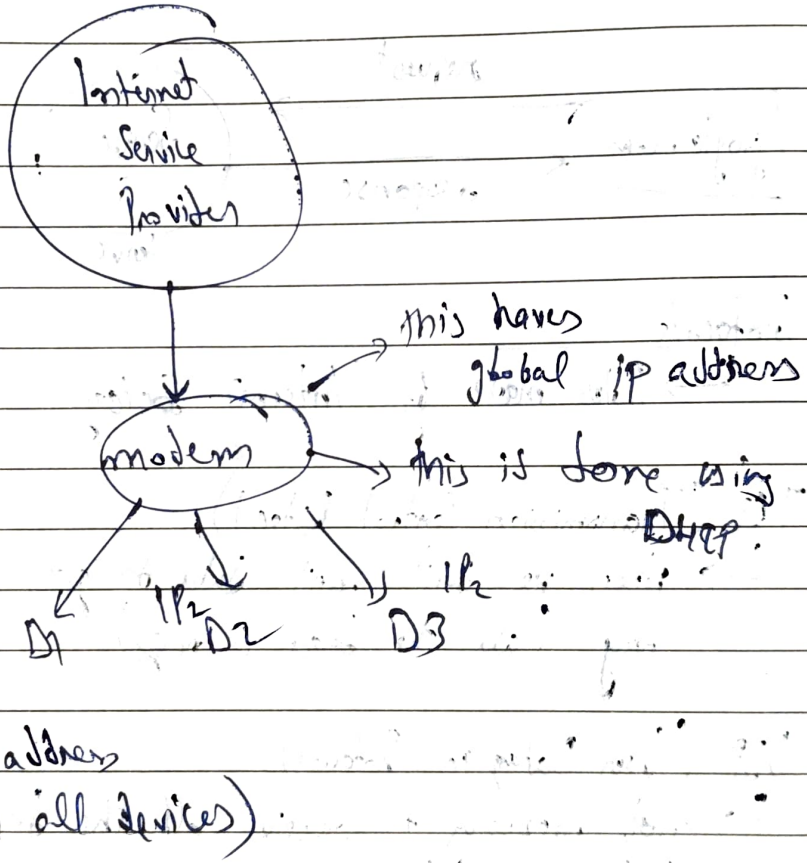
Date

Page No

IP address

→ Its like a phonebook





→ modem decides who sent the request using NAT (Network Access Translation)

→ computer decides which app to send data using Port Number

↓

identifies the application

Port Number → 16 bit number

Total = 2^{16} ≈ 65,000

HTTP stuff happens on Port 80

MongoDB 27017

0 – 1023 reserved ports

1024 – 49152 applications

remaining → can be used

Internet Speed (1000 000) bits

→ 1 mbps = 1 mega bits you transfer per sec

1 gbps = 10^9 bits / sec

1 Kbps = 1000 bits/sec

Download speed

→ getting data from 1 computer to other computer

Upload speed

→ sending

Workmate

Date | Page No

How communication b/w computers happens ?

1) Guided Way
connected with wires
Ex: ethernet

2) Unguided Way
~~Wifi~~
no path defined
Ex: Wifi

→ Every country is connected with wires underwater (Physically using optical fibre ...)
Check (submarine cable map)

LAN (Local Area Network)
Small house / Office
Using: Ethernet, wifi

MAN (Metropolitan Area Network)
→ Across a city

WAN (Wide Area Network)
→ Across countries
Using optical fibre cables

INTERNET

→ SONET (Synchronous Optical Networking)
→ Data using optical fibre cables

→ Frame Relay
→ Local area network to wide area

MODEM
→ Converts digital signal to analog signal

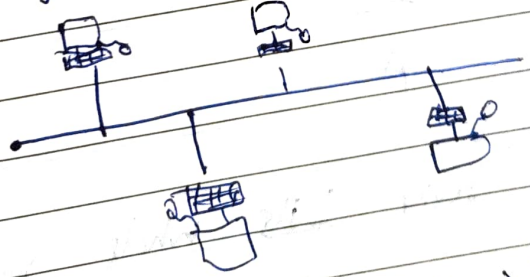
Ex: Computer data to electrical ~~data~~ signal across (Image)

Router
→ Routes data packets based on there IP address

ISP (Internet Service Provider)
→ ISP are connected to larger ISP
Ex: Tier 2 service provider (Airtel)
↓
Tier 1 service provider (TATA)

Topologies
→ Various ways computers are connected

① BUS
→ Every computer connected to a backbone



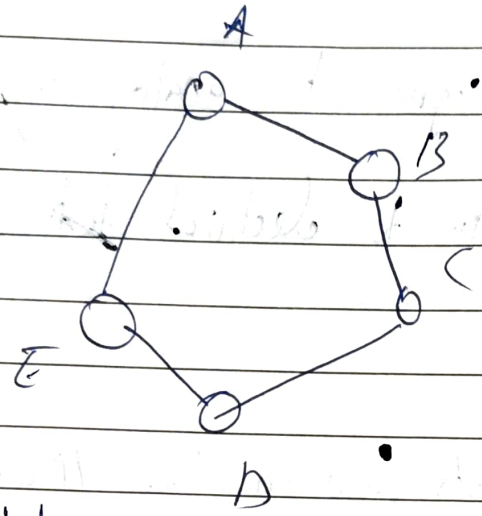
→ only one person can send data at a time

192.168.2.1

~~IP~~ data

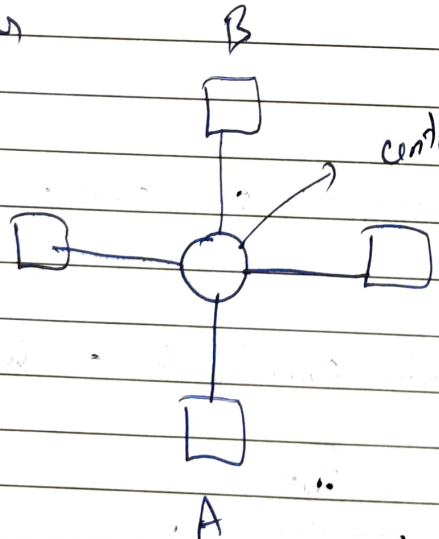
MAC addresses

(11) Ring



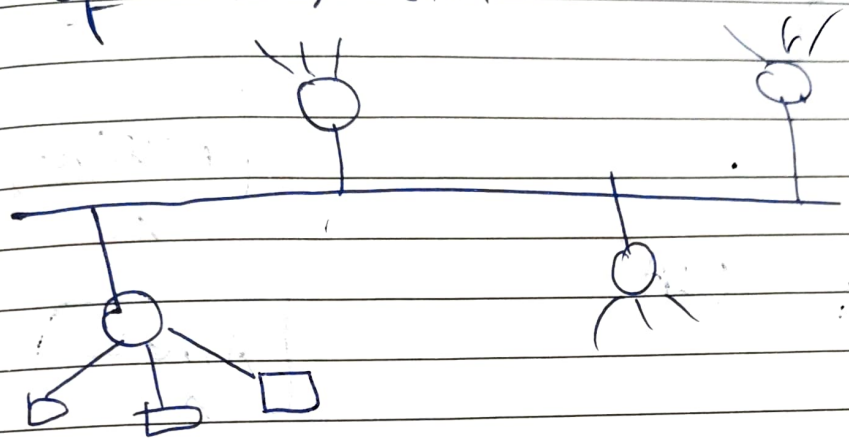
Limitations
→ If cable breaks you can't send data.
→ Unnecessary calls are made
Ex: Sending data A → D, it had to go through B & C

(3) ~~Star~~ Star



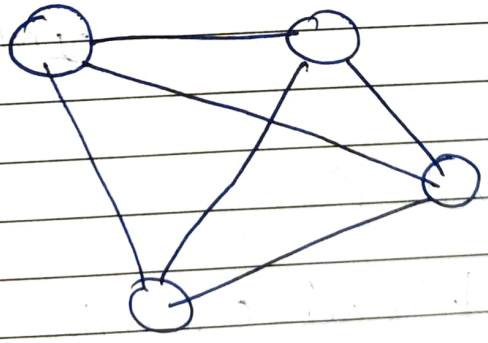
→ if centre device fails whole system goes down

④ Tree
Comb^{n} of bus & star



⑤ Mesh

Every single computer connected to every single ~~comptr~~ computer



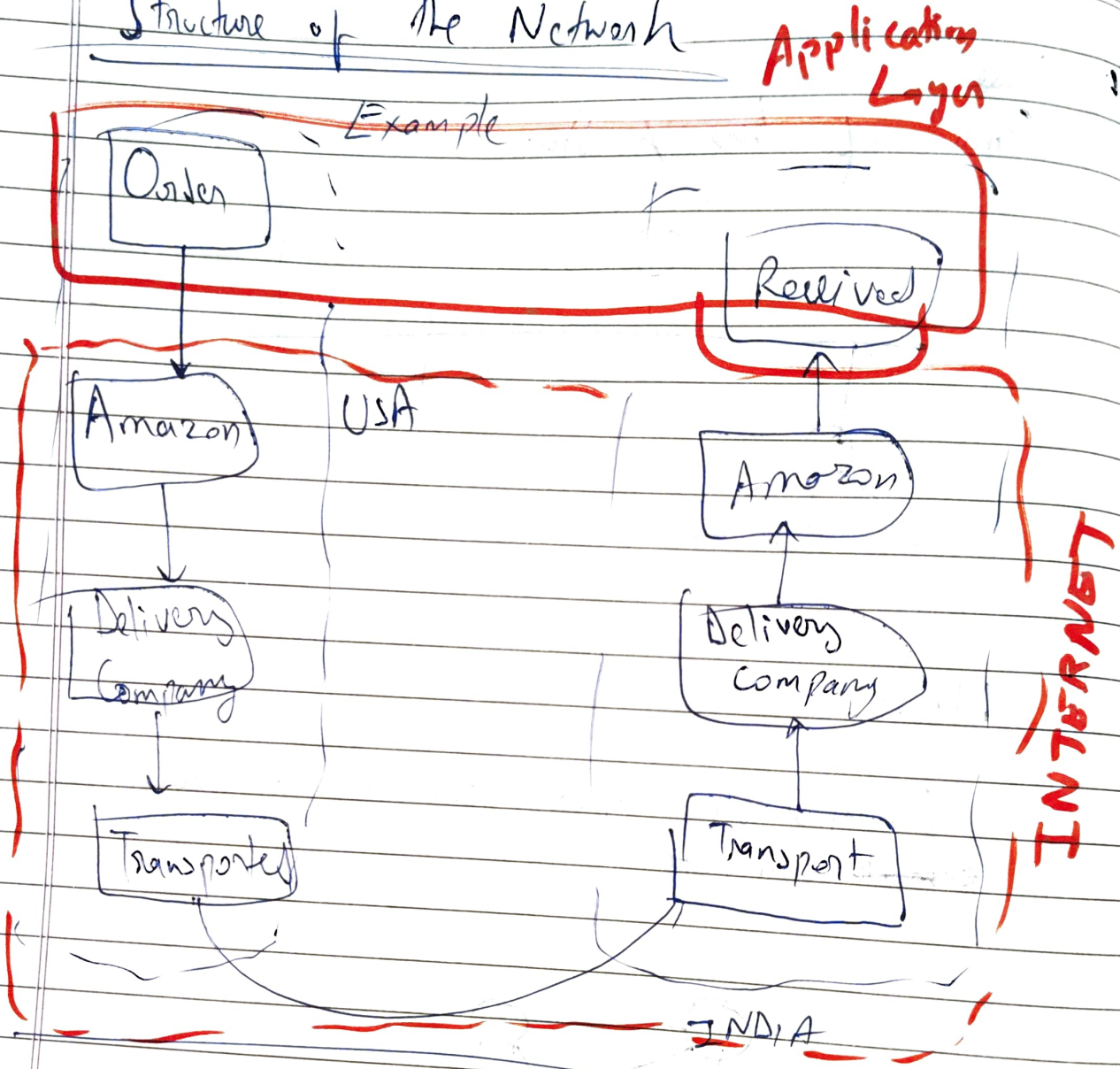
→ Limitation
→ Expensive → Scalability issue

Workmate

Date

Page No

Structure of the Network



OSI Model

→ Open System Interconnection Model

→ Server communicate with each other

→ ~~7 layers~~ 7 layers in OSI Model

Imp

① Application ② Presentation ③ Session
④ Transport ⑤ Network ⑥ Data Link
⑦ Physical

Application Layer

→ Implemented in software, contains application

→ your message is sent to presentation layer

Presentation Layer

→ Converts the data into binary format

ASCII → ~~EBC~~ EBCDIC

Translation

first data is encripted & also provides abstraction

→ goes to session layer

Session Layer

→ Helps in setting up and managing the connection

Does authentication then authorisation

→ Session layer assumes transport layer ~~will~~ to if i

computers

IP address

rect destination

destination

Workmate

Date | Page No

establishes session transport layer will do the transportation

Transport Layer

→ Does it in 3 ways

↳ Segmentation

→ Data received from session will converted into small layer called segments.

→ Every ~~session will~~ segment will contain source & destination port number and sequence number

↳ helps to reassemble segments in correct order

→ Flow control

controls the amount of data that is being transport

→ Error control

will be done later

Network Layer

→ Works on transmission of the received dataset from one computer to another that is located in a different network

→ Router lives over here

From Application to Transport network, we were talking about our own network but from here

we are communicating with other computers

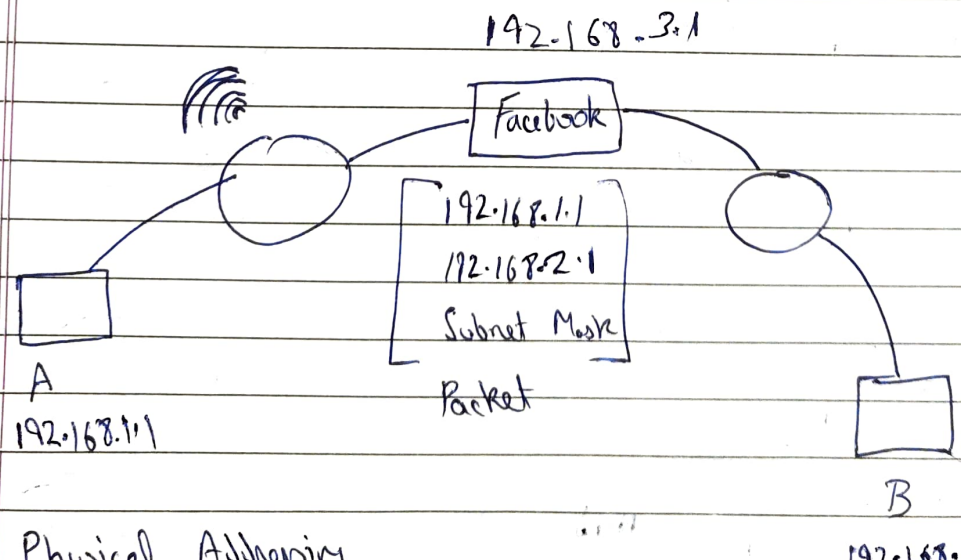
Func^n of network layer
→ Logical addressing
IP addressing done in network layer

→ Network layer assigns ~~network~~ senders & receivers IP address to every segment and forms IP packet.
→ It that every data packet reaches correct destination

→ It also performs routing
→ moving one ~~data~~ packet from source to destination

→ Also does load balancing

Data Link Layer



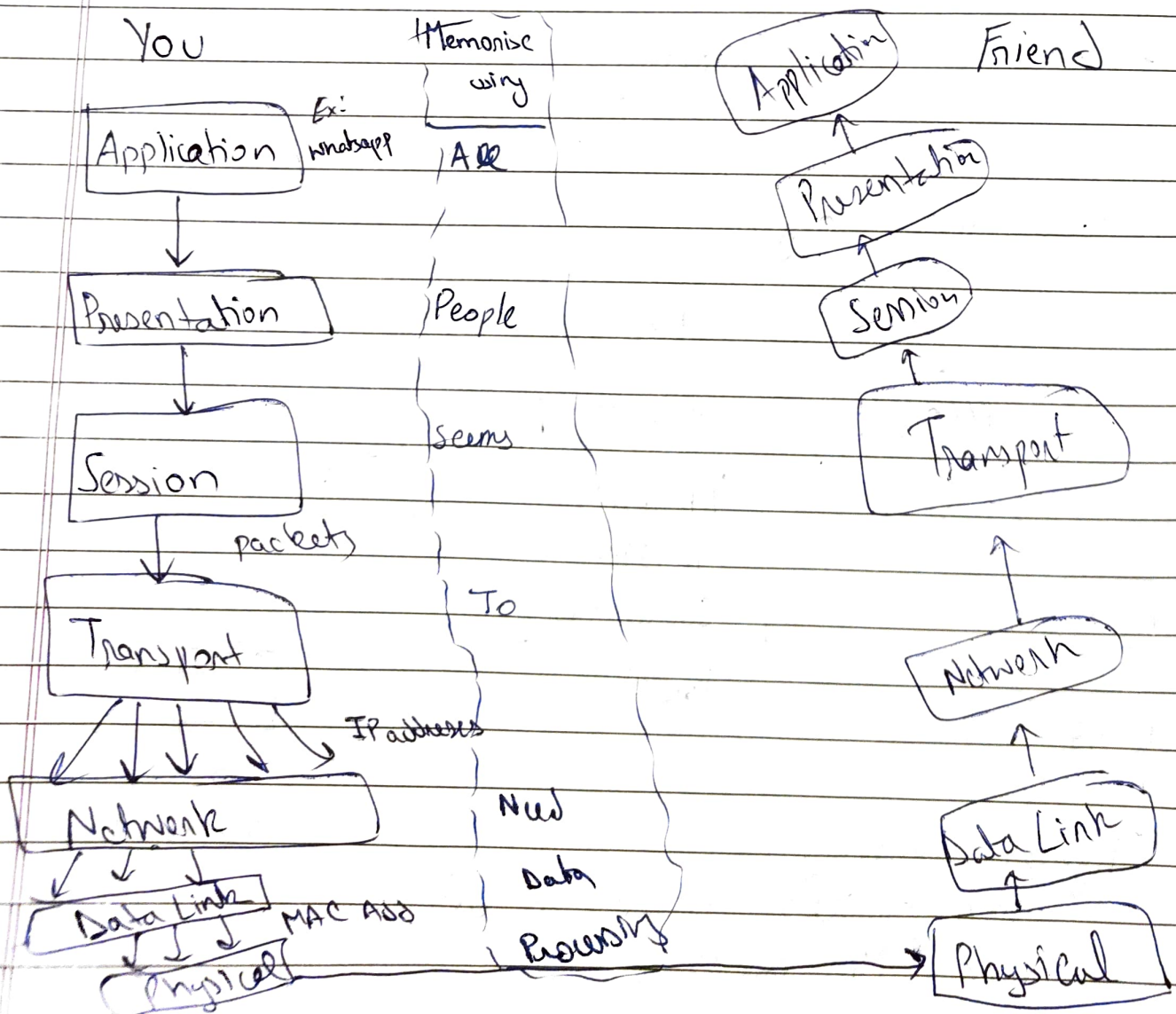
Physical Addressing
Which application to send the data
called - MAC addresses

Basically data link layers adds MAC addresses in a packet called Frame and transports that frame.

WILL BE COVERED LATER

Physical Layer

Data got from above in from 0 & 1, will be converted in maybe electrical or light (optical) or radio signal



Another Model

TCP/IP Model

→ Internet Protocol Suit
  → 5 Layers
    (I) Application (II) Transport (III) Network
    (IV) Data Link (V) Physical

→ Used Practically

DEEP DIVE
IN
THESE LAYERS

Workmate

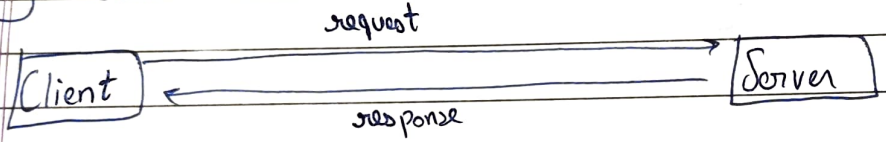
Date
Page No

~~(1) Application Layer~~

(1) Application Layer

→ User interaction like apps, whatsapp, browsers...
→ Lies in our devices
What we learning?
→ Protocols
→ Client - Server Architecture

(1) Client - Server Architecture



Server → System controls the website you hosting

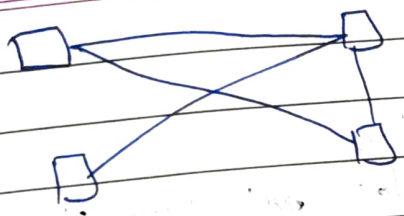
Collection of servers in big companies is known as Data Center

Ping - ~~Time taken~~ Round trip time from host to the client's computer
(You can't reduce ping time)

Pier to Pier (P to P) Architecture

→ Devices connected with each other (no data center)

→ Adv
→ Scalled rapidly
→ Decentralized network



Every computer here can be termed as a client & server as well
Ex: Bittorrent (Seeding)

Protocols

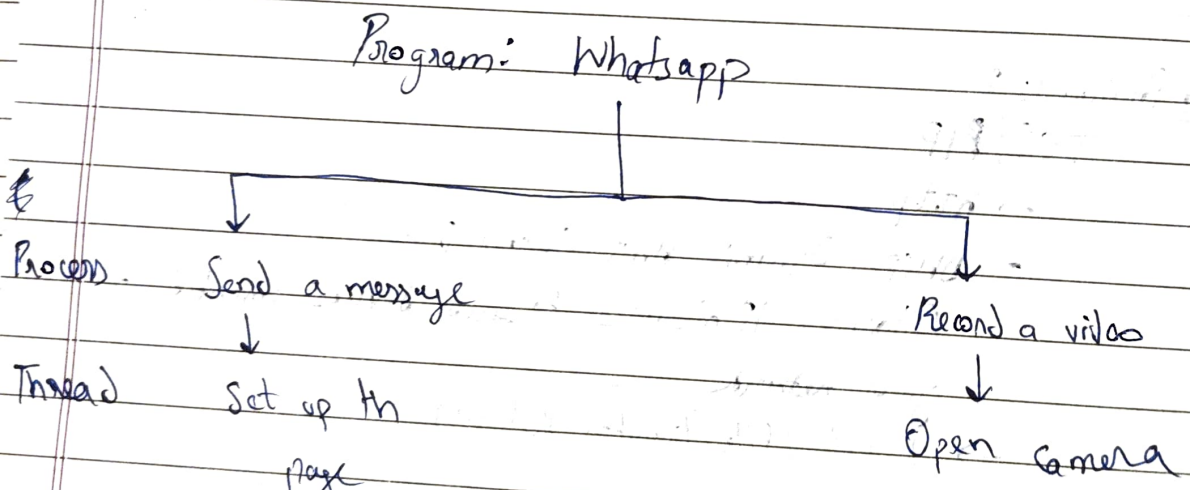
→ Web Protocols:
TCP/IP
→ http
→ DHCP (Dynamic Host Control Protocol)
Allocates IP addresses to devices connected to your network
→ FTP (File Transfer Protocol)
→ SMTP (Simple Mail Transfer Protocol)
→ to send emails
→ POP3 & IMAC
→ to receive emails
→ SSH
to login into terminal of someone device
→ VNC (Virtual Network Computing)
Graphical control

Telnet
→ Terminal emulation that enables user to connect to remote host or device using Telnet client:
Port 23

Thread → lighter version a process
1 process can have multiple running process
Thread vs process: thread only does 1 single job

→ UDP
- stateless connection
- data may be lost during connection

Q→ How these application talk to each other?
Ans→ Lets take Whatsapp



Socket
→ Used when you need to send a message from one system to other system.
→ It a interface b/w the processes & internet.
→ Not a hardware, more like a software

Port:
→ How to determine in where the data should go inside the application, ex: data received need to go to which chrome tab, For that we use Ephemeral Ports.

Workmate
Date
Page No

Workmate
Date
Page No

Ephemeral Port

→ When multiple ~~ap~~ same application are running, it assigns random port number. Once process is done it free's.

→ They exist on the client side but servers side you have to know the port number.

HTTP

→ Client-Server protocol which tells client how you request data from the server & tells server how you sends the data to the client.
→ When client makes req it called HTTP req
→ " server sends back response its called HTTP response.
→ Its a application layer protocol also requires some transport layer protocol.
→ Ex: HTTP.get(), HTTP.post()...
→ HTTP uses TCP (Transmission Control Protocol)
↳ stateless protocol (server will not store any client info. by default)

HTTP → application Layer
TCP → transport Layer

Date
Page No

HTTP Methods

→ GET
* requesting the data

→ Post
* giving some data

→ Put
* Puts data at specific location

→ Delete

Error/Status Codes

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

Cookies

→ Unique String stored in client browser
→ When you visit a website first time it will set a cookie
→ Then you revisit cookie will be sent to website

Workmate

Date

Page No

~~How it works?~~

Third Party Cookies

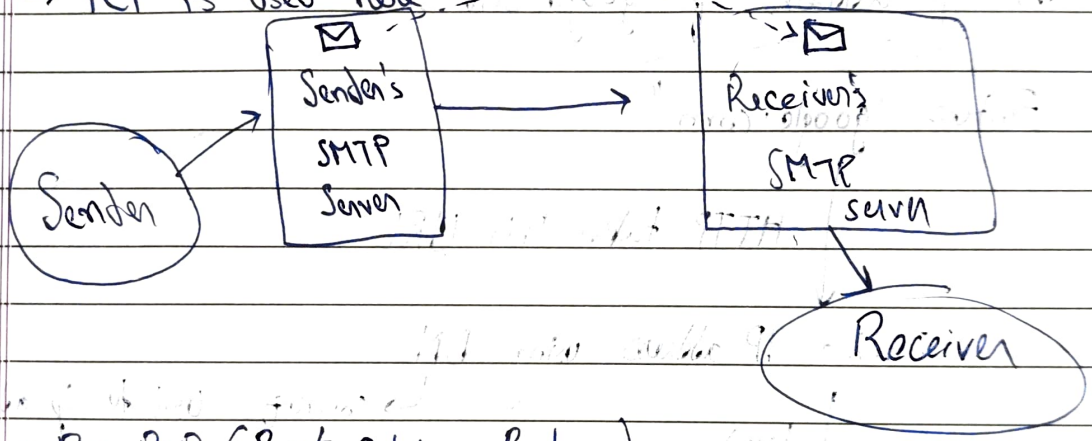
→ Cookies that are set for the urls that you don't visit.

How Email Works?

→ SMTP (Simple Mail Transfer Protocols)

→ for receiving - use POP3.

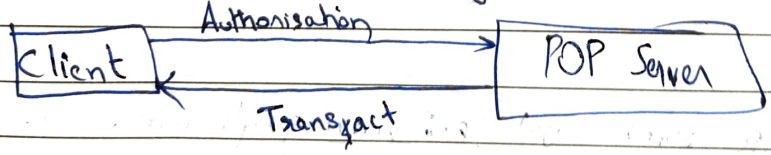
→ TCP is used here



→ ~~Pop~~ POP (Post Office Protocol)

client connects to the POP server, does all the authorisation & client asks for all the email.

: Used for downloading emails



(ISP) ← 5 : IP add 4 → [TLD]

→ IMAP (Internet Message Access Protocol)
Allows to view email on multiple devices
If you delete from any device email will be removed from the server

DNS

→ Domain Name ~~Server~~ System
→ ~~DNS~~ Domain Name are mapped to IP addresses
→ When you enter "google.com" it will use DNS to find the ip address of google servers.

Enter "google.com"
↓
HTTP takes this URL
↓
Finding IP address using DNS → converts URL to ip address
↓ found
129.0.10.1
↓
Connected to google server

When you type some url (ex: www.google.com), it will go to some database, So to reduce the traffic and backups these urls in these databases divided in various classes & domains.

Second level domain SLD

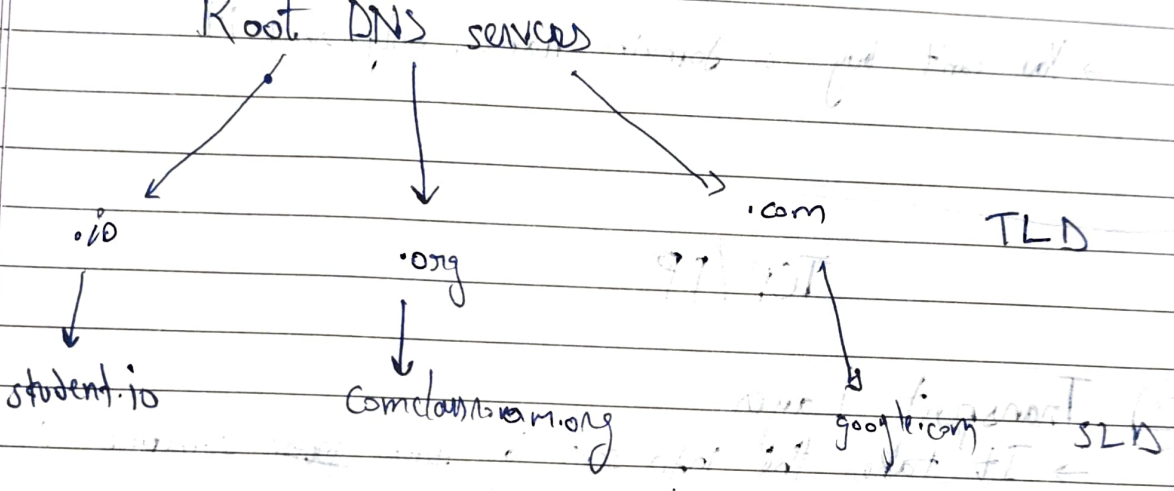
mail.google.com

sub domain

top level domain (TLD)

→ There are multiple categories for these 3 databases.

~~* Top level domain~~

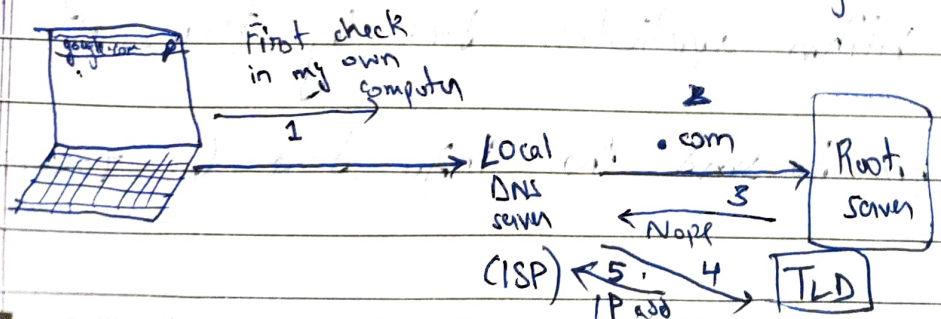


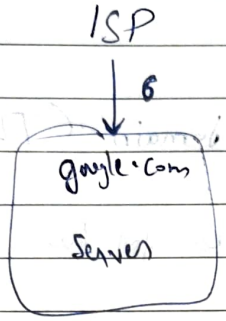
TLD

→ ~~for~~ organisation type specific

* .com (commercial)
* .edu (education)
* .org (non profit org)
* .in, .uk, .us (country specific)

→ managed by ICAN (Internet Corp For Assigned Names & Number)



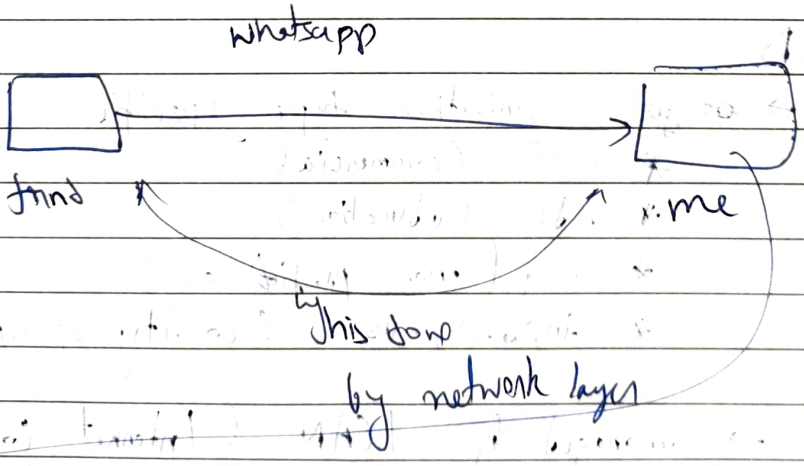


→ You can't buy a domain name, only rent it.

TCP/IP

② Transport Layer

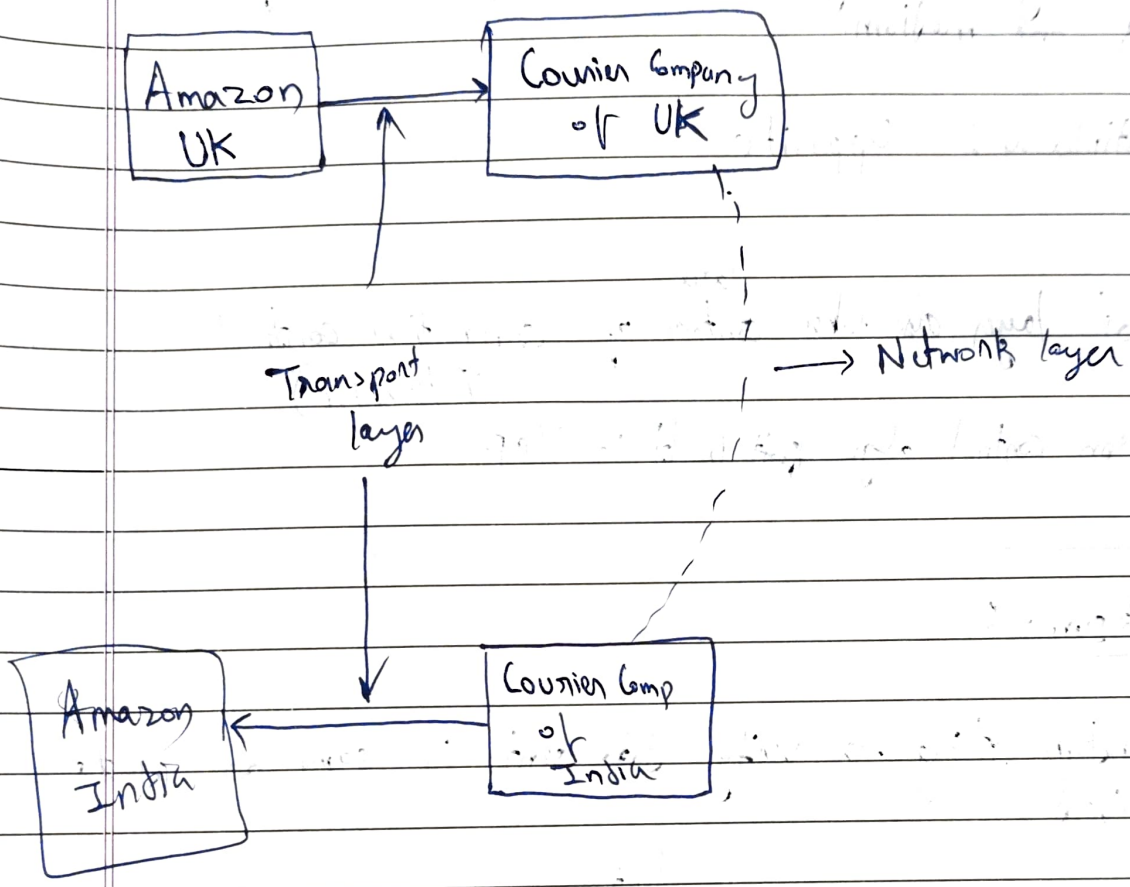
→ It takes the info sent by ~~you~~ ~~food~~ someone from network layer to the application



→ within application done by transport Layer

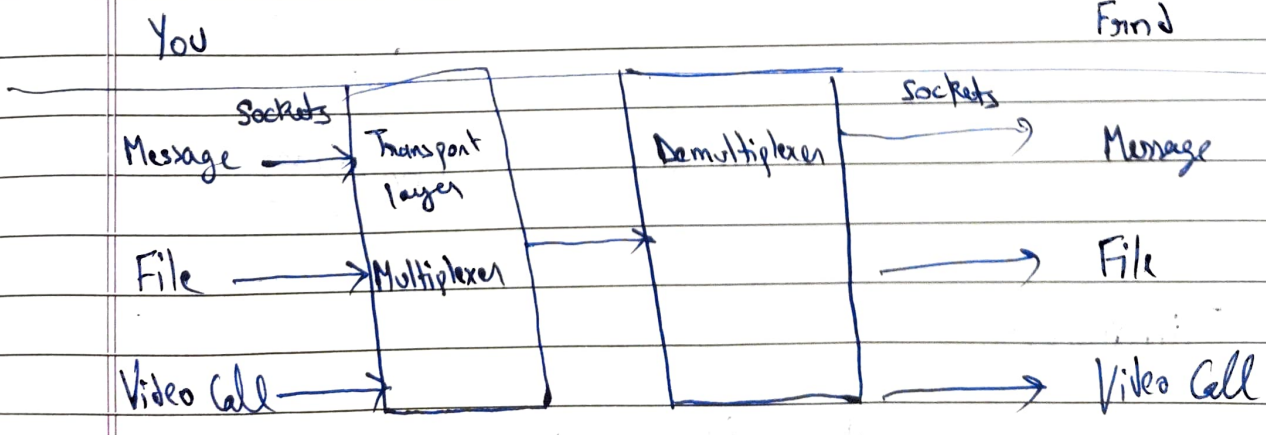
→ which app to send the info done by transport layer

Real life example



Transport layer protocols → TCP & UDP

Ex:



Workmate

Date

Page No

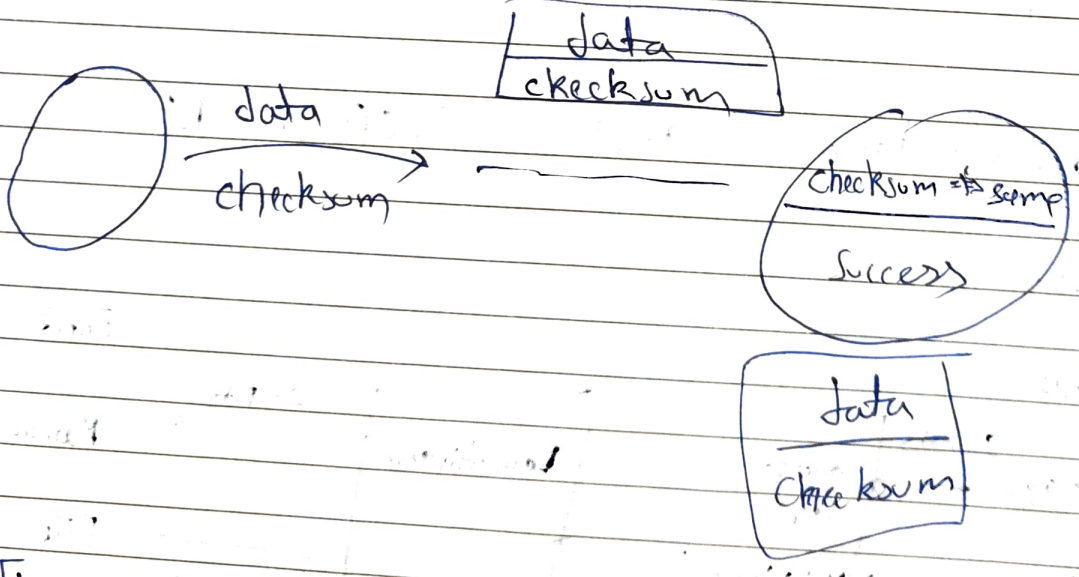
Multiplexing: Allowing messages or something to lot of destination using one medium.

Demultiplexing: Opposite

* Transport layer also takes ~~control~~ care of congestion control (traffic)
* Congestion control algo ~~fast~~ built in TCP

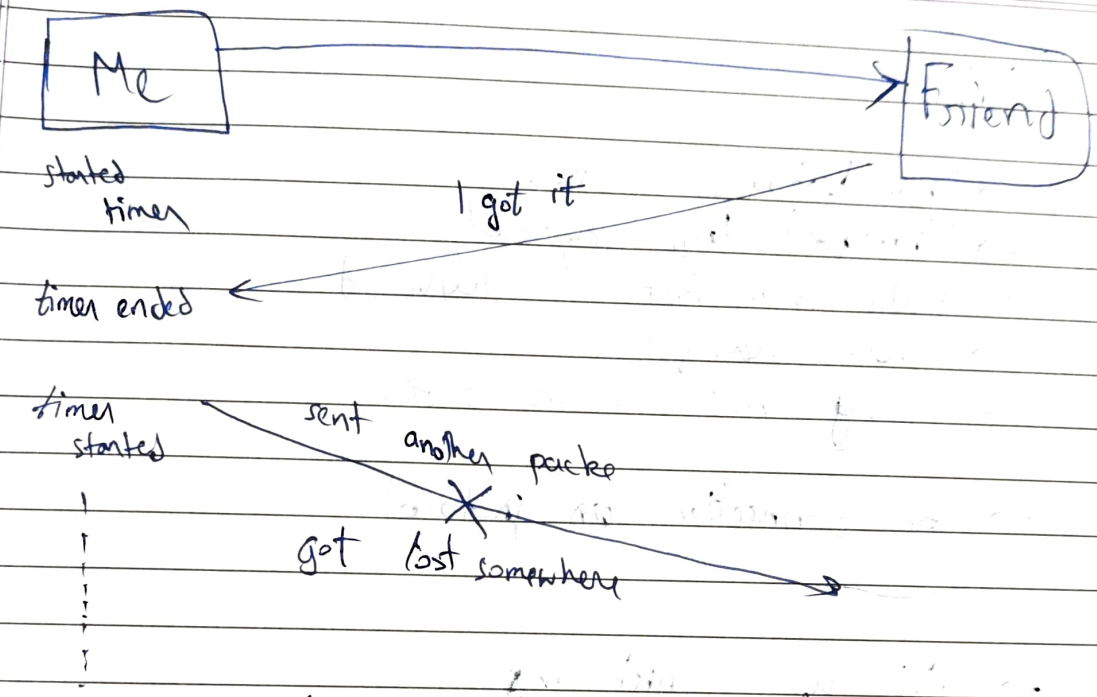
Checksums:

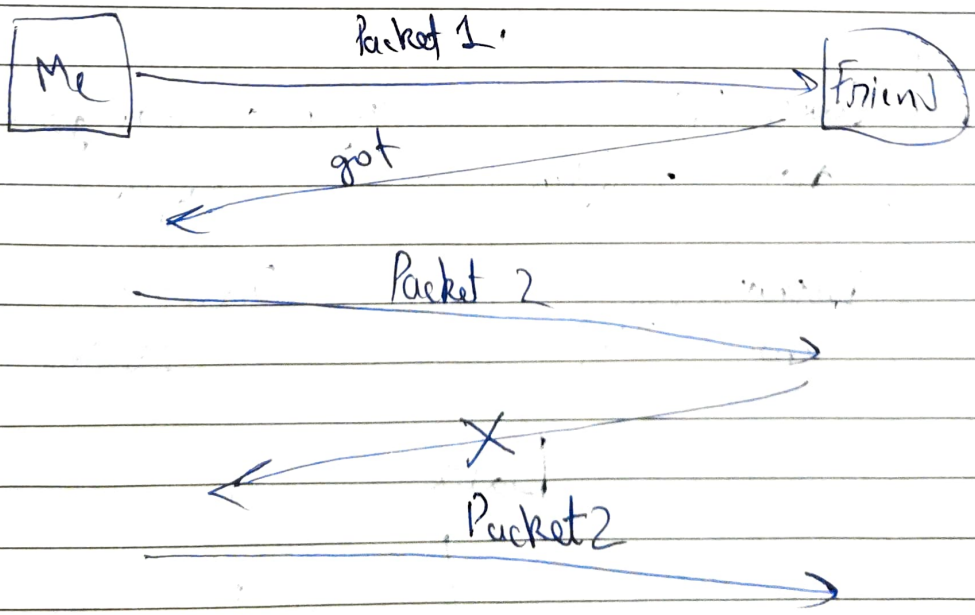
→ Random string to verify data sent is same as data received



Timmers:

→ to confirm the data packet is received to the destination





→ This problem is handled using sequence numbers