Computer Network

- Interprocess communication done by OS (kernel)
  - ↳ to share the data

functionalities

- Mandatory
  1) Error control
  2) flow control
  3) MUX, DEMUX
  - ↳ for all these there is a standard model → OSI
- Optional
  - encryption/decryption

http
- ↳ direct send the data

https
- ↳ send data encrypt

COMPUTER NETWORKS

OSI Model

- Application
- Presentation
- Session
- Transport
- Network
- Data link
- Physical link

Types of Networks

1. PAN → Personal Area Network
   - 1 - 1000 Meter
2. LAN - Local Area Network
   - upto 2 Km
3. CAN - Campus Area Network
   - 1 - 5 Km
4. MAN - Metropolitan Area Network
   - 5 - 50 Km
5. WAN - Wide Area Network
   - Above 50 Km

TCP/IP Protocol suit
OR
Internet Protocol
↳ Developed by ARPANET funded by DARPA

OPen System Interconnection (OSI) → (user interaction)

Application layer	Application layer	Application layer	Process to process
	Presentation layer		→ (decryp / encryp)
	Session layer		→ (authentication / authorization / session magn)
Transport layer	Transport layer (segment)	Transport layer	Host to Host → (segmentation / flow / error)
Network layer	Network layer (Packet)	Internet layer	Source to Destination
Data link layer	Data link layer (frame)	Network Access layer	Node to Node → (Physical addressing / error detection)
Physical layer	Physical layer (BITS)		UDP > TCP (no feedback)

5 layer — OSI — 4 layer — Practical

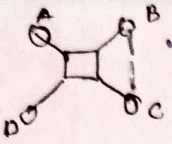
(path determi / Routing / logical addressing)

(1) Physical layer and its functionalities:

- Cables and connectors
- Physical topology → Mesh, Star, Bus, Ring, Hybrid
  - No. of Cables → n_{C_2}
  - No. of Port → $n(n-1)$
  - Reliability → Best
  - Cost → ↑
  - Security → High
- Hardwares (Repeaters, Hubs)
- Transmission mode.
- Multiplexing
- Encoding

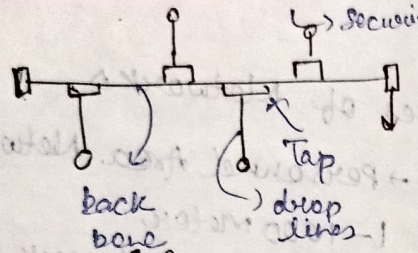
Point to Point communication

Star (Hub)



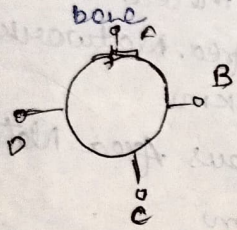
- no. of cable = n
- Port = n
- Reliability = X
- Cost < Mesh
- Security → X

Bus

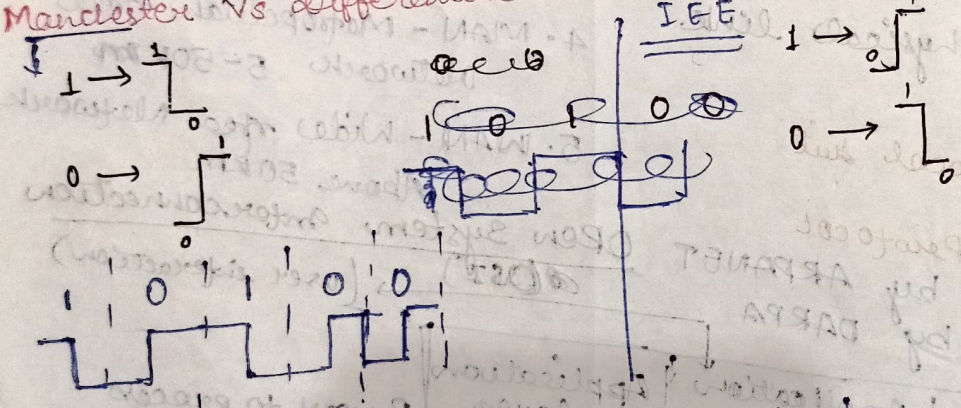


- no. of lines = n + 1
- no. of ports = n
- Reliability = X
- security = X

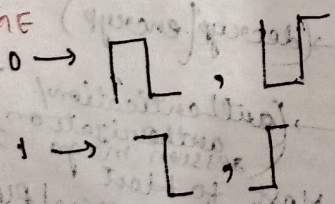
Ring

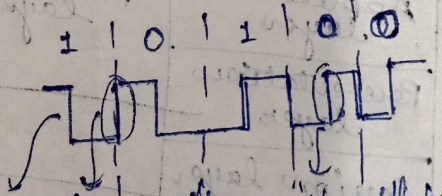


Manchester Vs Differential Manchester Encoding



DME





for 1st we can draw any of these

always there should be an edge to choose accordingly

Various Devices in Computer Networks

1) Cables
2) Repeaters
3) Hubs
} H/W

4) Bridges
5) Switches
6) Routers
} → H/W + S/W

7) Gateway
8) IDS
9) Firewall
} → Security

10) Modem

Base Band → only one data can transmit at a point
Broad Band

1) Cables → Unshielded twisted pair cable → 10 Base T, 100 Base T
- 10 Base T → 10 Mbps
- 100 Base T → 100 meters

- Coaxial cable → 100 Base 2 ↘ 200m, 10 Base 5 ↘ 500m
- Fibre optic → 100 Base FX → fibre channel → 2km

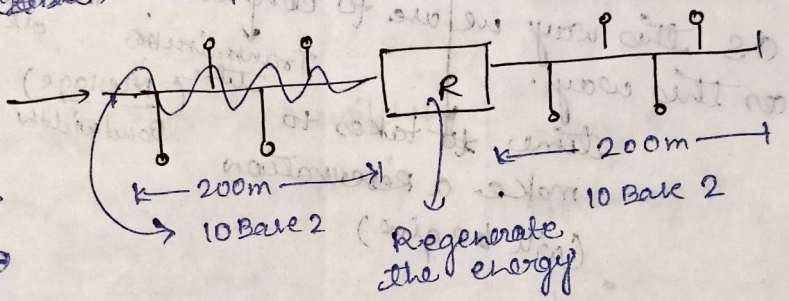
Used in Ethernet LAN

Use in Physical Layer

Attenuation → for n devices → n no. of max collisions are possible.

Cables are pure Hardware.

2) Repeaters

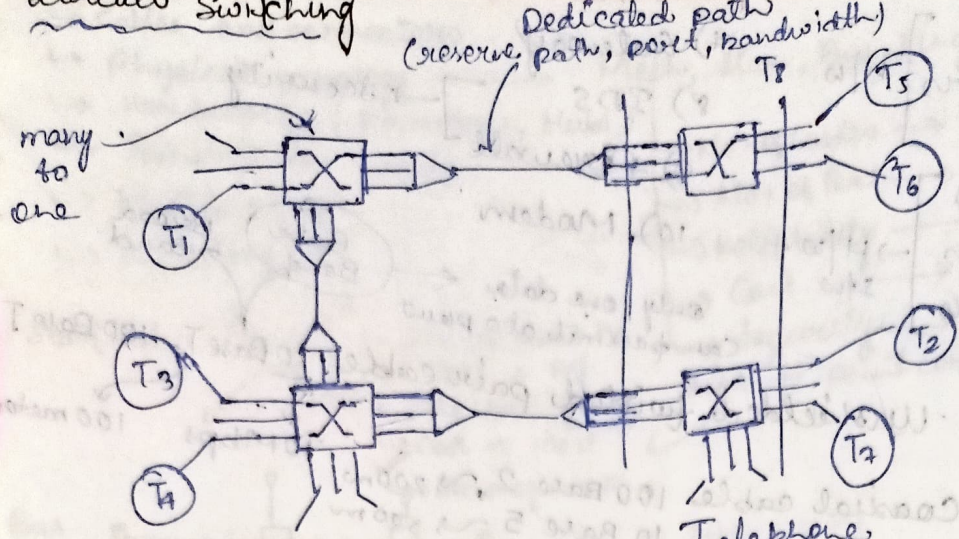


2 port device

~~2 port~~
forwarding
No filtering
Collision Domain
↳ n → no. of devices

↑ generally designed for telephone network

Circuit Switching



- Physical layer
- Contiguous flow
- No headers
- Efficiency less
- Delay less
- Reservation

Total time = Setup time + T.T + PD + Tear down time

- Before OS this way we are to connect devices this way.
- Setup time: time takes to make a Reservation (will be give)
- T.T: Transmission time (Message / Bandwidth)
- PD: propagation delay (dist / speed)
- Tear down time: time takes to free the resources after using

Packet Switching

- we transmit the data in form of packets.
- Application layer → T.L → N.L → Physical
- Network layer, Data link layer divides data into packets

Datagram → work on Network layer

Virtual → work on Data link layer

work as
1) Data link and Network layers
2) store and forward
3) Pipelining used
4) Efficiency ↑
5) Delay ↑

coz of this

can increase η, delay and no. of packets

Total time = n(T.T) + PD

n: no. of switches used inside network

(Packet switching)

Datagram Switching

- Connectionless
- No Reservation
- Out of order
- High overhead ↑
- Packet lost ↑
- Used in Internet IP network

no reservation (like B.D, Port)

coz all packets are using diff path (header) use will be high

Virtual Circuit

Connection Oriented (~~Reserve~~ Before)

Reservation → (like circuit switching)

Same order → (packets with arrive in order → it use routing table)

because path is reserved

less overhead → (header → to store the details about path) their is one path so less overhead

Packet lost → coz all packets are using same path

X.25, ATM

Cost ↑

Efficiency ↓

Delay ↓

coz of Reservation

Messages switching

↳ slightly diff than ckt switching

Predecessor of packet switching

store and forward

Hop by Hop delivery

Delay ↑

efficiency ↑

(2) Data link layer → within a network we can provide comm.

frame: header | data | tail

(1) hop to hop delivery
or
node to node

(2) flow control (speed of message)
- stop & wait S/W (help for node to node, diff than transport layer → source to destination)
- Go back N GBN
- selective reppate SR

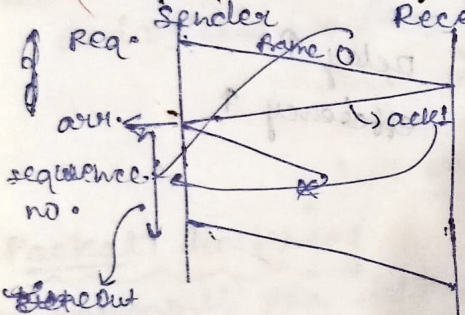
(3) Error control (hop to hop)
- CRC (in data link layer), checksum (Transport layer), Parity

(4) Access control
- CSMA/CD, Aloha

(5) Physical address (MAC)

Stop & Wait ARQ protocol

window size (1) for sender
window size (1) for receiver
= 2 sequence no.
0 1



- less efficiency
- $\eta = \frac{1}{1+2x}$ $x = \frac{T_p}{T_t}$
- Avil. seq. no. → 2

no. of bits
seq. no. ↓

(2) Go-back-N ARQ (Arithmetic Repeat Request)

- Out of order
- window size of sender $2^m - 1$
- window size of Receiver 1

(3) Selective Repeat ARQ

- window size of sender 2^{m-1}
- window size of Receiver 2^{m-1}

Sliding window is implemented selective reject ARQ, go-back-n-ARQ → error control

- Multiple frames
- $\eta = \frac{2^{m-1} \times 1}{1+2x}$
- Commulative & Independent ACK
- Out of order can be accepted
- Retransmission = 1

- only inorder packet will be accepted
- $\eta = \frac{(2^m - 1) \times 1}{1+2x}$
- commulative ACK
- Retransmission $= 2^m - 1$

Framing in Data link layer

- A frame in a character-oriented protocol

Flag A C B Flag → data

Flag A B esc Flag C Flag → data

- Byte stuffing & unstuffing → process of adding 1 extra byte whenever there is a flag or escape char. in the text

Redundancy → extra

Error detection and correction

- Single bit error
- Burst error

Detection
- Simple Parity (Even, ODD)
- 2D parity check
- Checksum
- CRC (Cyclic Redundancy check)

Correction
- Hamming Codes

*. If the B.D of channel is 1 Kbps then how much duration the error should last?

A —— 1Kbps —— B

→ 1000 bits — 1 sec

1 bit — $\frac{1}{1000}$ sec

1. Single Parity
   - (m + 1) bits (one extra bit)
   - Even parity → (no. of '1' even)
   - can detect all single bit errors in code word
   - can detect all odd no. of errors also

- Hamming distance → XOR of two o even parity no. of
  - no. of ones is hamming distance = d
  - can detect (d-1) bits of error.

2. Cyclic Redundancy check (CRC)
   - Based on binary division

total bits = (m+r)

→ Polynomial should not be div. by (x).

→ also not with (x+1)

→ can detect all odd errors, single bit, burst error of length equal to polynomial degree.

(= Hamming code for error detection & correction

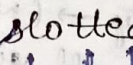
Multiple Access Protocols (MAC)

- Random access protocols
  - Aloha
  - CSMA
  - CSMA/CD
  - CSMA/CA
- Control Access
  - Polling
  - Token Passing
- Channelization protocols
  - FDMA
  - TDMA

Pure Aloha

→ Random access protocol
→ ACK is there
→ LAN based
→ Only transmission time
No propagation time
→ Vulnerable time = $2 \times T_t$
→ Efficiency, $\eta = G \times e^{-2G}$

Slotted Aloha



$VT = T_t$
$\eta = G \times e^{-G}$

(no. of station)

Carrier - Sense Multiple Access (CSMA)

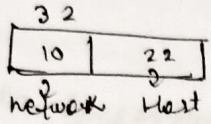
check the persit only

- 1 - persistent (High level of collision)
- 0 - persistent (less)
- p - persistent

(3) Network layer

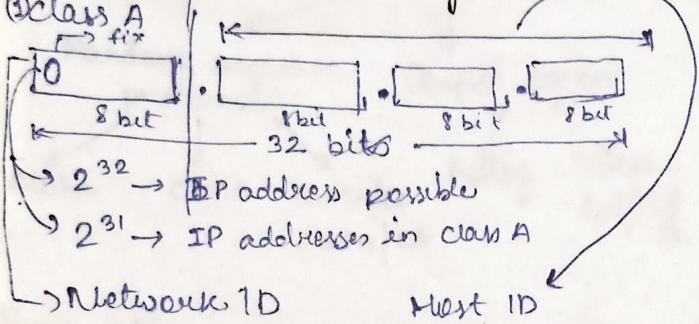
Responsibilities
→ source to destiny delivery
host to host
machine to machine
(1) logical address (IP) — using
(2) Routing
(3) fragmentation
(4) Congestion control

before 1980s



Classes of IP addressing

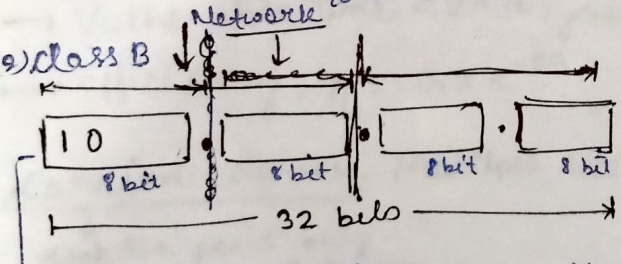
(1) Class A



- 2^{32} → IP address possible
- 2^{31} → IP addresses in class A
- → Network ID, Host ID
- No. of networks in class A = $2^7 = 128$
  use = 128 − 2 = 126
- No. of Host possible in every Network
  useable = $2^{24} - 2$
  '64.0.0.0 to 64.255.255.255
  ↓ directed broadcast address

to find network of IP take (A id) of the IP address with
255.0.0.0

(2) Class B

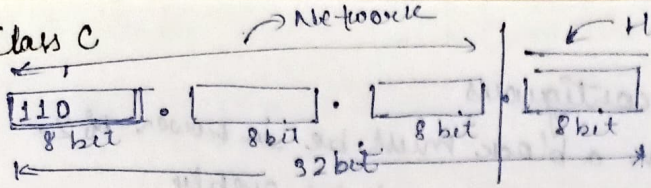


→ 10 000000 → 128
10 111111 → 129
Range → 128 − 191
* No. of Networks = 2^{14}
* No. of Host possible = 2^{16}
  useable = $2^{16} - 2$
128.0.0.0 to 128.0.255.255

No. of add. → 2^{30}

Q. to find the network of a IP address (host)
130.2.3.4
255.255.0.0 (default mask)
130.2.0.0 ← network

(3) Class C



110 00000 → 192

⋮

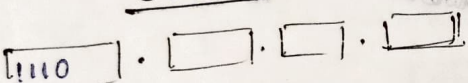
110 11111 → 233

- Range = 192 – 233
- No. of address in class C = 2^{29}
- No. of Networks = 2^{21}
- No. of Hosts in each network = 2^8
  useable = $2^8 - 2$
  = 254

* To find network of a host
  Subnet mask = 255.255.255.0
  and with ↗

(4) Class D & E

Class D



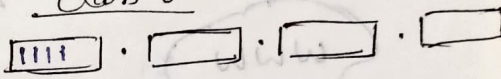
1110 0000 → 224

⋮

1110 1111 → 239

* Range = 224 – 239
* No. of IP addresses = 2^{28}
* No networks / host
* used for Multitasking / Group Gmail / Broadcast

Class E



1111 0000 → 240

⋮

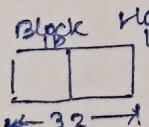
1111 1111 → 255

* Range = 240 – 255
* No. of IP address = 2^{28}
* No networks / host
* used for Military Purpose

Classless Addressing (CIDR)

→ No classes

→ Only blocks



→ Notation

x.y.z.w/n → mask 1111

→ no. of bits represents block/network

IP → 200.10.20.40/28 → network id → Host ID → 32 – 28 = 4 = 2^4

→ 2^8 'one'

11111111.11111111.11111111.11110000

255.255.255.240

To find network of IP address
↳ first find mask
/28 → 255.255.255.240
↳ take & with ↗

Rules —

- Addresses should be contiguous
- No. of addresses in a block must be in power of 2.
- First address of every block must be evenly divisible with size of block.

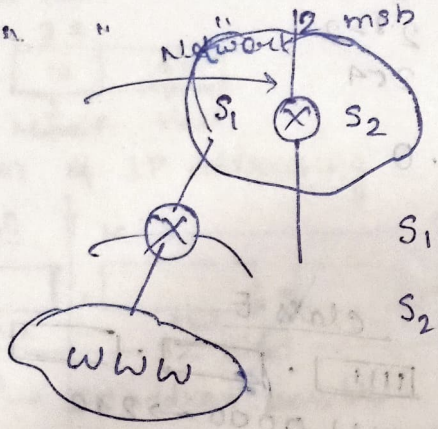
Subnetting → Dividing the big network into small networks

↓

network ID should be same

↳ if we reserve 1 msb of host ID → 2 part

" " " 2 msb " → 4 part



200.10.20.0

↓

$S_1 \rightarrow 200.10.20.\boxed{0}\,0\,0\,0\,0\,0\,0\,0$

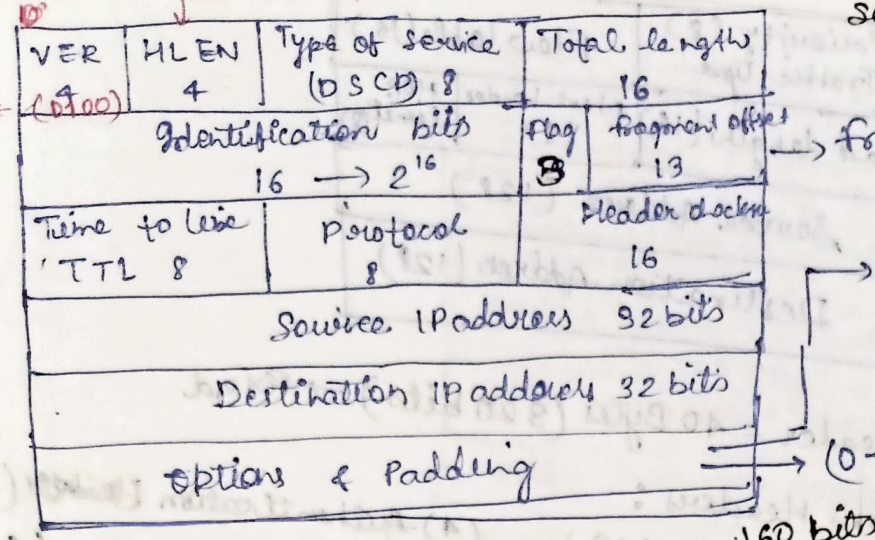
$S_2 \rightarrow 200.10.20.\boxed{1}\,0\,0\,0\,0\,0\,0\,0$

200.10.20 → Network

1 → 2 Host

IPv4 Header Format

→ Connection less, Datagram service



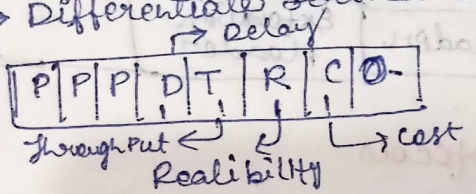
→ fragmentation

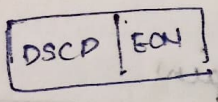
Header size = 20-60 Bytes

Payload = 0 - 65515 Bytes

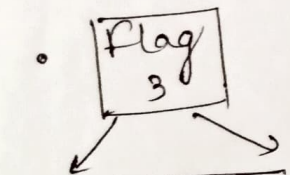
0 - 2^{16}

- HLEN × 4 = Header size
- Type of service → Differentiate services code point (DSCP)



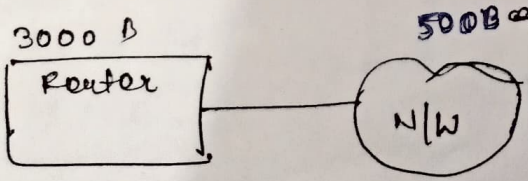


- Flag 3



→ More fragment
Yes → 1
No → 0

Q.



$\frac{2980}{480} = 7$

20	2980

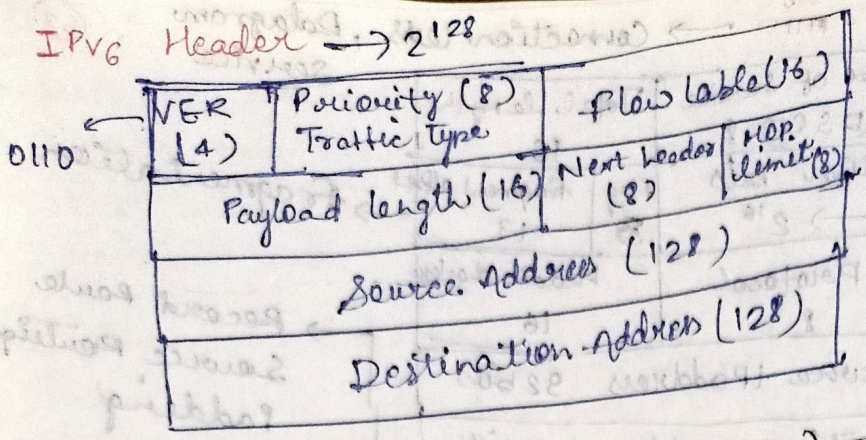
500
20 + 480

P_1 P_2 P_3 P_4 P_5 P_6 P_7

1
20 + 480

- Offset → No. of data ahead of a data

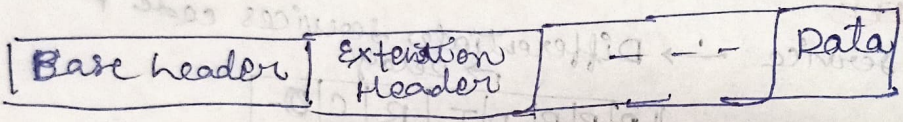
IPV6 Header → 2^{128}



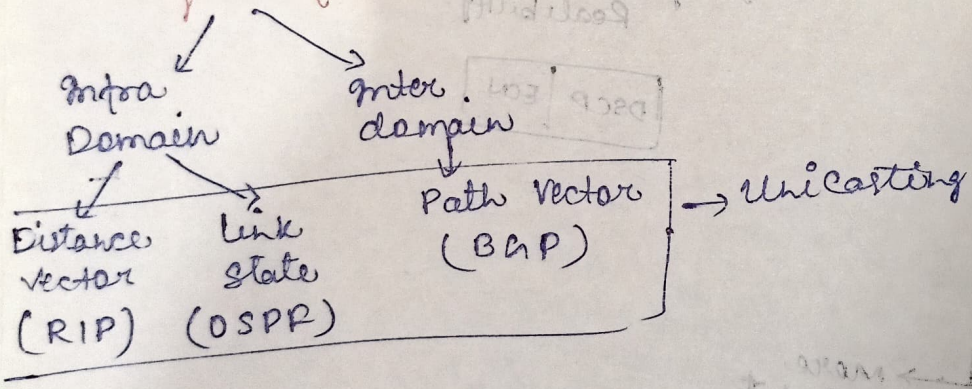
Base Header = 40 Bytes (320 bits) → fixed

Extension Headers:

1) Routing Header (43)
2) Hop by Hop option (0)
3) fragmentation Header (44)
(4) Authentication Header
(5) Destination Header
(6) Encapsulating Header
Payload (59)



Routing Protocols



ARP (Address Resolution Protocol)

Hardware Type		Protocol Type
Hardware length (6)	Protocol length	Operations Req-1, Rep-2
Sender Hardware address (6B for Ethernet)		
Sender Protocol address (4B for IP)		
Target Hardware address		
Target Protocol address		

NAT (Network address Translation)

↳ Translate → Private IP to Public IP and vice-versa

NAT Translation Table		
Private IP	Public IP	Port
10.10.0.2	25.25.25.10	—
⋮	⋮	

(4) Transport layer

Roles → (1) End to End delivery of process to process

(2) Use two protocol mainly (SHTP, - - -)
- TCP → Transmission control
- UDP → User datagram protol

(3) TCP → Reliability
- Inorder, no-loss of data

(4) TCP → connection-oriented

(5) Error control (TCP)
- (checksum)

(6) Congestion control (using AIMD)

(7) flow control

(8) Segmentation

(9) Multiplexing & demultiplexing

Port No. 80 ← Simple Mail Transfer protocol

* MAC address → physical address of a system
  - always constant
* IP address depend on Router
  - Length 48
* Define port number

Socket Address → to identify any connection uniquely.
- IP + Port
- IP address (32 bit) + Port No. (16 bit)

TCP : Transmission control Protocol

Source Port 16 bit	Destination port 16 bit
Sequence Number 32 bit	
Acknowledge No. 32 bit	
HLEN 4 bit | URG | ACK | PSH | RST | SYN | FIN	Window size 16 bit
Checksum 16 bit	16 bit urgent Pointer

Transport layer → create segments

Reliability
- Byte Streaming
- Connection oriented
- Full Duplex
- Piggybacking → CVBR → SR
- Error Control
- flow control
- Congestion Control

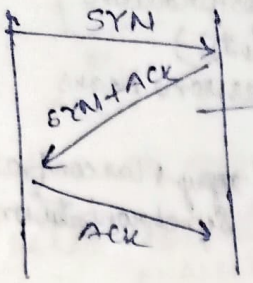
Use 3-Way

TCP Header - (20-60B)
- 20×8 = 160 bits

~~Transport Layer~~

Webpage port no. 80

TCP connection Establishment and connection Termination



→ 3 Way communication

- It is Duplex Mode ⇄ at the same time
- FIN → finish the connection

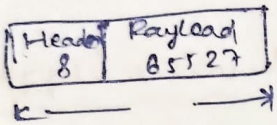
- Connection Termination
  - ↳ 3 way or 4 way

User Datagram Protocol [UDP] →

- ↳ Connectionless protocol
- ↳ like Postal service
- ↳ Unreliable
- ↳ No order

Source Port (16)	Destination Port (16)
Length (16)	checksum (16)

UDP header (8B fixed)



$2^{16} = 65535$

* checksum = UDP header + UDP data + Pseudo header of IP → Error control

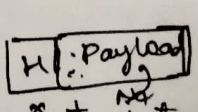
UDP Applications

(1) Query Response Protocol (one request one reply) [DNS, DHCP]

(2) Speed (online games, voice over IP)

(3) Broadcasting / Multicast [RIP]

(4) Continuous streaming [Skype, Youtube



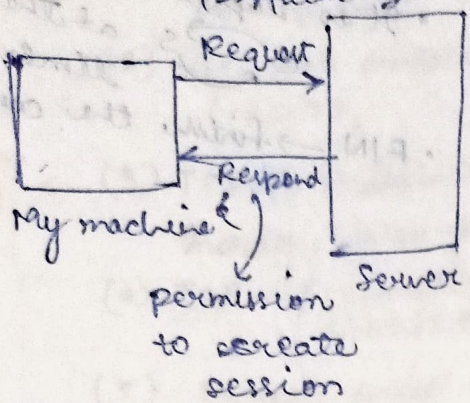
H + Data

TCP	UDP
(1) Connection Oriented	(1) Connectionless
(2) Reliable	(2) less Reliable (No order)
(3) Error control is mandatory	(3) Error control is optional
(4) Slow transmission	(4) fast transmission
(5) More overhead (20-60) Header	(5) less overhead (8B) Header
(6) flow control, congestion control	(6) No FC, CC

(5) Session layer: uses :-

(1) Authentication
(2) Authorization (set of privileges)
(3) Session Restoration (Check Points)
↳ use session beans

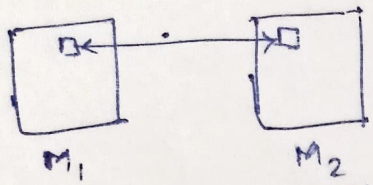
(4) Hibernate
↳ provides (Data Flow control, Synchronization)



* Session & Presentation layer are not provided by operating system

Presentation layer

(6) Presentation layer → (1) code conversion (formatting)
(2) Encryption | Decryption
(3) Compression



* does not provided by OS provided by application layer
* helps Application layer in presenting

* HTTPS works on (SSH) → secured socket layer

(7) Application layer → use in transport layer 16 Bit 2^{16}

Protocol Name	Port No.	Transport protocol
Echo	7	TCP/UDP
File Transfer Protocol FTP	20/21	TCP
~~Secure shell(ssh)~~		
Secure Shell (SSH)	22	TCP
Remote login Telnet	23	TCP
Simple Mail transfer protocol SMTP	25	TCP
Domain Name System DNS	53	UDP
Dynamic host control protocol DHCP	67/68	UDP
Trivial file transfer Prot TFTP	69	UDP
Hyper text transfer Protocol HTTP	80	TCP
Post office protocol POP	110	TCP
Network time protocol NTP	123	UDP
HTTPS	443	TCP
Routing Information P RIP	520	UDP