21/8/19
Tuesday

COMPUTER NETWORKS

- [Avg: 8-10 M] (Vimp)

- Layer: [Set of all functionalities] must for effective computer networking.
- OSI: theoritical model (7 layers).

imp

Application layer	AL
Presentation layer	PL
Session layer	SL
Transport layer	TL
Network layer	NL
Data link layer	DLL
Physical layer	PL

AL, PL, SL → TCP/IP.
TL, NL, DLL, PL → (actual imp).

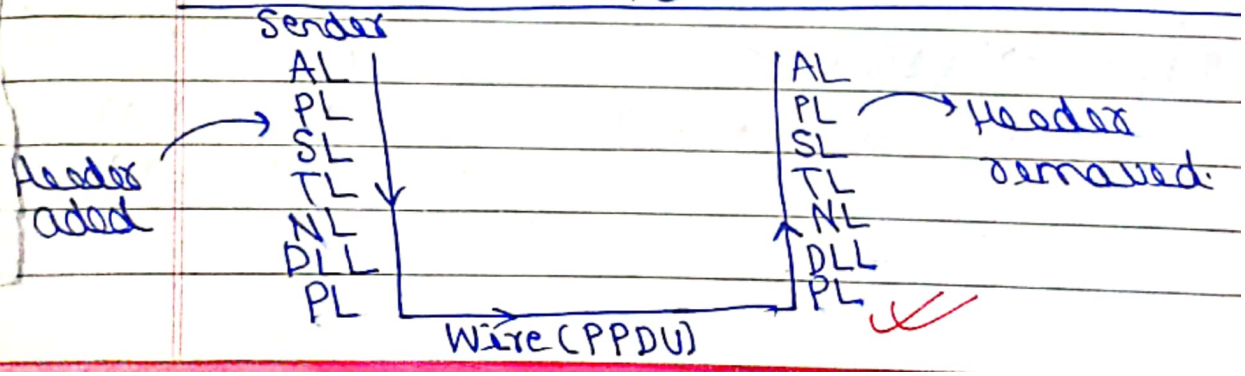
- Suppose in PL: Size = 240B.

240B = [AH + PH] (40B, overhead) + (200B, payload).

payload (actual msg).

$$\text{payload \%} = \frac{200}{240} \times 100$$

$$\text{overhead \%} = \frac{40}{240} \times 100$$



Introduction to OSI

(1) Application Layer
- user interface to n/w (imp)
- Server side synchronization
- [All] Message : APDU
  - length of data

(2) Presentation Layer (imp)
- data conversion, encryption & decrypt
- data compression & decompression

(3) Session Layer
- provides session to end user & manages it
- Provides check Pointing

(4) Transport Layer
- end to end communication (TCP)
  - connection oriented (establish conn, data transfr, close conn)
  - connection less (UDP)
    - fast, but loss of data
- Process to Process
- Port no required
- Segmentation & reassemb
- flow control, error control

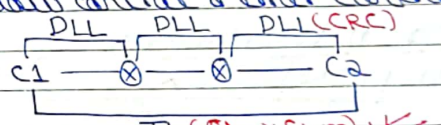
eg → Pg 15 [Segmentation]

Congestion control → n/w Layer
error/flow → TL/DLL control

(5) Network Layer (imp)
- IP → connectn less
- Source to destination communication
- Logical Addressing
- Routing
- Fragmentation (Provide data in order to send to down layers) (MTU)
- Congestion Control

n/w layer (imp)
- Message Switching: connection less
- Packet Switching: connection oriented

(6) Data Link Layer
- Node to node communication
- Physical Addressing (MAC) → [Media Access Control]
- Framing [ethernet/frames]
- flow control & error Control] TL



(7) Physical Layer (imp)
- conversion of bit to signal
- multiplexing/demultiplexing
- encoding/decoding

eg → Pg 22 (Q10: Pg 2 Material)

* Basics of error Control

1. Damage : Single bit error, burst error
   eg: S R
   101010 100101 ⇒ Burst error L = 4
   101010 011001 = BE. Length = 6.
2. Loss & Duplicate

* No. of bits in Error (due to noise)

eg → Pg25 (BW=10Kbps)

imp

* 1 bit delay = bit error rate = $\frac{1}{\text{Band width}}$

* Avg no. of transmission per packet = $\frac{1}{\text{Success Rate}}$ (imp)

eg → Pg30 (Q12)

* Probabilities in error control

1) P_b : Bit error Rate ($1/BW$)
2) $1-P_b$: Prob. of 1 bit without error
3) $(1-P_b)^L$: Prob of frame without error.

4) $1-(1-P_b)^L$: Prob of frame with error. (imp) → atleast 1 error (in a bit) in frame.

eg → Pg32 (2 Chunks)

* CodeWord = msg + CheckBits (imp)
  - CW received by receiver.
  - Payload = CW
  - Header | Payload | Trailer
    - Trailer → error control algo
    - Payload → CW = Msg + CB
  - Trailer applied on CW to check received msg is correct/not

* HAMMING DISTANCE

d = min{all HDs pair} (imp)

eg → Pg35 (Q13)

Given Hamming dist

(ii) Max no of errors that can be detected ⟹ d − 1.

(iii) Max no of errors that can be corrected ⟹ $\left\lfloor \frac{d-1}{2} \right\rfloor$ (imp)

imp

eg → Pg 36 [Probability] (Tree)

ERROR CONTROL MECHANISMS

* Error detection techniques
(i) Parity.
(ii) CRC → DLL
(iii) CheckSum → TL

(i) Parity
- 1D Parity → even / odd
- 2D Parity.

eg → Pg 40 [Block division | ram LHS] imp

Size of CW in 1D Parity : m+1

Size of CW in 2D Parity : m + m/K + K + 1
[Block Size → K] Size of CB

(ii) CRC [Cyclic Redundancy Check]

TL NPDU (TPDU)

DH | NPDU | DT : DLL

DT → Generator Polynomial (imp)

NPDU → Msg + CRC
CRC → CB.

Key → GP

Valid GP → divided by x+1
→ not divided by x.

Size of CheckBit / CRC = degree (GP)

eg → Pg 45 (Adv of CRC) → single bit (imp), burst → odd, ≤ deg, deg (prob).

eg → Pg 47 (CRC technique)

(iii) CheckSum : TL

- Key → Block Size (imp)
- TH | APDU | TT
  - APDU → CW = msg + CS
  - TT → Block Size

eg → 49 [CheckSum]

Receiver rec. BS & CW, performs the same steps, if result → CS, msg.

imp → Size of CheckBit / CheckSum = Block Size

CRC is more complex than CS.
- node to node (DLL)
- end to end (TL)

* Error Correction Techniques

1. retransmission
2. Hamming Code (Error detect & corr^n)

$$m + K + 1 \leq 2^r$$ (imp)

$m \rightarrow$ msg bits
$K \rightarrow \lceil \log_2(m + 1) \rceil$
$r \rightarrow$ no. of check Bits.

[Find the code Word sent]

eg → Pg 57-58 (imp)

eg → Pg 61 [Problem with HC] (1 bit error corrected)
[Shortcut]

* FLOW CONTROL

If sender Capac. > rec. capacity
(To avoid loss of data : flow C.)

$$\text{Throughput} = \text{Channel Utilization} * \text{BW}$$ (imp)

- Throughput: rate of data moved successfully per unit time (bps)
- Channel Utilization: how much time channel is utilized from total time.

* DELAYS

1. Transmission Delay (TD)

$$TD = \frac{\text{Length of Packet}}{BW}$$

2. Propagation Delay (PD)

$$PD = \frac{\text{distance}}{\text{Speed}}$$

C1 → data rate ↑ ⇒ TD ↓ (BW Matters)

Speed of Signal ↑ Matters ⇒ PD ↓

eg → Pg 66 (3 points) (imp)

eg → Pg 67 (Satellite Q) : PD

* Flow Control Mechanisms

(A) Noiseless Channel
1. Stop & Wait.
2. Sliding window Protocol

(B) Noisy Channel
1. Stop & Wait ARQ.
2. Go Back N.
3. Selective repeat.

(A) Stop & Wait Protocol → receiver sends ACK (1 bit)

Total: $TD_S + PD_{S\to R} + \text{Rcvr Process Delay} + TD_R + PD_{R\to S}$ (1 bit ACK)

imp: Channel Utilization $\Rightarrow \frac{TD}{TD+RTT}$ → time taken to send 1 packet.

$CU = \frac{1}{1+2a}$; $a = PD/TD$

(B) Stop & Wait ARQ → error and ACK/NAK [1 bit]. p → error rate

$CU = \frac{TD}{\left(\frac{1}{1-p}\right) * (TD+RTT)}$ → total time taken for transmission of 1 packet; $\frac{1}{1-p}$ → avg no of transm. per packet

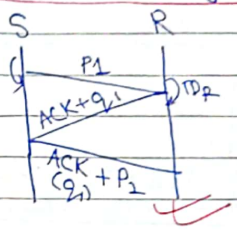
imp: $CU = \frac{(1-p)\,TD}{TD+RTT}$

eg → Pg 72 [Stop & Wait ARQ use seq no (0,1) to control errors]

* PIGGY BACKING → duplex communication.

(i) Stop & Wait

→ Receiver sends same data along with acknowledgement.



$CU = \frac{TD_S + TD_R}{TD_S + TB_R + RTT}$

(ii) Stop & Wait ARQ — p → error rate.

$CU = \frac{(1-p)(TD_S + TD_R)}{TD_S + TD_R + RTT}$

eg → Pg 74 (BW = 4 Kbps)

BW delay Product = BW * RTT → imp

AGAIN Sliding Window Protocol → CU ↑ (than S nW)

No. of packets that can be kept in buffer before getting acknowledgement.

imp: Optimal window = $\left\lceil \frac{RTT}{TD} \right\rceil$

SWS = 1 + optimal window

Sliding Window (ARQ) = SWS + RWS
max no. of seq no

noisy channel

classmate Date Page 12

imp

No. of bits required to represent a seq no in SW = $\log_2$ [SWS + RWS]

* $CU = \frac{SWS * TD}{TD + RTT}$

eg → Pg 81 (Q. 11) HB: Pg 6

B) 2. Go Back N → can lead to congestion.

- SWS = N, RWS = 1
- for less Noisy channel → before error packet
- rcvr discards N Packets & sender retransmits N Packets. (Go Back N)

imp

No. of bits req. for Seq no ⇒ $\lceil \log_2 (N+1) \rceil$

eg → Pg 85 (Go Back N)

$\log_2 (SWS + RWS)$

diag → Pg 86 (GBN) & [Diag]

imp

B) 3. Selective Repeat

(i) SWS = RWS
(ii) Suitable for more noisy channel
(iii) for each error, rcvr discards only error packets & sender transmits only error packets.

Go Back N ww Window of size 15 / 15 from seq

classmate Date Page 13

⇒ SWS = 15

imp

(iv) Search, Sort used

eg → Pg 87 (diag)

eg → Pg 87 (Go Back N)
eg → Pg 89 (Selective repeat)

imp

eg → Pg 90 (Q 13, Q 14) → Pg 7 (Met)

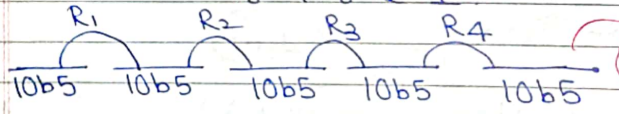
imp

* BASICS OF NETWORKING DEVICES

b/w 2 country	L4 Device (Gateways) (AL/TL)
b/w 2 cities	L3 Device (Routers) (NL)
b/w 2 lans	L2 Device (Switch / Bridge) (DL)
b/w 2 cables (within LAN)	L1 Device (Repeater/Hub) (PL)

L-1 Repeater/Hub (PL) → between 2 cables within LAN

* repeater: - bus topology, datarate 5-4-3-2-1



imp

eg → Pg 94 (54321)

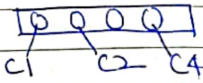
Error control

- Broadcast devices (doesn't check for errors).
- boosts signals / regenerates & forward.
- CD = 1 (Collision domain)

Repeater delay = 1/BW + Processing delay (imp)

eg → Pg 97 (repeaters)

* Hub.



- multiport repeater
- Also broadcast device, doesn't look for error control.

eg → Pg 100 & Pg 99 (for diag) (imp)

L2 Bridges / Switch → DLL (b/w 2 LANs) (can build RT also)

- store & forward device.
- error control done.
- uses MAC Address. (imp)

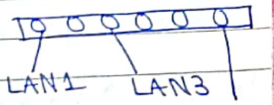
* Bridge (imp)

Bridge delay = Processing delay + Avg Queue delay + Transmission delay

Length of CD = max(L1, L2, L3)

eg → Pg 105 (Bridge Qs)

* Switch



- multiport bridge.
- Star topology

L3 router recog. n/w using SM → L2 Switch recog LAN using MAC Ranges (imp) → L1 Hub broadc. to all PCS

* Cut through Switch doesn't perform error control: only TD. (imp)

eg → Pg 108 (Switch)

* IP HEADER

- 5 Words / 20 B fixed + Padding.

1 W = 4 B = 32 bits (imp)

eg → Pg 110 (IP Header) → VITT IPsrd | R|M|E|DF|FO TPH

- Version: (4 bit): IPV4 / IPV6
- Internet Header Length (IHL): length of Header in words. (4 bit)
- TOS: service to be used by down layer (indicated by NL) (8 bit).

consider TPDU (imp)

Or 118

eg → Pg 114/115 (FO, MF, DF, TTL)

- TTL: To avoid looping of packets [max Hops]

eg → Pg 121 [Vimp] (imp)

* TCP Protocol

- connection oriented / byte oriented / max reliability / cumulative ack / full duplex commun & piggy backing.

eg → Pg 123 [No. of bits] [Bit Length] (imp) bit delay = 1/BW

* TCP Header

- 20B Header : 5W (imp)

diag → Pg 126 [HRCRCU]

(i) * 16 bit port no : 0 to 65,535 ports.

(imp)
0-1023	Special Purpose (21, 80)
1024-49151	Unused (FTP, http.)
49152-65535	Normal Users

Port no + IP = Socket

- If 1 tcp conn is already established on a pair of sockets, other conn. can't be established (imp)

eg → Pg 127 [Virtual Circuit]

(ii) Seq no. (32 bit) → word 2

eg → Pg 127 (1000B)

S —Seq 1500→ R: [I have received 1500-99, and no 1600 packet.]

S ←ACK 1600— R

next byte to be transmitted.

(iii) Hlen : like IHL [words].

(iv) Control Bits : (9 bits)

(1) SYN : (for establishing connection)

eg → diag (Pg 131) [4 States] (imp)

(2) FIN : (closing conn)

eg → diag (Pg 132) [6 States]

(3) RST : reset conn

(4) URG : URG = 1 (urgent byte to proceed)

eg → Pg 134 (Single Byte & range)

(5) PSH : push data to down layer [imp data], supports Nagle's Algo. (not required payload) (insufficient)

Payload / MSS = APDU - 20 (at TL)

* 20 | APDU (imp)

Max Segment Size = 65,536B

∴ Max. Payload = 65,536 - 20 - 20
⇒ 65,496B (TL) (AL)
or [0, 65,495B]

NOTE: If all control bits are 0, invalid connection.

eg → Pg 140 [Pg 10 - Q8] [Segment exch.] (imp)
(3 way handshaking in TCP Protocol)

* min. TCP header = 20B,
* efficiency at TCP level = payload / (abv.)
* Wrap around 32 bit - seq no Space.

eg → Pg 142 [Pg 11 - Q10] (Gate)
[min time, a seq no can be reused].

* FLOW CONTROL IN TCP

16 bit ⇒ RWS explains other party to understand their capacity.
⇒ Receiver use this formula:
RWS = Buffer Size - [(A) - (B)] (imp)
A: last byte received
B: last byte read.
eg: read 1000B, out of 2000B recv.

In Stop & Wait: SWS = Advertise window

(imp) [SWP]

* Sender: SWS = RWS - (Last Byte Sent - Last Byte Acknowledged)

eg: sent 2000, recvd ACK of 1000, 1000 still in Buffer

eg → Pg 146. (Stop & wait Flow Control)
[Pg 11 / Q12] → SWS = Advert. Window

* Calculation of TIME OUT IN TCP.

- Retransmit packet, if not getting ack. till timeout.

Timeout = 2 * RTT

where, RTT = α * Prev. RTT + (1 - α) * Curr RTT (imp)

α → Smoothening factor

eg → Q13 [Pg 148] (Prev. RTT = 30, α = 0.9, Curr. RTT = 40).

eg → Pg 149 [continuous 2RTTs]

* UDP: (User Datagram Protocol).

(low of data access)
- connection less (no connection / flow control / no reliability, no ACK / SYN)

* Header: 8 bytes (imp)

	00 to 15	16 to 31
W1	Source Portno.	Dest Portno.
W2	length	Checksum
	Data	

used by receiver first to check data, no ACK / NAK involved.

→ less overhead [faster].

diff → Pg 151 [TCP V/S UDP]

DNS, VOIP, TFTP → use UDP
HTTP, SMTP, FTP → use TCP

imp

* Routing Protocol → Static routing
→ dynamic routing → DVR, LSR.

[which path to be selected b/w C1 & C2 is decided by routing algorithms]

(ii) Static routing (1980s)
- Path to particular dest is fixed
- next Path is computed using Static routing Table (by IP of neighbours)

eg: $R_1 \rightarrow R_4$ (192.:
$R_1 \rightarrow R_5$ (:182.:

disadv: if link is cut, router cant send packet to destn, through another path.

imp * Flooding: packet will be forwarded to all the connected neighbours, but will result in duplicate packets.

eg → Pg 157 [Flooding diag]

* Time to Live (TTL): Max no: of Hops [8 bit]

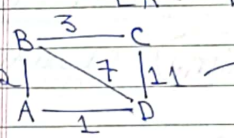
eg → Pg 158 [diag] [$C_1 \rightarrow R_1 \rightarrow R_4 \rightarrow R_2 \rightarrow R_5 \rightarrow C_2$] = 5

NOTE: TTL is used, to avoid looping of Packets (if TTL = 0, & not reached destination; packet discarded).

(iii) Distance Vector Routing → Dynamic Routing.

- Every node knows only abt neighbours and maintain their distance vectors.
- Every router exchange their DV with neighbours parallely.

[A ← D, B; C ← B, D]



RT at B:

D	dist.	NH
A	2	A
B	0	B
C	3	C
D	7	D

DV

eg → Pg 3 (Ravi N.) [Pg 159] exam sc

eg → Pg 4 [Shortcut] (RT of A and B)

imp If n edges → max n iterations → Shortest Path. (to stabilize)

NOTE: if n nodes → (n−1) iterations, to compute the final routing table.
[in exam, compute by observ.]

imp * Which edge not used?
(i) alternate path exists, never used).

eg → Pg 6 [Ravi Notes]

eg → Pg 166 [Cost of link N2-N3 reduces to 2], compute new DV at N3.

eg → Pg 167 [After link goes down, N2 reflects this change immediately in its DV]

→ [Issue in DVR]

* Count to Infinity Problem (DVR)

(ii) Link A-B is added; B updates its DV immediately, & share with nbrs.

eg → Pg 1 (Ravi) → Pg 166 [Good News]

eg → Pg 2 [Bad news] (imp) [takes time to reach ∞]

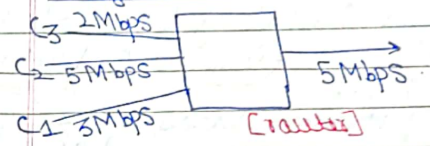
* In DVR, we are sending only the DV, not hops (problem)
If we send full RT, BW ↑ (not in 1980s)

(iii) Link State Routing (after 95)

- In LSR, each node sends 'Hello' Packet to all nodes in n/w, but DVR sends only to neighbours
- By looking at LSP, each node constructs its own RT using dijkstra. (imp)
  ∴ [1 iteration only]
- When change in n/w: LSP sent again.
- If mult. LSP at node, LSP with highest seq no selected

* In LSR/DVR, Shortest Path Algo is run at all nodes.

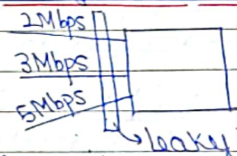
* Congestion Control: (n/w layer func^n)



- Precautions for congestion control
  → cumulative acknowled. in TCP
  → Selective Repeat instead of Go Back N.

* Algorithm for Congestion Control

(ii) Leaky Bucket



(imp) [take all data & provide to router with constant rate] [Bursty → uniform traffic]

eg → Pg 176 [Q12-Q1] (Min. Size of Queue to prevent loss of data)

eg → Pg 179 [Good Q on Leaky Bucket]

Q. Output = 5Mb/min
Input = 100Mb/min for 12S; & no input for 48S.
∴ 20Mb of data → input (12S)
1 Mb of data → output (12S)
→ ∴ [19Mb → Size of Queue]
[Problem with Leaky Bucket]

C / M-T

(iii) Token Bucket — Token allocator
- tokens gener. at regular intervals.
- if packet is ready, a token is removed from bucket, & packet is sent.
- if no token: Packet cant be sent to network.

imp → [advert. W]

Gate: SWS = min { ~~receiver~~ window size, ~~congestion~~ congestion window }

Router can handle 4 KB
ww if, RWS = 6K ∴ SWS = 4KB.

* Congestion Control Algo Phase
  - → Slow Start Phase
  - → Congestion avoid Phase
  - → Congestion control Phase.

(ii) Slow Start: Start with MSS, Send Packets exponentially, cal. CW, Threshold [when error Packet], after threshold reaches, additive increase algo.

eg → Pg 184 [imp] ← Max Traffic

(iii) AIMD [Additive inc. multip. decrease]
Start with MSS, send Packets AI,
if error, start with threshold.

eg → Pg 185 ← ww Traffic

[Inc by MSS?]

eg → Pg 186 / 187 → Vimp [Pg 13/Q6]

[Only such Q will come]

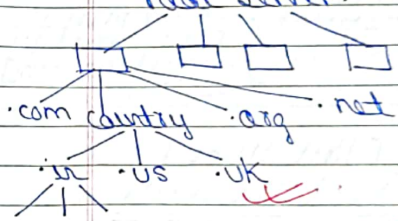
* APPLICATION LAYER PROTOCOLS [Structured]

① DNS Protocol (Domain Name Server)

domain name ←map→ IP Address (DNS function) — imp

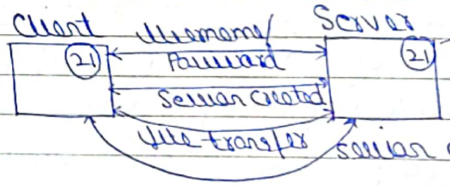
dns record (IPAdd., domainName, Validity)

eg → Pg 191 [Steps to get IP for a domain Name] (Root Server → TL Server) [DNS lookup]



ww UDP

② FTP [File Transfer Protocol] → needs Port 20 & 21
- ww TCP. (Secure) (fast) [Stateful]
- transfer data b/w 2 comp on the same local n/w (imp) [Port no 21]
- User must authenticate an FTP Server.



FTP → can copy data from remote server

Telnet → can actually access & modify program on remote server

eg → Pg 194 [Diag: Telnet > FTP]

⑤ SMTP [Simple Mail Transfer Protocol]
- Not Suitable for client authentication.
- TCP/IP Protocol for send/rcv mail
- 2 Parts → message: prepares msg
  → transmitter: transmits msg
- SMTP Server (act like post office): handles routing of msg.
→ MAIL FROM : Sender
  RCPT TO : recipient
  DATA : message
  QUIT : quit
  Codes and their purpose [to simplify mail comm.]

④ POP Protocol [Post office]
- used to download the mails when user comes online; Maps user to msg and allocate into cloud Space
- useful for checking emails from comp that is in specific location. mailloc → IMAP [multiServer]
- retrive email from mail Server.

imp

eg → Pg 197 [Pop Protocol / DNS / email]
[needs underlying TCP]

⑤ HTTP PROTOCOL [Hyper Text Transfer]
- (request, response protocol)

Stateless

Web Browser (Client) ⟷ http request / http response [html files] ⟷ applic^n running (Server)

HTTP Working: Passing URL → Sending request → Server Response

- used in e-commerce transaction.
- Passing URL = Port No / IP Address.
- Sending request → GET: get URL data
  POST: post on Server
- HTTPS → Secure Version of HTTP.

eg → Pg no 200 [M1, M2, M3]

TL	→ Segment
NL	→ Datagram
DLL	→ Frame
AL	→ Message

protocol data unit

PDU

SECURITY
- Symmetric Algo: Key used by client & server are same.
- Asymmetric Algo

eg → Pg 202 [Symmetric Algo → Cipher Texts]

- Asymmetric Algo
  - Public Key: Known to everyone
  - Private Key: Known only to authorised ones.

imp

(i) Encryption
Sender: A, Receiver: B
Sender encrypts with Public Key of B
Rcvr decrypts with Private Key of B.
Public Key of rcvr.

(ii) Digital Signatures
Sender signs with private Key of A
Rcvr accepts with public Key of A
Private Key of Sender

eg → Pg 204 [MCQ] → 2 Security Aspects

1) * RSA Algorithm : to generate the pair of private & public Key. (imp)

Public Key = (e, n)
Private Key = (d, n)

$d * e = 1 \bmod \phi(n)$

$M' = M^e \bmod n$
$M = M'^d \bmod n$
[encrypt always with Public Key]

eg → Pg 206 [Numerical]

2) * Fermat's Theorem : $a^{p-1} \bmod p = 1$ [DM]

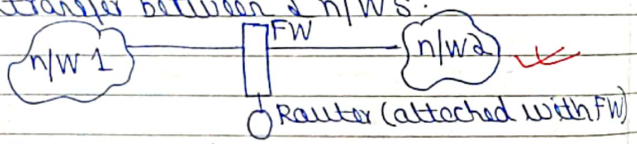
Q $3^{52} \bmod 5$.

$\Rightarrow 3^4 \bmod 5 = 1$ (Fermat's theorem)

$\therefore (3^4)^{12} \cdot 3^3 \bmod 5$

$\Rightarrow 2$ (Ans) $\quad 27 \bmod 5$.

* FIREWALLS : Security s/w that will be used to have a secure communic^n transfer between 2 n/ws. [PDA with 2 Stack → TM]



→ only authorised traffic can pass.

Firewalls (imp)
- → Packet Filters : use Selection criteria to filter out packets.
  - Rule : based on IP/port
  - Action : if this IP, then :
- → Application Gateway

* Application Gateway acts as Proxy Servers. (remove Source IP, attach its own IP)

eg → Pg 58 (diag) [Rev Copy]

* IPV4 Addressing [32 bit]

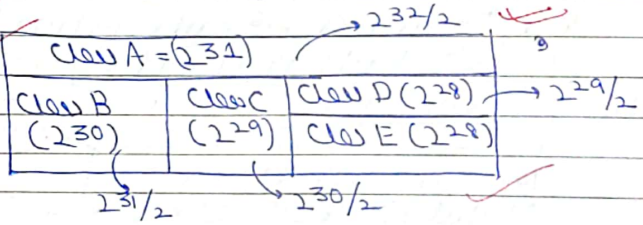
→ IP Address : logical add: at n/w layer, used to identify Systems

0-255 . 0-255 . 0-255 . 0-255

32 bit → 4 octates

∴ 2^{32} IP Addresses are difficult to manage, ∴ divided into classes :

Classful Addressing (imp)
- Class A : 0.0.0.0 to 127.255.255.255
- Class B : 128.0.0.0 to 191.255.255.255
- Class C : 192.0.0.0 to 223.255.255.255
- Class D : 224.0.0.0 to 239.255.255.255
- Class E : 240.0.0.0 to 255.255.255.255



1) CLASS A (1st bit Kept constant) (imp)

* 2^7 networks, 2^{24} hosts/network.
∴ 2^{31} hosts (ip)

eg → Pg 60 (Rev) [Class A – diag]

First ip of any n/w = n/w Address
last ip of any n/W = Broadcast Add

NOTE: First & last n/w in Class A: not used.
↳ given to local host

imp → Spcl IP add = $2^{24} + 2^{24} + (2 \times 126)$
Usuful = $(2^{24} - 2) \times 126$

* IP = 100.127.224.5 [Class A] → n/w Add = 100.0.0.0 [found using SM]
→ BA = 100.255.255.255

2) CLASS B: (1st 2 bit kept constant)

imp → 10 . . . .
n/w | Host

* 2^{14} networks, 2^{16} hosts/network
∴ 2^{30} hosts or IP address

eg → Pg 218 [Class B – diag]

Spcl IP = $2 * 2^{14} = 2^{15}$ (imp)
Usuful IP = $(2^{16} - 2) * 2^{14}$

3) CLASS C:

imp → 110 . . . .
n/w | host

* 2^{21} networks, 2^8 hosts/network
∴ 2^{29} hosts or IP

Spcial = $2 * 2^{21} = 2^{22}$ (imp)
Usuful = $254 * 2^{21}$

eg → Pg 219 [Class C]

eg → Pg 220 [Host imp Table] (RL)

* SPECIAL ADDRESSES

(i) N/W Address: used by router to find the destination n/w, where packet has to be sent.

Q Dest ip: 172.40.60.20 (N/W Add: 172.40.0.0) [Class B]
∴ router sends to this n/W.
→ Router uses Subnet Mask to determine n/W Add & host Add within n/W.

imp → Class A: 255.0.0.0
Class B: 255.255.0.0
Class C: 255.255.255.0
} default SM. [without subnetting]

eg → Pg 223 [Router ANDs, SM with dest ip to get the n/W Address]

(ii) Directed Broadcast Address (make all host bits 1)
– has to send to all hosts in other n/W.
– last IP Add of n/W = DBA.
if router ip: 172.40.60.20
n/W Add: 172.40.0.0
DBA: 172.40.255.255 ← Class B

OS → IS

(iii) Limited Broadcast Address (LBA)
- If a packet has to be sent to all the systems in the same network. (imp)
- IP Header:

150.20.40.60	255.255.255.255	Payload
Source IP	default LBA	

(will be sent to all hosts in n/w with n/w Add: 150.20.0.0) [Class B].

(iv) Loop Back Address (127.0.0.1)
- First n/w of class A; 1st host/IP (valid)
- Server is reading on your m/c, want go to wire, loop back from n/w layer (where IP header is checked)

Diag → Pg 225

* SUBNETTING: Process of dividing larger n/w into smaller n/w.
disadv: wastage of add will be more.
eg → Pg 236 [Class C → 8 subnetworks].

i). Max no. of hosts possible = max no. of useful IP. (imp)
[1st & last S n/W not used, 1st & last IP of each S n/W not used]
[like class A]

* Design Subnet Mask

Subnet bits = $\log_2$(No. of Subnetworks)

Useful S n/w = 254

* Class B is divided into 256 subnetworks [16 host bits]. S n/w bit = 8.
8 bit (1s) [Subnet] ($\log_2 256$)
8 bit (0s) [256 hosts / S n/w]
∴ default SM - B: 255.255.0.0
SM: [255.255.255.0]
↳ [to get S n/W Address]
Possible = 254 / S n/w

eg → Pg 230 [Class A is divided into 4096 hosts].
eg → Pg 231 Valid SM, 40s/41s in any order].
eg → Pg 232 (Q1) Class B (Given)
Subnet Mask = 255.255.248.0 (imp)
Max hosts/subnet ?

NOTE: 255.255.255.240 → Subnet Mask [Class C]
11110000
→ when no of 1s = no of 0s (host bit) [max utilization (imp)
Class B: 255.255.255.0
Class A: 255.255.240.0
} SM for max utilization

eg → Pg 239 / Pg 240 [Range of Sub n/ws, given n/W Address]

* disadv of classful Addressing: many IP is wasted.
if req: 300 ip, Class B → 2^{16} IP (many wasted)

* SUPERNETTING : Process of combining 2 or more subnetworks.

eg: if my 1000 ips, combine 4 class C.
∴ 256 X 4 = 1024 hosts

32
22 10 ← log(1024).

* CIDR Notation [Classless Interdomain Routing].

⇒ 172.48.7.5/22 → 32 [IP size]
22 10.

[IP Address is in a n/w which consists of 2^{10} hosts] imp

eg → Pg 243 [CIDR Q-2]

eg → Pg 244 [Company has been allocated block of IP space: 128.119.152.0/21]

imp [SC]

/26 → /28 (+2) divided into 4 SNs [2^2].
/26 → /24 (-2) combine into 4 SNs.
[CIDR Prefix]

* /28 divided into 2^{12} Subn/w
(28+12 = 40 > 32) [Not poss]

* VLSM [Variable length Subnet Mask]

⇒ I have /24 n/w → 28 hosts

For rough Qs, calculate SM first.

A (128) → /25
64 → B → /26 [VLSM]
64 → C → /26.

eg → Pg 251 [(a)] Vimp

* BASICS OF IPV6
- Range of IPV4 = 2^{32} (not sufficient)
- Range of IPV6 = 2^{128}.
- [Faster routing, Security ↑, No Broadcast]

imp

* 8 double octets separated by :
16b : 16b
4b 4b 4b 4b
(7CF1) [Hexadecimal no]

eg → Pg 254/255 [to avoid length of IPV6 address]

* Conversion of IPV4 to IPV6
- for compatibility (∵ 5% Systems use).

IPV4: 2 double octets
↓
IPV6: ——:——::

* Subnetting in IPV6.

ex: [2853:7EF2::/48] → This IP is in a n/w of 280 Hosts

imp [CIDR notation]

* Fields in IPV6 → indicates quality of Pkt
[Version, Priority field, Flow label, Hop limit (TTL)]

* ETHERNET (imp)

(Thick) 10 base 5	→ 10Mbps / 500m	Bus
(Thin) 10 base 2	→ 10Mbps / 185m	
(Twisted Pair) 10 base T	→ Star.	

* Ethernet uses Manchester Encoding at PL
[Bit 0: (low → high)
Bit 1: (high → low)] (Rev of diff.)

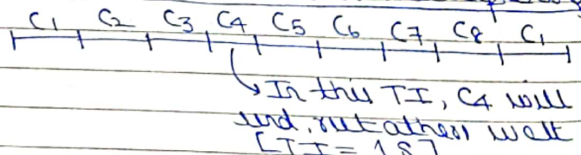
eg → Pg 259 (Manchester encoding)

* Signal levels represented in baud/s.

imp → [baud rate = 2 * bit rate] (ethernet)

* Ethernet Operations.

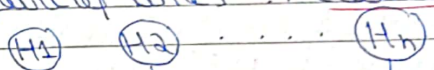
(i) Slotted Aloha: [Particular time to send data]: time division multiplexing



→ In this TI, C4 will send, not other will [TI = 1s]

→ Discrete time frames / No Possibility of collision. (imp)

(ii) Pure Aloha
→ continuous time frame.
→ Transmission of data doesn't depend upon time. (Frame can be sent in any part of time] ∴ (collision ✓) (imp)



S1: If H1 wants to send data, it prepares ethernet frame (imp)

DLL: [LLC | NPDU] LLCPDU

LLC ← DLL → MAC (sublayers)

[MAC | LLCPDU | MACT] : frame

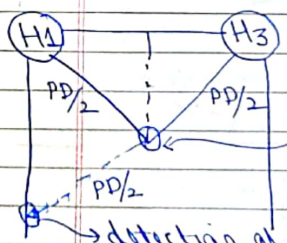
14B (46–1500B) 4B (CRC)

∴ Smallest ethernet frame = 64B
Largest ethernet frame = 1518B (don't stop)

S2: H1 checks whether data is flowing on wire/medium

S3: If data not flowing, H1 sends data.

S4: If another Hi sends at that time, collision occurs → · Voltage ↑ · power div. · collision signal will reach to one host.



S5: After collision detected, Jam Signal sent to alert other station. → after time: PD.

S6: Collision recovery (Binary exp rece alg) starts [reduce prob of collision on retransmission]

if, collision no. → i = min(10, # no. of collisions)
K = {0, 1, 2 ... $2^i - 1$}
∴ Waiting time = K * Slot time (imp)

eg → Pg 264 (Q3/Pg7) [Prob that node choose K=4] K = {0, 1 .. $2^i - 1$}

* Time taken by bit to travel from beginning to end of n/w = PD.
if, PD = 225 bit time
⇒ 225 x bit delay = 225/BW

eg → Pg 266 [Q6/Pg8]
Time = Collision detection time + TD of Jam Signal + TD of packet

NOTE: Transmission delay of JaM Signal (if not mentioned size) = 1 bit delay (imp)

eg → Pg 266 [Q7]
eg → Pg 268 [Q8] [A's Packet completely delivered to B: time]

* MINIMUM PACKET SIZE IN ETHERNET
MPS = RTT * BW (Gate)
RTT = 2 * PD [Like BW delay]

* There are n Stations. Each Station attempts to transmit with prob 'p' in each time slot.
eg → Pg 270 $(nP(1-P)^{n-1})$.

eg → Pg 271 [Vimp Q] (imp)
(Station with lower token value wins)

Previous Years.

* Delimiter pattern/Flag = 01111110 (2 times (Prev Yrs))
o/p bit string after stuffing = 011111001 01
Now, [break b4/last one, in stretch of 1s]
remove this 0 (∵ o/p string).
add 0 (∵ i/p string).
(imp)
∴ String i/p: 01111101 01

* if, Max. window size for data transmission using Selective repeat protocol is asked: eg. = SWS.

* If 2 invalid Satellite: [b/w 2 stations] [$PD_{S_1} + PD_{S_2}$]
data → Satellite → data
Sender 1 ← ACK ← ACK ← Sender 2
∴ PD ↔ 2PD
∴ Distance = $\frac{4}{2}$ * (dist b/w S1 & Satellite)

imp

* When word 'TRANSIT' : (no. of packets in the link which has not reached the dest') ⟶ WS = (OW) $\left\lceil \frac{PD}{TD} \right\rceil$.

When word 'TRANSMIT' :
⟶ Optimal Window = $\left\lceil \frac{RTT}{TD} \right\rceil$
[atleast 1 packet will have reached dest^n]

* DES (Data Encryption Standard) performs permutation on the elements of i/p alphabet.

* FTP : PROMPT, PORT [Portno 21]
(Use it, if you dont want to be prompted for each file transfer)

HTTP : HEAD, GET, POST, PUT
(to get metainfo)

SMTP : RCPT, HELO, EHLO, SIZE, VRFY
(recipient) [first SMTP command]

[Network Security]

* SHA-1, SHA-512, MD-2, MD-4, MD-5 : (these are used to generate Message digest.

* Bit Synchronization done at Physical L. → Sublayer of DLL
MAC Sub layer does channel Sharing.

* [Same n/w = Similar n/w address]

eg] ⟶ Q1.26

imp

* Let SWS = 32 TDal 1 frame = 1ms
∴ 32 Packets sent ∴ Time = 32 msec T_1

Waiting time = Total time − T_1
Total time = $TD_S + TD_R + RTT$ (piggy backed)
⇒ 50+2 = 52
20 msec

* Base 64 encoding ⟶ last 64 = 6
∴ multiples of 6.

10100011 00110111 11101001
10101011 ⟶ (Given: 32 bits)
∴ Pad 4 bits [36], at LHS
101000 110011 011111 101001
101010 110000
⇒ (0 z f p q w)
Base 64 encoding Table : [a = 26, A = 0]

imp

Bit Stream 1 0 0 0 0 1 0 1 1 1 1
(Manchester coding & differential Manchester encoding)
eg ⟶ Q1.18 (read from Graph)
[Imp formula list Grp]

* IEEE 802.11 wireless LAN uses CSMA/CA, ethernet uses CSMA/CD.
Contention possible in CSMA/CD.
(∵ Pure Aloha)

imp

* Satellite network ⟶ high PD n/w
[CSMA/CD not suitable]

[1 frame transmitted in 1 slot]

prob = 0.2

* In a Time Slotted Aloha: Each host has to provide min. 0.16 frames
imp
$0.16 = (0.2)(0.8)^{N-1}$
$N = 2$ ← Max hosts practical can support.

* Router & Bridge (L3, L2 device) Selectively forwards data Packets [Self Learning bridges build up routing table by inspecting incoming Packets. (uses Spanning tree Algo)]

* Source Routing: (Gate) allows Sender to Partially/completely specify the route of the packet [S → D]. (allowed in IP)

* Routing Algo, doesn't differentiate between TCP/UDP.

* ARP [Address resolution Protocol], is used to find the MAC Address, for a given IP Address. RARP = opp

* Optimum Packet Size → less no. of packets required to deliver a msg.

* AL can send any amt of data to TCP Layer.

NA = host bits are 0
BA = host bits are 1
all host bits are 1.

classmate Date Page 43

* Subnetted class B network.
DBA = 144.16.95.255 (n/w, host). SM = ?
imp
16 bit host
Subnet bit, host bit.
Now, 01011111.11111111
(subnetid) (anything)
∴ 255.255.224.0 / 255.255.240.0 / 255.255.248.0 / 255.255.252.0

* TCP can use both Selective and cumulative acknowledgement.

* n/w: 200.10.11.144/27 (third DBA)
Last IP add. of this network, that can be assigned to host?
DBA (max host bits 1).
SM: 255.255.255.224.
⇒ AND → N/W Add: 200.100.11.
n/w 100 00000
S n/w bits
imp
1st IP: 200 100 00001
last valid IP: 100 11110

* RIP uses DVR uses UDP. imp
2017 OSPF uses LSR uses neither TCP/UDP.
BGP used to scale OSPF efficiently.

* In IPV4 frame format, there are 40B of option field (38 bytes used, 2 reserved). (max 40B in option field)

* In TCP, when the Bandwidth delay Product increases beyond 64K, receiver window Scaling is needed. During Window Scaling, Max RWS goes from: $(2^{16}-1)$ to $(2^{16}-1)\times 2^{14}$: 65535 to 65535×2^{14}. (Window Scaling)

* UNIX SOCKET API (convert to passive Socket) (Text)
  - Bind: Binds a Socket to address (already created)
  - Listen: waits for connects to Socket
  - Accept: Accepts a connect to Socket
  - rcv: receives data.
  (order)

* A process with a connected UDP socket can call connect() again & again. A process with a connected TCP Socket can call connect() only one time.
  A connected UDP Socket can comm. with only one peer at a time.

* [Symmetric Key]
  1] Secret Key cryptosystems:
  (2 people, 1 Key ; 3 people = 3 Key).
  $\Sigma(n-1) = \frac{n(n-1)}{2}$ Keys. (nC2) (imp)

* For Public Key cryptosystems, [1 Pub, 1 Priv.] with each Person : 2n Keys. (RSA)
  $g^a \bmod p$ = Shared Key of A.

* Diffie Hellman Key exchange Algo: $(g^{ab}) \bmod p$ = DH Key (imp)
  p: modulus
  g: primitive root.

* (Gate)

Packet Switching	Circuit Switching
- Path is dynamically decided, Paths are shared. (utilizn ↑)	- Path is Pre-decided, and noone can take the path, unless packet reaches (utilizn ↓)
- More delay	Less delay
- More per Packet processing.	Less per Packet processing.
- Can lead to reordering.	don't lead to reordering.

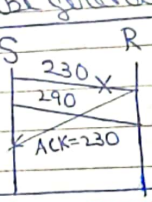
* PPP (Point to Point Protocol) is a DLL Protocol.
* Fast retransmit / Fast recovery is part of TCP congestion control algorithms.
* Slow start Algo [Congestion + Window] Flow Control

Q [Destination, Mask, Interface] (imp) (2 time repeat)
Given: IP Address.
Interface?

* Start with largest Subnet Mask [if matched, to dest; forward in that interface] (if not, check next)

[ICMP errors]

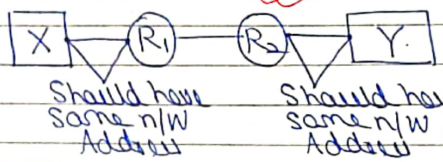
* ICMP Port Unreachable error will be generated for both TCP & UDP.

* 

TCP
Data = 60B.
ACK = 230.
[host subnet bits = 1]

Q SN mask = 255.255.255.224. (imp)
Given 6 IPs (Class C - observ.).
How many distinct SNs?
[Check 1st 3 bits of last octet]

eg → Q3.37 [ME sam] (Q3.38)



* Layer 4 Firewall (can look at all headers of PL, DLL, NL and TL.
- It can block entire Http traffic in a time Interval will just block Port 80. (But can't manipulate a Specific web; we need AP for that).

* Checksum is recomputed & stored again at each router.
∴ may change

(imp)

(RL)

* In Slow Start Phase of TCP cong.
(i) The cWnd increases by 1 MSS on every successful acknowledgment
(ii) The cwnd increases exponentially on every RTT. ∴ [2^i RTT]

* Bridge routing (self learning bridge) builds up RT, using Spanning tree algo, to avoid loops.

[Ethernet Address]

* MAC Address of 2 Systems (in any network), can never be Same.
∴ MAC Add. → always unique.

* DHCP is used to assign IP dynamically [doesn't require addresses]
DNS: Text Address ⟷ IP address.
ARP: converts IP to MAC.

(imp)

* Stateless Protocol: no info retained; by either sender/receiver → FTP
[I] some authorization req; stateful
Now, TCP → Stateful, POP3 → Stateful,

* TraceRoute reports a possible route that is taken by moving from Host A to Host B by using ICMP reply packet to A, by each router; in ascending order of hop dist.

(imp)

at least 3 non-overlapping channels are used. [X, Y, Z]

Ethernet	Wireless LAN
- uses CSMA/CD Protocol.	uses CSMA/CA Protocol.
- Collision detection, avoidance & recovery (Exp Back off Algo)	- Collision detection not possible. Collision avoidance done by RTS-CTS
- We don't use ACKed frames. (imp)	- We use ACK frames.

* In DNS lookup; we get [IP Address from Domain Name], but in Reverse DNS lookup; we get domain name from IP.
→ Check in add.arpa

* Count to ∞ Problem: V (imp)

A — X — C — D — F

(t+100); Value of EA.

∞	2	3	(t+1)
3	4	3	(t+2)
5	4	5	(t+3)
5	6	5	(t+3)

Eg → Q 4.23 [Grp]

* BW, PD, [Packet length]
Processing delay, ack frame bytes given. (imp)
∴ CU = TD's / (TDs + TDR + Process. Delay + RTT)
∵ Piggy Backing not mentioned.

* S — R1 — R2 — D

NL visited: 4 times.
DLL visited: 6 times.

* TCP Connections
SMTP/TELNET: 1 TCP Connection.
HTTP: multiple TCP connections.
FTP: 2 TCP connections. (imp)
→ data connection (P-20)
→ control connection (P-21)

eg → Q1.32 [Switch delay]
(TD, PD, Switch delay)
(ME soln) → GO (Grp) (Vimp)

* Efficiency = transmission time / (transmission time + Polling time) [CU] (imp)

⇒ A node can max. transmit 1000 B
∴ Tt = 10 Mb — 1 s [TD]
1000×8 — x
BW = 10 Mbps

∴ Throughput = Efficiency × BW

* Goodput = Successfully deliv. Packets (ack received) / Time for transmission.

Eg → Q3.25 [Grp-Formula]
1+N time → ACK of P-1
2+N → ACK of P-2
i time → ACK of (i−N) packets
[x+N] x

[Not Application layer data (Encrypted)]

* Q — R_1 — R_2 — R_3 — H (IMAP)
  By Sniffing at R_2 we can get IP address (S.IP & D.IP of Q, H), Port No (SPno, DPno), Link Address of (R_1, R_3) only.

* The web Browser first need to figure out IP from URL using DNS, then establish TCP connection at port 80. Once TCP connectn estab; browser sends HTTP req using GET. Finally web Server → HTTP response. [HTTP needs an underlying TL Protocol] (IMAP)

* With digital Signature, only Sender can Encrypt / Birthday attack; [No other party can; no other party has its Private Key]

* If n is a power of 2 (2^n), min. no of multiplication needed to compute a^n. [with temporary variables] (Exponent)
  a^8 = Max mul = 8 (n)
  a^8 = b = a x a, c = b x b, Ans = c x c
  ∴ 3 mul ($\log_2 8$) ∴ [log n]

* Firewall = PDA with 2 Stack (Turing m/c)

* If there are 15 routers, Total no. of messages = [$^{15}C_2$ x 2!] [b/w every pair] (total pairs)

* Token Bucket Problem

Capacity of Bucket (C) = 1MB
Max output rate (M) = 20MBps
Token rate (T) = 10MBps
∴ Min. time required to send 1MB of data: (Bucket Size)

$$S = \frac{C}{M - T} = \frac{1}{10} = 0.1S$$

0.1 — 1MB
x — 1MB

[Given empty bucket]
(i) bucket full, & we need to send 12MB), ∴ (12 − 8S) * 0.1

TEST [2019 (Subject Test)]

* Sender email → receiver. Client. [Mail first goes from Sender to (1) SMTP Server, then DNS lookup of receiver's IP, SMTP over TCP to (2) encrypt mail from Server to receiver's mail Server, & receiver uses HTTP (4) to read this email (∵ browser).] (mail Process)

[Mail Process]

when it gets RWS=0, ... Sender immediately starts Persistant timer.

imp

* Timer
1. Retransmission timer: used when expecting ack from receiver
2. Persistent timer: Keeps Window open even if other end closes its window.
3. Keep alive timer: detects when other end crashes/reboots.
4. Time-wait timer: measures the time a connection has been in timed-Wait.

* BW = 10 Mbps
detects collision during trans. (12th bit on wire) L = 12
Speed of light = 2×10^8 m/s
imp
distance of station from detector m/c: $d \leq \left(\frac{L}{2}\right)\left(\frac{V}{B}\right)$
[collision detection formula]

* AIMD Algo.
WS at start of Slow SP = 2MSS
Threshold = 12MSS
Time out occurs during 7th Transm.
CW at 12th Transmission:
eg → Q12

* In Slow Start Algo, Size of CW inc exponentially until it reaches threshold after that, there is additive increase.

Leaky Bucket

* Capacity of Bucket = 200MB
o/p BW = 2MBps
Hart data = 450MB
i/p or Hart BW = ?
Soln. Bucket Size = i/p data − o/p data
imp → 200 = 450 − o/p data
∴ o/p or network data = 250MB
∵ 2MB — 1S ∴ x = 125S
250MB — x
Now, i/p BW: 450MB — 125S
x — 1S
∴ Hart BW = 3.6 MBps

Inverse Mod Int

* e = 7 (RSA) ∴ Public Key
Now, $d * e = 1 \bmod \phi(n)$
$d \times 7 = 1 \bmod 160$
$\therefore d = 7^{-1} \bmod 160$ [modulo inverse]
imp
[Calculate mod−1]
$\therefore 7x = 160 + 1$
$7x = 161$
$x = 23$
$\therefore d = 23$
∴ Crypt = 11 ∴ M = $(11) \bmod n$ priv.
$\Rightarrow (11)^{23} \bmod 187$
$\Rightarrow 88$

Account: R S M
HR: S P V
But = 7
Worst = 9
eg → Q17

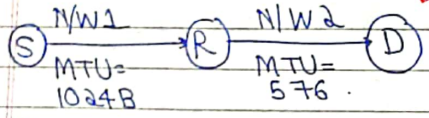
* TCP supports multiplexing.

* ICMP errors
  - Redirection error: packet is redirected to a n/w, ∵ host doesn't belong here
  - Source Quench: packet discarded due to congestion in n/w.
  - Parameter prob: Packet discarded due to error in Header of IP datagram

imp

* TCP → flow, error, congestion control
  UDP → error control.
  Simple Stop & Wait → flow control
  Stop & Wait ARQ → flow, error control

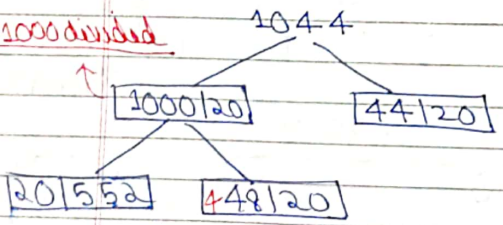
* TPDU = 1044B.



imp

Srcw fragmentation: how many Bytes delivered to IP layer?

1000 divided

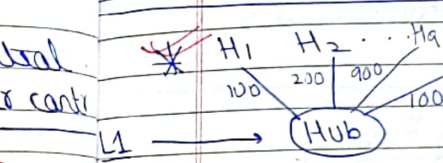


∴ Total = (20+48+44) + 60 = 1104.
552

* DMA bandwidth = 2Gbps
  I/O bus BW = 1Gbps
  workstation limited by I/O bus
  effective BW = 1000 Mbps
  ⇒ 500 Mbps²

Q Bus Bandwidth becomes the limiting factor?

* Packets or ACKs can get reordered in the network. (possibility)
  [win if no packet lost/ACK lost]

* 

CHq → H10 [1900m]

Mistake [Concept]

Packet length = RTT * BW.

⇒ Max end to end PD = (1000+900)/(2×10⁸) = 9.5μs

⇒ Hub delay = 2.5μs ∴ 1 2 RTT = 24

eg → Q20 [Vimp Q]

* Max. efficiency obtained from Pure Aloha = 1/2 * e⁻¹
  Max. efficiency obtained from Slotted Aloha = 1 * e⁻¹

imp

Difficult Qs

* Host A —(Rbps)— R —(Rbps)— Host B

- No. of Packets = F/S.

Packet 1 will reach destination in:

R — 1s.

40+S — x ∴ x = (40+S)/R. TP

∴ ((40+S)/R) x 2 sec.

Next Packets will reach in intervals of (40+S)/R [Back to back]

Now diff. to get minima of S.

eg → 28Q

* Max line Speed of Router.

IP Datagram ID number Space = 65535

Packet lifetime = 10s.

∴ 6553 packets/sec.

eg → Q31

* MSS = 1KB. imp [Slow start]

At the end of i^{th} RTT, it will send 2^i KB

2^i KB = 10 MB

$2^i \cdot 2^{10}$ B = $10 \cdot 2^{20}$ B

$\frac{2^{i+10}}{2^{20}} = 10 \Rightarrow 2^{i-10} = 10$.

∴ i = 14 RTTs

eg → Q29

imp

* RSA: Public Key Cryptography
Diffie-Hellman: Session Key
Digital Signature: Public & Private Keys for Encryption & decryption.
Symmetric Key Cryptog: $^nC_2$ Keys for n People

Token Bucket

* capacity of Bucket: c Tokens
Token rate: r token/s
Max no. of packets that can enter in n/w, in t sec?

Soln:

[c+rt] Packets.

max o/p rate: if c+rt — t
x — 1

M = c+rt/t packets/s

∴ M = (c+rt)/t imp

Host → N/W