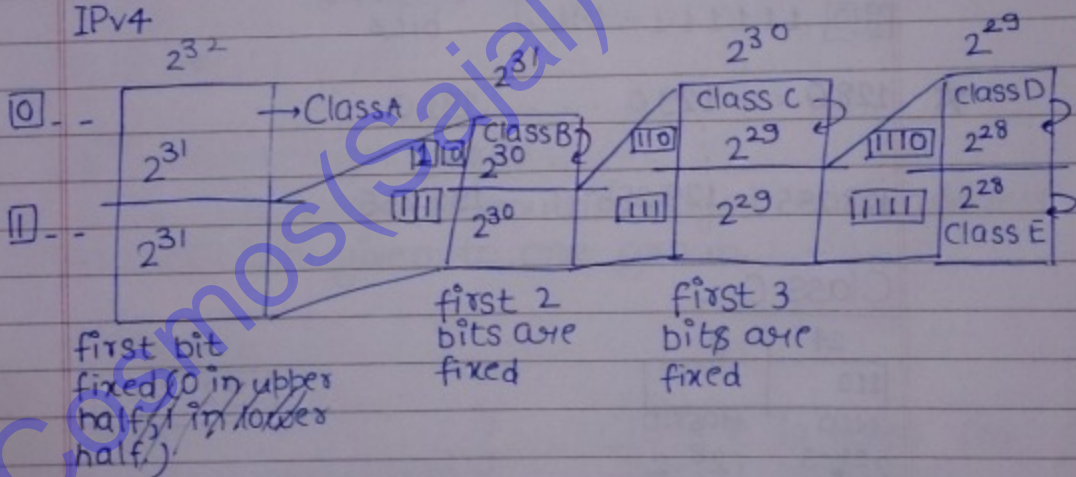


09.12

- iplookup/ALL
- NSlookup

→ If we type the title of a website, e.g. "www.gmail.com", then to get its IP address of the webpage, then DNS is taken use of to get the IP address, DNS gives the IP address of that webpage & is composed of netid & hostid which is used to access the network containing that server which contains the webserver. & hostid contains address of that webserver.

IPv4



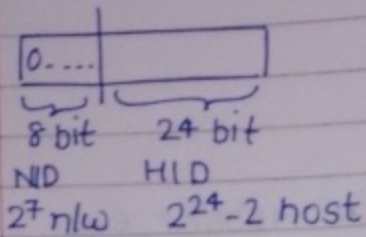
★ Class A :-

0 000 0000 (0) → not taken as any network id 1-126
 0 000 0001 (1)
 0 000 0010 (2) (1-126 n/w id's are allotted).

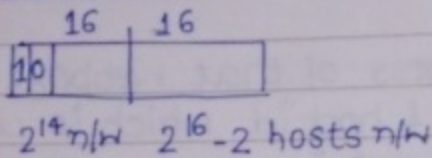
⋮

⋮

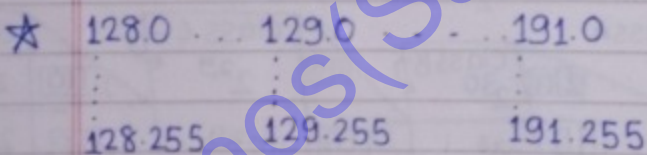
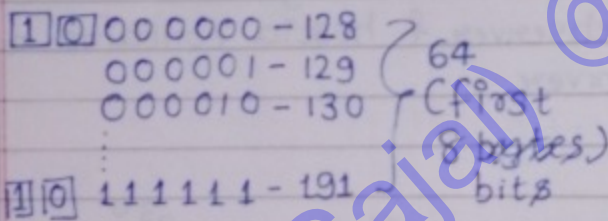
0 111 1111 (127) → not taken as any network id (taken as loopback address)



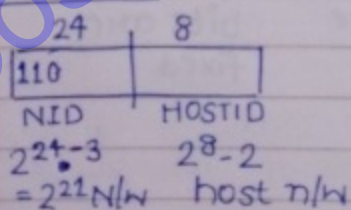
★ Class B :-



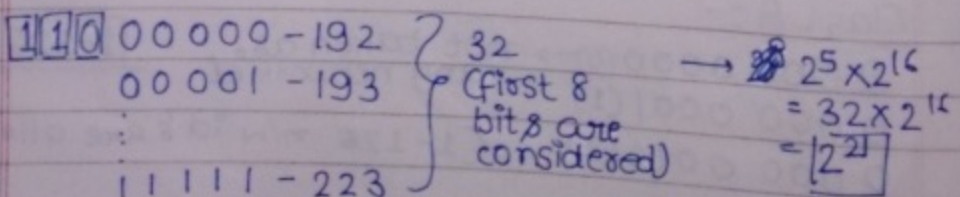
128-191



Class C :-



192-223



192.0.0	223.0.0
192.0.255		223.0.255
192.255.255		223.255.255

Class D

→ Used for multicasting

1110

4 bits

28 bits

224-239

1110 0000 → 224

1110 0001 → 225

⋮

1110 1111 → 239

 $2^{28} = 256$ million groups.

- Class D is used for multicasting & each address in class D is given to one group.

Class E

1111

4 bits

(fixed)

28 bits

- Class E is used for special purposes.

1111 0000 → 240

0001 → 241

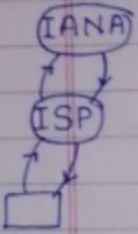
⋮

1111 → 255

240-255

IANA

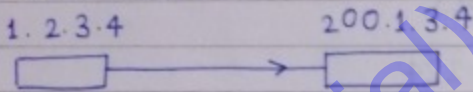
(Internet assigned Numbers Authority)



IANA provides IP addresses to different requesters.

Types of message casting

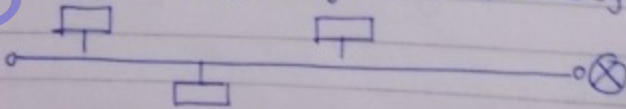
- ① Unicast → Sending a message from one host to another (1:1).



- ② Broadcast → sending a message from 1 host to all other hosts is called broadcasting.

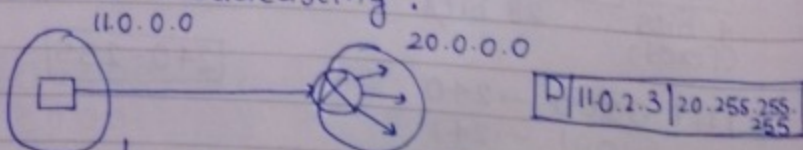
a. limited broadcasting: sending a message from 1 host to all other hosts in the same network is called limited broadcasting.

LAN is a switch; everyone sees everything else.



router blocks the traffic.

- (b) Directed broadcasting:



IP	NID	DBA	LBA
1.2.3.4	1.0.0.0	1.255.255.255	255.255.255.255
192.1.2.3	192.1.2.0	192.1.2.255	"
173.1.2.3	173.1.0.0	173.1.255.255	"

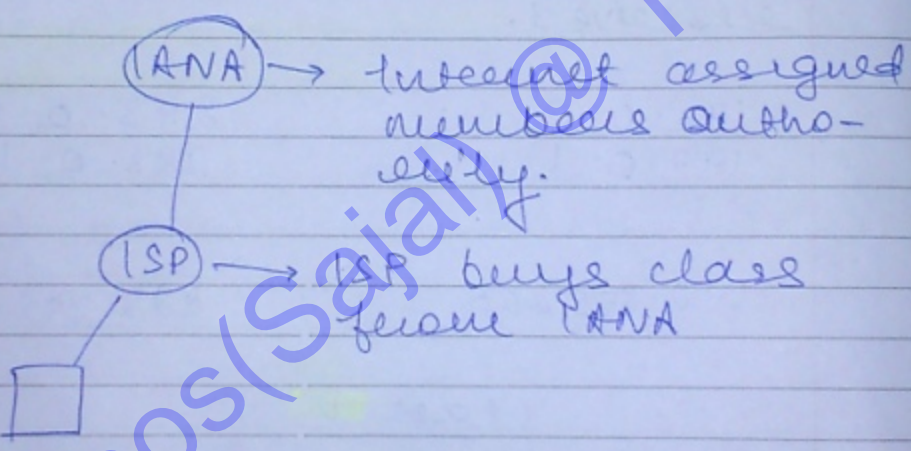
- each address in class D is given to one group.

Class E

- ~~240-255~~ 240-255

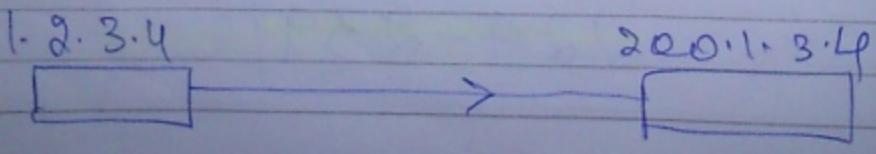
- there is no concept of n/w id or host id.

- Reserved



Types of message casting

① Unicast → sending a message from 1 host to 1 host. (1:1)



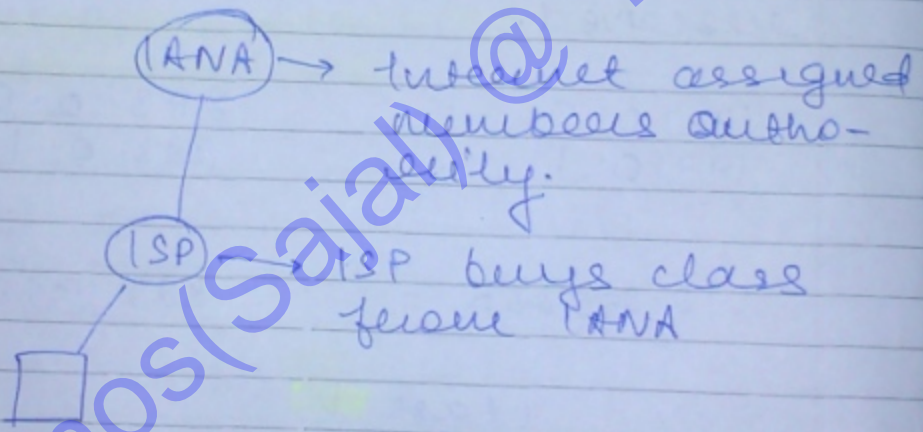
- each address in class D is given to one group.

Class E

- ~~249-255~~ 240-255

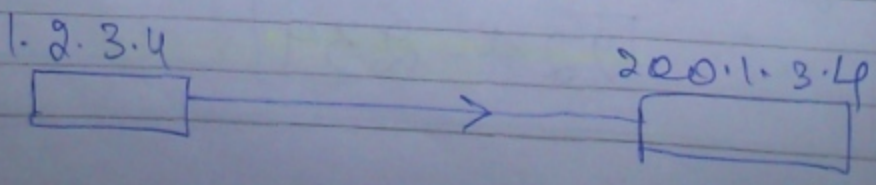
- there is no concept of n/w id or host id.

- Reserved



Types of message casting

① Unicast → sending a message from 1 host to 1 host. (1:1)

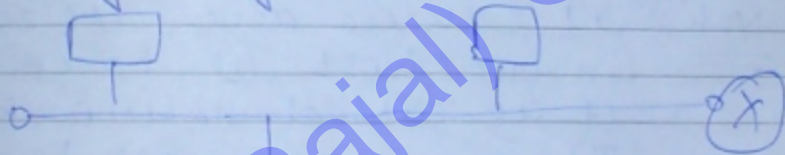


② **Broadcast** → sending a message from 1 host to all other hosts is called broadcasting.

a) **limited broadcasting**: sending a message from 1 host to all other hosts in the same network is called limited broadcasting.

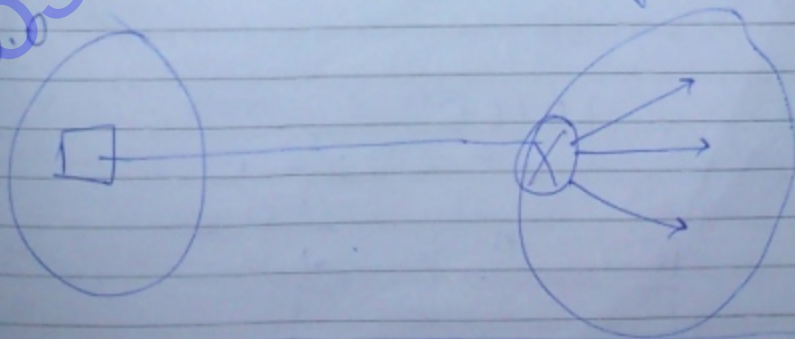
(all 0s signify entire n/w to)

CAN is a switch; everyone sees everything else.



router blocks the traffic.

b) **Directed broadcasting**: 20.0.0.0



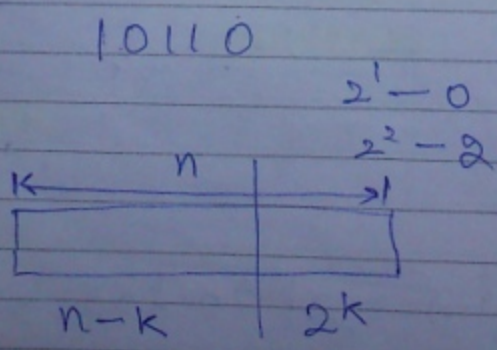
D	11.0.2.3	20.255.255.255
---	----------	----------------

IP	NetID	DBA
1.2.3.4	1.0.0.0	1.255.255.255
192.1.2.3	192.1.2.0	192.1.2.255
173.1.2.3	173.1.0.0	173.1.255.255

LB A
255.255.255.255
u
u
same

Note: if there are all zeros in the host id part, then it is called network id. If there are all 1s in the host id part, then will be called directed broadcast address for that network.

Thus, 2 IP addresses are reserved, first and last.



CIDR representation of a block

Date: _____

Page No.: _____

Q

a.b.c.d/n where a.b.c.d is IP addresses and n is no of bits used for network id part.

$32-n$: host id bits

$2^{32-n}-2$: no of hosts. again, first address identifies block, and last address is reserved for directed broadcast.

• for the eg above, mask is $\frac{28}{2^{32-28}}$ addresses for hosts.

eg. 100.100.100.100/28

01100100

01100000 = 96

→ [100.100.100.96 →
!
100.100.100.127 →]

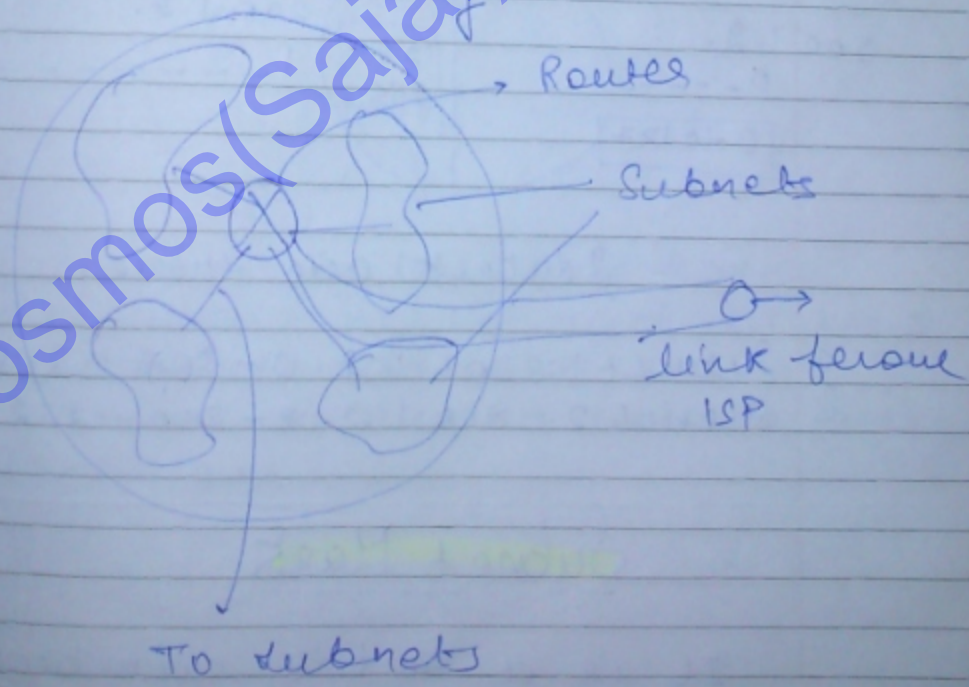
• any address in the block may be given.

9
120.250.250.850
N + 1110101111010
1111
180.240.0.0
1
1111011

N/w id.
120.250.240.0
to
120.250.255.255
directed broadcast

Subnets

dividing a big network into many smaller networks is called subnetting



Advantage:

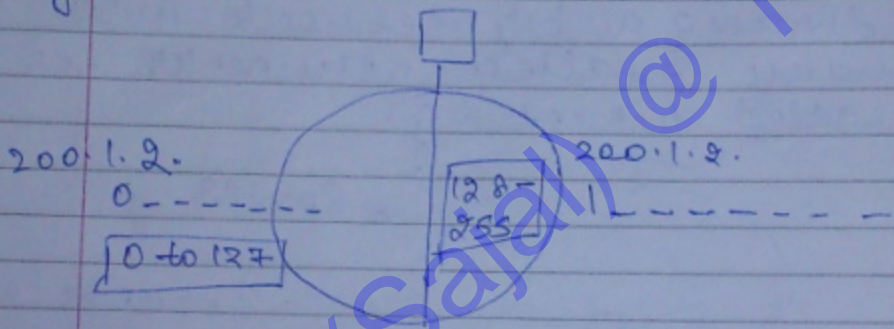
- ① Maintenance
- ② Security

Disadvantage:

- ① Routing becomes difficult.
 - Identification of N/A
 - Idⁿ of subnet within N/A.
 - Idⁿ of host within subnet
 - Idⁿ of process within host

Eg

200.1.2.0



2 subnets are there.

- { Subnet 1: 200.1.2.0 - 200.1.2.127
- { Subnet 2: 200.1.2.128 - 200.1.2.255

Subnet Mask

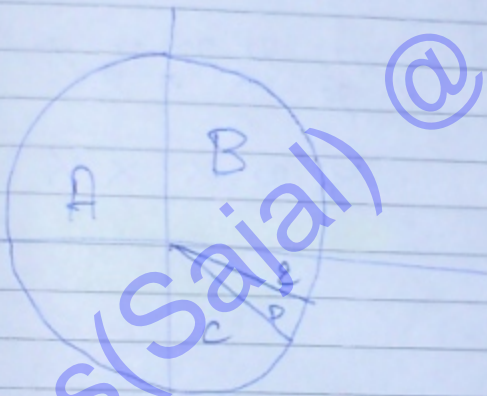
It is a 32 bit number in which no of ones indicate n/a id part plus subnet id

part and number of zeros indicate host id part.

Eg. for a class C address with 4 subnets, the subnet mask will be the following

255.255.255.192

for dividing the n/w into 8 parts, use 3 subnetting bits.



for each subnet, we have different net id and directed broadcast address.

for the n/w as whole, these are unique.

Computer Networks

Date: _____
Page No.: _____

IP Address	Subnet Mask	CA	CB	CC
255.0.0.0	2 ²⁴ - 2	1	X	X
255.255.0.0	2 ¹⁶ - 2	28	1	X
255.255.255.0	2 ⁸ - 2	24	28	1
255.128.0.0	2 ²³ - 2	21	X	X
255.240.0.0	2 ²⁰ - 2	24	X	X
255.255.192.0	2 ¹⁴ - 2	20	22	X
255.255.248.0	2 ¹¹ - 2	23	25	X
255.255.255.128	2 ⁷ - 2	27	29	2
255.255.255.240	2 ⁴ - 2	28	32	4
255.255.255.255				

hosts # CA # CB # CC

Note: Common mistake

Number of subnets

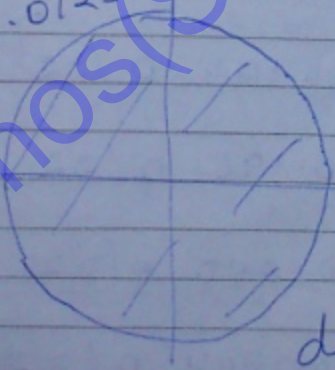
all 0s and all 1s in the subnet id are possible.

if k bits are chosen for subnet id then $2^k - 2$ subnets is wrong.

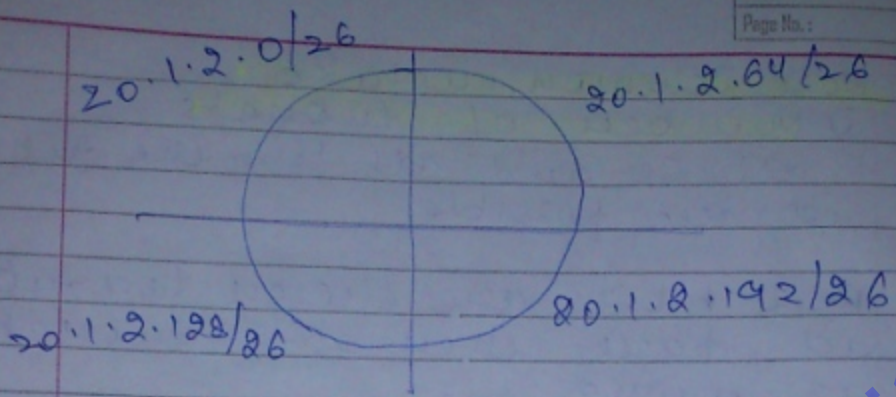
2) Theoretically, we can choose any bits from any position in the host id part for subnetting. Practically, we should always choose from first few bits. we can have 8th subnet masks.

Subnetting in classless

20.1.2.0/25 20.1.2.0/24
80.1.2.128/25



division into 2



division in 4 parts

20

$$15.20.198.100/20$$

$$11000110$$

divide in 4 parts

N/w id → 15.20.192.0

1st → 15.20.192.0/22

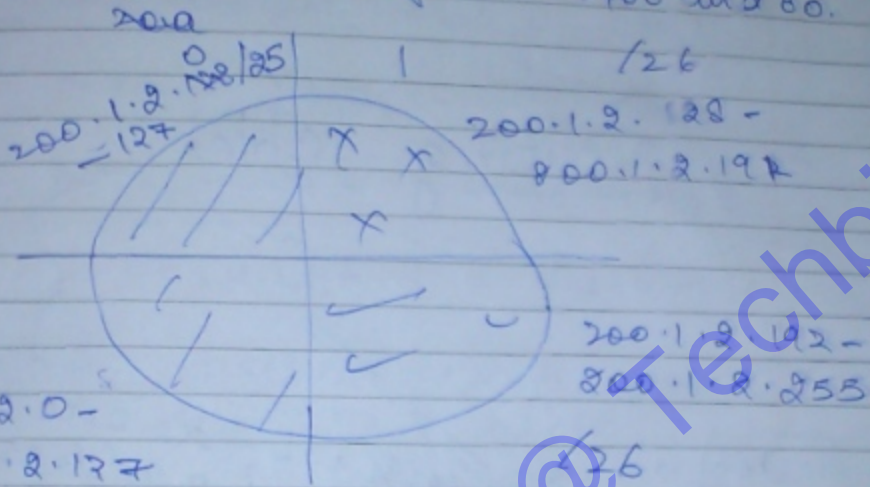
2nd → 15.20.196.0/22

3rd → 15.20.200.0/22

4th → 15.20.204.0/22

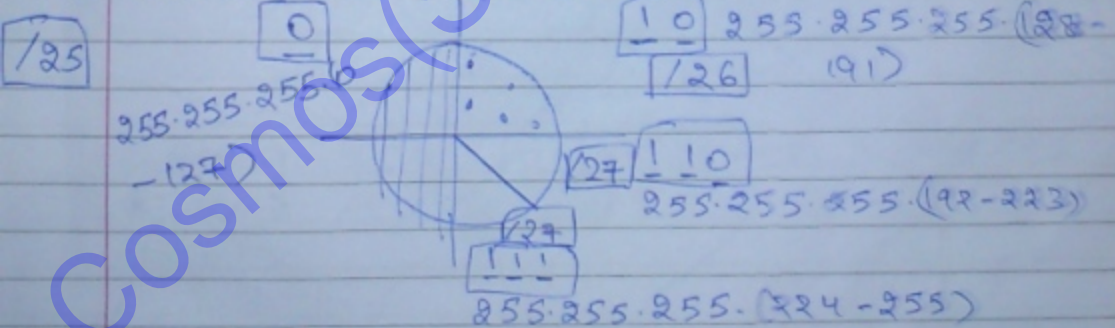
Variable length Subnet Masking (VLSM)

∴ 200.1.2.0/24 divide into 3 subnets such that we get 120, 60 and 60.



* we can also divide it the other way.

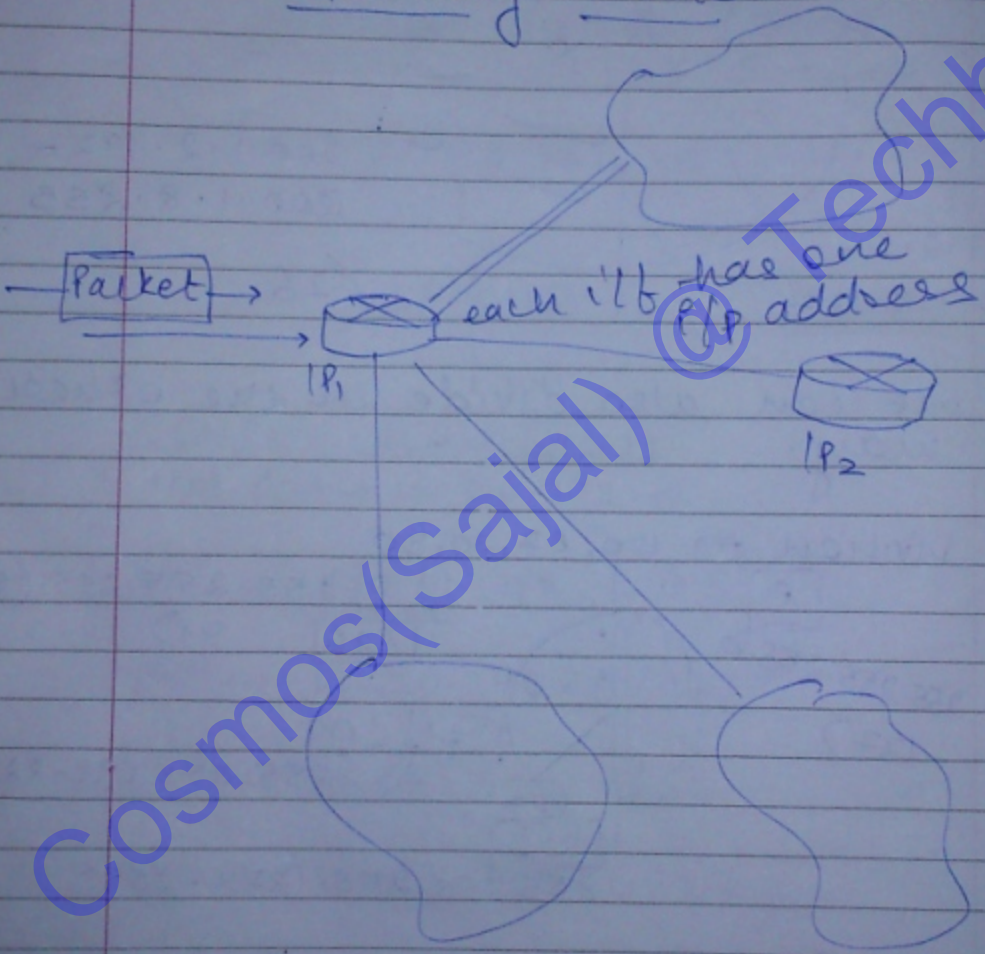
∴ Division → 120, 62, 30, 30



20.1.198.100/20 divide into 1/2, 1/4, 1/4

- 20.1.192.0/21
- 20.1.200.0/22
- 20.1.204.0/22

Routing Table

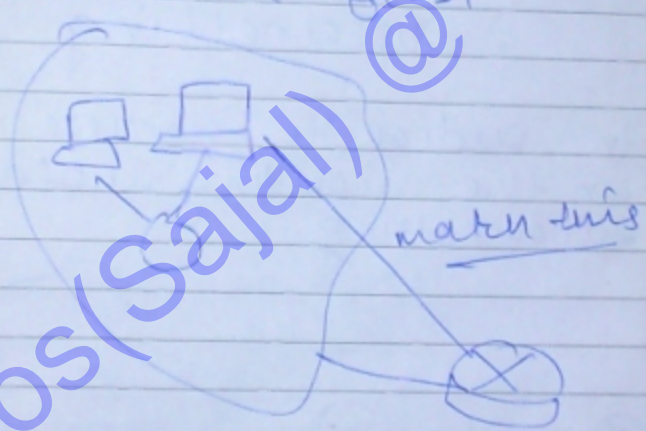


Whenever a packet comes to the router, it masks the destination IP and mask,

and searched the table.

NID	Subnet Mask	Interface
NID ₁	SM ₁	eth0
NID ₂	SM ₂	eth1
NID ₃	SM ₃	eth2
NID ₄	SM ₄	eth3
default		eth3

Note: largest mask should be matched first



using for largest mask makes search easy.

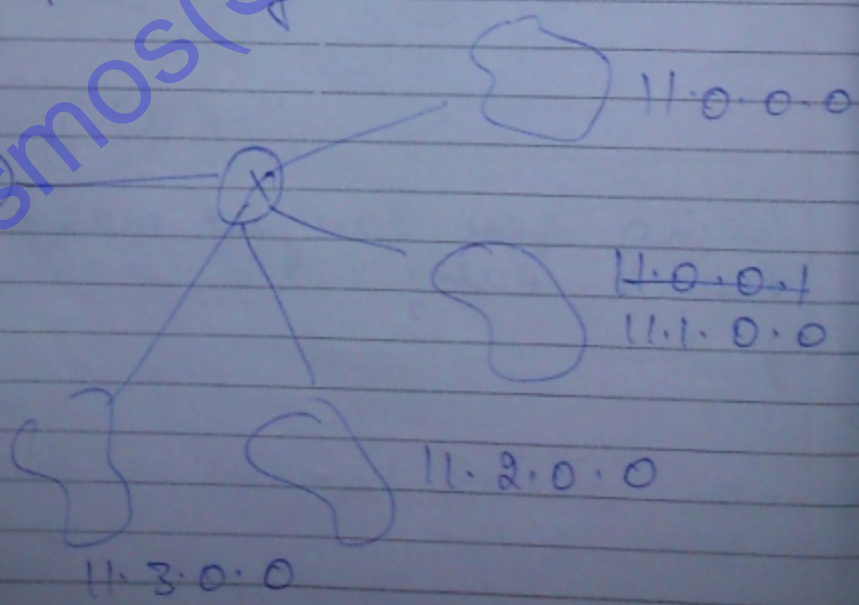
Eg

20.0.0.0	255.0.0.0	eth1
20.128.0.0	/9	eth2
20.128.0.0	/10	eth3
20.160.0.0	/12	eth4
	default	eth5

- 20.168.3.1 matches all, forwarded to eth1
- 120.x.y.z matches none, sent to default.

Supernetting

In order to reduce the # entries in the routing table, we need supernetting.



Supernetting rules:

1. Network IDs should be contiguous.
2. number of subnets should be of same size.
3. Number of networks should be a power of 2.
4. first n/w id should be a multiple of supernet size.

finding supernet ID

- ① find out supernet mask, AND with any IP address.
- ② AND all of them
- ③ first ID.

Supernet Mask

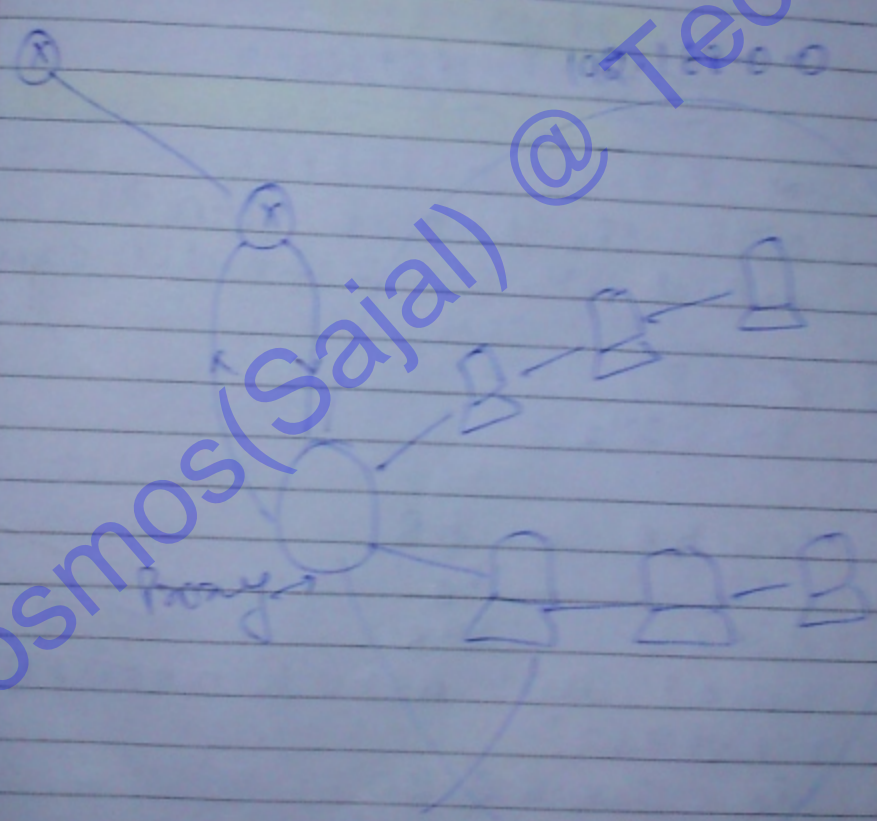
it is a 32 bit number in which number of 1s indicate fixed part and number of 0s indicate variable part.

```

hexa  173 0. 0. 0
       173.1. 0.0
       173.2. 0.0
       173.3. 0.0
       -----
       255.255.2.
       255.252.0.0
  
```


1 CA
 10.0.0.0 - 10.255.255.255
 16 CB
 172.16.0.0 - 172.16.0.31-255.255
 256 CC
 ① 192.168.0.0 - 192.168.

Network Address Translation



Private addresses can be assigned to all the nodes. Proxy does the job of deciding who takes

what. Depends on what
is being requested.

IPv4 exhaustion

→ IPv4 addresses are getting used fast, this problem is called getting consumed fast.

SM: ↓

IPv6

NAT

Gate Questions

Q7) SM

a) IP_1, IP_2 same n/w?

→ And SM, check if n/w is same.

Q8)

b) (IP)
SM₁, SM₂

Q9) Comp A, Comp B

IP_A

IP_B

SM_A

SM_B

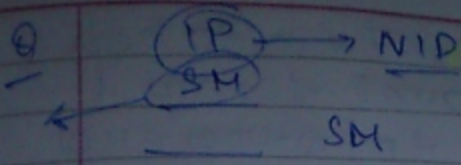
what will A think about B & vice versa?

$IP_A = 200.1.2.20$

SM_A: 255.255.255.128

$IP_B = 200.1.2.100$

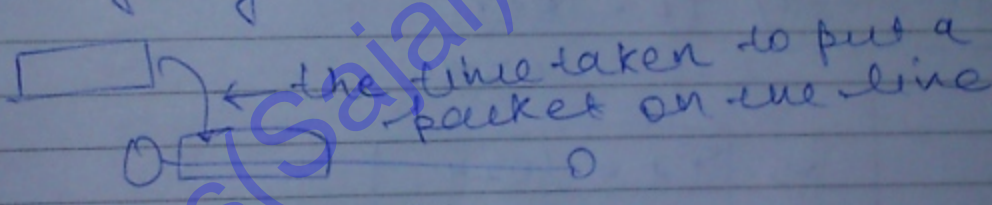
255.255.255.192



Note:

Delays in Computer Networks

Transmission delay: Time taken to transfer a packet onto the outgoing link. This means



factors: \leftarrow size
 \leftarrow BW

$$B \text{ bps}$$

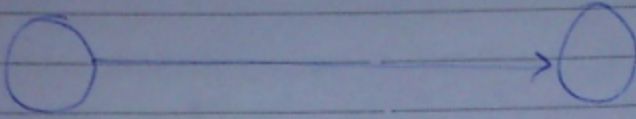
$$1 \text{ sec} \rightarrow B \text{ bits}$$

$$B \text{ bits} \rightarrow 1 \text{ s}$$

$$1 \text{ bit} \rightarrow \frac{1}{B} \text{ sec}$$

$$L \text{ bits} \rightarrow \frac{L}{B} \text{ s}$$

- **Propagation delay:** Time taken by a bit to travel from 1 end of the wire to other end of the wire.



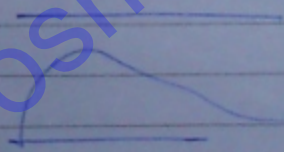
$$T_p = \frac{d}{v}$$

- If BW is 1000 bps & $L = 10000$ then what is the T_p time?
→ 1s

- If $B = 1 \text{ kbps}$ & $L = 1 \text{ kb}$

$$T_p = \frac{1000}{1024}$$

$$\Rightarrow T_p = \frac{1024}{1000}$$



- If $d = 2 \text{ km}$, $v = 2 \times 10^8 \text{ mps}$

$$T_p = 10 \mu\text{s}$$

$$10^{-3} \quad 2 \times 10^3 + 10^{-3} \quad 10^{-3}(1 + 0.01)$$

B $L = 1000 \text{ bits}, B = 1 \text{ Mbps}$

$$d = 2$$

$$1.01 \text{ ms}$$

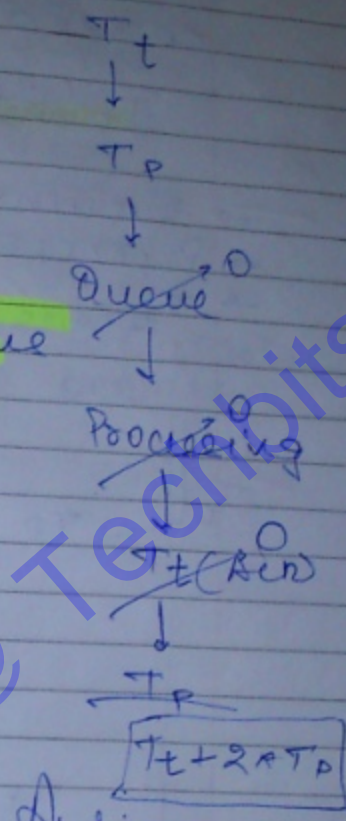
$$\text{Time} = T_{\text{del}} + T_{\text{trans}}$$

$$T = \frac{d}{v} + \frac{L}{B}$$

$$= \frac{2 \times 10^3}{2 \times 10^8} + \frac{10^3}{10^6}$$

$$= 10^{-3} + 10^{-3}$$

$$T = 1.01 \text{ ms}$$



Flow Control Mechanisms

* A fast sender should never send more than what a receiver can receive.

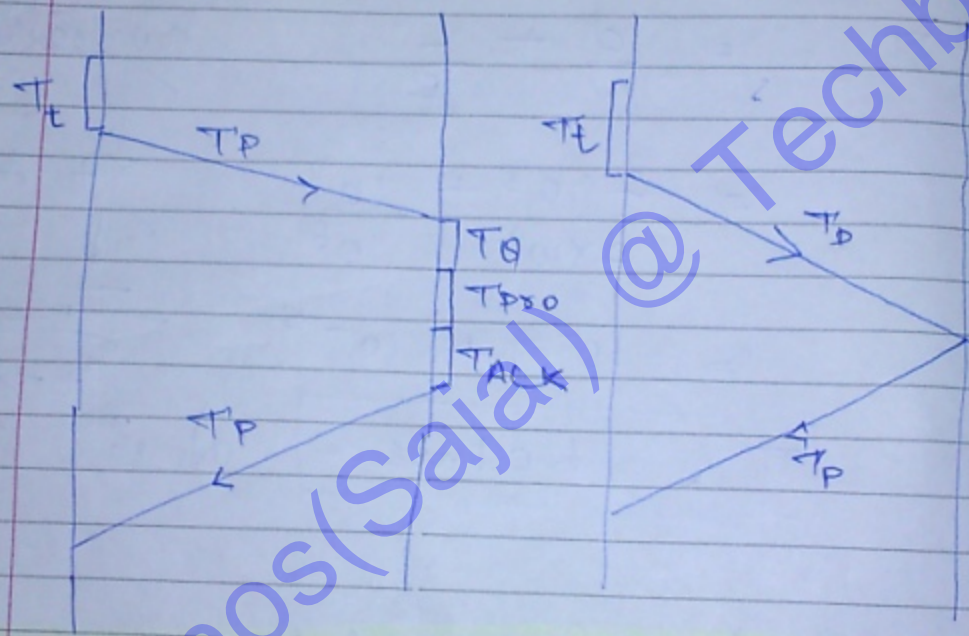
STOP AND WAIT

In this strategy, a sender will send data and wait for ACK, before sending next data.

Therefore, total time taken to send one data packet is

$$T_{transfer} + 2 \times T_{prop}$$

- Queuing, uncertain, assumed 0
- Processing (also) = 0
- $T_{transfer}$



- (a): detailed timing diagram
- (b): timing with all extra delays assumed 0.

$$N = \frac{\text{Total time spent transmitting}}{\text{Total cycle time}}$$

$$n = \frac{T_t}{T_t + 2 \times T_p}$$

$$n = \frac{1}{1 + 2a}$$

$$a = \frac{T_p}{T_t}$$

Q. if $n = 1/2$ in stop and wait, then what is relation between T_t and T_p .

Ans =

$$\frac{1}{2} = \frac{1}{1 + 2a}$$

$$1 + 2a = 2$$

$$a = \frac{1}{2}$$

$$a = \frac{1}{2} = \frac{T_p}{T_t}$$

$$T_t = 2 \times T_p$$

$$\frac{1}{1 + 2a} \geq \frac{1}{2}$$

$$2 \geq 1 + 2a$$

$$a \leq \frac{1}{2}$$

$$\frac{T_p}{T_t} \leq \frac{1}{2}$$

$$T_t \geq 2 T_p$$

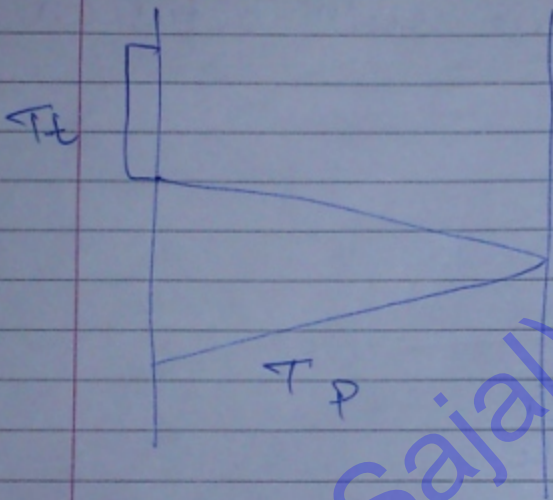
if T_t is very less and T_p is very large, more time will be spent travelling.

Q if $T_p = 1\text{ms}$ and $BW = 1\text{Mbps}$
what is min length of
packet for 50% eff

Ans

$$\frac{L}{10^6} \geq 2 \times 10^{-3}$$

$$L \geq 2000 \text{ bits}$$



to increase eff, you
increase T_t , which can
be done by increasing
length.

Throughput:
packets / Time

$$\frac{BL}{T_t + \theta TP}$$

Link utilization
 see
 BW utilization \rightarrow
 see
 Effective BW
 = nB

$$\frac{(L/B)(CB)}{T_t + \theta XTP}$$

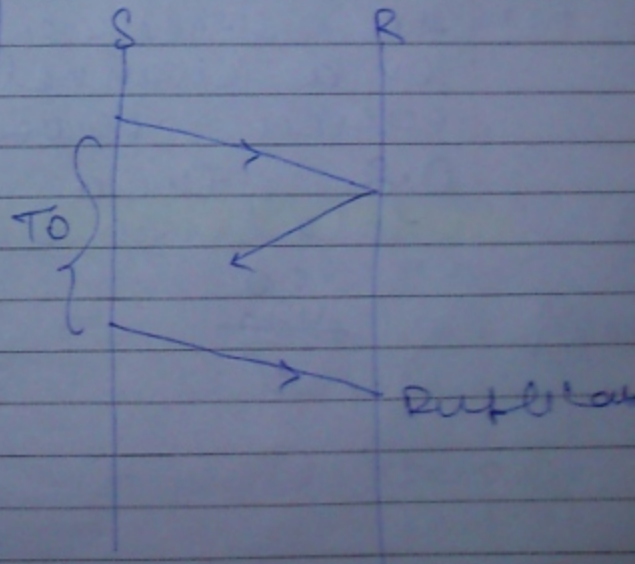
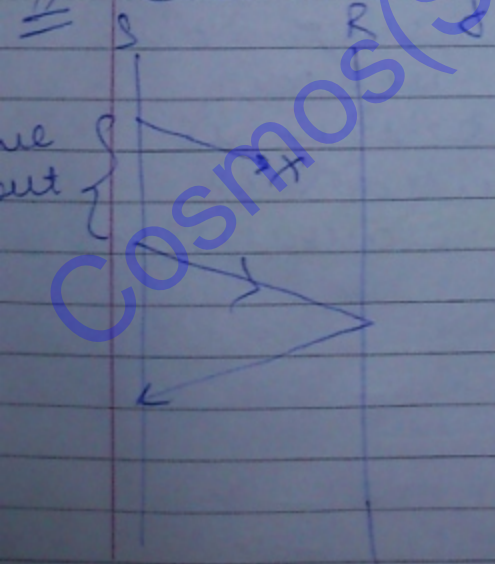
$$\text{throughput} = nB$$

if $T_t = 1 \mu s$, $T_p = 1 \mu s$, $BW = 3 \text{ Mbps}$,
 what is TP.

$$= \left(\frac{1}{1 + 2 \cdot 1} \right) (3 \text{ Mbps})$$

$$\text{Ans} = 1 \text{ Mbps}$$

Data missing

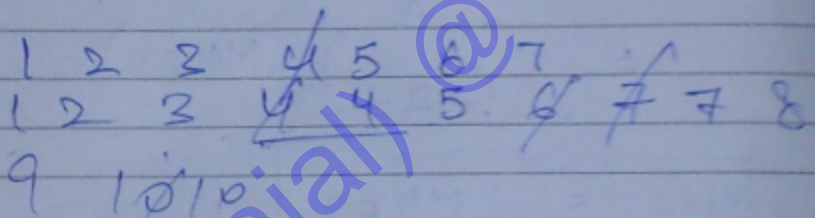


now, we need to add seqⁿ no in sender to correctly identify packet.

$$S \& W [S \& W + IO + \text{Seq}^n \text{ number}]$$

Q If in stop and wait, a sender is sending 10 packets, in which every 4th packet is lost then what is the total number of ~~10~~ trans req

Q



Q

If 400 packets are retransmitted before sender to receiver, using S&W, on a channel where error probability is 0.2, then what is total no of transmission.

Q

$$\frac{400}{0.2} = 2000$$

$$n + np + np^2 + \dots$$

$$= \frac{n}{1-p}$$

$$= \frac{400}{1-0.8} = 500$$

$$\eta = \frac{1}{1 + 2 \times \frac{d}{v} \times \frac{B}{L}}$$

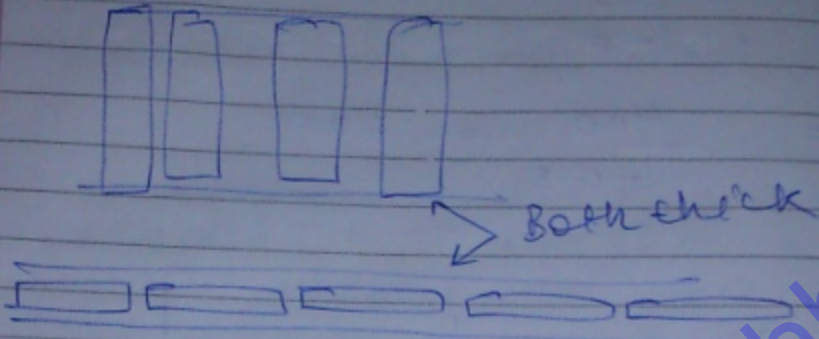
- As length increases, efficiency decreases; stop and wait is suitable for frames of big size.
- As distance increases, efficiency decreases; stop and wait is efficient in LANs, but not in WANs.

Capacity of a channel

→ Number of bits a channel can hold at any time is called capacity of the channel.

$$[\text{Capacity} = \text{Bandwidth} \times \text{Delay}]$$

if **BW delay product** is high then it is called thick wire // thick channel.



• Basically a measure of how much can be stuffed into the wire.

low BW delay \rightarrow thin wire

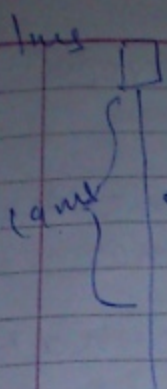
Note:

In thick wire, stop and wait fails, therefore, in order to increase efficiency we use pipelining.

Pipelining

Q if $T_t = 1 \mu s$, $T_p = 9.5 \mu s$, then what is efficiency of ΣW

$$\eta = \frac{1}{1 + 9.5} = \frac{1}{10}$$



← i could have sent additional packets here, this gives motivation for pipelining.

- 1. $T_t \rightarrow 1p$
- 2. $1p \rightarrow T_t + t_e$
- 3. $1sec \rightarrow$

$$\left(\frac{T_t + 2 \times T_p}{T_t} \right) = WS$$

we need $\lceil \log_2 WS \rceil$

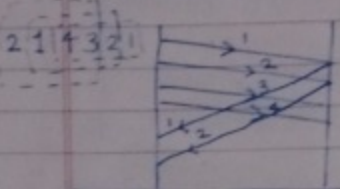
$T_t + 2 \times T_p \rightarrow$ Total time spent in sending of the packet and receipt of ACK

why $/T_t$: gives us # packets that can be sent. each packet takes T_t to be placed on line

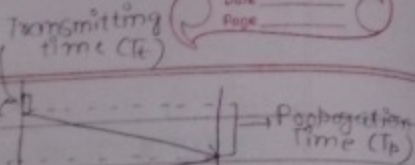
Date

23.09.12

Quicklinks (Q-44 to 62)



This time period can be used to send more T_p packets.
this time = $T_t + 2T_p$, now, the no. of bits that can be put on the wire in this time are



Window size $W_s = \frac{T_t + 2 \times T_p}{T_t}$

W_s :- window size
 T_t :- transmission time
 T_p :- Propagation time.
 $T_t + 2T_p$ which is called the window size
now, the no. of packets that can be sent in the time $T_t + 2T_p$ are $\frac{T_t + 2T_p}{T_t}$
so the no. of bits in the sequence no. field are $\log_2 \frac{T_t + 2T_p}{T_t}$

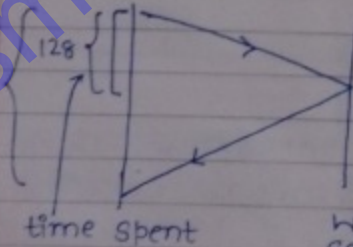
- the min. no. of bits req. in sequence no. for window = $\lceil \log_2 W_s \rceil$

Q. If $T_t = 1 \text{ ms}$, $T_p = 100 \text{ ms}$, then in a sliding window protocol, what is the min. no. of bits req. in sequence no. field.

Ans $W_s = \frac{201}{1} = 201$

no. of bits req = $\lceil \log_2 201 \rceil = 8$

* If we have only 7 bits in sequence no. field, then we can send only 128 bits.



* 128 ~~bits~~ frames will be sent & not complete 201, so we can't send frames 129 to 201 in same window because we have to repeat the sequence no. 000000 for 129th frame before

receiving ack for 0th frame, \therefore we have to wait for ~~20~~ ack of 0th frame which comes after $T_t + 2 \times T_p$

Waiting time = $T_t + 2 \times T_p - 128 \times T_t$
 $= 1 + 2 \times 100 - 128 \times 1$
 $= 201 - 128$
 $= 73$

Why we can't send 129th frame before receiving ack for 0th frame?
 → If we have sent 129th frame before receiving ack for 0th frame, then we have to repeat seq. no. 000000 for 129th frame.
 Q. If the 129th frame arrives at the receiver, as before 0th frame then receiver will not be able to distinguish that whether it is a dup frame or a new frame.

$$W_s = \left(\frac{T_t + 2X T_p}{T_t}, 2^n \right)$$

n :- no. of bits req. in sequence no. field.

efficiency:-
(of stop & wait).

$$\frac{1}{1+2a}$$

$$a = \frac{T_p}{T_t}$$

sending only 1 frame in a period of $1+2a$.

[efficiency is total time spent in transmission divided by total time cycle $(T_t + 2X T_p)$]

Sliding window protocol (efficiency):-

$$\frac{W_s}{1+2a}$$

efficiency:-

$$\frac{W_s \times T_t}{T_t + 2X T_p} = \frac{W_s \times T_t}{T_t + 2X T_p} = \frac{W_s}{1+2a}$$

total no. of packets sent in the interval of $1+2T_p$

Q. In above question, what is the efficiency.
Ans $\frac{128 \times T_t}{T_t + 2 \times 100} = \frac{128 \times 1}{1 + 2 \times 100} = \frac{128}{201}$

Bandwidth Utilization/

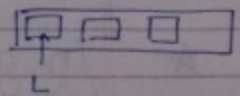
$$\eta \times B \quad \left[\begin{array}{l} \eta:- \text{efficiency} \\ B:- \text{bandwidth} \end{array} \right]$$

Throughput:- no. of bits

derivation:- $(W_s \times L)$ → total no. of bits in each window

L :- no. of bits in each frame

$$\frac{W_s \times L}{T_t + 2X T_p} \rightarrow \text{no. of bits sent per unit of time}$$



$$\left(\frac{W_s \times L/B}{T_t + 2X T_p} \times B \right) \quad \frac{L}{B} = T_t$$

$$= \boxed{\eta \times B}$$

★ Sliding window protocol is used for flow control only, if there are any errors, then we need error control also. In sliding window protocol, error control is implemented in 2 ways:-

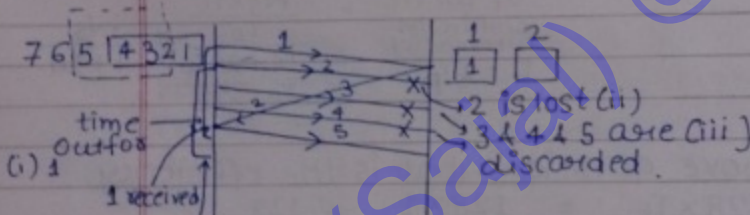
(i) GBN

(ii) SR

for GBN:-

$$\frac{N}{1+2a} = \eta$$

$$WR = 1$$



(iv)

time-out for 2nd packet, so

sender knows that packets in window

2, 3, 4, 5 are lost,

so sender will

go back 4 from the time out i.e.

5, from the last frame

sent, e.g. in this case from 5.)

★ In go back N, N indicates sender window size, if N=10, then it is go back 10.

★ efficiency of go back N = $\frac{N}{1+2a}$

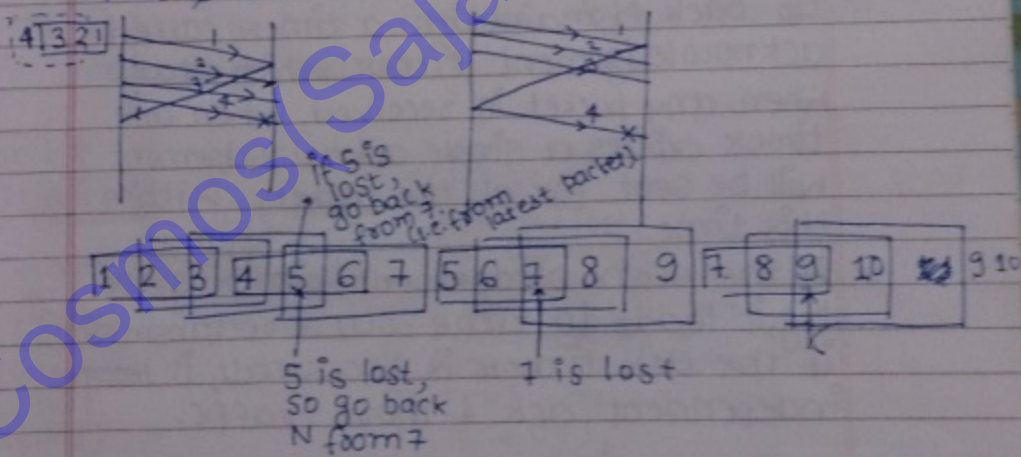
[from sliding window protocol, $\eta = \frac{W_s}{1+2a}$ & $W_s = N$ in this case.]

Q. If $T_t = 1 \text{ ms}$, $T_p = 19.5 \text{ ms}$, then what is the efficiency of GB 10.

Ans $\eta = \frac{10}{1 + 2 \times 19.5} = \frac{10}{1 + 39} = 0.25$

Q. In GB N, receiver window size is 1, which always mean that receiver will be waiting for in-order packet, which means that any out of order packet will be discarded, so sender has to go back N & retransmit entire window if there is any time out.

Q. If in GB N, $N=3$, 10 packets are to be transmitted & every 5th packet is lost, then what is the total no. of transmissions req.

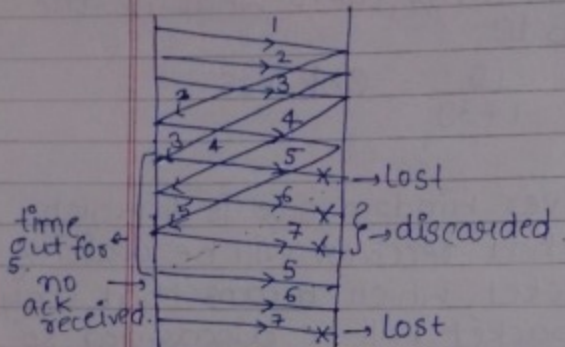


(18) transmissions.

In case of stop & wait:-

1 2 3 4 5 5 6 7 8 9 9 10

(12) transmissions.



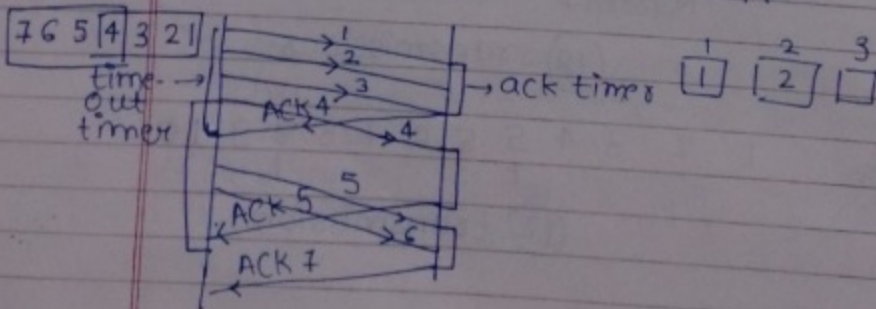
★ Acknowledgements are of 2 types :-

- (i) Independent.
- (ii) Cumulative.

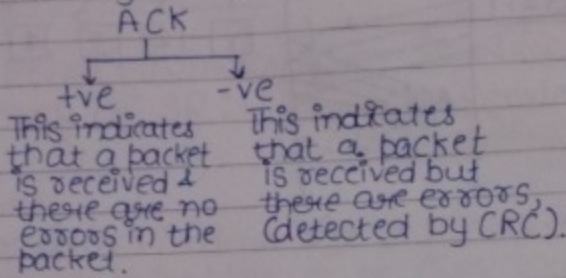
★ In Go back N, the acknowledgements are cumulative.

Go back N maintains a timer called acknowledgement timer which starts when any packet is received, when ack timer expires a single acknowledgement will be sent for all the packets within this time.

If the acknowledgement timer is too big, it leads to time-out & retransmission, if the ack timer is too small, it becomes independent ack & more traffic.

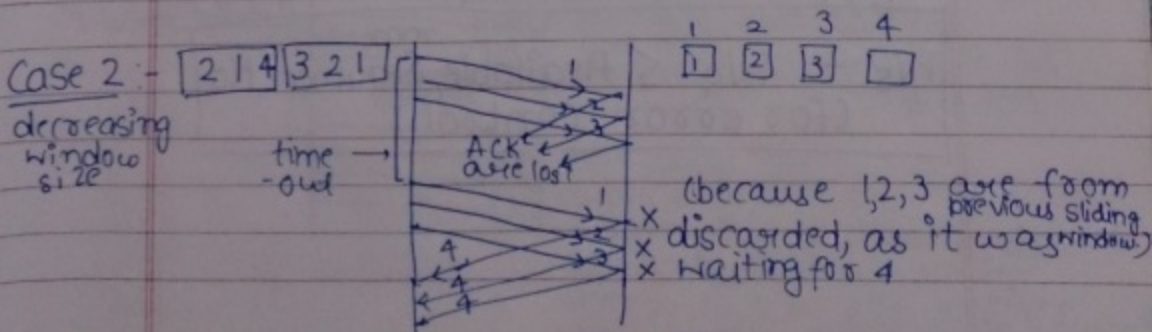
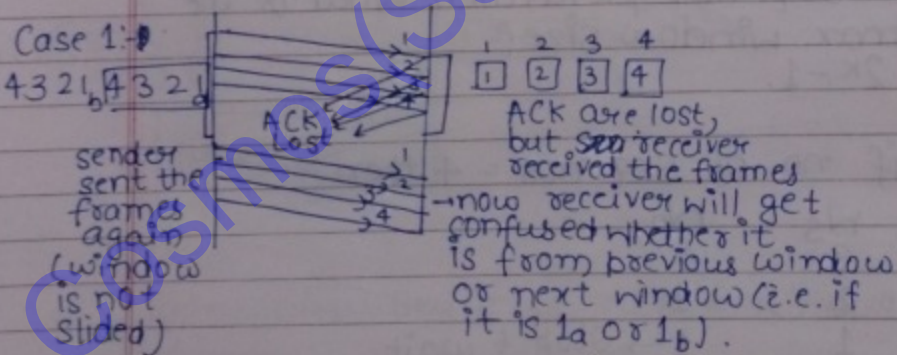


★ There are 2 types of Acknowledgements:-



★ Go back N receiver uses +ve ACK only, i.e. if a packet is corrupted, Go back N receiver will silently discard & all subsequent packets will also be discarded, so sender will retransmit entire window after time-out.

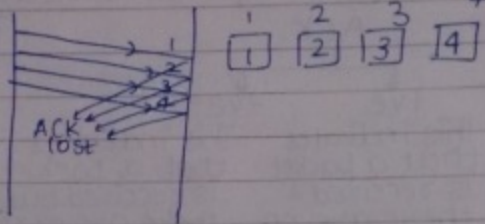
★ Relation b/w window sizes & sequence nos. in go back N:-



Case 3:-

4 3 2 1 5

4321



Q1. If max. no. of Sequence no. available is N , then what is the max. window size?

Ans. $W_s = N - 1$.

Q2. If sender window size is N , then what is the min. no. of seq. no. req.?

Ans. $N + 1$

Q3. If 'k' is max. no. of bits available in seq. no. field, then what is the max. window size?

Ans. $2^k - 1$.

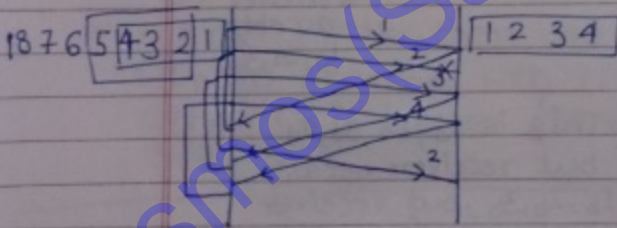
* If no. of seq. nos. = 4, then

W_s	W_R
3	1
2	1
1	1 → Stop & wait.

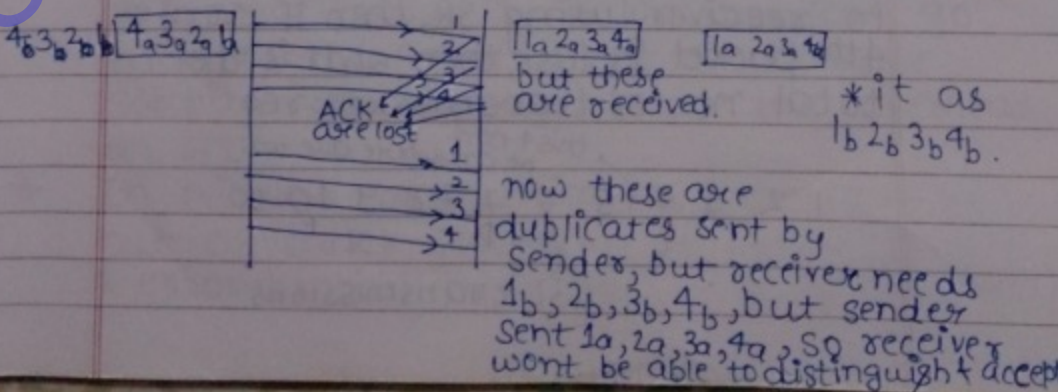
$W_s + W_R \leq \text{Available, Sequence No. (for error control)}$

Selective Repeat

- ★ In SR, sender window size is $N (> 1)$.
- ★ efficiency of SR :- $\frac{N}{1+2a}$ N :- sender window size.
- ★ ~~Whenever~~
In SR, receiver window size > 1 (equal to sender window), which implies a receiver can even accept out of order packets, so whenever a packet is lost there will be a time out at the sender & sender will send ^{cor packet} only lost packet selectively.
- ★ In SR, acknowledgements are independent, so acknowledgement timer is zero.
- ★ In SR, a receiver will send -ve acknowledgement if a packet is received but it has 'bit errors'.



Rein. b/w sequence nos. & window sizes:-



* If sender window size = receiver window size = N , then what is the min. no. of seq. no. required

Ans. $2N$

* If max. seq. no. is N , then sender

* If k is the max. no. of bits in the sequence no. field, then what is the max. h_s & h_r .

$$h_s = 2^k - 1$$

$$h_r = 2^k - 1$$

these 2 windows should have completely diff. seq. nos.

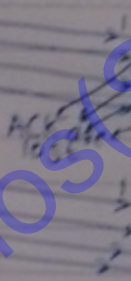
3 2 1 5 4 3 2 1

1 2 3 4

5 1 2 3
1a 2a 3a

* receiver accepts
1a, 2a, 3a:

* So receiver must have two windows of entire diff. seq. no.



This is 1a, 2a, 3a, 4a but receiver needs 1b, 2b, 3b, but receiver won't be able to distinguish b/w 1a, 2a, 3a & 1b, 2b, 3b, so*

Q. If 10 packets are sent from sender to receiver using SR, then if every 4th packet is lost, then what is the total no. of transmissions req.

1 2 3 4 + 5 6 7 7 8 9 10 10
 ↑ ↑ ↑
 this 4 can be anywhere else too.

(13) transmissions

★ Comparison b/w Stop & Wait, SR, Go back N.

(i) Sequence no. :-

2 in Stop & Wait

$N+1$ in Go back N.

⇒ $2N$ in SR, $k=0,1,2,\dots$ [why &?]]

(ii) buffers req. :-

2 in Stop & wait

$N+1$ in GBN (N for sender & 1 for receiver).

$2N$ in SR.

(iii) Retransmissions are less in S4W, & SR

more in GBN,

B/w req. is more in GBN.

(iv) Sorting logic & searching logic is req. in SR,
so more CPU time is req. in SR.

Q If B/w is moderate, buffers are sufficient & CPU's are powerful, then SR is preferable.

→ If B/w is sufficient, buffers are moderate & CPU's are slow, then Go back N.

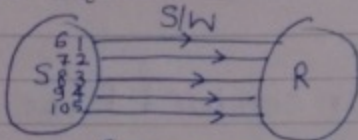
→ In a channel with high error probability, then SR is better because no. of retransmissions are less.

★ In wireless communication, out-of-order sequencing of frame arrival won't happen, & error probability is low, therefore go back N is preferred. (& not SR, because searching & sorting will increase the overhead on CPU).

★ In wireless communication, there may be out-of-order sequencing of frame arrival & error probability is high, ∴ SR is preferable.

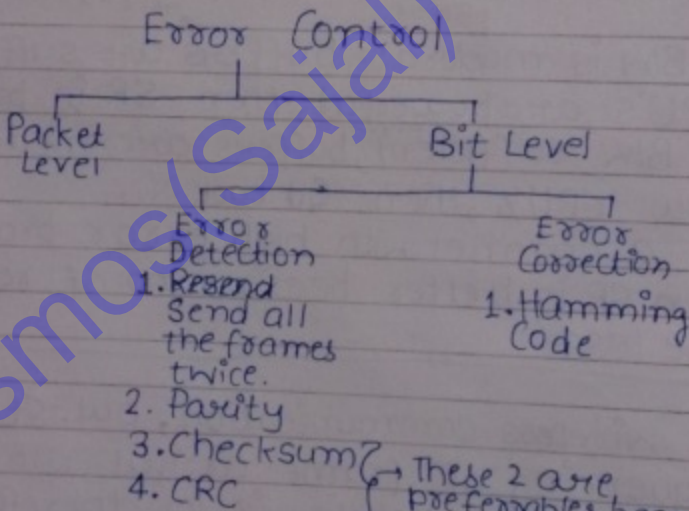
★ If there are N channels b/w sender & receiver & every channel is operated using stop & wait, then what is the overall effect equal to?

→ It is equal to SR .



If there is an error in receiving '1', then only frame 1 is resent & not all 1 to 5, $\therefore SR$.

★ Go back N is also called conservative protocol.



→ These 2 are preferable, because Hamming Code increases the overhead, & Checksum & CRC let forces the sender to retransmit the packets only at the time of detection of errors

CRC (Cyclic Redundancy Check) :-

Cyclic codes

we are sending extra bits.

for checking purposes.

Exclusive-OR (XOR) (Modulo-2 sum)

e.g.

$$\begin{array}{r} 1 \\ \hline 10 \end{array}$$

$(10)^2 \text{ mod } 2 = 0$

$$\begin{array}{r} 1 \\ \hline 11 \end{array}$$

$(11)^2 \text{ mod } 2 = 1$

CRC Generator = 1101, if we have to send 101011

$$\begin{array}{r} 1101 \overline{) 101011000} \rightarrow \text{added 3 bits} \\ \underline{1101} \\ 011111000 \\ \underline{1101} \\ 00101000 \\ \underline{1101} \\ 011100 \\ \underline{1101} \\ 00110 \rightarrow \text{add this to } 101011000 \end{array}$$

$$\begin{array}{r} 1101 \overline{) 101011110} \rightarrow \text{so this no. will be sent.} \\ \underline{1101} \\ 011111110 \\ \underline{1101} \\ 00101110 \\ \underline{1101} \\ 011010 \\ \underline{1101} \\ 00000 \end{array}$$

remainder is 0, so there is no error.

in case of error

$$\begin{array}{r} 1101 \overline{) 111011110} \\ \underline{1101} \\ 1111110 \\ \underline{1101} \\ 001011 \\ \underline{1101} \\ 01100 \\ \underline{1101} \\ 1 \end{array}$$

remainder is not zero. (so there is an error.)

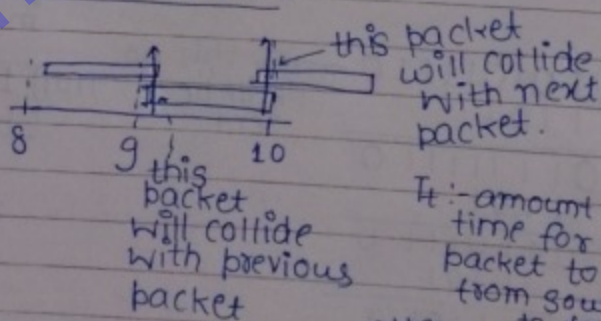
Access Control :-

In a shared link, many stations will share a common medium & try to transmit their data at the same time, \therefore some access control methods are required to control the access to the shared medium.

1. Aloha

In this protocol, any station can send data any time, \therefore collisions are possible. acknowledgements are used in aloha, so if an ack is not received, it indicates that the data might have collided & so retransmission is required.

Before retransmitting, a sender must wait for random amount of time called backoff time.

Vulnerable time :-

T_t :- amount of time for a packet to go from source to destination.

→ So, during 9 & 10, no packet should be there b/w time period > 8 & < 10 .

So, vulnerable time is $2 \times T_t$.

→ Vulnerable time means in this period, if there is some other packet, then collision will happen.

$$\text{Load} = \lambda \times T_{\text{slot}}$$

\downarrow \downarrow
 no. of time req. / sec. slot

$$\text{efficiency :- } \eta = G \times e^{-2G}$$

where G :- no. of requests per time slot
 where time slot = T_t .

$$\frac{d\eta}{dG} = e^{-2G} + G e^{-2G}(-2)$$

$$0 = e^{-2G} - 2G e^{-2G}$$

$$\Rightarrow \boxed{G = \frac{1}{2}} \quad (\text{so, max. efficiency is when } G = \frac{1}{2})$$

i.e. 1 request per 2 time slots,

& max. efficiency $\eta_{\text{max}} = 0.184$.

$$\begin{aligned} \star \text{ Blw utilization} &= \eta \times \text{Bandwidth} \\ &= 0.184 \times 100 \\ &= 18.4 \end{aligned}$$

Q1. If Blw of a shared medium is 100 Mbps, then what is actual bandwidth available in Aloha.

$$\begin{aligned} \text{Ans. } &0.184 \times 100 \\ &= 18.4 \text{ Mbps.} \end{aligned}$$

Q2. In the above ques., in that LAN if every station wants 1 Kbps, then how many max. stations can be placed in the LAN.

$$\begin{aligned} \text{Ans. } &\frac{18400 \text{ kbps}}{1 \text{ kbps}} = 18,400 \text{ stations.} \end{aligned}$$

If every station wants 1 Mbps, then max. no. of stations = $\boxed{18}$.

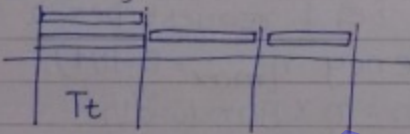
classmate

Date _____

Page _____

Slotted ALOHA :-

Time is divided into slots where each slot is T_t & all the stations are forced to transmit only at the beginning of a time slot, ∴



Vulnerable Time = T_t

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

$$\therefore G = 1$$

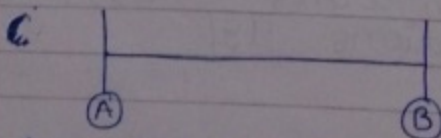
$$\therefore \eta_{\max} = 36.8\%$$

Q1. If B/w = 100 Mbps in a slotted aloha & every station needs 1 Kbps, then what is the max. no. of stations that can be placed in the LAN.

Ans. 36,800.

[Note] :- Aloha is obsolete.

★ In worst case :-



When the data from A is about to reach B, if at that pt. collision occurs, then it will take another T_p collision time

to reach B.

$$\text{So, } T_t \geq 2 \times T_p$$

$$\frac{L}{B} \geq 2 \times T_p$$

$$L \geq B \times 2 \times T_p \quad \text{CSMA/CD}$$

CSMA/CD

In this, any station can transmit data at any time, but before transmitting the data a station should sense the carrier. If the carrier is free, then data should be transmitted, else the station should refrain.

There are no acknowledgements, ∴ a sender should detect a collision while transmitting the data (if there are any). The condition for collision detection is $T_t \geq 2 \times T_p$.

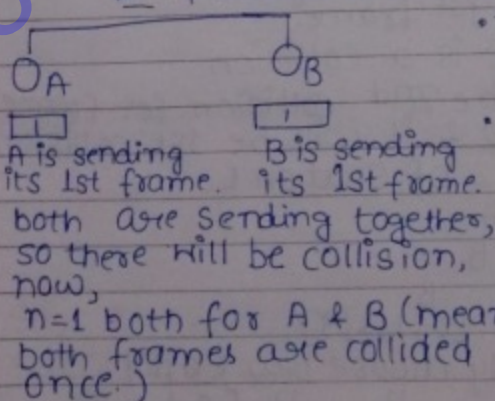
$$\Rightarrow L \geq 2 \times T_p \times B$$

if $T_p = 1 \text{ ms}$ & $B = 1 \text{ Mbps}$, then what is min. L for collision detection.

$$L \geq 2 \times 1 \times 10^{-3} \times 10^6$$

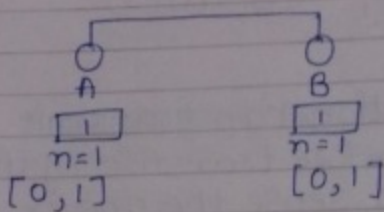
$$L \geq 2000 \text{ bits}$$

Exponential Backoff Algo:-



- This algo gives waiting time for stations involved in collision.
- This algo works for only 2 stations, so it is called binary backoff algorithm.
- Waiting time for a station is $K \times T_{slot}$. Where K belongs to $[0, 2^n - 1]$, where n is collision no. for a frame.

now, the algo will randomly choose K value for frame at station 1 in b/w $[0, 2^1 - 1] = [0, 1]$ & $[0, 1]$ for frame at station 2.

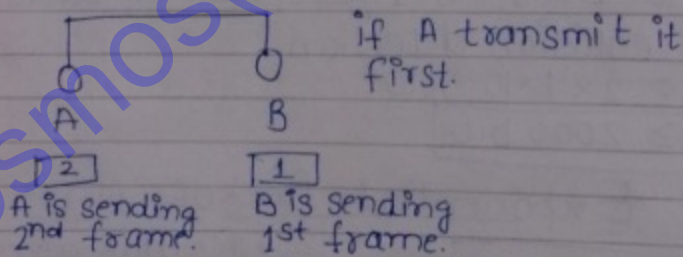


A	B		
0	0	→ Collision	$p(C) = \frac{1}{2}$
0	1	→ A	$p(A) = \frac{1}{4}$
1	0	→ B	$p(B) = \frac{1}{4}$
1	1	→ Collision	

$$W_T = K \times T_{slot}$$

\downarrow constant waiting time
 K value chosen from $[0, 1]$

★



suppose there is a collision.

then it will be 2nd collision for frame 1 at station 2 & it will be 1st collision for frame 2 of station 1.

so $n=1$ for A
& $n=2$ for B

A $[0, 1]$ B $[0, 1, 2, 3]$
 A will randomly choose b/w $[0, 1]$
 B " " " " " " $[0, 1, 2, 3]$

A	B
0	0
0	1
0	2
0	3
1	0
1	1
1	2
1	3

$$P(C) = 25\%$$

$$P(A) = 62.5\%$$

$$P(B) = 12.5\%$$

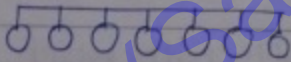
this means
A will transmit
in 0x1x0

this means
A will transmit in
1x0x1

★ So, collision probability decreases as no. of collisions increases.

★ The main disadvantage is Capture Effect, in which 1 frame at one station collide again & again & won't be transmitted at all.

★



The probability of successful transmission = $n p (1-p)^{n-1}$ [when only 1 station transmit & other (n-1) stations doesn't.]

$$\text{Prob} = n p (1-p)^{n-1}$$

$$\frac{d(\text{prob})}{dp} = 0 \text{ (for max. } p)$$

$$dp$$

$$\Rightarrow p = \frac{1}{n}$$

$$\text{now, max. prob} = n \times \frac{1}{n} \left(1 - \frac{1}{n}\right)^{n-1} = \boxed{\frac{1}{e}}$$

max. probability of success.

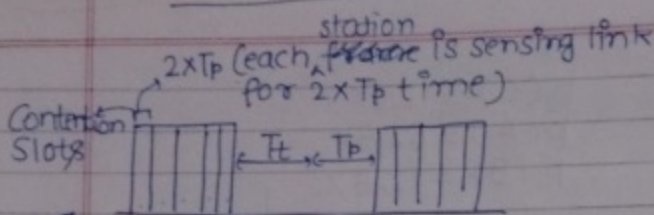
★ So, e tries are required for 1st successful transmission.

Efficiency of CSMA/CD

classmate

Date _____

Page _____



the frame will be sent e -times.
 & will suffer collision & successful transmission will take place after e trials.

during this time the media sender is transmitting its data for at least $2xT_p$ time & collisions are taking place for e times.
 total time = eT_p

★ If there are n stations, connected by shared medium, then medium will be successfully used only when one station transmits the data & remaining station refrain.

let ' p ' be the probability with which a station ~~is sending~~ ^{wants to} send data, then

prob. of success p

$$p = np(1-p)^{n-1}$$

the value of p is maximum when

$$p = \frac{1}{n}$$

$$P_{s, \max} = \frac{1}{e}$$

we need on average e no. of tries before 1st success.

∴ η can be analysed as follows

worst case time for contention slots = $2xT_p$

$$\eta = \frac{T_t}{T_t + T_p + e \times 2 \times T_p}$$

$$= \frac{1}{1 + (2e + 1)a} = \boxed{\frac{1}{1 + 6.44a}}$$

$$\eta = \frac{1}{1 + 6.44 \times \frac{d}{v} \times \frac{B}{L}}$$

- ★ if distance increases, efficiency decreases (no. of collisions are more.)
- ★ if ~~dist~~ Length increases (of packets), then no. of collisions decreases.
i.e. if we have small sized packets, no. of contention slots increases.

TDM (Time Division Multiplexing) :-

- ★ In TDM, time is divided into slots & each slot is given to one station in a round robin manner.

	T_{t1}	T_{p1}	T_{t2}	T_{p2}	T_{t3}	T_{p3}	T_{t4}	T_{p4}
total $N T_t$ in $N(T_t + T_p)$	①		②		③		④	

$$\eta = \frac{N \times T_t}{N(T_t + T_p)} = \frac{1}{1 + a}$$

- ★ If $T_t = 1 \text{ ms}$, $T_p = 1 \text{ ms}$, then what is the η ?
in TDM :-
 $\eta = 50\%$
- ★ If $B/w = 4 \text{ Mbps}$, then what is B/w utilization?
→ 2 Mbps .
- ★ If every station wants 1 Kbps , then how many stations can be placed in the LAN
at max.
→ 2000 .

Date

classmate

Date

Page

29.09.12

Q1. (i) In a TDM n/w if $T_b = 2\text{ms}$ & $T_t = 1\text{ms}$, then what is the efficiency? (33.33%)

(ii) If $B/w = 3\text{Mbps}$, then eff. b/w.

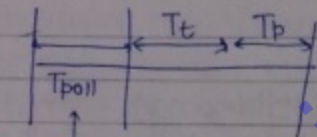
If every station needs 1Kbps , then how many stations?

Ans. (i) 33.3%

(ii) eff. b/w = 1Mbps

(iii) no. of stations = $\frac{1\text{Mbps}}{1\text{Kbps}} = 1000$

★



time taken by algo to decide which station to transmit next

$$\eta = \frac{T_t}{T_{poll} + T_t + T_b}$$

Note: - If T_b is not given, then consider it as zero.

★ Delays:

→ 10 bit time

it means that the time it takes for 1st bit to reach the destⁿ, we can transmit 10 bits.

Token passing in Token ring:-

• Bit Delay:-

If time is given in bits, we can convert into seconds by dividing with blw, b bit delay indicates the time taken to transmit b bits.

Conversions:-

- bit delay $\xrightarrow[\text{divide by blw}]{\text{multiply by blw}}$ delay (in sec.)
- delay (in meters) $\xrightarrow[\text{divide by velocity}]{\text{multiply by } v}$ delay (in sec.)

• Ring Latency:-

It is the time taken by a bit to go around the ring & return to the same point. total

$$\text{ring latency} = \frac{d}{v} + \frac{N \times b}{B} \text{ sec.}$$

$$= \frac{d}{v} \times B + N \times b \text{ bits}$$

d:- length of the wire

v:- velocity

N:- no. of stations.

B:- Bandwidth

b:- bit delay at each station

(i.e. it takes for each station to transmit the bit back to the wire).

Time taken for the token to come back to its initial position:-

- (i) Time taken for 1 station to hold the token =
time taken for all the bits of that station to come back to the sender

$$\text{Station} = T_t + T_{RL}$$

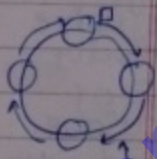
↓ transmission time ↗ Ring Latency

- (ii) If we assume all stations are willing to transmit, then:-
time taken for all stations to hold it = $N \times (T_t + T_{RL})$

- (iii) Now, token has to travel from one station to other station & come back to original position:-

$$= T_p \text{ (Propagation Time)}$$

$$= \frac{d}{v} \quad \begin{array}{l} d \rightarrow \text{length of the wire} \\ v \rightarrow \text{velocity} \end{array}$$



time taken for token to travel from one station to next along the wire.

So, total time = $N(T_p + T_{RL}) + T_p$
if $T_b = 0$

$$\therefore N(T_p + T_t) + T_p$$

(sum up all the time.)

$$\therefore \text{efficiency} = \frac{\text{total transmission time}}{\text{total cycle time}}$$

$$= \frac{N \times T_t}{N(T_p + T_t) + T_p} = \frac{1}{1 + \frac{(N+1)T_p}{N}}$$

Delayed Token Reinsertion

In this strategy, data is transmitted & allowed to take a round & then removed & only after that token is released.

In this case, token holding time = $T_t + T_{RL}$

→ if bit we assume bit delay (b) = 0.

$$T_{RL} = T_p \left(\frac{d}{v} \right)$$

$$T_{HT} = T_t + T_p$$

→ Total cycle time :- is the time taken by the token to be seen by all the stations & coming back to the same point :-

$$N \times (T_{HT}) + T_p$$

→ Total Transmission time in this total cycle

$$\text{time} = [N \times T_t]$$

$$\eta = \frac{N \times T_t}{N \times (T_{HT}) + T_p} = \frac{1}{1 + \frac{(N+1)a}{N}}$$

Early Token Time

In this strategy, a station will hold the token, & transmit the data & immediately release the time.

→ Token holding time = T_t

→ Total cycle time = $N \times T_t + T_p$

$$\eta = \frac{N \times T_t}{N \times T_t + T_p} = \frac{1}{1 + a/N}$$

Q1. If $T_p = 1 \text{ ms}$, $T_t = 1 \text{ ms}$, $N = 1$, then what is the efficiency in early & delay?

Ans. In delay

In early token reinsestion:-

$$\eta = \frac{1}{1 + a/N} = \frac{1}{1 + 1/1} = 50\%$$

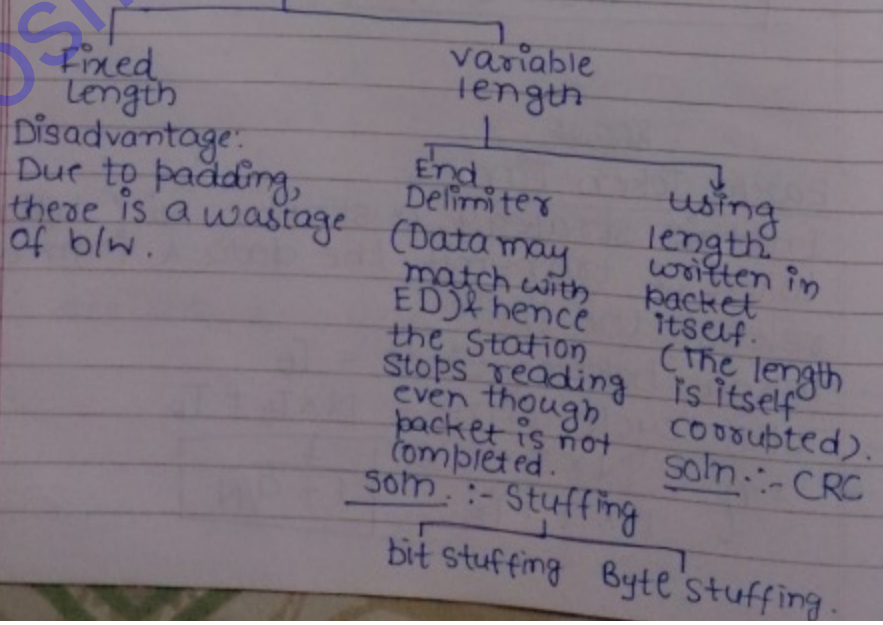
In delay token reinsestion:-

$$\eta = \frac{1 \times 1}{1 \times (1+1) + 1} = \frac{1}{3} = 33\%$$

Note:- Default strategy is early token reinsestion

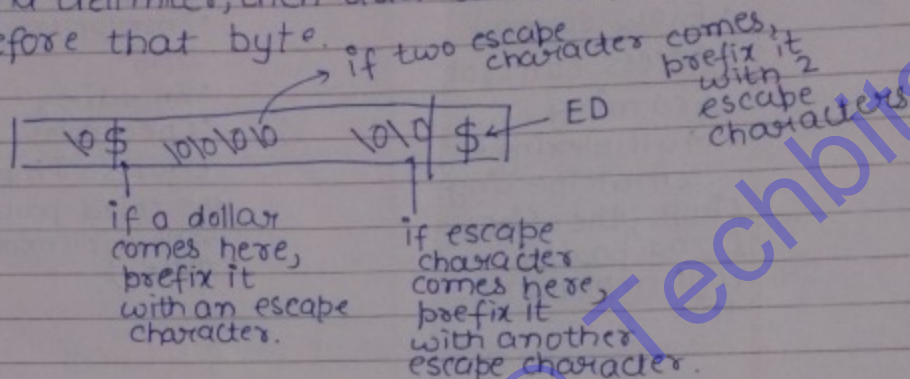
Under heavy load condn. (when all are transferring data) early is better.

Framing (Dividing data into frames/packets)



Byte Stuffing:-

Whenever data matches with the byte used for end delimiter, then add an escape character before that byte.



If two dollar character comes, then prefix both of them with escape character.

$\backslash 0 \$ \backslash 0 \$$.

* Byte stuffing is obsolete.

Bit Stuffing:-

If any pattern matches with end delimiter, then break the pattern by stuffing 1 bit

e.g. (i) ED = 01111

then add a 0 after 3 ones.

(ii) ED = 0111

Message = 011101100

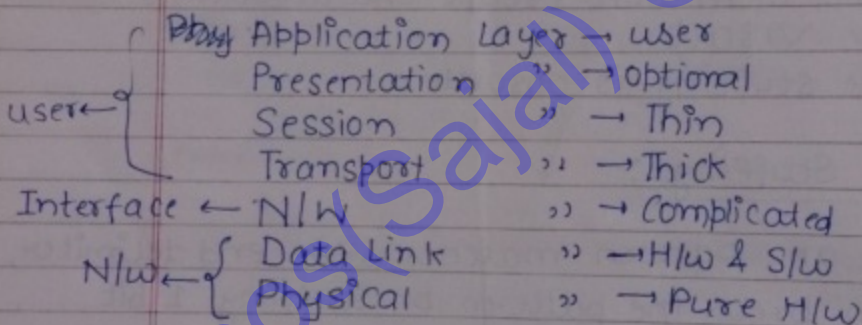
Message sent = 01101011000

[add a zero after 3 ones, no matter whether sequence is 0111 or 0110]

Functions of CN

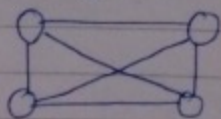
- Error Control
- Flow Control
- Access Control
- Framing
- Multiplexing & Demultiplexing (imp., the first & 2nd party will implement this.)
- Compression
- Encryption
- DNS
- Encoding
- Checkpoint (not so imp., the third party will implement these.)

ISO-OSI:-

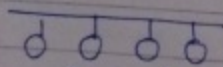


Physical Layer:-

It deals with electrical, mechanical, procedural & functional characteristics of physical links.



Point-to-Point



Broadcast

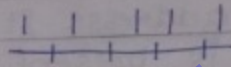
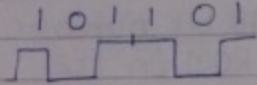
Will take care whether to add start delimiter (preamble).

Modes of transmission:-

- ① Simplex e.g. T.V. service provider
- ② Half Duplex e.g. HAM radio
- ③ Full Duplex e.g. Mobile Communication

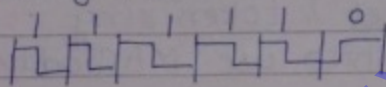
Encoding:-

- Simple encoding:-

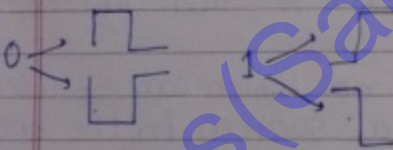


[unable to detect the no. of 1's.]

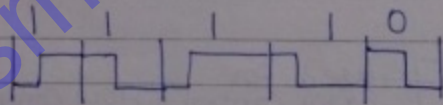
- Using Manchester encoding:-



- Using Differential Manchester:-



e.g.

2. Data Link Layer:-

- (i) Flow Control :- Sliding Window Protocol
- (ii) Error Control :- CRC
- (iii) Access Control :- Aloha, Slotted Aloha, Polling, CSMA/CD
- (iv) Framing :- putting SD & ED
- (v) Physical Addressing

• Flow Control:-

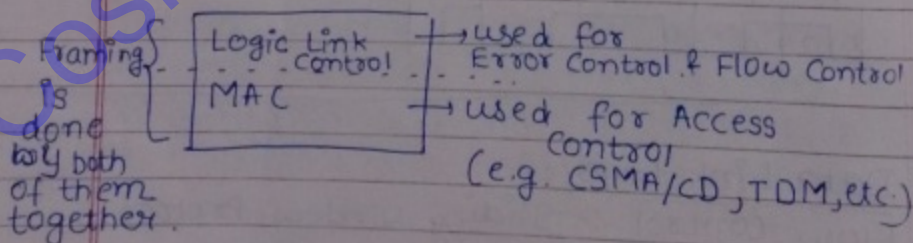
Physical Address:-

Any no. which can be used to identify a station uniquely in a LAN is called Physical address.

MAC address:- It is a 48-bit number which is present in the ROM in the NIC which is unique globally. MAC address can be used as physical address. e.g. ethernet & token ring are the LAN's which use MAC address as physical address.

Apple Talk is a LAN which uses randomly generated nos. as physical address.

Logical Address:- Any no. which can be used to uniquely identify a system in the entire world (or globally) is called Logical Address. e.g. IP address is used address in TCP/IP.



Network Layer :-

- Routing
- Logical Addressing
- ✓ Congestion Control
- Fragmentation

Transport Layer :-

- ✓ End to End connectivity .
- ✓ Service point addressing
- Segmentation.
- ✓ Error Control
- ✓ Flow Control

→ Service point address:- any no. which can be used to identify a process uniquely within a host is called service point address, e.g. port number.

Session Layer :-

- Synchronisation / Checkpointing

Dialog Control :-

Even though the channel is full duplex, we use it as half duplex.

e.g. Web conference

- Checkpointing :-

Presentation Layer

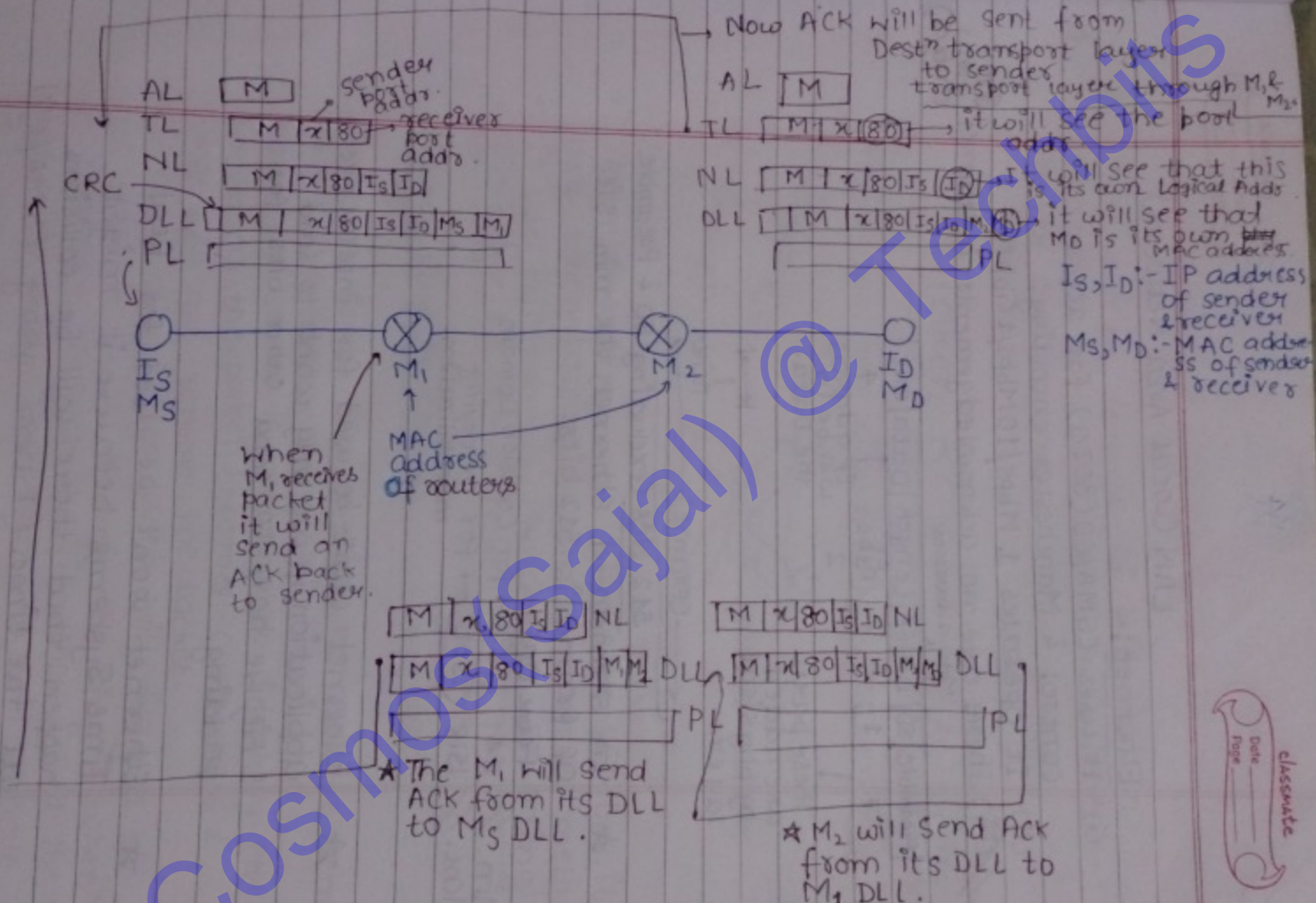
- Encryption
- Compression
- Translation
(e.g. ASCII \rightarrow EBCDIC)

Mainly Session & Presentation layer functions are optional & not needed by all the applications, \therefore these are implemented at Application Layer by the concerned application.

- Application Layer :- Message
- Transport " :- Segment
- Network " :- Datagram
- Data Link " :- Frame
- Physical " :- Single PDU.

★ Diff. b/w flow control of Transport layer & Data link layer:-

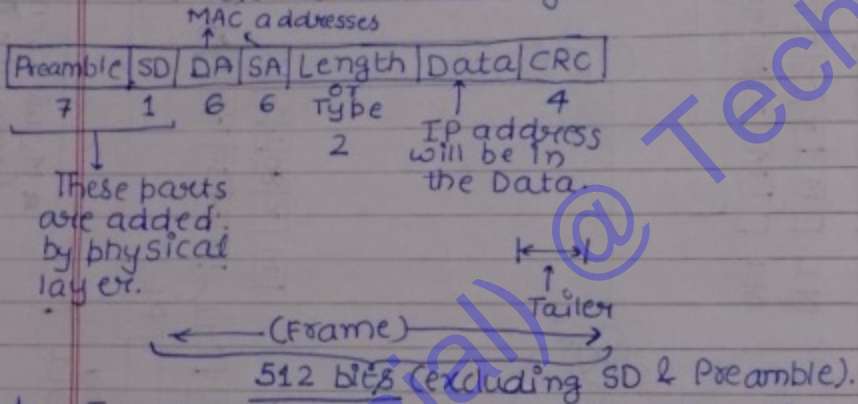
\rightarrow Let assume if the packet is lost at M_2 , though M_2 have sent Ack to M_1 , depicting it has received the packet, but sender's transport layer runs a timer which times out if it doesn't receive Ack from receiver's transport layer.



LAN (Local Area Network)

Ethernet:-

- (i) It uses CSMA/CD (802.3) for access control & Manchester encoding.
- (ii) It operates 1 Mbps/10 Mbps/1 Gbps.
- (iii) There are no acknowledgements in ethernet. (No flow control)



- ★ In standard ethernet, the min. size of frame is 512 bits.

	Frame Data	
Min. ✓	64 (46)	→ for Collision Domain
Max. ✓	1518 (1500)	→ for removing monopolization.

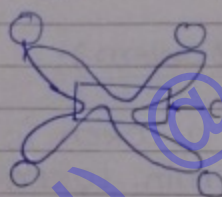
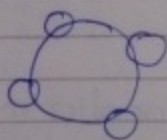
- ★ Ethernet can't be used for interactive application, if we only want to send 1 bit, we have to send other bits for padding.

- ★ Ethernet won't be used for Real Time Systems because it might happen that there will be collisions all the time. (When using CSMA/CD)

- ★ There are no concept of priorities in ethernet, so we can't use it in client-server architecture.

Token Ring (802.5) [not used for practical purposes]

There are two diagrams for token ring:-



central switch

- Token ring operates at 4 Mbps or 16 Mbps, it uses token passing as access control method.
- It uses differential Manchester encoding.
- There are no ACK in token ring.

Token Ring Problem:-

1. When the sender is down & is not able to remove the packet from the ring.
 2. When the sender is not able to identify the packet as the packet becomes corrupted & hence will not remove the packet.
- To solve this, Master comp. will flag a bit 1st time it passes through the ring & when it sees that bit set, it will know that the packet is doing 2nd round & will remove it.

Sender	Receiver
Available	C
0	0 → Initial bit
1	0 → if it is set, then packet is corrupted check error bit if not, then receiver might be busy.
0	1 → copied at sender
1	1 → Invalid

- Initial bit pattern ← 0
- sender has sent, but receiver didn't receive ← 1

- sender has sent & receiver has received ← 0

• sender has not sent & receiver has received.

Token Problem:-

- Whenever a token is lost, monitor will wait for min. token return time to max. token return time.
- Min. token return time = RL (when no station transmits.)
- Max. token return time = Cycle time
= R.L. + N(CH.T.)

Monitor Problem:-

- When monitor itself is down or corrupted.
- Monitor should send AMP packet at regular intervals of time & if AMP packets are not received for sometime then all stations will conduct election & elect the next monitor.

Date

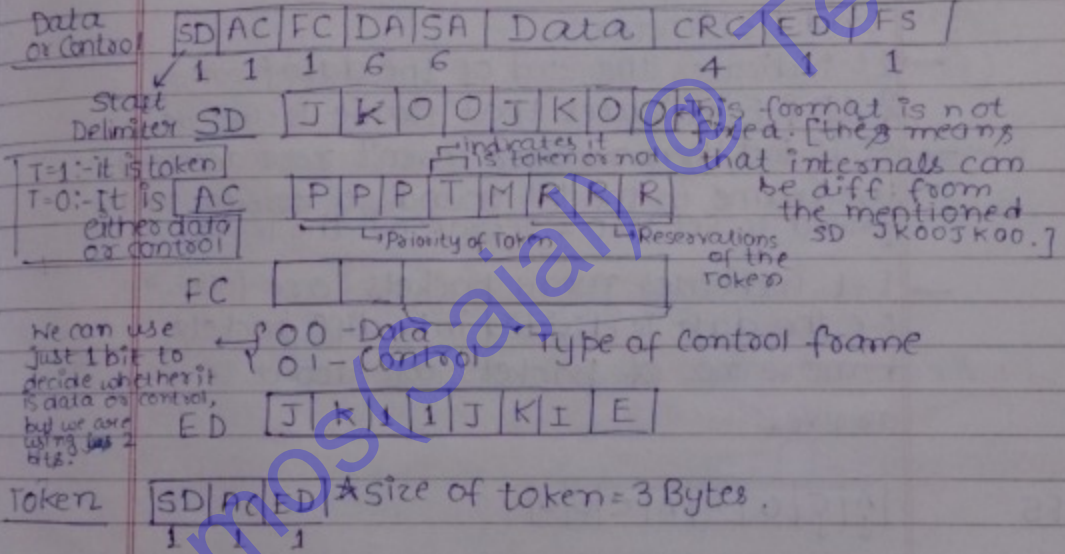
classmate

Date _____
Page _____

30-03-12

- The slw of a monitor could get corrupted & hack in such a way that monitor will be sending just AMP packet & will not do any other task.
- For this problem manual intervention is required.

Frame format (Token ring)



SD:-

Start Delimiter which indicates the beginning of a frame. J & K are line codes which are not used for any valid encoding.

AC:-

This byte is used for access control.

Even though a station has a token, it can't send the data because another station with high priority wants to send data.

M:- indicates that about to a stamp by monitor, if M=1, it indicates monitor has stamped on the packet i.e. the packet has already made a round.

around the ring & is now making a 2nd round).

if $M=0$, then packet will be making its 1st round & monitor will strip it to '1'.

Frame Control:-

End Delimiter:-

- It indicates the end of the data frame & ED & E indicates that frame is corrupted.
- So if $E=1$, then receiver haven't received the frame in the 1st round, so monitor sender need to set $M=0$ & retransmit the packet.
- $I=1$ indicates more packets are following, i.e. the data is divided into diff. packets & more no. of packets are about to arrive.

FS:

A	C	0	0	A	C	0	0
---	---	---	---	---	---	---	---

Q1. Why two copies of A & C in FS?

Ans. Because FS is not included in CRC computation

Q2. Why FS is not included in CRC?

Ans. Because CRC is computed at sender & FS is computed/changed at receiver

Q3. If b/w of a token ring is 4Mbps & token holding time is 1 ms, then what is the max size of frame that can be sent

$$\text{Ans. } \frac{4 \times 2^{20}}{10} \times 4 \times 10^6 \text{ bps} \times 10^{-3} \text{ s} = 4 \times 10^3 \text{ b} \\ = 4000 \text{ bits}$$

$$\& \text{ max size of data} = 4000 \text{ bits} - 21 \times 8 \text{ bits} \\ = \frac{500 \text{ B} - 21}{[4799]}$$

now for 16 Mbps

$$16 \times 10^6 \text{ bps} \times 10^{-3} \text{ s} = 16 \times 10^3 \text{ bits} \\ = 2000 \text{ Bytes (of max frame size)}$$

$$\& \text{ max. data size} = 1979 \text{ Bytes} (2000 - 21)$$

- ★ Min. data size in token ring could be zero bytes (because there will not be any collisions).

$$\text{Capacity of a wise} = \text{Bandwidth} \times T_p$$

- ★ Now, min. length of the wise in token ring can be calculated using capacity of wise. A token ring should be capable of holding atleast 1 token. capacity \geq token size.

$$\therefore \text{length} \geq 24 \text{ bits}$$

$$\text{Blw} \times T_p \geq 24 \text{ bits}$$

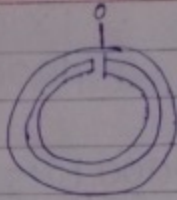
$$\text{or } B \times \frac{d}{v} \geq 24 \text{ bits}$$

- Q4 If a token is of 24 bits, blw is 4 Mbps & velocity of signal in the wise is $2 \times 10^8 \text{ m/s}$, then what is the min. length of token ring

$$\text{Ans. } 24 \times 10^6 \times \frac{d}{v} \geq 24 \times 12$$

$$\frac{24 \times 10^6 \times d}{2 \times 10^8 \times 10^2} \\ d \geq 1200 \text{ m}$$

$$\begin{aligned} B &= 4 \times 10^6 \\ v &= 2 \times 10^8 \text{ m/s} \\ 4 \times 10^6 \times \frac{d}{2 \times 10^8} &\geq 24 \times 12 \\ 16 \times 10^6 \times d &\geq 2880 \\ d &\geq \frac{2880}{16 \times 10^6} \\ &= 18 \times 10^{-6} \text{ m} \\ &= 18 \mu\text{m} \end{aligned}$$



- This means that atleast 200m of wire is req. for the 1st bit to reach the sender & last bit to be transmitted from the sender.
- If wire is less than 200m, then the 1st bit will come back to sender even when the last bit is not transmitted & hence it will be an overlap & hence error takes place.

Q. If we have only 1 km wire, then what is the bit delay that has to be introduced in order to compensate 200m?

Ans. The time req. to travel 200m = $\frac{200}{2 \times 10^8} = 10^{-6}$ s.

now, 4×10^6 bits takes 1 s

10^{-6} s will transfer 4 bits

\therefore 4 bits of buffer has to be introduced in b/w the wire.

Sender

Transport Layer

TCP (reliable)

Network Layer

IP (unreliable)

Receiver

★ The main responsibility of n/w layer is switching.

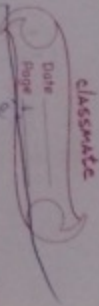
★ The transport layer will first send a notification to receiver telling about the frames/packets that has to be received in future.

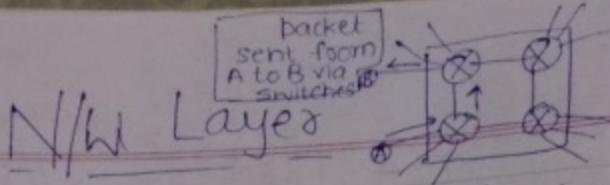
★ The IP which is an unreliable protocol will fragment the packets & send them via the n/w, they may or may not arrive & many-a-times arrive out of order.

★ In this case, the transport layer at receiver will send a +ve or -ve ACK about the data received & the transport layer at sender will send the packets if there was a -ve ACK.

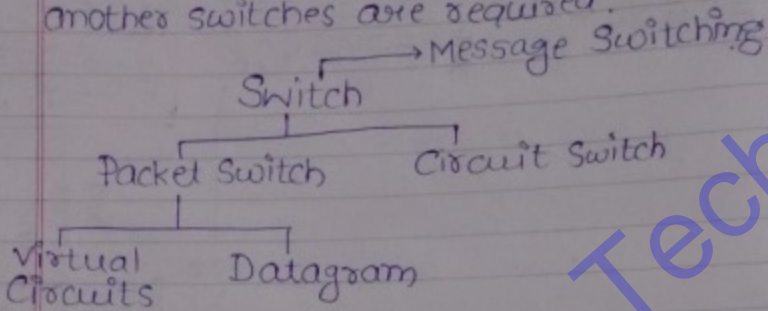
The packets at receiver will arrive -

- out of order
- corrupted
- duplicated
- delayed





- ★ To send data from one n/w to another switches are required.



Circuit Switching

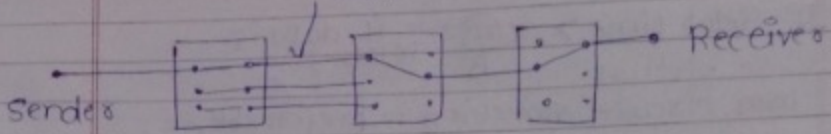
Total time taken for a packet to be sent from sender to receiver:-

$$S + \frac{m}{Bw} + \frac{(x+1)d}{v} + T$$

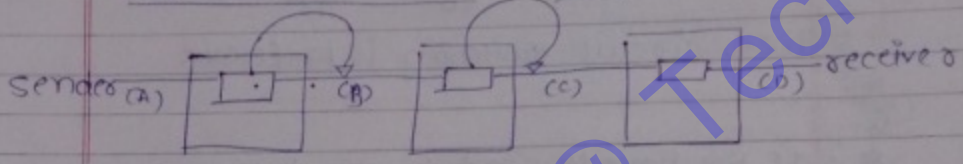
S ↓ time req. to set up the connection (i.e. setting up of the wire path)
 $\frac{m}{Bw}$ ↑ Bw (time req. to put the message on the path.)
 $\frac{(x+1)d}{v}$ ↑ v (velocity) ↓ d (distance b/w two switches) (time req. for packet to travel from sender to receiver)
 T ↓ tear-up time (to tear up the path)

- ★ Header is only req. in setting up time. when we need sender & receiver address to switch & establish the path & header is no longer req. after that.
- ★ We ~~don't~~ The packets will arrive in-order because there is only one path.
- ★ Circuit switching is implemented at physical layer.

Establishment of path.

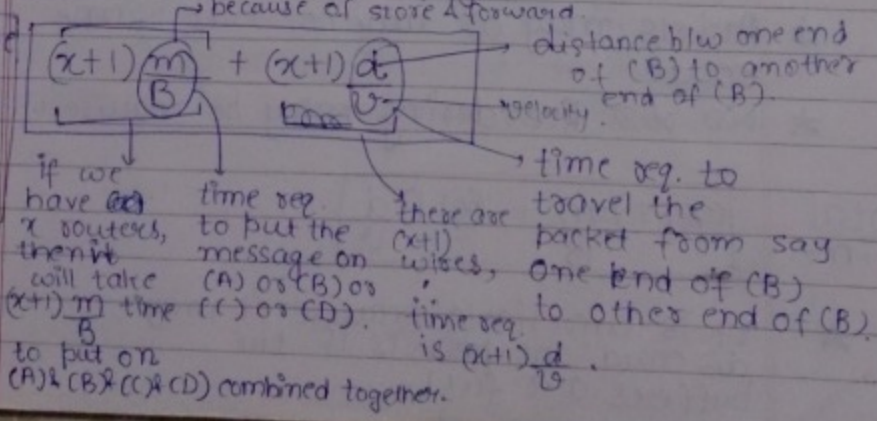


Virtual Circuits (ATM)



- ★ Whenever the 1st packet goes, it will contain header with sender & receiver address & will establish a connection & will tell the routers to allocate some part of buffers for the packets to be arrived in future.
- ★ Header is req. only for the 1st packet & all the remaining packets will follow the same route.
- ★ All the packets will take the same route.
- ★ Out of order is not possible.
- ★ If we have a connection-oriented service at network layer, we don't need connection-oriented service at transport layer.

Setup time & Tear-up time is negligible.



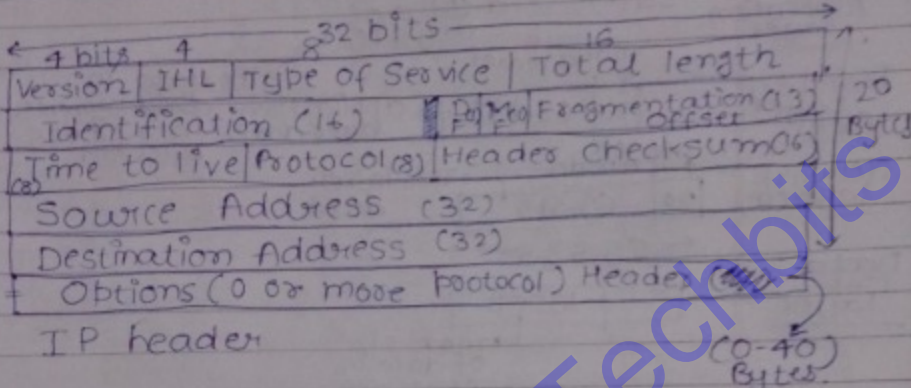
- ★ If Setup time $> x \frac{m}{B}$ i.e. if data is if less
virtual then, circuit switching is preferable.
- ★ if setup time $< x \frac{m}{B}$ i.e. data is more
then, circuit switching is preferable.
- ★ If we have to send bulky data, then circuit switching is better.
- ★ If we have to send small amounts of data, virtual circuits are better.

Datagram Circuits (IP)

- ★ No resource reservation is done, i.e. buffers are not reserved for future packets.
- ★ All packets contain header (as they may take diff. paths, so they must contain sender & receiver address for routers.)
- ★ Packets may or may not take same route.
- ★ No reseq. Reordering may be required.

Total time = $\boxed{\frac{(x+1)m}{B} + \frac{(x+1)d}{v}}$

- ★ It is unreliable, because it may discard the packets if the buffers are full.

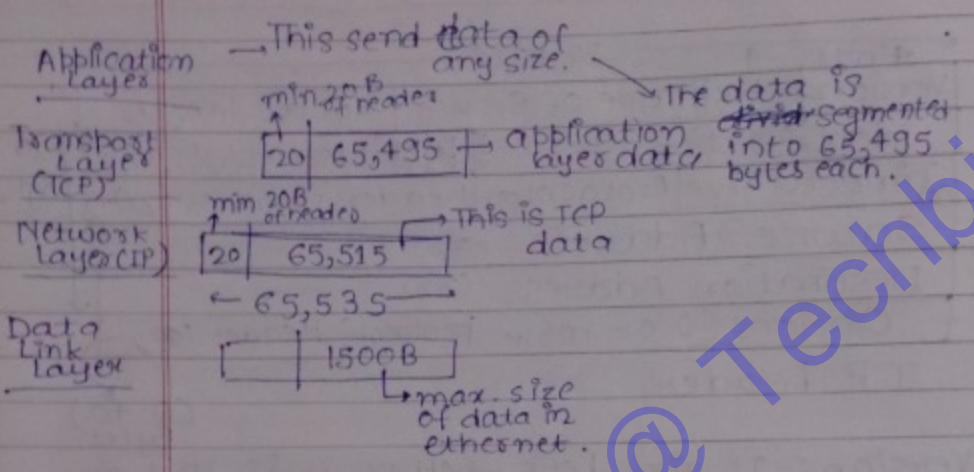


- (i) Version :- It gives tells whether it is IPv4 or IPv6.
- (ii) IHL (IP header length) :- 1 bit indicates 4 Byte length, because max. header size = 60B, & 15 bits are req. to represent them, each bit indicates 4B.
 \therefore IHL = 1010 means header is of $10 \times 4 = 40$ bytes.

This field indicates size of the header in terms of 4 bytes.

- if IHL = 26, IHLF = 7
- if IHL = 24, IHLF = 6
- if IHLF = 10, IHL = 40
- if IHLF = 7, IHL = 28.

- (iii) Total Length :- it is a 16 bit field which indicates total size of IP header + IP data which can be a maximum of $2^{16} - 1$.



★ If the max. no. that can be put into offset is 65,514, when the last frame size is just 1B & 65,514 B is ahead of it, so, we need to store 2^{16} B data in 13 bit field, so using scaling $\frac{2^{16}}{2^{13}} = 8$. so, take the actual

201 + 1000 + 200 | 20 Transport Layer

MF=0
Offset=0
Length=1421

To Network layer

MF=0
offset=122
length=445

425 | 20

488 | 20

488 | 20 N/w layer

but last bit need not to be multiple of 8, as there is no frame after it, hence its size won't be written in any frag offset.

MF=1
off=188-61
length=506

MF=1
off=0
length=506

→ This is further fragmented into frames at data link layer

120 | 20 Data link layer

MF=1
offset=107
length=110

1184 | 20

MF=1
offset=84
length=204

164 | 20

MF=1
offset=61
length=204

Since n/w layer has IP implemented in it, & IP has fragmentation offset of just 13 bits & the max size it can hold is 65511, so we need 16 bits, but we have only 13 bits,
 $\frac{2^{16}}{2^{13}} = 8 \text{ bits}$
 So 8 bits will depict 8 bits of data.

∴ we have to divide the data in the multiples of 8 only.

Now, the packets will be made in-order at receiver using Offset no.
 e.g, if we have
 0 84 61 107

Overhead of fragmentation

1401+20 at transport layer at sender
 & at receiver we have 1401+20x5
 ∴ overhead =
 $1401 + 20 \times 5 - 1401 - 20$
 $= 4 \times 20 \text{ B}$

classmate
 Date _____
 Page _____

• Efficiency (η) =
$$\frac{\text{Total useful bytes}}{\text{Total bytes received}} = \frac{1401}{1401 + 5 \times 20} = \frac{1401}{1501}$$

• B/W utilization/effec. b/w/throughput:

$$\eta \times B/W$$

Q. Where should the fragments be reassembled?

Ans. At the destination.

Q. Why not at immediate routers?

Ans. ① Further fragmentation may be required.

② Since ^{it is datagram service} every ~~packet~~ datagrams may not take same route.

Q. How can a receiver know that a datagram is not fragmented?

Ans. If MF=0 & offset=0 for some datagram.

Q. If a datagram is fragmented, how can a receiver ~~understand~~ identify all the fragments of a datagram?

Ans. Identification no.

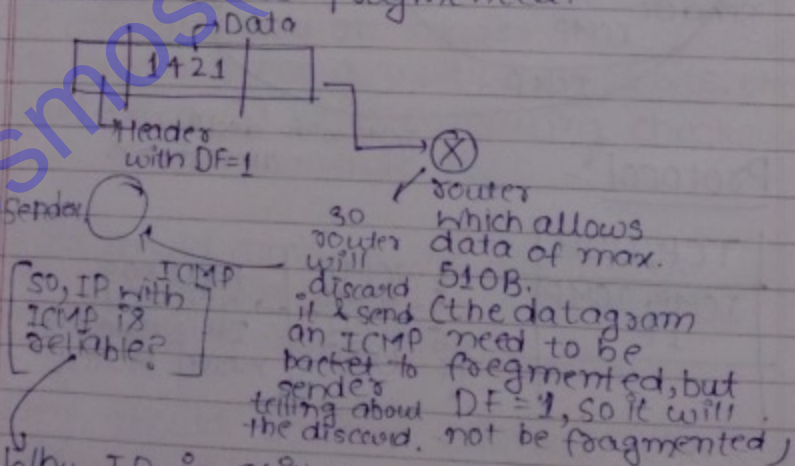
	MF	off
not fragmented	0	0
first fragment	1	0
last fragment	0	!0
intermediate fragment	1	!0

Reassembly Algorithm:-

- (i) Identify that the datagram is fragmented (MF \neq 0, or, Offset \neq 0, or, both \neq 0).
- (ii) Collect all the fragments having same identification no.
- (iii) Identify the 1st fragment. (Offset = 0).
- (iv) Count the no. of data bytes in that fragment (Total length - Header length). Divide it by 8. (Let it be x).
- (v) Search for the fragment having x as its offset.
- (vi) Repeat the above two steps until MF = 0.

DF (Do not fragment)

- If DF is set, it indicates that the datagram should not be fragmented.



- Why IP is still unreliable even with the support of ICMP?

Ans. Because if ICMP is discarded/lost, then no ICMP for it is generated.

Time to Live:

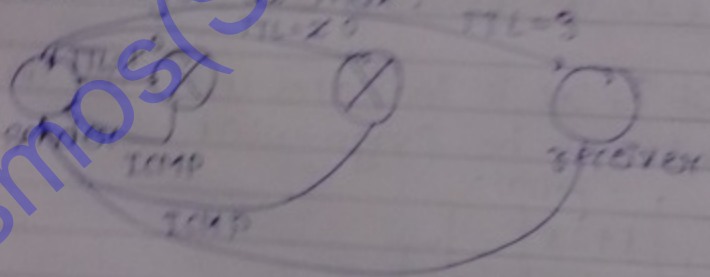
- It is used to prevent a packet from infinite looping.

* This pro

Prevent packet looping
It will

Prevent will give all the intermediate routers b/w a source & a destination.

- Keep incrementing TTL until we get ICMP packet with destination unreachable message (then we give wrong port no.) & port no. can't be read by transport layer, & routers won't be able to forward because it has 1 hop lower TTL.



Protocol:-

TCP, UDP
ICMP, IGMP
IP

- 1-ICMP
 - 2-IGMP
 - 6-TCP
 - 17-UDP
- These are the protocols that tell IP is carrying which protocol.

Checksum

0000 0001 | 0001 0001 | 0001 1000 | 11 01 01 01

1

17

24

add them together

$$1 + 17 + 24 = 42$$

$$(42)_{10} = (00101010)_2$$

now take its 1's complement

$$(11010101)$$

append it to the data

now, sum up these.

$$\begin{array}{r} 00101010 \\ 11010101 \\ \hline 11111111 \end{array} \text{--- all 1's (so no error)}$$

★ It is called header checksum because checksum is calculated only on headers.

★ If header is not a multiple of 16 bits, then we pad extra 0's while computing checksum but we will not transmit it.

Q. Who should compute the checksum?

Ans. ① Senders, Receivers

② All the routers on the way

Q. Why should routers calculate checksum?

Ans. Because TTL changes at each router & some other fields like offset, MF, Total length & Options may change at the routers.

<u>NID</u>	<u>HID</u>
valid	valid → IP
"	all 0's → NID
"	all 1's → Direct. broadcast address
all 1's	all 1's → Limited "
all 0's	valid → Host within a n/w
all 0's	all 0's → I don't have IP.
127	!0, !1 → loopback.

★

ping 127.0.0.1

★

ping 127.0.0.0

127.255.255.255 }

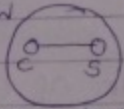
@Techbits

10.11.12

ARP (Address Resolution Protocol)

It is used to find out the physical address of machine whose IP address is already known.
We have 4 cases where ARP is used:

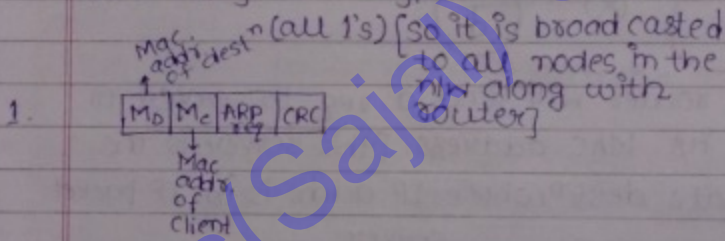
Case 1:

N/w  • when both client & server are in same n/w.

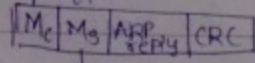
To check whether they are in same n/w, we check the subnet id of client acc. to client & subnet id of server acc. to client. (when both subnet id are same they are in same n/w)
When both client & server are in same n/w, client will forward message directly to server in a frame. For this client has to find out physical address of the server & so it generates ARP request.

ARP Note:- ARP request is always broadcasted at data link layer.

ARP reply is always unicast.

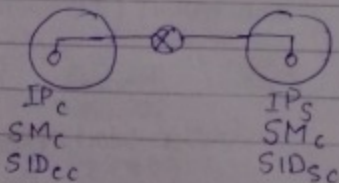


2. The node having the IP addr. of destn. will accept it & put its MAC addr. in ARP reply.

3. 

[Now, this is unicast to just one node.]

Case 2:



When subnet id of client acc. to client \neq subnet id of server acc. to client [then client & server are in diff. n/w.]

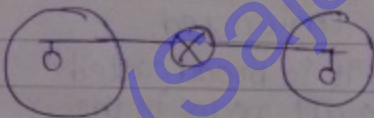
In this case frame has to be 1st forwarded to the router. ∴ client should find out MAC addr. of the router

Note:- Any broadcast message at DLL can never cross n/w boundaries.

So, to obtain the MAC addr. of the router the ARP req. is broadcasted, & the router will reply with its MAC addr.

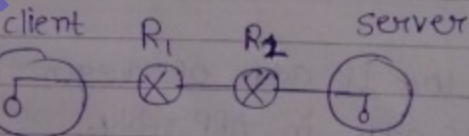
- Though the default router is same always, but being so much loaded with traffic, so NIC cards are changed many-a-times, ∴ we need to send ARP req. periodically.

client Server



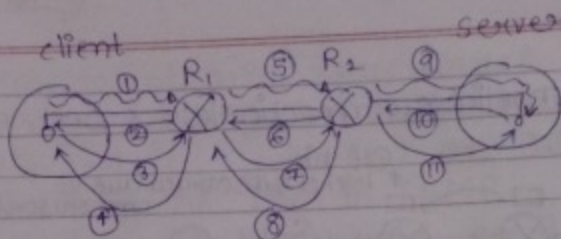
Case 3:-

When router will send a req. to server to know its MAC address. [this server is the ultimate destⁿ whose IP addr. is in IP packet.]



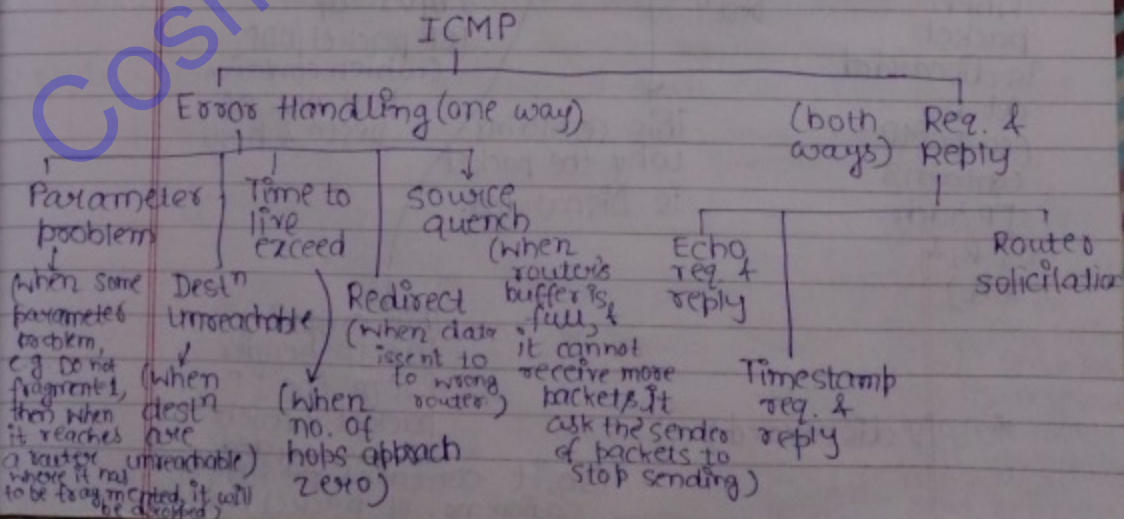
Case 4:-

When router R₁ will send ARP req. to R₂ to know its MAC addr. for next hop.



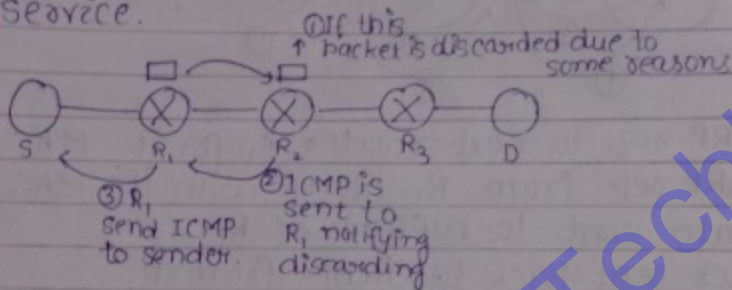
- ① ARP req. to next router R_1 for its MAC addr.
- ② ARP rep from R_1 along with its MAC. addr.
- ③ Data sent to buffers of R_1 .
- ④ Ack sent back to client from R_1 .
- ⑤ R_1 send ARP req. to R_2 for its MAC addr.
- ⑥ R_2 send ARP reply to R_1 .
- ⑦ Data sent to buffers of R_2 .
- ⑧ Ack sent to R_1 .
- ⑨ R_2 send ARP req. to server for its MAC addr.
- ⑩ server send its MAC addr. to R_2 via ARP reply.
- ⑪ Data sent to server from R_2 .

★ ICMP:- It is a n/w layer protocol. It is used for error handling & feedback messaging at n/w layer. (It doesn't use any protocol at transport layer.)



★ For both TCP & UDP, ICMP packets will be generated.

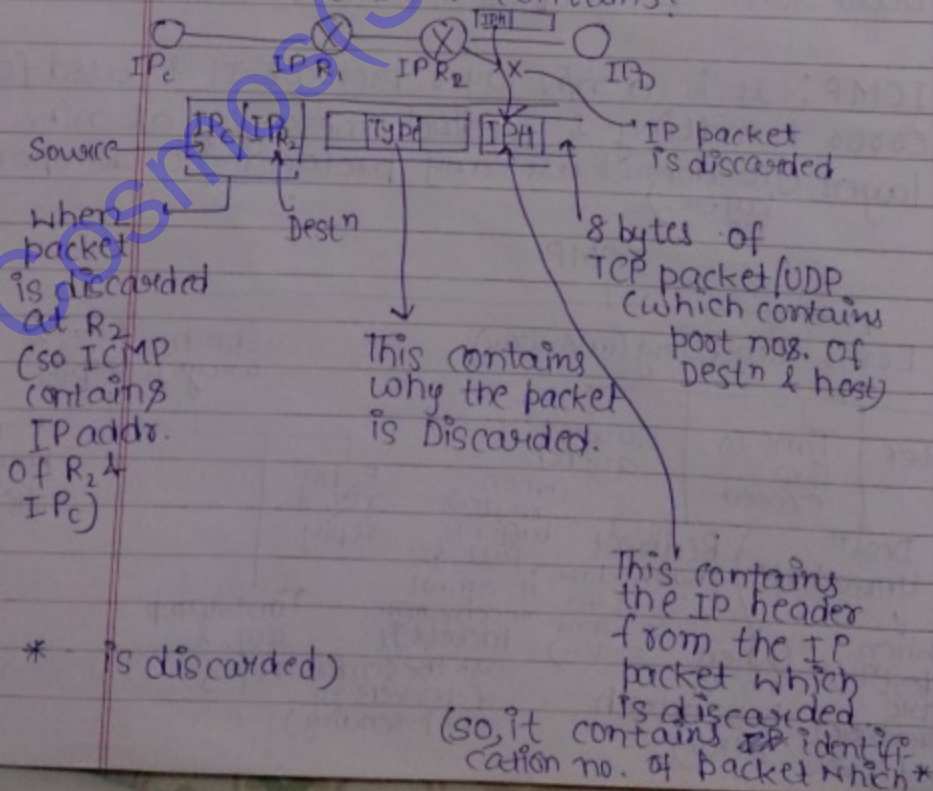
★ IP is unreliable, connectionless & best effort service.



★ Though IP send ICMP packets, but still it is unreliable, because if ICMP itself is lost, no ICMP's are generated for it.

Trace Route:

→ What the ICMP contains?



How to obtain trace route?

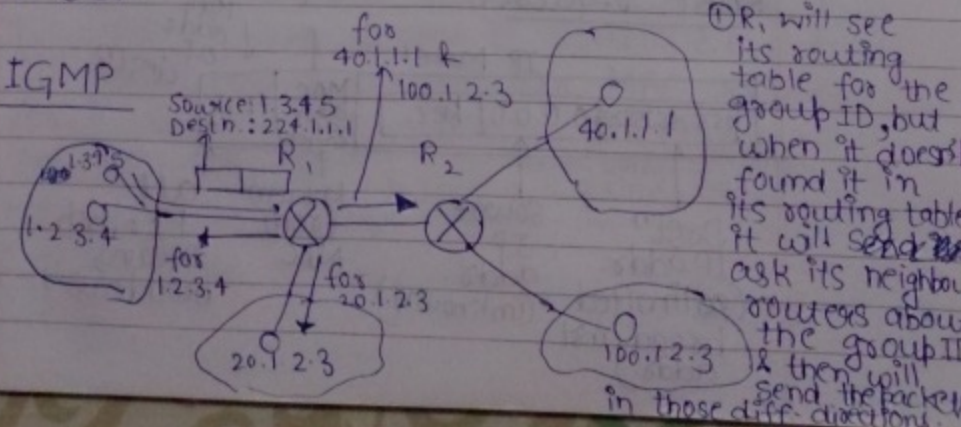
- ① First send a packet with $TTL=1$, so R_1 will discard it & will send an ICMP packet specifying the IP_{R_1} in ICMP.
- ② Now, increment TTL by 1, so R_1 will forward it to R_2 , & R_2 will discard it & send ICMP to R_1 Client via R_1 (specifying its IP_{R_2}).
- ③ Now, increment TTL by 1, so $TTL=3$, & it will reach the Destⁿ, no ICMP will be sent, in this case, we specify destⁿ port no. as one which is wrong, so destⁿ will send a destⁿ unreachable ICMP to sender.

Algo for Traceroute :-

- ① Generate a UDP packet (UDP is used because header size of UDP is only 8 Bytes.) with $TTL=1$ & keep incrementing TTL till we get destⁿ unreachable message.

Note: UDP packet must be sent to a port which doesn't exist

Note: ICMP will be generated both for TCP as well as UDP.



Class D is used for Group messages.

- (i) Can create a group.
- (ii) Can join the group.
- (iii) Can unjoin the group.
- (iv) Transfer info. about the groups to all the routers.

GID: 224.1.1.1

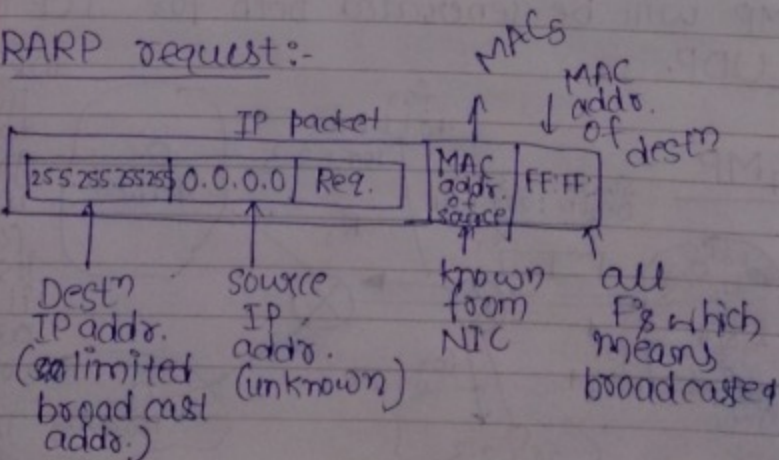
1.2.3.4
20.1.2.3
40.1.1.1
100.1.2.3

- ★ 256 million (2^{28}) Class D addresses are reserved for group id's, but till date we are using only few thousands of groups.

RARP

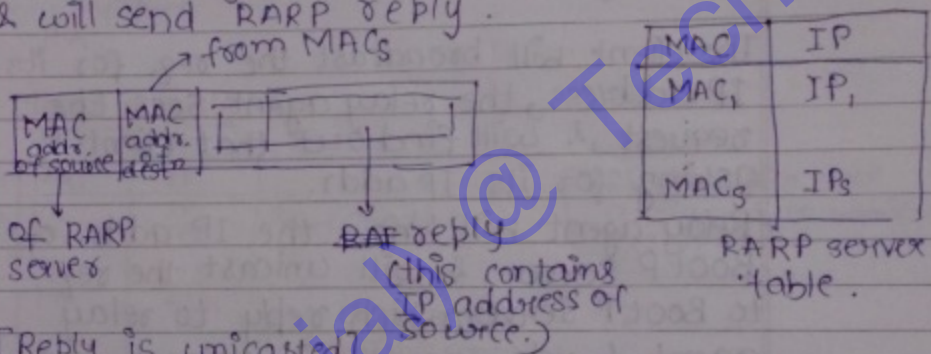
- When the source doesn't know the IP address of itself, so it ask the RARP server about its IP address.

RARP request:-



This req. is seen by all nodes in the same n/w, the router will not allow it to leave the n/w.

- The RARP server finds out that it's a RARP req. & consult its table.
The table will have an IP address of source corresponding to source MAC addr. & will send RARP reply.



[Reply is unicasted]

Problem:-

[Request is broadcasted]

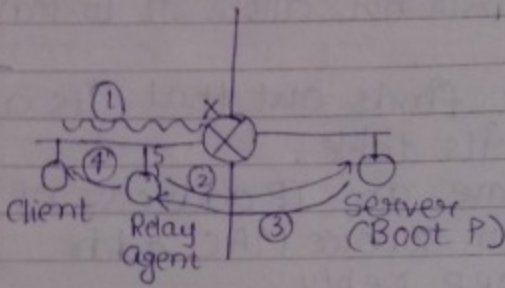
- ① Don't know my IP address (so $IP_{source} = \text{all } 0\text{'s}$)
- ② Don't know whom to ask (so $IP_{dest} = \text{all } 1\text{'s}$ & $MAC_{dest} = \text{all } 1\text{'s}$)

★ Disadvantage of RARP:-

- If there are subn/w in the same n/w, then we must have RARP server for each subn/w.
 - The table of RARP server is static, i.e. one IP addr. for a particular MAC address. (so, if we have 200 machines; then we need 200 IP addresses). [if we have only 100 machines working at a time, then we need only 100 IP addresses]
 - We should have RARP server in all n/w.
- no. of IP \geq no. of machines

- RARP is obsolete.

BOOTP:



The client will broadcast the req. for its IP address, the relay agent sees the request, & will find out that client is asking for its IP addr.

Relay agent will know the IP addr. of Boot P Server & will unicast the req. to Boot P Server, which reply to relay agent, & relay agent will unicast the reply to client.

Advantage:-

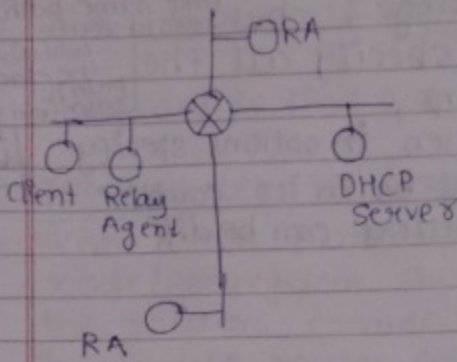
- For a large no. of n/w, we can have only one Boot P server.

Disadvantage:-

- The Boot P server table is static.

Format:

DHCP (Dynamic Host Configuration Protocol)



Mapping table:-

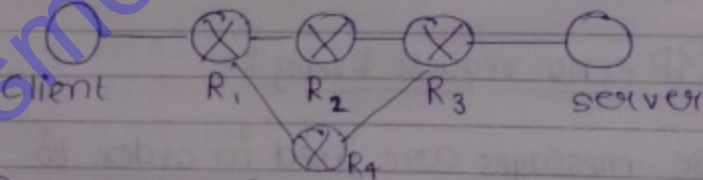
MAC	IP
M ₁	IP ₁
M ₂	IP ₂
...	...
M _k	IP _k

→ static

→ dynamic

(also this contains pool of IP addrs, whenever a machine asks for IP addrs, it will be allocated to that machine with a particular lease period.)

Static IPs are assigned to various servers.



• Source Routing:-

When at the source site, the route to the destⁿ is fixed.

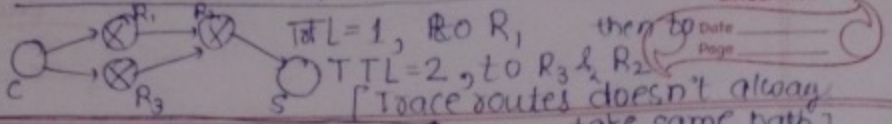
→ Strict Source routing:-

When the complete route is mentioned in the Options field of IP.

• Max. no. of IP addrs. that can be specified in options field is 9. (as options is 40 bytes max.)

∴ $40/4 = 10$, but 4 bytes are used for inter gap b/w 2 IP.

Record Route is diff. from Trace route:-

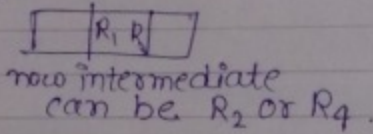


Loose Source Routing:-

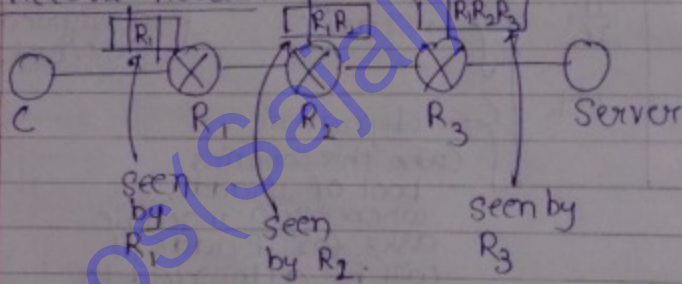
When we doesn't specify all the routers in options, the routers written in options specify those routers must be in its complete path, the other routers can be diff.

will give the exact path taken from source to dest.

e.g.



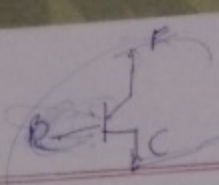
Record Route:-



but we need trace route, because record route is not received by source.

ICMP echo request & reply :-

- These messages are used in order to test whether the n/w layer of the destⁿ & all the routers on the way are working or not.
- Ping ~~uses~~ uses echo request & reply.



Timestamp req. & reply :-

- It is used to find out time as well as delays.

ICMP router solicitation :-

- When a n/w is connected to many routers, a node/system should know what are the routers connected to the n/w, for this it will use router solicitation req. & all the routers will reply to this request.

ICMP routes advertisement :-

- Whenever any new routes come up, everyone should know about it & so the routes advertises itself.

Special IP address (127.)

- ★ To check whether the sender's NIC is working properly, we use loopback address.
- ★ 127 is loopback addr. which is used to test self connectivity.

```
Ping 127.0.0.1 ✓
Ping 127.0.0.0
Ping 127.255.255.255
```

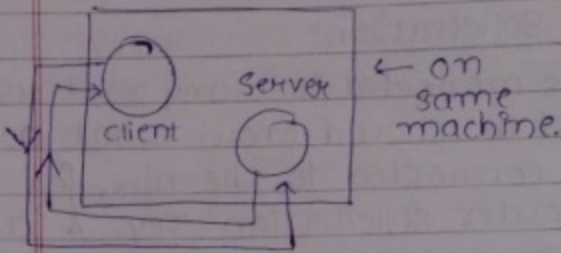
	S.A.	D.A.
Data	1.2.3.4	127.0.0.1

↳ other than
127.0.0.0
127.255.255.255

★ if we write S.A. & D.A. same, then the packet will go to the router & then come back to same machine.

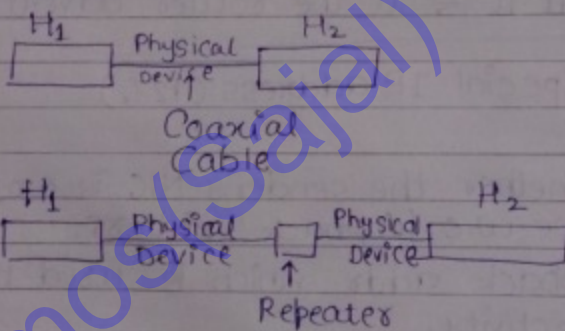
- ★ interprocess comm. within the same machine.

- * In order to test client & servers, which are running on same machine, we use special address 127.



Hardware in Comp. N/w (Devices)

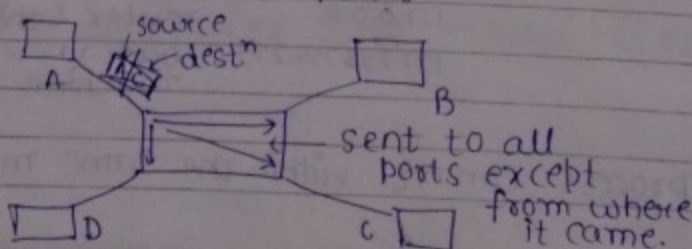
1. Coaxial Cable:-



Repeater is used to decrease attenuation when coaxial cable is too long.

2. Hub:-

- Hub is multipoint repeater.
- It is a pure electronic device with no SW associated with it.



Disadvantage of Hub:-

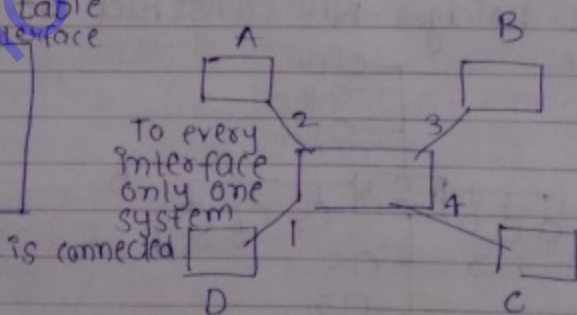
- It doesn't have a lookup table, \therefore it will have a lot of traffic.
- Collisions are possible inside a hub because it is not store & forward device, \therefore collision domain doesn't change.
- Hub has only physical layer, it doesn't have DLL or NL.
- If a device has to stop broadcasting done
- The broadcast domain is also not changed in hub because it has only physical layer.

Switch :-

- Switch is an active device (store & forward s/w), using the s/w a switch will construct lookup table. A switch is a store & forward device

lookup table

MAC	interface
A	2
B	3
C	4
D	1



- ★ Switch contains Physical layer & data link layer.
- ★ Doesn't contain n/w layer.

- Since, switches have a lookup table, the traffic will be less.
- Since, switch is a store & forward device, there will be no collision. \therefore collision domain is used.

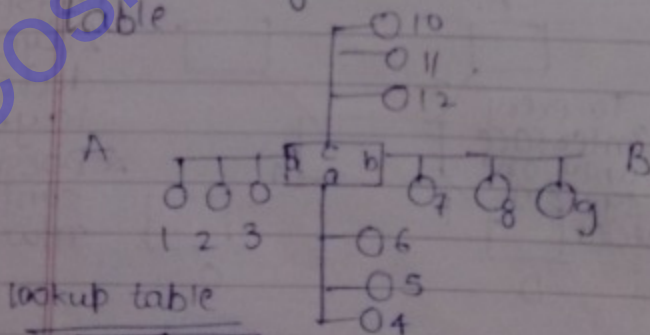
- It will not stop the broadcasting done at data link layer, because switch doesn't contain n/w layer.

Disadvantage:-

- Switch is 4-5 times costlier than hub.

Bridge :-

- Bridge is a switch with less no. of ports. It is used to connect many LANs (instead of system).
- Bridge contains physical layer & data link layer.
- Broadcast domain is not changed by a bridge because it doesn't contain n/w layer.
- Collision domain is decreased because it is store & forward device.
- Even bridges will construct lookup table.



lookup table

MAC	Interface
1	a
2	a
3	a
⋮	

- but if we move ③ from A to B, lookup table becomes

- Bridge will perform 3 tasks:-
 - (i) Forwarding - take a packet & send to other interface.
 - (ii) Filtering - when both source & destⁿ are on same n/w, the packet will be filtered.
 - (iii) Fill the lookup table.

Router :-

- Router is a device which is used to connect various n/w or subnets.
- Router will contain physical layer, DLL, & n/w layer.
- Router is a store & forward device.
- Broadcast domain & collision domain are reduced by routers.
- Responsibilities of routers are :-

(i) Forwarding.

(ii) Filtering

(iii) Routing.

- Routing - The process of preparing the routing tables is called routing.

- Forwarding - The process of choosing one outgoing links among all the available outgoing links is called forwarding.

[if the device have routing table, forwarding means choose one interface among all, but if the device doesn't have routing table forwarding means send to all the interfaces except from which it comes, also known as flooding.]

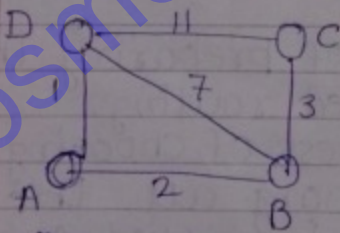
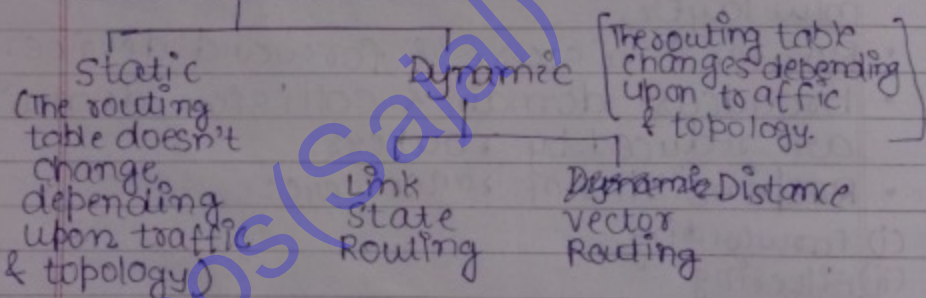
- Advantages of flooding :-

- High reliability (packet delivery is guaranteed even if there is atleast one path.)
- Shortest path is guaranteed (always the 1st packet is the one which has taken shortest path.)

Disadvantages:-

- Lot of traffic & duplicate packets are generated.

Routing algorithms



At A Destⁿ Dist. Next Hop

	Dest ⁿ	Dist.	Next Hop
A	0	A	
B	2	B	
C	∞	-	
D	1	D	

Distance Vector

At C

	Dest ⁿ	Dist.	Next Hop
A	0	-	
B	3	B	
C	0	C	
D	11	D	

At B

	Dest ⁿ	Dist.	Next Hop
A	2	A	
B	0	B	
C	3	C	
D	7	D	

At D

	Dest ⁿ	Dist.	Next Hop
A	1	A	
B	7	B	
C	11	C	
D	0	D	

- In the 1st step, create the routing table for all the routers (when they know only about their adjacent neighbours.)

Step 2:- All the nodes will exchange distance vectors with their neighbours. At every node new routing table will be constructed using the new information from neighbours.

At C:-

length of shortest path of max. edge 1

A	∞	-
B	3	B
C	0	C
D	11	D

from B (distance vector)
A 2
B 0
C 3
D 7

from D (distance vector)
A 1
B 7
C 11
D 0

new routing table:-

length of shortest path of max. hops 2
edge

A	5	B
B	3	B
C	0	C
D	10	B

C to A = $\left\{ \begin{array}{l} C \rightarrow B \rightarrow A = 3 + 2 = 5 \checkmark \\ C \rightarrow D \rightarrow A = 11 + 1 = 12 \end{array} \right.$

C to B = $\left\{ \begin{array}{l} C \rightarrow B = 3 \checkmark \\ C \rightarrow D \rightarrow B = 11 + 7 = 18 \end{array} \right.$

C to C = 0

C to D = $\left\{ \begin{array}{l} C \rightarrow D = 11 \\ C \rightarrow B \rightarrow D = 3 + 7 = 10 \checkmark \end{array} \right.$

At A

A	0	A
B	2	B
C	∞	-
D	1	D

from B
A 2
B 0
C 3
D 7

from D
A 1
B 7
C 11
D 0

new table

A	0	A
B	2	B
C	5	B
D	1	D

$$A \rightarrow B = \begin{cases} 2 \\ A \rightarrow D \rightarrow B = 1 + 7 = 8 \end{cases}$$

$$A \rightarrow C = \begin{cases} A \rightarrow D \rightarrow C = 1 + 11 = 12 \\ A \rightarrow B \rightarrow C = 2 + 3 = 5 \end{cases}$$

$$A \rightarrow D = \begin{cases} A \rightarrow B \rightarrow D = 2 + 7 = 9 \\ A \rightarrow D = 1 \end{cases}$$

- Even though we have calculated new routing table for C in step 2, we will use old routing table of C in this step & will use the new table in step 3. [the step 2 will complete when new routing tables for all routers will be made.]

★

Compute Step 3 similarly.

At A			from B	from D
A	0	A	2	1
B	2	B	0	7
C	5	B	3	11
D	1	D	7	0

AB = 2

AD = 1

A	0	A
B	2	B
C	5	B
D	1	D

3rd method

from diagram (do directly)

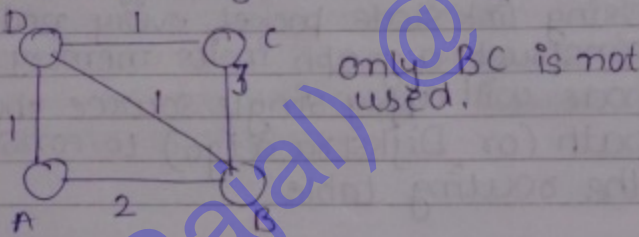
A	1	A
B	3	A
C	6	A
D	0	D

(final routing table).

Q. In the above question, how many edges are not used?

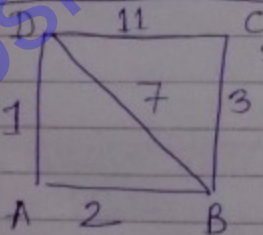
Ans. 2 (CD & BD).

Q. If those unused edges are ^{changed} made to 1.



★ Disadvantage of Distance Vector Routing is count-to-infinity.

Link State Routing :-



step 1: Every node will construct link state packets using local knowledge (knowledge about the neighbours)

Step 1: Link State packet at A:-

Seq No.	Age
B	2
D	1

at B:-

Seq No.	Age
A	2
C	3
D	7

at C:-

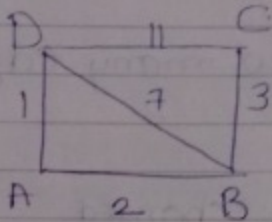
Seq No.	Age
B	3
D	11

at D:-

Seq No.	Age
A	1
B	7
C	11

Step 2:- All Link state packets will be flooded to all other nodes.

At B:- B will get Link state packet from A, C, D (adjacent neighbour)



B will know the topology of the entire n/w & similarly other nodes will know about the entire topology.

Step 3:- Using link state packet, every node will construct a graph in its memory. Every node will apply single source shortest path (or Dijkstra algo) to construct the routing table.

★ In link state routing, there is no count-to-infinity problem.

13.11.12

GatewaysGateways

It is a connecting device which has all the 5 layers & so a gateway is capable of Deep Packet Inspection (DPI) i.e. at gateway we can even look into the application layer.

Transport Layer :-

- Main responsibility of transport layer is end-to-end connectivity.
- If N/w layer is providing unreliable & connectionless service, then transport layer should provide reliable connection oriented service. Two popular protocols used at Transport layer:

(i) TCP

(ii) UDP

TCPHeader Format :-

Source port address (16 bits)				Destination port address (16 bits)			
Sequence Number (32 bit)							
Acknowledgement Number (32 bit)							
Header length (4 bits)	Reserved (6 bits)	W G	A K	P S	R E	F L	Window size (16 bits) [adv window]
Checksum (16 bits)				Urgent pointer (16 bits)			
Options (if any) (0-40 byte)							
data							

- Min. size of TCP header is 20 Bytes.
- Max. " " " " " 60 Bytes.

(1) Port Numbers:

0 to $2^{16}-1$,
Out of which 0 to 1023 are well known,
& 1024 to 49,151 are reserved, &
49,152 to 65,536 are available.

Well known ports:-

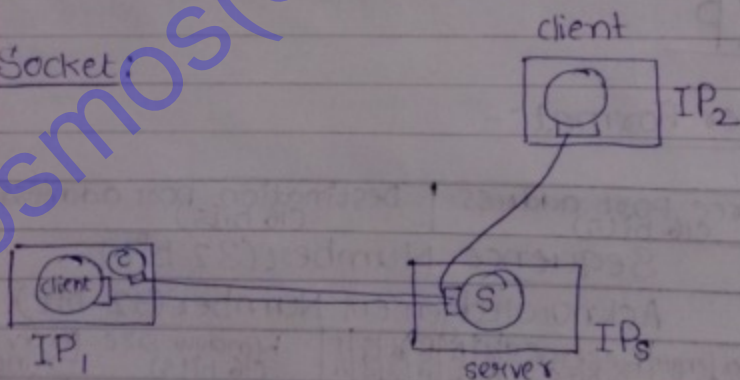
All popular services runs on well-known port numbers & these port nos. are fixed.

Reserved ports :-

These port nos. are with IANA (Internet Assigned Authority) & they can be used for any new protocols that will come up in future.

★ TCP is connection oriented protocol.

Socket:



client wants a web service, so it connects to port no. 80 at server.

- diff. port nos. are req. to distinguish b/w diff. processes.
- If two clients connects to same server's port no. (eg. 80 in this case) & choose same port at client's site (eg. x), then port no. alone can't distinguish b/w two clients.

- So, we need IP address.
- But, if the same machine sends req. to same server, then IP address alone can't distinguish b/w the 2 requests.
- So combination of IP address & port no. is req. for distinguishing them.
- A socket is 48 bit no. (IP+port).
- TCP is byte stream protocol, i.e. every byte is numbered in TCP.

Sequence No. :-

- Since every byte in the stream is numbered, the sequence no. of the ^{first byte} segment is seq. no. of the 1st byte in that segment. Sequence no. is 32 bit field which means 2^{32} sequence nos. are possible.
- We can send out only 2^{32} bytes with unique sequence nos.
- Wrap around :- The process of using up all the sequence numbers & repeating a previously used sequence no. is wrap around.
- (WAT) Wrap around time :- The time taken to wrap around is called wrap around time.

Q. If b/w is 1 Bps, then what is the wrap around time?

Ans.

$$\frac{2^{32} \text{ Byte}}{2 \text{ Byte}} \rightarrow \frac{2^{32} \text{ B}}{2 \text{ B}} = 2^{29} \text{ B.}$$

$$1 \text{ seq.} \rightarrow 1 \text{ s}$$

$$2^{32} \text{ seq.} \rightarrow 2^{32} \text{ sec.}$$

- Lifetime :- It is the time for which a packet can be there in the internet before being discarded. (practically lifetime = 3 min.)
- So $\boxed{\text{WAT} \geq \text{Lifetime}}$

Q. If B/w is 1 MBps, then what is WAT?

★ Every byte takes one sequence no.

classmate

Ans. 1M Bytes $\rightarrow 1\text{B}$

1M seq. no. $\rightarrow 1\text{B}$

1 seq. no. $\rightarrow \frac{1}{1\text{M}}\text{B}$

2^{32} seq. no. $\rightarrow \frac{2^{32}}{1\text{M}(10^6)}\text{B} = 4096\text{B} = 4096 \cdot 962368$
 $> 180\text{Bec}$
 (3min)

Q. B/w = 1GBps
WAT = ?

Ans. 10^9 Bytes $\rightarrow 1\text{B}$

10^9 seq. no. $\rightarrow 1\text{B}$

1 seq. no. $\rightarrow \frac{1}{10^9}\text{B}$

2^{32} seq. no. $\rightarrow \frac{2^{32}}{10^9}\text{B}$

$= 4.294967296\text{Bec.}$

$< 180\text{sec}$

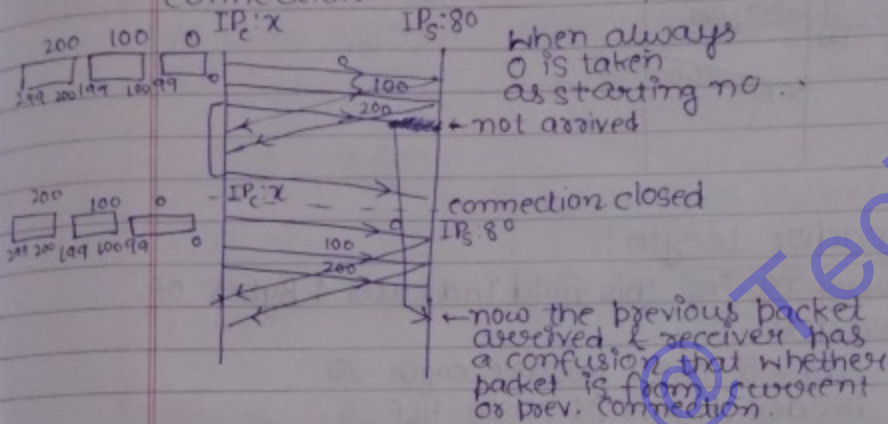
[So, we are sending packets frequently]

★ Since, rtp around time $<$ lifetime, in order to distinguish b/w 2 packets with same sequence no., we use time stamp.

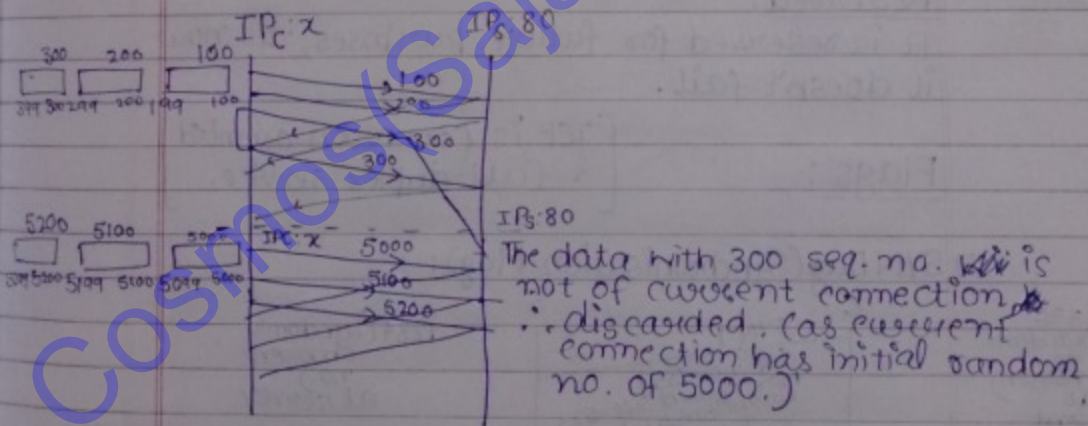
• Timestamp:- is used in the options field.

★ All the segments made by Transport Layer need not be of same size.

- We use random initial sequence nos. in order to avoid accepting packets from previously closed connection as a packet from current connection.



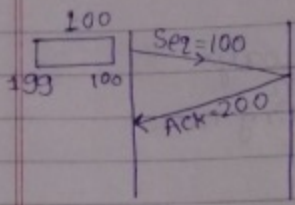
To avoid this problem, we use random initial sequence no.



- ★ The probability of two processes on same computer to pick same seq. no. = $\frac{1}{2^{32}}$ (very small).
- ★ WAT will not change even if we use random initial sequence no.

Acknowledgement No. :-

It is the sequence no. of the byte that the receiver is expecting next.



Header Length :-

Every no. in this field indicates 4 Bytes of header.

If HLF = 5, then header length = 20

If header length = 21, then HLF = 6

$\frac{1}{24} \leftarrow$ padding (in options)

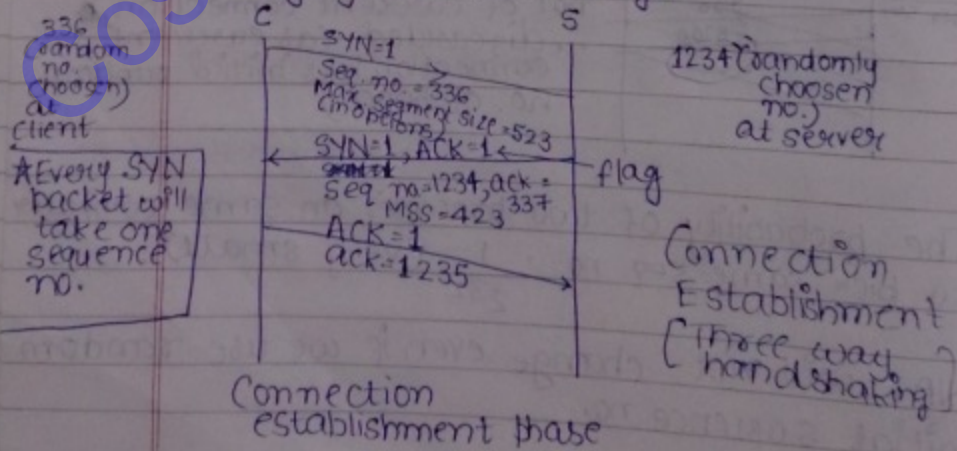
Reserved :-

It is reserved for future purposes, till now it doesn't fail.

Flags :-

[TCP is connection oriented & full-duplex service.]

(i) SYN Flag (Synchronisation Flag):

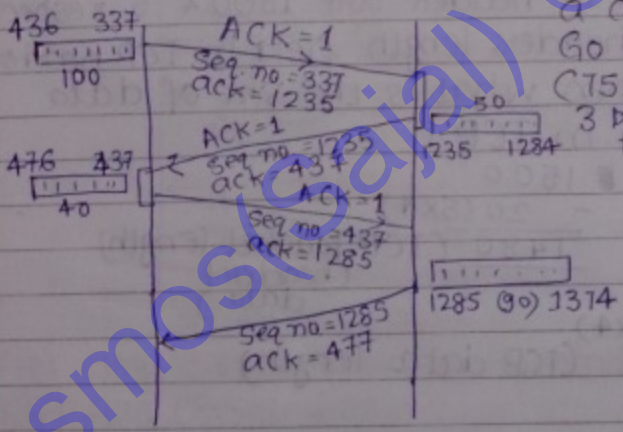


(ii) ACK flag This flag indicates that acknowledgement field is being used & it is valid.

Note: Only the request segment will have ACK=0 & all other segments will contain acknowledgements (ACK=1).

SYN	ACK	
1	0	- Request (First packet)
1	1	- Reply
0	1	- Pure & Piggybacking
0	0	- X (Not possible)

TCP uses a combination of Go back N & SR (75% SR + 25% GBN) 3 principles & 1 principle from SR GBN

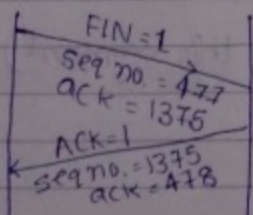


How to Data transfer phase.

(iii) FIN Flag: It is used to close the connection.

SYN=1
FIN=1
ACK=0
Data=0

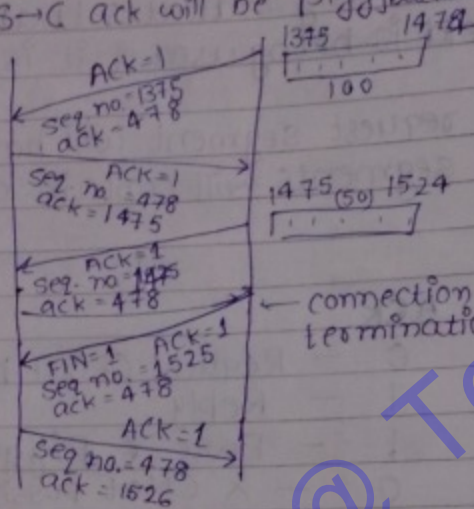
- SYN will take 1 Seq. no.
- FIN will take 1 Seq. no.
- Every Data byte will take 1 Seq. no.
- Pure ACK won't take any Seq. no.



← FIN is sent
← FIN is acknowledgement [so connection b/w client to server is closed.]
Now, data can't be sent from C→S, can be sent from S→C
ack can be sent from C→S
Ack can be sent from S→C

but S→C ack will be piggybacked & not pure.

* All packets other than the first SYN packet will have ACK=1.



Q. If total length field & header length field in IP header are 1500 & 5 respectively & header length field in TCP header is 5, then what is the size of data present in TCP?

Ans.

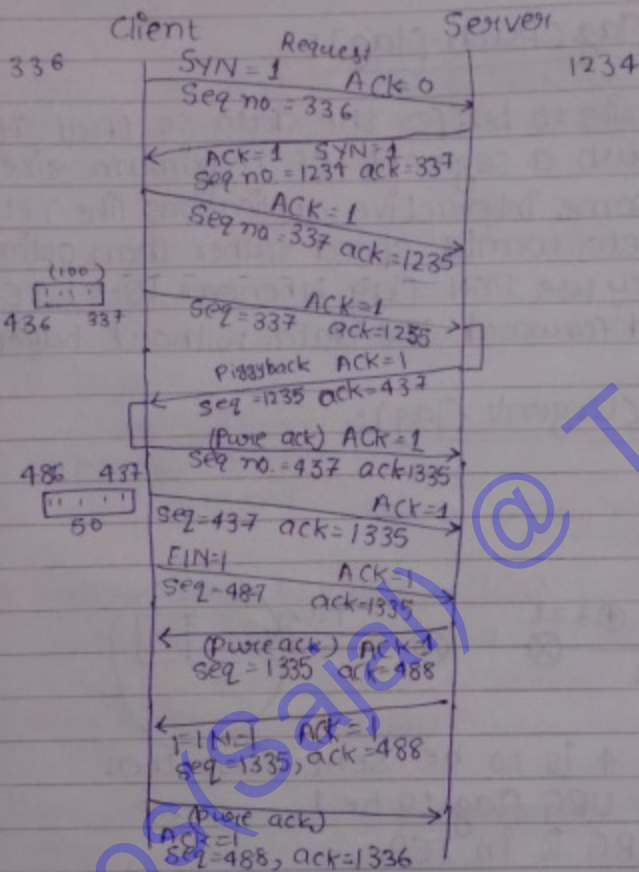
$$\begin{array}{r}
 1500 \\
 - 20(5 \times 4) \\
 \hline
 1480 \quad \text{(TCP packet length)} \\
 - 20(5 \times 4) \\
 \hline
 1460 \quad \text{(TCP data length)}
 \end{array}$$

Q. If in above question, seq. no. of the TCP segment is 1234, then what is the ack no.?

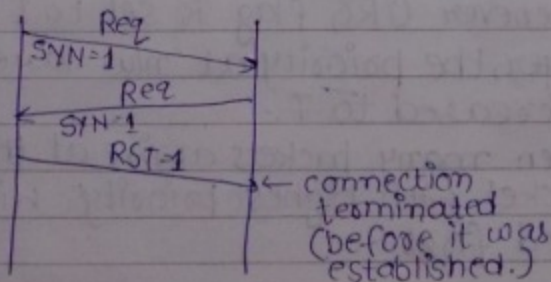
Ans.

$$\begin{array}{r}
 1234 \\
 + 1460 \\
 \hline
 2694 \quad \text{(last byte of this segment)}
 \end{array}$$

2694 is the ack no.



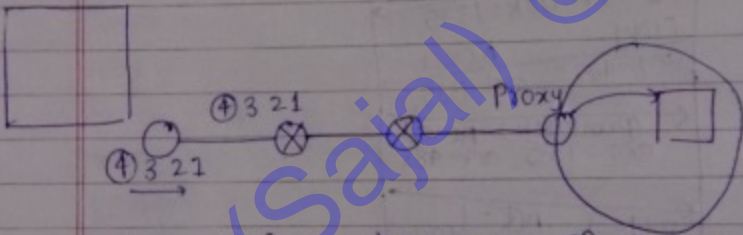
- FIN flags : It is used to terminate the connection.
- RST flag (Reset flag) :
Reset is used to terminate the connection during the connection establishment phase.



- PSH flag (Push flag) :-

TCP tries to buffer the data so that it can push a segment of maximum size, but some interactive applications like TELNET, Chat, etc. want speed rather than optimality, so they use PSH flag. Whenever PSH=1, TCP should forward the data without buffering.

- URG (Urgent flag) :-



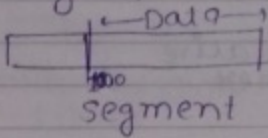
- When 4 is to be sent first, then we set URG flag to be 1.
 - But, URG is in TCP.
 - So, routers won't be knowing it as it works at n/w layer max., & URG is in Transport layer, so to tackle it, whenever URG flag is set to 1 at transport layer, the priority at n/w layer will be increased to 7.
- When many packets arrive at the router, the packet with highest priority will be forwarded first.

Urgent Pointer:-

- When URG = 1, this field is valid.
- This field indicates till what part of segment is the data urgent.

Q. If urgent pointer = 100 & sequence no. of the segment is 1000, then what is the sequence no. of the last byte in the segment which is urgent.

Ans.

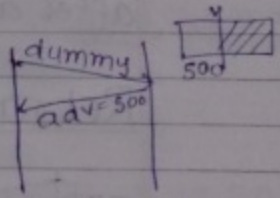
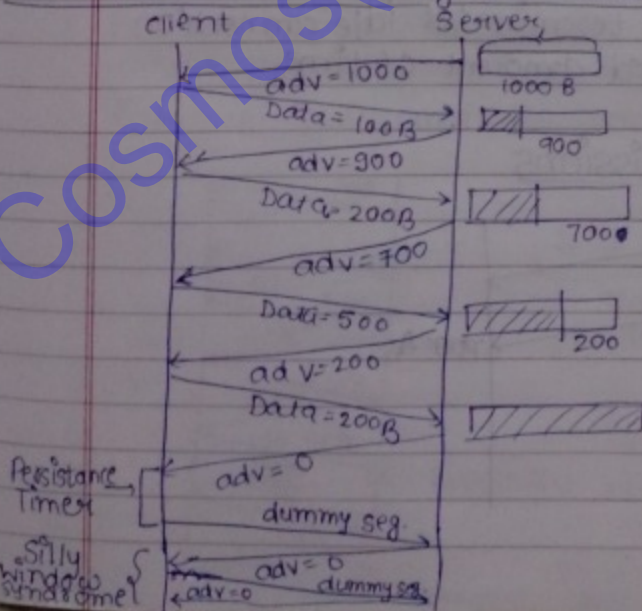


if urg pts. = 100,
then no. of urgent bytes = 101 B

last urgent bytes = 1100
urgent data = 1000 to 1100 (101 B)

Advertisement Window:-

TCP Flow control using advertisement window:-



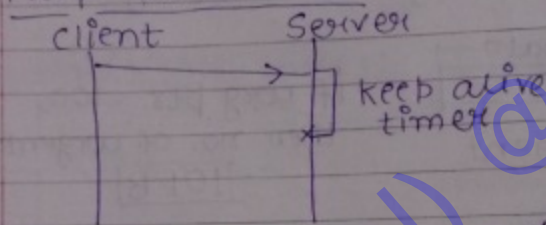
• Dummy segments are sent when client can't send normal segments

Persistence Time

Silly window syndrome

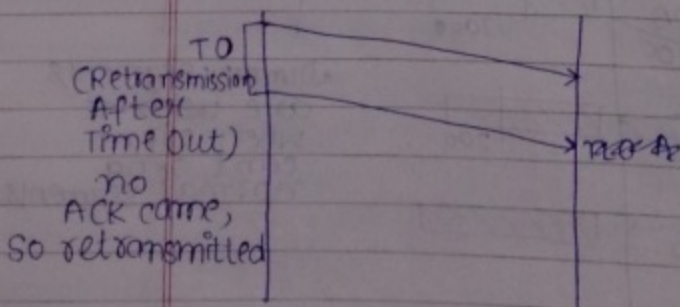
- Since advertisement window is 16 bits, a server can't advertise more than 64 KB even if it has free buffer, ∴ to overcome this 14 bits are added (appended) to this field to make it 30 bits (1 GB). These bits are in Options.

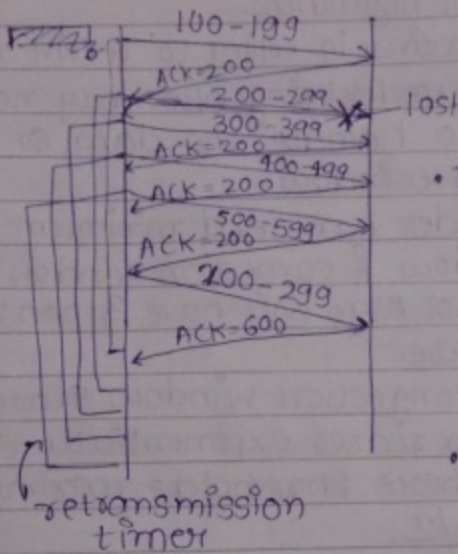
Keep alive timer



- In this case, whenever the client establishes a connection with server, the server starts keep alive timer, if ~~the~~ client doesn't send any data / or perform any activity within the timer expiration period, then connection will get terminated.
- This timer terminates idle connection after a fixed amount of time.

Retransmissions



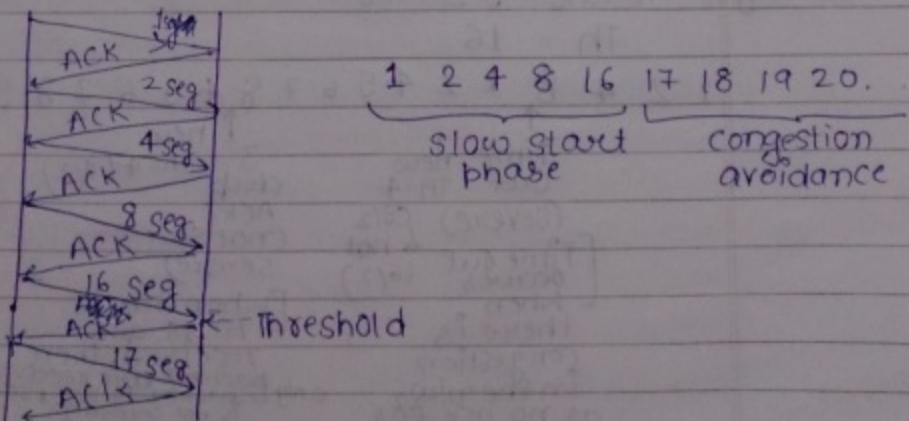


- Three duplicate ACKs are received, so it is a "fact" that if 3 duplicate ACKs are received for a packet, then that packet might be lost in the the n/w, so retransmit it before time out.
- If last packet is lost, then it will be retransmitted after time out.

• TCP Congestion Control:-

Buffer size = 32000 B (Receiver)
MSS = 1000 B

- Now, we can send 32 segments at once w/o waiting for ACK.
- But we can't send all 32, as the n/w might get congested due to it (the routers may cause timeout) & hence TCP controls congestion.



Congestion Control Algorithms:-

- Even though a receiver is willing to receive more than 1 MSS, the underlying n/w may not be in a position to handle the data, so TCP uses congestion window.

Therefore, a sender can send minimum of advertise window & congestion window.

- Congestion Control Algo will have 3 phases:-

(i) Slow start phase.

In this phase congestion window starts with 1 MSS & increases exponentially till the threshold where threshold = maximum sender window/2.

(ii) Congestion avoidance phase:-

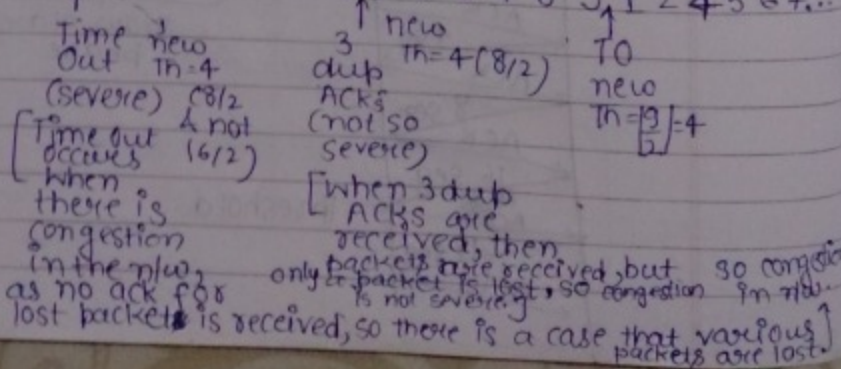
In this phase, congestion window will grow linearly till it reaches maximum window size.

e.g. let receiver's buffer size = 160,000 B, MSS is 1000 B, then after how many RTT's (Round Trip time)/Rounds, sender can send 16 MSS in one window

Ans. ~~12~~ after 11 rounds.

e.g. Max = 32 MSS
Th = 16

1 2 4 8 1 2 4 5 6 7 8 4 5 6 7 8 9 1 2 4 5 6 7 ..



Congestion Detection Phase:-

Congestion could be detected in 2 ways:-

- (i) 3 duplicate ACKs.
- (ii) Time Out.

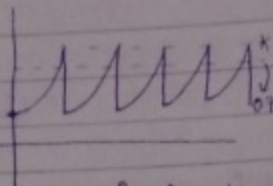
- 3 duplicate ACKs:- Whenever congestion is detected because of 3 duplicate ACKs, it will indicate that congestion is not severe, \therefore new threshold is set to half of current congestion window size & algo enters congestion avoidance phase.
- Time Out :- this indicates that congestion is severe & so new threshold is set to half of the current congestion window size & algo enters ~~cong~~ slow start phase.

Time Out Timer at link-to-link protocols in at data link layer like "HDLC" uses static time out timer, but end to end protocols like TCP should not use static time out timers as it may lead to n/w congestion when there is heavy traffic.

Fin RTT at 4:00 A.M. is less than that at 6:00 P.M.

7.11.17

Congestion Control



• If the sender is sending data at rate of 1Gbps, but user is allowed only 100Mbps. In this case we set a value k (max. limit of no. of packets that receiver will receive), when k is obtained, we simply starts discarding the packets, we don't send ACK's for them, so timer will time out at server, so the server will again start from 1, & hence in this way we can set k & set the receiver to receive data at 100Mbps (by setting k).

★ Second soln. :-

In this soln., we send adv. window size acc. to 100 Mbps from client to server.

Checksums

1. TCP checksum is calculated at for TCP header, TCP payload & pseudoheader.
2. IP Layer but checksum calculation for IP header.
3. If in transit, the IP header gets corrupted & the checksum for IP gets modified in such a way that the router/receiver can't catch the error.

In this case, TCP will catch the error by checksum on pseudoheader, so TCP will discard the packet as the IP header gets corrupted & sent to wrong machine.

★ The TCP doesn't calculate checksum for entire IP header, just a part of the header, as the IP header changes through the transit.

The fields that change during transit in IP header:-

- TTL
- Offset
- Checksum
- Options
- MF

Q. Why should routers compute the checksum of IP for every packet?

Ans. Because TTL changes at every router with each hop. In this case if there is fragmentation, fragment offset, MF & total length may change & Options could change.

Q. Why TCP is computing checksum only on pseudoheader from IP & not on actual header?

Ans. Because many fields in IP header may vary.

★ We need CRC at DLL even though we use checksum at TCP & IP because sometimes we can send data ~~to~~ from b/w two machines only, so we don't need n/w & transport layer, so error will be detected by DLL's CRC only as TCP & IP layers are absent.

★ Why can't use static ^{round trip} time?

Ans. Because the round trip time changes ~~is~~ at different time of day, we need ~~the~~ round trip time timer to ~~be~~ change from time-to-time.

$$ND = \alpha(ID) + (1-\alpha)(AD)$$

$$= 7.5 \quad - \textcircled{2}$$

$$NRTT = \alpha(IRTT) + (1-\alpha)(ARTT)$$

$$= 15 \text{ms}, \quad - \textcircled{1}$$

$$\text{SO Next Time Out (TO)} = NRTT + 4 \times ND$$

$$= 45 \text{ms}.$$

③ In next step,

we take $IRTT = 15 \text{ms}$ (from ①)

& $ID = 7.5 \text{ms}$ (from ②)

& let $ARTT = 25 \text{ms}$

$$\therefore AD = |IRTT - ARTT| = 10 \text{ms}$$

$$\& ND = \alpha(ID) + (1-\alpha)AD$$

$$= 8.75 \text{ms}.$$

$$\& NRTT = \alpha(IRTT) + (1-\alpha)ARTT$$

$$= 20 \text{ms}$$

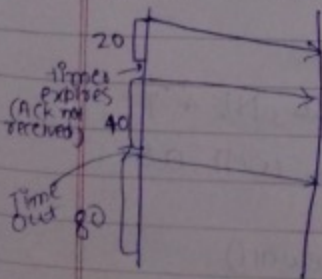
$$\& TO = NRTT + 4 \times ND$$

$$= 55 \text{ms}.$$

Options (in TCP)

KARN'S soln.

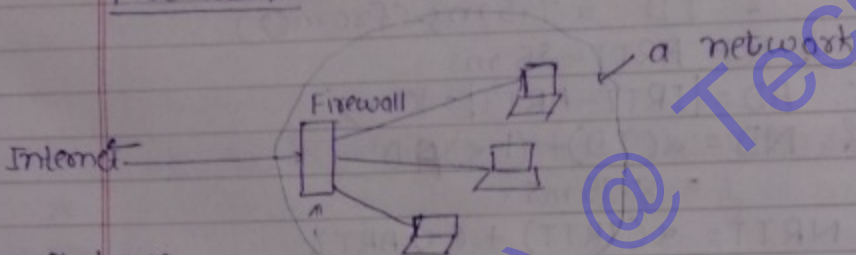
If we don't get ACK either in basic algorithm or in Jacobson's algorithm, the timeout for the next retransmission acc. to KARN is twice the previous time-out.



LAN connecting Devices:-

- ① Wires
 - ② Hub
 - ③ Switch
 - ④ Bridge
- These devices can't stop a broadcasting message.

* Routers will stop the broadcasting.

Firewalls

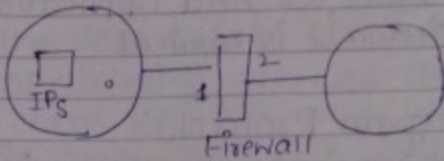
We can't have a layer 2 firewalls, because the packet from internet will be coming from through gateway, as all the packets will be coming from gateways so all the packets will be discarded.

All the devices in the n/w will be connected to the firewall (so all packets reaching the hosts in the n/w, & packets from hosts will pass through firewall).

Types of Firewall :

1. Layer 3 firewall :- (Packet filtering firewall)
This firewall will contain till n/w layer & can make a decision depending on the IP address. (It can filter out a host with a particular IP address.)
2. Layer 4 firewall :-
This contains physical layer, DLL, NL & TL, ∴ it can filter both host as well as particular service on the host.
3. Layer 5 firewall :- (Proxy firewall)

This contains all 5 layers. It can filter host, particular service on a host & particular user (User IP & password).



Firewall :-

Dest ⁿ IP	Source IP	Dest ⁿ Port	Source Port	Use type	Interface
-	IPs	-	-	-	1
IPs	-	23	-	-	2
-	IPs	-	-	-	-

When we want to block any packet coming from FB, then firewall will block it (no matter from which interface it comes from)

if the sender with IP add^s as IPs is requesting for some services through interface 1, then the firewall will block it.

if the destⁿ is a particular host with IP add^s as IPs & we don't want anyone to connect to its port 23 (for telnet services) & the org. (coming from interface 2), so firewall blocks it.

Q. What is the smallest firewall needed to block ICMP packet?

Ans. ICMP works at n/w layer, so we need Layer 3.

Q. What is the smallest firewall that can be used to block HTTP traffic?

Ans. Layer 4, we have to block port no. 80, we need layer 4.

Q. What is the firewall capable of blocking some of the users?

Ans. proxy firewall

In order to block users, we need application layer, so proxy firewall is required.

UDP (User Datagram Protocol):-

Whenever speed is required rather than reliability we use UDP.

Source port no.	Destn port no.
checksum	Total Length

& not the header length.

- Multimedia

- DNS

- NTP

TL

Reliable → (TCP)

Unreliable → (UDP)

NL

Connectionless (Datagram → IP)

DLL

Connection Oriented (Virtual Circuits → ATM)

PL

★ If the Datagram received at n/w layer has to be handed over to user at application layer w/o providing any reliability at transport layer, we use UDP at transport layer.

~~★~~ Need for UDP:-

(i) If an application needs speed rather than reliability, then UDP is better. (eg. Multimedia applications)

(ii) If an application needs one request & one reply kind of communication, then connection

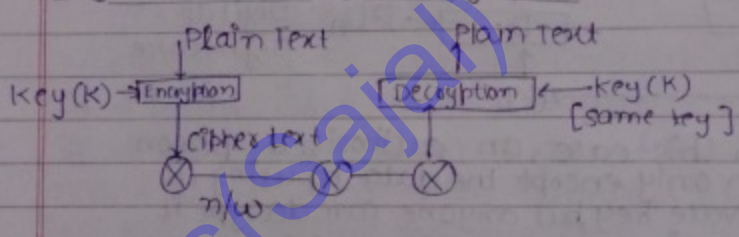
establishment is not required, ∴ UDP is better.
(e.g. DNS, NTP (New Type Protocol), Post of the Day of the day), n/w news

Note:- UDP is connectionless, unreliable protocol.

(iii) The data rate in TCP is not uniform because of differing sizes of advertisements which data rate (diff. no. of packets sent) during congestion control. if application needs uniform data rate, then UDP can be used.

Cryptography

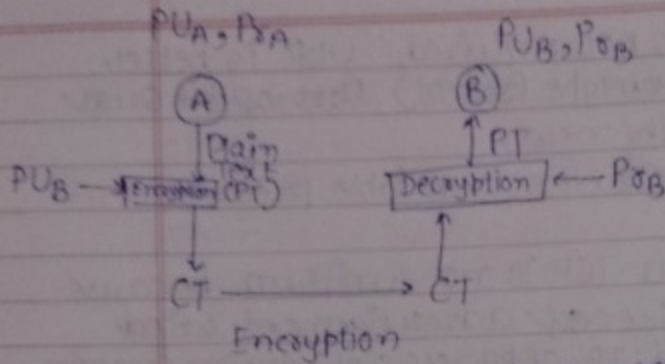
Symmetric-Key Encryption



Disadvantage:-

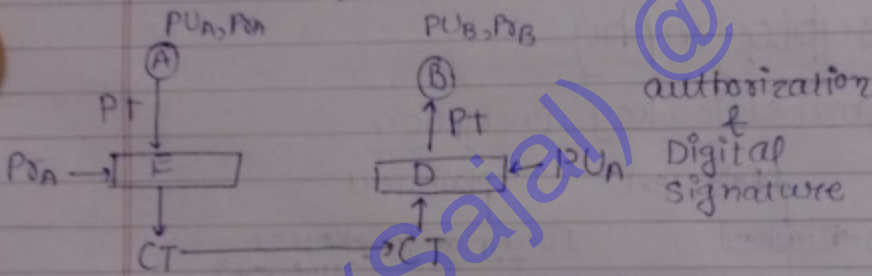
- ① Key transfer is not secure.
- ② if there are 'n' parties who want to communicate securely, then we need n^2 keys.

Public Key, Private Key Encryption

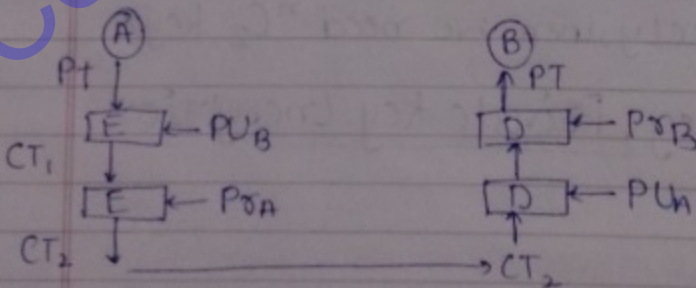


Encryption

In this case, anyone can send the data to B as everyone has P_{B} , so we can't identify whether packet comes from authorised person or not.



In this case, an authorised person can only encrypt the data by using its private key, but anyone can decrypt it by using the public key of A & hence no security, but in this case, authorization can be guaranteed.



In this case, the packet can only be sent by A & no other can send it (as only A has its private key) & only can also be encrypted by public key of B, so that only B can decrypt it using its private key & no other.

Basics Of Cryptography

① Euler's Theorem:-

If ~~an~~ 'p' is a prime no. & 'a' is any positive no. not divisible by p, then $a^{p-1} \equiv 1 \pmod{p}$.
 [a < p] → must means $\frac{a^{p-1}}{p}$ gives remainder

Euler-Totient no. (ϕ):-

It represents no. of the integers < "n" which are relatively prime to "n".

Relatively prime:- Two prime nos. whose gcd is 1.

$$\therefore \phi(5) = \{1, 2, 3, 4\} = 4$$

$$\phi(10) = \{1, 3, 7, 9\} = 4$$

$$\phi(35) = \{1, 2, 3, 4, 6, 8, 9, 12, 14, 16, 18, 21, 22, 24, 26, 27, 29, 31, 32, 33, 34\} = 24$$

Whenever any no. can be written as product of 2 prime nos. & the nos. are diff.

Note:- If When $n = p \times q$, such that p & q are 2 prime nos. & $p \neq q$, then $\phi(n) = \phi(p) \times \phi(q)$
 If p is a prime no., the $\phi(p) = p - 1$

Discrete Logarithms:-

If 'a' & 'n' are relatively prime nos, then there exist at least one integer 'm', which satisfy $a^m \equiv 1 \pmod{n}$

Q. If $a = 7$ & $n = 19$, then find m?

Ans $\frac{7^m}{19} \equiv 7^m \pmod{19} = 1$

Q. Find out the period of $3 \pmod{7}$

Ans. $3^1 \pmod{7} = 3$

$3^2 \pmod{7} = 2$

$3^3 \pmod{7} = 6$

$3^4 \pmod{7} = 4$

$3^5 \pmod{7} = 5$

$3^6 \pmod{7} = 1$ } \rightarrow period = 6

it becomes 1.

$\& 3^7 \pmod{7} = 3^1 \pmod{7}$ [as period is 6]

★ If period of $a \pmod{b}$ is $\phi(b)$, then a is called primitive root of b .

★ 3 is a primitive root of 7, 2 is not a primitive root of 7.

★ The period of $a \pmod{b}$ will definitely divide $\phi(b)$.

RSA Algorithm (to generate public key & private).

rsa - keygen

Step (i) Select 2 prime nos. 'p' & 'q' such that $p \neq q$.

Step (ii) Calculate $n = pq$

Step (iii) Calculate $\phi(n) = (p-1)(q-1)$

Step (iv) Select an integer 'e' such that gcd of $\phi(n)$ & e is 1
& $1 < e < \phi(n)$.

Step (v) Calculate 'd' such that
 $d \equiv e^{-1} \pmod{\phi(n)}$

$$ed \equiv 1 \pmod{\phi(n)}$$

Public key (e, n)

Private key (d, n)

RSA algorithm can be used to send a small no. for security & then it can be used as symmetric key for the subsequent communication.

Diffie-Hellman Key Exchange

- (i) If A, B wants to exchange a key, then there are 2 publically known nos. e.g. a prime no. 'p' & an integer α which is primitive root of 'p'.

A selects a random integer $x_A < b$ & compute Y_A as $Y_A \equiv \alpha^{x_A} \pmod{p}$.

B selects a random integer $x_B < b$ & computes Y_B as $Y_B \equiv \alpha^{x_B} \pmod{p}$.

Each side (A & B) keeps X as private & declares Y as public key.

$$\begin{array}{l} \frac{A \oplus A}{A \oplus B} \\ \text{key} = (Y_B)^{x_A} \pmod{p} \\ \text{key} = (Y_A)^{x_B} \pmod{p} \end{array} \quad \left. \vphantom{\begin{array}{l} \text{key} = (Y_B)^{x_A} \pmod{p} \\ \text{key} = (Y_A)^{x_B} \pmod{p} \end{array}} \right\} \text{both of these nos. are same.}$$

Application layer

HTTP

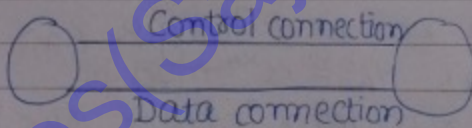
It is used for web service. Port no. is 80.

HTTP is stateless. (i.e. it/ do HTTP server doesn't remember the connection info. it asks client to store the info in cookies.)

HTTP uses TCP at transport layer (reliability is required).

FTP (File transfer Protocol)

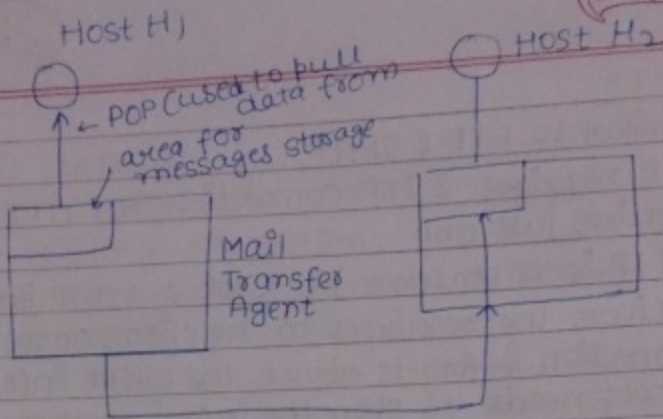
- ① It is used to transfer files b/w client & server.
- ② FTP requires reliability, so it uses TCP at transport layer.
- ③ Port Nos. used are 20, 21.



Using control connection, we can browse the file system & using data connection we can transfer the data.

- ④ FTP is out of band connection, i.e. data & control information follows 2 connections.

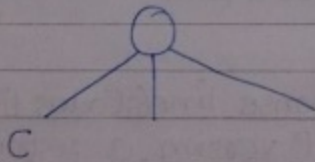
SMTP (Simple Mail Transfer Protocol)



- ① SMTP is used to push the emails & POP is used to pull the emails.
- ② Both the protocols are pure text based.
- ③ MIME is a set of s/w programs which will assist SMTP & POP in sending & receiving data which is not pure text.
- ④ Both SMTP & POP needs reliability & so they use TCP at transport layer.

DNS (Domain Naming Server)

- ① DNS is a one request one reply protocol, ∴ it uses UDP at transport layer.



HTTP

- Similar to FTP & SMTP.
- FTP requires 2 TCP connection, but HTTP requires just one.
- HTTP is a stateless protocol, so HTTP just delivers the resources to the client, closes the connection & forgets about the client info, so clients needs to store the info. in cookies

HTTP request/response format

Initial Line

Headers

Blank Line

Body

- The initial line differs for both request & response.

initial request line:-

method name URL HTTP version

e.g.

GET /path/to/file/index.HTML HTTP/1.0

method name

local path

HTTP version

initial response line(status line):-

- it contains HTTP version, a response status code that gives the result of the request & an English reason phrase describing the status code,

e.g. HTTP/1.0 200 OK

or HTTP/1.0 404 NOT FOUND

FTP

1. It is stateful
2. Uses TCP for reliability
3. Out of band connection
(which means that FTP uses 2 different ports for control & data connection.)

The FTP client sends an FTP request to FTP server on port 21 of TCP (this occurs through a command connection), initially authorization is required which takes place through username & password.

After the authorization, some command is sent through common control connection (e.g. to download a file from server), the FTP server in response opens a data connection with the client on port 20, & downloading takes place.

After that the connection terminates (the server closes the connection, but remembers about the client i.e. authorization, etc. & hence is stateful), so whenever client ask for next file transfer, he/she don't need to start the authorization step again.

It is called Out of band connection because control & data connection takes place through 2 different ports.

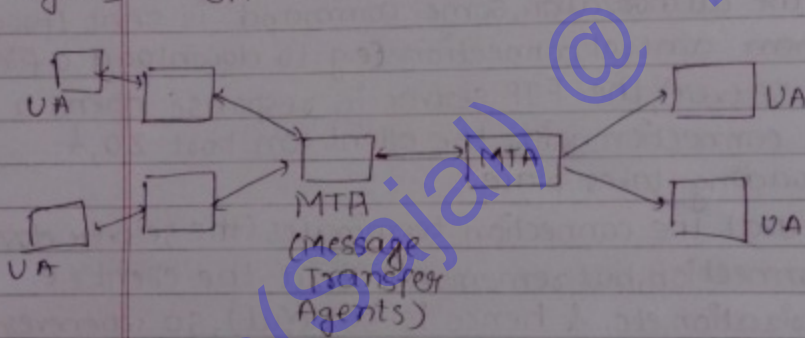
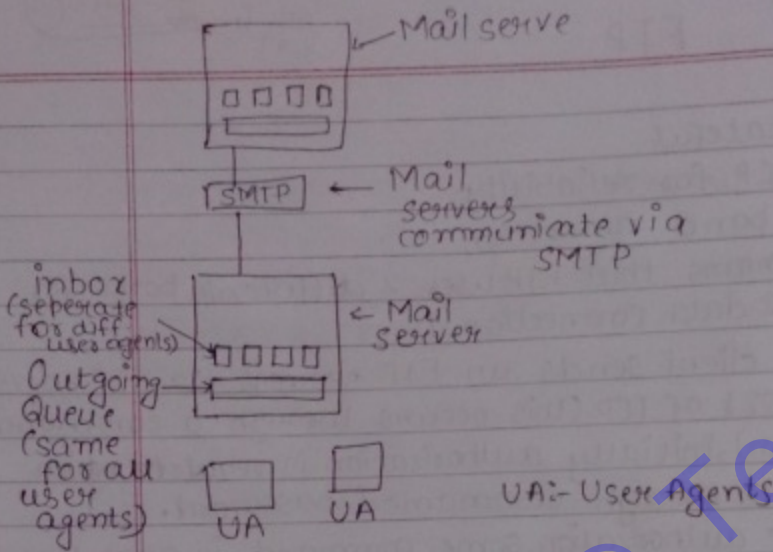
SMTP

Supported by TCP/IP suite.

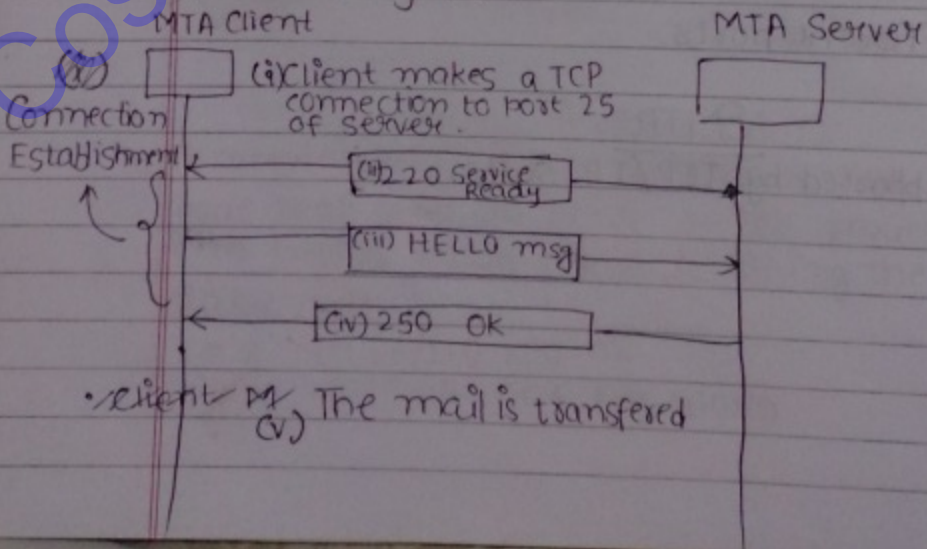
80
(17)
60
289

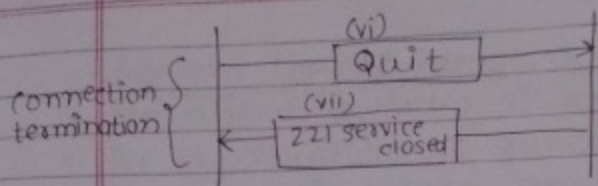
classmate 17

Date _____
Page _____



- MTAs are used to transfer email from one m/w having some format say f1 to another m/w having same or different format (say f2).





- SMTP doesn't allow non-textual data to be sent via a mlw.
- But MIME (Multipurpose Internet Mail Extensions, an extension to SMTP) can be used to transfer non-textual data to be transferred via Internet. (non-ASCII)

POP3

- Used for interacting with the mail box.
- Used for downloading the mails from mail server to the user's machine.
- Users can't create folders on mail server.

IMAP4

- More features than POP3.
- Can check email header prior to downloading.
- Can search contents of email prior to downloading.
- Can partially download email.
- Can create, delete or rename mailboxes on mail server.
- Can create hierarchy of mailboxes in a folder for email storage.