CN & Security.

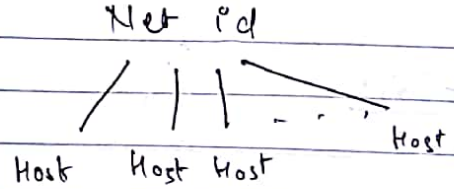
Physical Address
(OR)
NIC address
(or)
MAC address
(or)
Ethernet address
or
LAN card address

48 bit address
Implicit address.

→ Using MAC address alone can not be used in transmitting the data, because every company transmitting the data, because every company has its own representation.

Unicasting → one to one
Multicasting → one to many
Broadcasting → one to all

→ For IP, there are 2 notations:
(i) Dotted decimal notation → 71.55.69.87
(ii) Binary notation → 01001



→ Entire network will be represented by a no. known as net id.

→ In binary notation, first few bits will decide the type of class.
→ In dotted decimal notation, first octet will decide the type of class.

*Class A:
(Unicasting) 0 (2^7-2) | 2^{24}-2
8 bits (Net id) | 24 bits (Host id)
0000 → DHCP client
127.x.y.z → loopback address.

0.0000000 = 0
⋮
01111111 = 127
(0-127) → (1-126)

→ In class A, there are (2^7-2) networks.
→ In which, each network will have (2^{24}-2) hosts.

*Class B:
(Unicasting) 1 0 (2^{14}) | (2^{16}-2)
16 bits (Net id) | 16 bits (Host id)

10 000000 = 128
⋮
10111111 = 191
(128-191)

*Class C:
(Unicasting) 1 1 0 (2^{21}) | (2^8-2)
24 bits (Net_id) | 8 bits (Hostid)

110 00000 = 192
⋮
110 11111 = 223
(192-223)

*Class D:
1110

1110 0000 = 224
⋮
1110 1111 = 239
(224-239)

*Class E
1111

1111 0000 = 240
⋮
1111 1111 = 255
(240-255)

① IP_1 = 201.55.73.89
Net Id : 201.55.73.0
Direct Broadcast address of network: 201.55.73.255.

*Network Mask (or) Default Mask:
Class A → 11111111 00000000 00000000 00000000
255.0.0.0
Class B → 255.255.0.0
Class C → 255.255.255.0

→ Network mask is a mathematical tool, which is used for solving the networking problems.

IP = 201.55.73.89 (Class C)
Mask = 255.255.255.~~255~~0 (Bitwise AND)
Net Id = 201.55.73.0

201 : 11001001
255 : 11111111
201 : 11001001

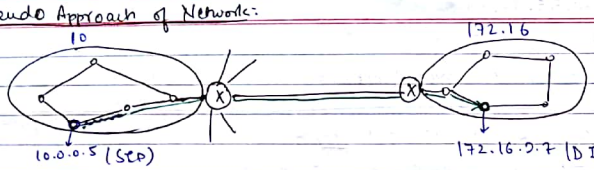
201.55.73.255 ⇒ DBA (Direct Broadcast Address)
(Used to send broadcast to all hosts.

Net_id
201.55.73.0
201.55.73.1 | .2 | ... | .254

→ For a network id, host bits are all 0s.
→ For DBA of the network, host bits are all 1s.

→ We subtract 2 addresses in the number of hosts because the is used for net-id and the other one is used for DBA of n/w.

Pseudo Approach of Network:



	SIP	DIP	
①	D | 10.0.0.5	172.16.2.7	→ It is a unicast packet between n/ws.
②	D | 10.0.0.5	172.16.255.255 (DBA)	→ It is a broadcast packet

→ DBA will always be used as destination IP address.

E.g.

	SIP	DIP	
	D | 172.16.255.255 ✗	10.0.0.5	⇒ will not exist

③	D | 10.0.0.5	10.0.0.9	→ It is a unicast packet within the n/w.

Special Address:

Class E

↳ 255.255.255.255

↳ Limited Broadcast Address

↳ Scope is local.

	SIP	DIP
D	10.0.0.5	255.255.255.255

→ Limited Broadcast Address will always be used as destination address.

IP Address

Private IP address	Public IP address
① Scope is local	① Globally unique
② Works only in LAN	② Controlled by ISP
③ free of cost (Loading n/w OS)	③ Internet Service
④ Not get internet service	④ Not free of cost.

⑤ Range: 10.0.0.0 – 10.255.255.255
172.16.0.0 – 172.31.255.255
192.168.0.0 – 192.168.255.255

Assigning Private IP address in LAN:

(10.0.0.1 – 10.0.0.240)
↳ group of private address.

	SIP	DIP
D	0.0.0.0	10.0.0.7

+

MAC Address
16:1A:12:14:16:18

D	10.0.0.7	255.255.255.255

Client → DOS, XP
Server → Windows NT, 2003
(DOS + Networking protocols)

(Stateful)

IP Address	MAC Address
10.0.0.15	16:1A:12:14:16:18

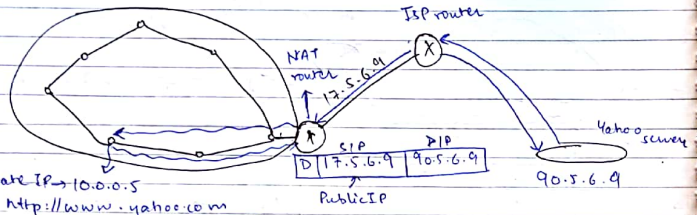
→ Once the server is loaded with n/w O.S., it will get group of private IP addresses. Out of which, one IP is assigned to the server.

→ The server's IP is informed to all clients, using limited broadcast address.

NOTE → Every client, when it is not having an IP address, still it wants to transmit the data, then it uses "0.0.0.0" as the source address.

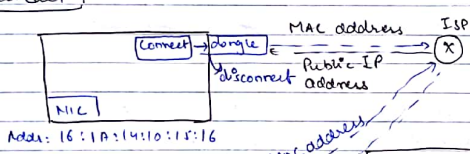
→ Whenever the client is requesting, alongwith it MAC address is transmitted, so that the server can understand which computer is requesting for the IP address.

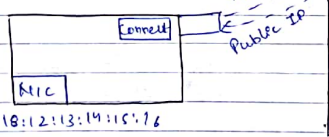
Public IP Address : Internet Service



→ NAT router converts private IP into public IP, when the packet is going out of the network.
→ It converts public IP into private IP, when the packet is coming inside the network.

Practical-1 :





MAC	Public IP	Enable
16:1A:14:10:15:16	6.5.6.1	✓ ✗
	free	
18:12:13:14:15:16	6.5.6.1	✓
16:1A:14:10:15:16	8.6.7.1	✓

→ A computer can have multiple IP addresses at different instances of time. (Dynamic IP)

① Which of the following IP can be used as source as well as dest. IP

(a) 10.255.255.255 — (c) 10.5.6.1
(b) 172.16.255.255 — (d) None

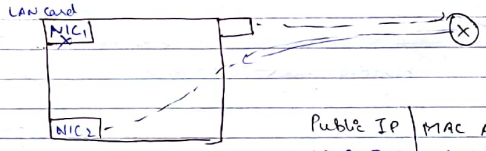
② Which of the following IP can be advertised to Internet?

(a) 10.0.0.5 — (c) 121.9.6.3
(b) 172.16.0.3 — (d) None

12/09/2017

Practical-2 :

16:14:1A:12:13:16



Public IP	MAC Address	Enable
16.9.5.7	16:14:1A:12:13:16	✓ ✗
16.9.5.7	20:1A:1B:1C:1D:1D	✓

→ A computer can have multiple MAC addresses to support fault tolerance.

Q. $IP_1 = 201.55.66.73$, Calculate hosts on this network:

Hosts = 0.0.0.73

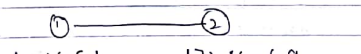
$IP_1 = 201.55.66.73$

Def. Mask = 255.255.255.0 → complement → Bitwise AND

Wild card Mask = 0.0.0.255

Hosts on the n/w = 0.0.0.73

Eg.



172.16.5.1 — 172.16.6.9

D	172.16.5.1	172.16.6.9

✓

D	10.0.5.1	0.0.6.9

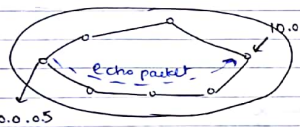
✓ Hosts on this W/w

Host on this n/w: When you transfer data within the network, you can make network bits as "0".

e.g. 172.16.5.1 — 10.5.6.7

	SIP	DIP	
D	172.16.5.1	10.5.6.7	✓
D	0.0.5.1	0.5.6.7	✗

Ping: Packet Internet Gropher:



Start
↳ Run cmd C:\> ping -t 10.0.0.7 ↵
positive ack ← TTL: 2ms, RTT: 4ms
(or)
C:\> ping -t 10.0.0.7
negative ack ← Destination unreachable

* Self troubleshooting:

C:\> ping -t 127.0.0.1 ↵ → loopback address
→ can be anything
TTL = 2ms, RTT = 4ms

10.0.0.5

→ Loopback address packet will never enter into the network.

D	10.0.0.5	127.0.0.1	✓
	SIP	DIP	

→ Loopback address will always be used as DIP.

D	0.0.0.0	127.0.0.1

ARP Request Packet → Request Packet (Broadcast)

SMAC	DMAC	SIP	DIP
16:10:12:1A:1F:12	↓ ?	10.0.0.5	10.0.0.7

FF:FF:FF:FF:FF:FF (Broadcast MAC)

Response: MAC address of the destination computer.

ARP → Address Resolution Protocol

→ ARP query packet is a broadcast packet.
→ ARP reply is a unicast packet.
→ The purpose of ARP is, if IP address is given, it is able to generate the MAC address.

Drawbacks of Classful Addressing:

Class A ($2^7 - 2$) networks ≈ 8000000
↳ each network → ($2^{24} - 2$) hosts ×
10000 wasted

Class B (2^{16}) networks → 65534
↳ each network → ($2^{16} - 2$) hosts ×

Class C 2^{21} networks
1000 → 254
↳ each net → ($2^8 - 2$) hosts ×

Subnetting:

→ Dividing a network in small parts for effective utilization of IP addresses.

Eg ① In class C, subnet mask is 255.255.255.224

No. of subnets
No. of hosts in each subnet N/w bits Subnet bits Host bits
Subnet mask = 11111111 11111111 11111111 111 00000
Default mask = 11111111 11111111 11111111 00000000 (255.255.255.0)

No. of subnets = $2^3 - 2 = 6$
No. of hosts = $2^5 - 2 = 30$

→ During subnetting, subnet bits are borrowed from host portion.

Q. In class C, Subnet mask = 255.255.255.240
= 255.255.255.1111 0000
S H

No. of subnet = $2^4 - 2 = 14$
No. of hosts = $2^4 - 2 = 14$

Q. In class B, Subnet mask = 255.255.254.0
255.255.1111111 0.00000000
S H

No. of subnets = $2^7 - 2 = 126$
No. of hosts = $2^9 - 2$ = ~~512~~ 510 (in each subnet)

④ IP = 201.55.76.89 (Class C) — 89 = 01011001
Subnet mask = 255.255.255.224 — 224 = 11100000
Subnet id = 201.55.76.64 — 64 = 010 00000
Subnet no. = 2nd subnet. (2)

⑤ IP = 202.89.99.113 — 113 = 01110001
Subnet mask = 255.255.255.240 — 240 = 11110000
Subnet id = 202.89.99.112 — 112 = 0111 0000 → 7th subnet
Subnet no. = 7th subnet — No. of subnets = $2^4 - 2 = 14$

→ for a subnet id, host bits will always be zeroes (0s).

⑥ IP = 200.99.89.121 — 121 = 01111001
Subnet mask = 255.255.255.224 — 224 = 11100000
(i) Subnet id = 200.99.89.96 — 96 = 011 00000 (3rd subnet)
(ii) first host of that subnet = 011 00001 = 97
= 200.99.89.97
(iii) Last host of subnet = 011 11110 = 126
= 200.99.89.126
(iv) DBA of that subnet = 011 11111 = 127
= 200.99.89.127

** We are subtracting 2 addresses in the no. of hosts in each subnet because 1 is used for subnet-id and the other one is used for DBA of the subnet.

⑦ IP = 201.55.79.99
Sub mask = 255.255.255.224 — 224 = 111 00000
S H
(i) 4th subnet id = 100 00000 = 128
= 201.55.79.128
(ii) 3rd subnet id = 011 00000 = 96
= 201.55.79.96

⑧ (Network bits (Class C) | Subnet bits | Mask = 111 00000)
IP1 = 201.99.88.93 — ⇒ 93 = 01011101
IP2 = 201.99.88.103 — ⇒ 103 = 01100111
IP3 = 201.99.88.113 — ⇒ 113 = 01110001
Sub-mask = 255.255.255.224
Identify the IPs belonging to same subnet: IP2 & IP3

⑨ IP = 202.55.99.87
Sub-mask = 255.255.255.224 → 111 00000 (S H)
(i) Net id = 202.55.99.0 → 00000000
(ii) First subnet id = 001 00000
= 202.55.99.32
(iii) DBA of network = 202.55.99.255 → 11111111
(iv) Last subnet id = 110 00000
S H

000 ✗
001 ✓
010 ✓
011 ✓
100 ✓
101 ✓
110 ✓
111 ✗

NOTE.
** We are subtracting 2 addresses in the no. of subnets because one is used for net id.
& other one is used for DBA of the network.

** If we use, 111 00000 as subnet then, DBA of that subnet = 111 11111, which is same as DBA of N/w

(10) IP = 204.99.89.119

Sub-mask = 255.255.255.240 → 1111 0000 (S H)

(i) First host of 2nd subnet = 0010 0001 = 33
= 204.99.89.33

(ii) 3rd host of 4th subnet = 0100 0011 = 67
= 204.99.89.67

(iii) 2nd host of 3rd subnet = 0011 0010 = 50
= 204.99.89.50

(11) IP_1 = 204.89.99.124 (N/w bits, 4 bits) ⇒ 124 = 0111 1100 (3rd subnet, 28th host)
IP_2 = 204.89.99.118 ⇒ 118 = 0111 0110 (3rd subnet, 22nd host)
IP_3 = 204.89.99.79 ⇒ 79 = 0100 1111 (2nd subnet, 15th host)
Sub-mask = 255.255.255.224 = 1110 0000
Identify which host of which subnet are they?

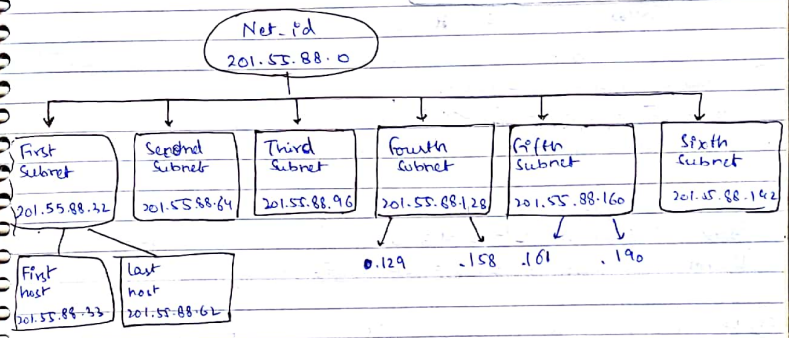
(12) Which of the following will be the last host of subnet?
(a) 201.99.89.31 ⇒ (0001 1111) DBA ~~(b)~~ 201.99.89.14 (0000 1110) Last host
(b) 201.99.89.63 ⇒ (0011 1111) DBA (d) None.

**(13) DBA of subnet is 201.15.33.31 (N/w → 0001 1111). Which of the following can be subnet mask?
(a) 255.255.255.192 = 1100 0000 (S H) → DBA = ~~00~~ 0011 1111 ✗
(b) 255.255.255.128 = 1000 0000 → DBA = ~~0~~ 111 1111 ✗
(c) 255.255.255.240 = 1111 0000 → DBA = ~~000~~ 0001 1111 ✓
(d) None

**(14) DBA of subnet is 199.16.89.63 → 0011 1111
Which of the following will be subnet mask?
(a) 255.255.255.240 = 1111 0000 (0011 1111)
(b) 255.255.255.224 = 1110 0000 (001 1111)
(c) 255.255.255.248 = 1111 1000 (0011 1111)
(d) ~~Above~~ Any of above.

(15) DBA of subnet is 198.17.89.31 → 0001 1111
Which of the following can be subnet mask.
(a) 255.255.255.224 ⇒ 1110 0000 (S H) 000 1111 1 ⇒ Subnet bits cannot be all 0s
(b) 255.255.255.240 ⇒ 1111 0000 000 1111 1
(c) 255.255.255.248 ⇒ 1111 1000 0001 1111 1
(d) both b & c
(e) All

(16) IP = 201.55.88.95 → 0101 1111 (S H) → DBA of 2nd subnet
Sub-mask = 255.255.255.224 → 1110 0000 → continuous mask
(i) Net id = 201.55.88.0
(ii) Second subnet id = 010 00000 = 201.55.88.64
(iii) First subnet id = 001 00000 = 201.55.88.32
(iv) First host of first subnet = 001 00001 = 201.55.88.33
(v) Last host of first subnet = 001 11110 = 201.55.88.62
(vi) DBA of first subnet = 001 11111 = 201.55.88.63



⇒ If we take a continuous mask, designing of a network will become simple and easy.

Discontinuous Subnet Mask:

(17) IP = 205.66.88.93
Subnet mask = 255.255.255.41 (Discontinuous Mask)
= 11111111 11111111 11111111 00101001
Host bits. Subnet bits
1's in subnet mask are subnet bits.

(i) First subnet id = 00000001 = 205.66.88.1
(ii) Second subnet id = 00001000 = 205.66.88.8
(iii) Third subnet id = 00001001 = 205.66.88.9

→ Discontinuous mask cannot be applied in networking.
→ But can be applied in security.

(18) Company requires 28 hosts. What is best suitable subnet mask?
No. of hosts = $2^5 - 2 = 30$ (can have 28 hosts) N + S + H
Subnet mask = 255.255.255.224 24 + 3 + 5
= 11111111 11111111 11111111 111 00000
N/w S H

(19) Company requires 62 hosts.
No. of hosts = $2^6 - 2 = 62$ ✓
Subnet mask = 11000000 = 255.255.255.196

(20) Company requires 500 hosts. (Class B) N + S + H
No. of hosts = $2^9 - 2 = 510$ 16 + 7 + 9
Subnet mask = 11111110 00000000 = 255.255.254.0

(21) Can network mask of class "C" be the subnet mask of class "B"?
True. 255.255.255.0 will look as subnet mask in class B.

(22) Can the network mask of class A (255.0.0.0) be the subnet mask of class B?
False F

(23) In the 20th problem, if the restriction is we have to use class C network. Then can we apply subnetting?
We can not use subnetting.
* Supernetting.

Supernetting: Joining two or more networks to form a larger network ~~is known as supernetting~~ according to the requirement of user.

(1) In class C, if supernet mask is 255.255.252.0
No. of networks that can be joined = $2^2 = 4$ → No. of supernet bits
N/w bits Host bits.
11111111 11111111 111111 00 00000000
Supernet bits

⇒ During supernetting, we can join power of 2 networks (2^n) only.

(2) In class C, if supernet mask is 255.255.240.0
No. of networks, that can be joined = $2^4 = 16$
240 = 1111 0000 ⇒ supernet bits = 4

(3) One of the address of supernet is 201.55.73.89. 73 = 01001001
Supernet mask = 255.255.252.0
Range of supernet = 201.55.72.0 to 201.55.75.255
Supernet id = 201.55.72.0
252 = 11111100 ← supernet bits
72 = 01001000

72.0 ⇒ 01001000 00000000
72.255 ⇒ 01001000 11111111
73.0 ⇒ 01001001 00000000
73.255 ⇒ 01001001 11111111
74.0 ⇒ 01001010 00000000
74.255 ⇒ 01001010 11111111
75.0 ⇒ 01001011 00000000
75.255 ⇒ 01001011 11111111

13/09/2017

Classless Addressing:

↳No Class
↳Block
↳(Group of IP address)

=> n.y.z.w/n ⇒ CIDR notation (Classless Inter Domain Routing)
(or)
Slash notation

201.55.77.89/26
↳mask = 11111111 11111111 11111111 11 000000
255. 255. 255. 192 ⇒ Mask

/30 = 255.255.255.252
/25 = 255.255.255.128
/27 = 255.255.255.224

① One of address of block is 201.55.77.89/26
(i) No. of addresses in a block = $2^{32-n} = 2^{32-26}$
$= 2^6 = 64$ IP addresses

89 ⇒ 01011001
64 ⇐ 01 000000
127 ⇐ 01 111111
(i) Range of block = 201.55.77.64/26 to 201.55.77.127/26
(ii) Net Id = 201.55.77.64/26
(iii) First host = 201.55.77.65/26
(iv) Last host = 201.55.77.126/26
(v) DBA = 201.55.77.127/26

Rules of Classless Addressing:

① Addresses in a block are continuous.
② The first address of a block should be exactly divisible by no. of addresses of a block.

② One address of block = 201.53.43.129/27
Range of block = 201.53.43.128/27 to 201.53.43.159/27
10000001 = 129 No. of addresses in a block = 2^{32-27} = 32 IP addresses.
100 00000 = 128.
100 11111 = 159

③ One of the address of block is 63.59.89.99/22.
No. of addresses in a block = 2^{32-22} = 1024 IP addresses. = 4 × 256
01011001 01100011
11111100 00 000000
01011000 00 000000 ⇒ 63.59.88.0/22 to
01011011 11 111111 ⇒ 63.59.91.255/22
Range ⇒ 63.59.88.0/22 to 63.59.91.255/22

4) 88 (22

④ One of the address is 35.73.89.86/20.
No. of addresses = $2^{12} = 2^4 * 2^8 = 16 * 256$
01011001 01010110
11110000 00000000
0101 0000 00000000 = 35.73.80.0/20
to 0101 1111 11111111 = 35.73.95.255/20

16) 80 (5

Range = 35.73.80.0/20 to 35.73.95.255/20

⑤ Block contains 32 IP addresses which of the following will be first address of block?
(a) 201.16.16.16
(b) 201.15.15.8
(c) 201.14.14.160 32)160(5
(d) None

⑥ One address of a block is 63.89.69.79/26. If this block is divided into 4 equal sub blocks, then, calculate the range of the block and range of ~~sto~~ sub blocks.

No. of IP addresses = $2^{32-26} = 2^6 = 64$

79 → 01001111

64 → 01000000

64 = 01000000 = 63.89.69.64/26 to 63.89.69.127/26

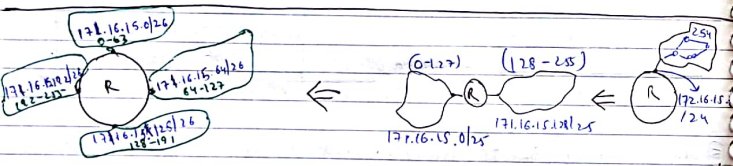
127 → 01111111

Subblocks: No. of addresses in each subblock = $\frac{64}{4} = 16 = 2^4$

$= 2^{32-28}$

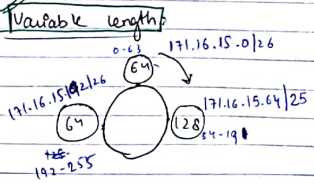
Subnet mask = 28.

I^st →	63.89.69.64/28 to 63.89.69.79/28	64 → 0100 0000 ; 79 → 0100 1111
II^nd →	63.89.69.80/28 to 63.89.69.95/28	80 → 0101 0000 ; 95 → 0101 1111
III^rd →	63.89.69.96/28 to 63.89.69.111/28	96 → 0110 0000 ; 111 → 0110 1111
IV^th →	63.89.69.112/28 to 63.89.69.127/28	112 → 0111 0000 ; 127 → 0111 1111



Equal length subnetting.

Variable length



① An ISP has a block 131.16.0.0/16.

I Group has 128 customers, each cust. requires 256 IP addresses.

II Group has 64 customers, each cust. requires 128 IP address.

III " " 16 " " " " 64 " "

I Group: 1st cust → 131.16.0.0/24 to 131.16.0.255/24 ($256 = 2^8 = 2^{32-24}$)

128th cust → 131.16.127.0/24 to 131.16.127.255/24

II Group: $128 = 2^7 = 2^{32-25}$

1st cust → 131.16.128.0/25 to 131.16.128.127/25 (10000000)

2nd cust → 131.16.128.128/25 to 131.16.128.255/25

64th cust → 131.16.159.128/25 to 131.16.159.255/25

III Group: $2^{64} = 2^6 = 2^{32-26}$

1st cust → 131.16.160.0/26 to 131.16.160.63/26

2nd cust → 131.16.160.64/26 to 131.16.160.127/26

16th cust → 131.16.163.192/26 to 131.16.163.255/26

Calculate the IP addresses that are leftover with the ISP after assigning to these groups and the answer should be scaling factor of power of 2. $(a * 2^b)$

$(164-255) * 256$ → $(256-164) * 256$

$= 92 * 256 * 2^8$ → $23 * 2^{10}$

(or)

$= 2^{16} - [128 * 256 + 64 * 128 + 16 * 64] = 2^{16} - [2^7 * 2^8 + 2^6 * 2^7 + 2^4 * 2^6]$

$= 2^{16} - 2^6(2^2 + 2 + 2)$ → $2^{16} - (2^{15} + 2^{13} + 2^{10}) = 2^{16} - 2^{10}(2^5 + 2^3 + 1)$

$= 2^{10}(2^6 - 41) = 2^{10} * 23$

* Special Case : Classful (Subnetting)

① IP_1 = 201.55.78.89

Subnet mask = 255.255.255.128 = 11111111 11111111 11111111 10000000 (N | H)

[explicitly configured zero subnet, DBA subnet (or) net work wishes to form subnets.] → No network

(i) Zero subnet id = 0 0000000 = 0
= 201.55.78.0

(ii) First host of zero subnet = 0 0000001 = 201.55.78.1

(iii) Last host of zero subnet = 0 1111110 = 201.55.78.126

(iv) D.B.A. of zero subnet = 0 1111111 = 201.55.78.127

(v) D.B.A. (name) subnet id = 1 000000 = 201.55.78.128

(vi) First host of DBA subnet = 1 000000 1 = 201.55.78.129

(vii) Last host of DBA subnet = 1 111111 0 = 201.55.78.254

(viii) D.B.A. of DBA subnet = 1 111111 1 = 201.55.78.255

② IP_1 = 201.55.44.99 ; Nw is divided (split) into

Subnet mask = 255.255.255.224 = 11111111 11111111 11111111 111 00000 (y) subnets.

No. of subnets = $2^n - 2 = 2^3 - 2$
= 6 subnets.

③ Network wishes to form subnets. → No network → Only subnets.

No. of subnets = 2^3 = 8 subnets.

IPv6 : (version 6)

→ 128 bits (Hexadecimal notation)

$\frac{128}{16}$ = 8 fields.

IPv6 ⇒ FE80 : 1234 : 1A12 : 0 : 0 : 0 : 71A0 : 1A1F ✓

⇓

FE80 : 1234 : 1A12 :: 71A0 : 1A1F ✓

IPv6 ⇒ 2001 : 0 : 0 : 0 : FF80 : 0 : 0 : 1A12

2001 :: FF80 :: 1A12 ✗

⇒ 2001 :: FF80 : 0 : 0 : 1A12 ✓

(or) ⇒ 2001 : 0 : 0 : 0FF80 :: 1A12 ✓

Rules for compression in IPv6 :

→ Whenever zeroes are present in IPv6 address in a continuous manner, then, they can be replaced by double colon (::).

→ Whenever zeroes are present in discontinuous places in IPv6 address, then at only one junction, zeroes are replaced with double colon (::).

(1) :: ⇒ 0:0:0:0:0:0:0:0 (Unspecified address) (default address) (IPv4 0.0.0.0)

(2) ::1 ⇒ 0:0:0:0:0:0:0:1 (Loopback address) (IPv4 127.x.y.z)

⇒ In IPv4, entire 127 series (127.x.y.z) is wasted for loopback testing, whereas, in IPv6, only one address is used for loopback testing.

* IPv6 128 bit address = Network Prefix 64 bits + Extended MAC (Interface Id) 64 bits

MAC address ⇒ 16 : 1A : 12 : 1F : 1A : 12

0001 0110 → unicast MAC address.

19 : 12 : 1E : 1F : 1A : 12

0001 1001 → Multicast MAC address.

FF : FF : FF : FF : FF : FF

↳ Broadcast MAC address.

#Extended MAC address:

48 bit address ⇒ 16:1A:12 : 1F:1A:12

⊕ FFFE

16:1A:12:FF:FE:1F:1A:12

1G1A:12FF:FE1F:1A12

64 bit extended MAC address.

#Global Unicast Address: (Ela Public IP in IPv4)

(3) 2000::/3 ↳ First 3 bits 001

0010 0000 0000 0000 : 16 0's : 16 0's : 16 0's : Extended MAC (64 bits)

↳ First address

0011 1111 1111 1111 : 16 1's : 16 1's : 16 1's

3FFF:FFFF:FFFF:FFFF ⇒ Last address.

*Range:

2000::/64 to 3FFF:FFFF:FFFF:FFFF (64) upto 64 bits

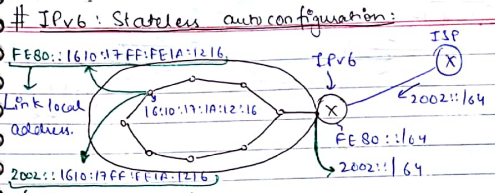
#Link Local Address: (Ela private IP in IPv4)

(4) FE80::/64

1111 1110 1000 0000

Starting 10 bits.

IPv6: Stateless autoconfiguration:



(i) Neighbor Solicitation
↳ neighbor request
↳ neighbor reply

(ii) Router solicitation
↳ Router request
↳ Router reply

ICMP Messages

→ IPv6 doesn't depend on ARP because MAC address itself is a part of IPv6 address.
→ By default, IPv6 is a stateless protocol, whereas IPv4 is a stateful protocol.
→ For mobile networks or wireless networks IPv6 is best suitable because of its stateless autoconfiguration.
→ In IPv6, by default it doesn't require DHCP server.
→ In IPv6, a computer can have multiple IP addresses at the same time.
→ By default, IPv6 doesn't require NAT (Network Addr. Translation) operations.

14/09/2017:

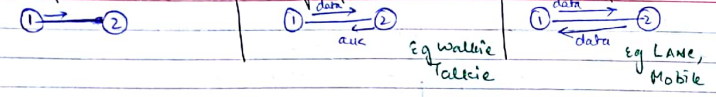
→ IPv6 doesn't support broadcasting, It supports only multicasting.

#Anycasting:

→ In anycast address, all nodes will have the same address, but the service will be provided by the nearest node.

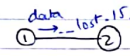
#Basic Concepts of Networking

① Simplex Transmission | ② Half Duplex Transmission | ③ Full Duplex Transmission

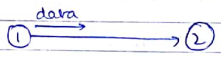


eg Walkie Talkie | eg LAN, Mobile

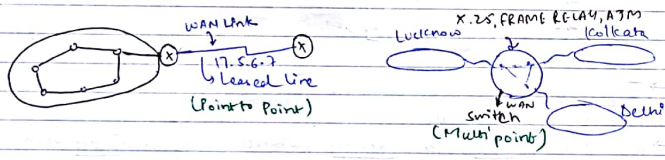
→ Timer =20s is waiting a finite amount of time

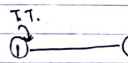


→ expires → resend Buffer [11010] ↳ data



→





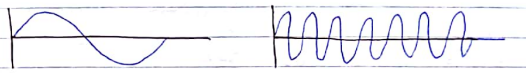
Transmission Time = Data size / Bandwidth

① Data size = 2 lc bits
B.W. = 10Mbps
T.T. = $\frac{2*10^3 \text{ bits}}{10^7 \text{ bits/s}}$ = $2*10^{-4}$ s
T.T. = 200 usec

kilo = 10^3	mi Mi = 10^{-3}
Mega = 10^6	micro = 10^{-6}
Giga = 10^9	nano = 10^{-9}

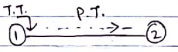
kilo = 10^3 ↳ data transfer
kilo = 2^{10} ↳ data storage

BW → bits/sec | BW → low freq. sig. | BW = 20MHz
cycles/sec | 2 bits (or) 2 cycles/sec
Hertz



*Transmission Time : The time taken to place the data on the network is known as transmission time.

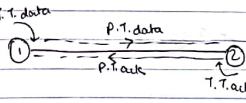
→ If Bandwidth is high, then signal will travel longer distances in short amount of time.
→ Low bandwidth signals will travel short distances.



Propagation time = Length of cable / velocity of medium

② l = 2.3 km
v = $2.3*10^8$ m/s
P.T. = $\frac{l}{v}$ = $\frac{2.3*10^3}{2.3*10^8}$ = 10^{-5} s

⇒



T.T. ack = ack size / Bandwidth

* T.T. ack is negligible — Ack size <<<< Data size
* P.T. data = P.T. ack

Total time = $TT_{data} + PT_{data} + TT_{ack} + PT_{ack}$
= $TT_{data} + 2PT$

* **Total time = TT + 2PT** *

* % link utilization of sender = $\frac{T.T.}{T.T. + 2*P.T.} \times 100$

(3) % Link utilization = 50%

$50 = \frac{TT}{TT+2PT} * 100 \Rightarrow TT + 2PT = 2TT$

* $TT = 2PT$

(4) % Link uti. = 50% $\Rightarrow TT = 2PT$

$l = 200m$

$v = 2*10^8 m/s$

$BW = 10Mbps$

Data size = ?

$\frac{Data\ size}{BW} = 2 * \frac{l}{v}$

$Data\ size = \frac{2 * 200 * 10 * 10^6}{2 * 10^8}$

$= 200 * 10^{-1} = 20$ bits

* Throughput = $\frac{Data\ size}{Total\ Time} = \frac{Data\ size}{T.T. + 2PT}$

(5) BW = 10Mbps

$l = 200m$

$v = 2*10^8 m/s$

Frame size = 200bits

$T.T. = \frac{frame\ size}{B.W.} = \frac{200}{10^7} = 20\mu s$

$P.T. = \frac{l}{v} = \frac{200}{2*10^8} = 10^{-6} = 1\mu s$

Throughput $\Rightarrow \frac{200}{20\mu s + 2\mu s} = \frac{200}{22\mu s}$

$\Rightarrow \frac{100}{11} Mbps = 9.09 Mbps$

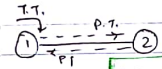
$\Rightarrow$ The rate at which user transmits the data is known as Throughput

Link utilization = $\frac{T.T}{TT + 2PT} \times 100$

$= \frac{Data\ size / B.W}{TT + 2PT} \times 100$

% L.U. = $\frac{Throughput}{Bandwidth} * 100$

L.U. = $\frac{9.09}{10} * 100 = 90.9\ 90.9\%$

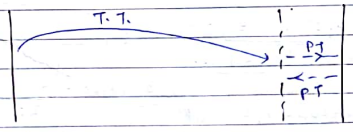


Round Trip Time = 2P.T.

LAN: l is small

$PT = l/w$, P.T. is small

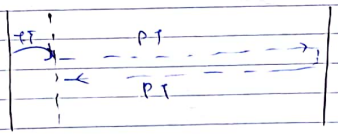
T.T. is larger.



WAN: l is large.

$PT = l/w$, PT is large

PT is larger.



⑥ BW = 10 Mbps, Calculate 1 bit delay?

1 sec = 10^7 bits

1 bit delay = 10^{-7} s

= 0.1 μsec ; TT. = $\frac{1}{10^7} = 10^{-7}$ s.

**

⑦ BW = 10 Mbps; V = $2*10^8$ m/s, Calculate 1-bit delay in metres of cable?

1 sec = 10^7 bits → 1 bit = 10^{-7} sec

1 sec = $2*10^8$ m

10^{-7} sec = $2*10^8*10^{-7}$

= 20 metres.

⑧ In LAN, BW = 10 Mbps; RTT = 50 μsec. Calculate no. of bits that can be transmitted in R.T.T.?

1 sec = 10^7 bits

RTT = 50 μsec = $(50*10^{-6}*10^7)$ bits

= 500 bits

⑨ In LAN, BW = 100 Mbps. RTT = 25 μsec. Data size = 25 bits. Calculate no. of data units in R.T.T..

1 sec = 10^8 bits

RTT = 25 μsec = $25*10^{-6}*10^8$

= 2500 bits.

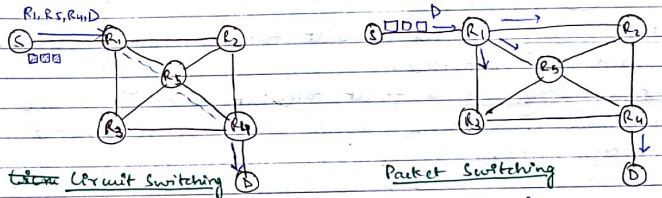
Data units = $\frac{2500}{25}$ = 100 units.

(Total bits / Frame/Data size)

Circuit Switching and Packet switching:

→ In circuit switching, there are 3 phases:
① Connection establishment ② Data Transfer ③ Connection Release.

→ In packet switching, directly data can be transferred.



Circuit Switching

① Connection Establishment
② Data Transfer
③ Connection Release

Packet Switching

① Data Transfer

→ In circuit switching, each packet will have the entire path address.
→ In packet switching, each packet will have only destination address, the intermediate path is decided by routers.
→ Circuit switching is not a store and forward technique because they bypass the queue.
→ Packet switching is a store and forward technique because packets are stored, routing algorithm is applied and forwarded on best path.
→ Resource reservation is a feature of circuit switching, because the resources are dedicated.
→ In packet switching, resources are shared among the users.
→ wastage of resources more in circuit switching.
→ Wastage of resources less in packet switching.
→ In circuit switching, the delay between the data units is uniform.
→ In packet switching, the delay between the data units is variable.

Congestion : If more no. of packets are transmitted in less time, then, the routers buffer will be full. Then, the router is congested.

→ In circuit switching, congestion can occur during connection establishment phase, whereas, in packet switching, congestion can occur during data transfer phase.

→ Circuit switching is not a fault tolerant technique, because the path is connected.

→ Packet switching is a fault tolerant technique, because wherever a link is broken, the data can be diverted via other paths.

→ In circuit switching, all packets will have same speed or same bandwidth.
In packet switching, packets might have different bandwidth or throughput.

→ Circuit switching is preferable for sending long messages, whereas packet switching is preferable for sending short messages.

→ Circuit switching is reliable.

→ Packet switching is unreliable.

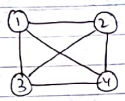
→ Circuit switching is slow.

→ Packet switching is fast.

Topologies (LAN)

- Physical Topologies → Mesh, Star, Bus
- Logical Topologies → IEEE 802.3, IEEE 802.11

(1) Mesh Topology:



4 devices 6 links

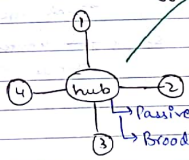
n devices, ⇒ $^nC_2$ ⇒ $\frac{n(n-1)}{2}$ links

College LAN : 100 computers = $\frac{100 \times 99}{2} = 4950$ links → cost is high

→ Maintenance difficulty.

Advantage : Small group, small project.
Highly secure, reliable.

(2) Star Topology:



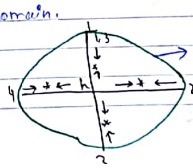
→ one collision domain
→ 4 devices, 4 links
→ n devices, n links required.
→ 100 devices, 100 links
↳ cost is less

↳ Passive (non-intelligent device)
↳ Broadcast device.

→ Hub is a broadcasting device because whenever a packet comes to hub, it is diverted in all directions except the point of origin.

→ Two or more systems' data interfere with each other, then there is a possibility of collision.

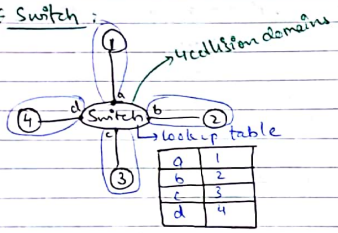
→ The place or area where the collisions are confined is known as the collision domain.

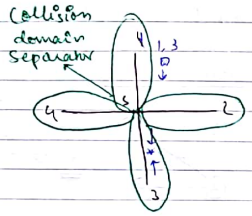


→ Not a collision domain separator.

→ If hub is used as a centric device, then entire network has the same collision domain.

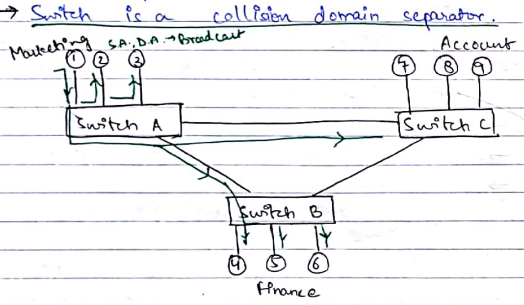
* Switch:





→ If switch is used as a centric device, then, each port has a separate collision domain.

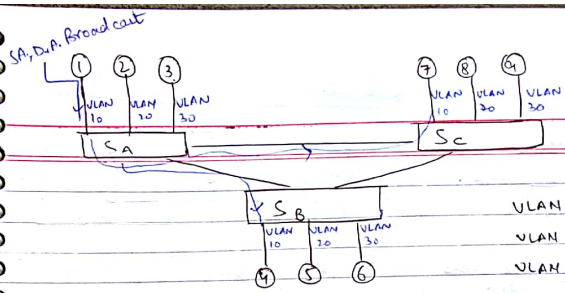
→ Switch is a collision domain separator.



→ By default, switch is not a broadcast domain separator.

* Virtual LAN (VLAN): Physically systems can be placed anywhere, but they are logically connected to their groups only.

LAN ⇒ vLAN
↳ By configuring the ports of switch.



→ If a LAN is converted into vLAN, then switch will act as a broadcast domain separator.

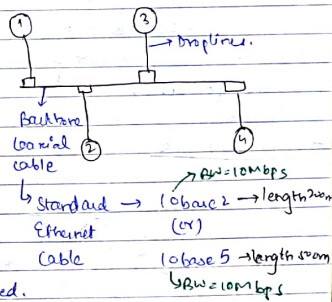
15/09/2017

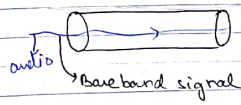
Bus Topology:

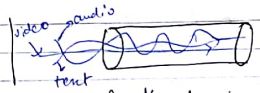
4 devices, 4 droplines + 1 backbone cable
n devices, n droplines + 1 backbone cable
↳ cost is less

→ Baseband Signal:
If only one type of ~~signal~~ data is allowed.

→ Broadband signal:
If more than one type of data is allowed.



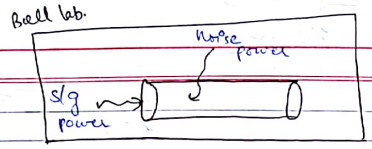




* Signal to noise ratio

$= \log_{10} \frac{\text{signal power}}{\text{noise power}}$ bels.

$= 10 * \log_{10} \left(\frac{S_p}{r_p}\right)$ decibels (deci + bels)



① Signal power = 100 milliwatts
Noise power = 1000 milliwatts

$\left(\frac{S}{N}\right)_{ratio} = 10 \log_{10} \left(\frac{100}{1000}\right)$

$= 10 \log_{10} 10^{-2}$

$= -20$ dB (-ve)

Noise power is dominating signal power.

② Signal power = 1000 milliwatts
Noise power = 10 milliwatts.

$\left(\frac{S}{N}\right)_{ratio} = 10 \log_{10} \left(\frac{1000}{10}\right)$

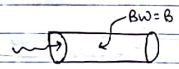
$= 10 \log_{10} 10^{2}$

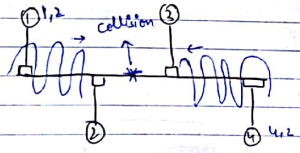
$= 20$ dB (+ve)

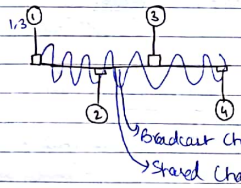
Signal power is dominating noise power.

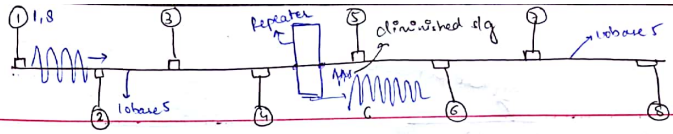
* Max data rate:

Max data rate = $B \log_2 \left(1 + \frac{S}{N}\right)$ bps









→ Repeater is used to increase the length of LAN.
→ Repeater is also known as regenerative device because it regenerates the signal to its original strength.
→ Both hub and repeater are passive devices.
→ Repeater is a 2 port device.
→ Hub is a multiport device.

Q. If lan required is 2000m, length of the cable is 500 metres. Then how many repeaters are required. 3 repeaters.

OSI Model: (7 layers)

	Sender	Receiver	
message ←	Application layer	Application layer	
	Presentation layer	Presentation layer	
TCP / UDP	Session layer	Session layer (all layer switch = Gateway)	
segment, datagram	Transport layer → Port address	Transport layer	(3 layer switch = Router)
packet	← Network layer → IP addr (WAN)	← Network layer	(2 layer switch = Bridge)
frame	← Data Link layer → MAC addr (LAN)	Data link layer	Active devices
bits	← Physical layer	Physical layer	hub, repeater: Passive devices

→ Data Link layer is responsible for node to node delivery within the LAN.
→ Network layer is responsible for source to destination delivery b/w the networks or across the networks.
→ Transport layer is responsible for process to process delivery or end to end delivery which is done by port address.

application header

Application Layer → [AH | Data]
Presentation layer → [PH | AH | Data]
Session layer → [SH | PH | AH | Data]
Transport layer → [TH | SH | PH | AH | Data]
Network layer → [NH | TH | SH | PH | AH | Data] → Packet
Data link layer → [DH | NH | TH | SH | PH | AH | Data] → Frame
Physical layer →

→ Network architecture is known as the protocol stack architecture because the last header that is added at the sender side is the first header that is removed at the receiver's side.

Q. "M" is a message that should be transmitted, which is the header that is added at every layer, "N" layers are present in hierarchy, then, calculate the fraction of the data in the whole content that is transmitted.

Fraction of data = $\frac{M}{NH + M}$

Total header = NH

H + M — [H | M] ← App^n
2H + M — [H | H | M] ← Presentation
. Session
. Transport
. Network
. Data Link
NH + M — Physical

Protocols functions:

Application → Services like http, ftp, SMTP, DNS, telnet, etc.
Presentation → Syntax and semantics of data
Session → dialog control & session management
Transport → flow control, error control, segmentation, congestion policies
Network → routing algo, routing, IP addressing, fragmentation, traffic shaping
Data link → flow control, error control, access control, framing
Physical → physical, electrical characteristics of cable.

→ 7 layers of OSI model have been reduced to 5 layers of TCP/IP model by including the functionalities of presentation and session layer into application layer.

Flow Control:

① Stop & wait ARQ
② Go back N ARQ
③ Selective repeat ARQ

→ flow control policies

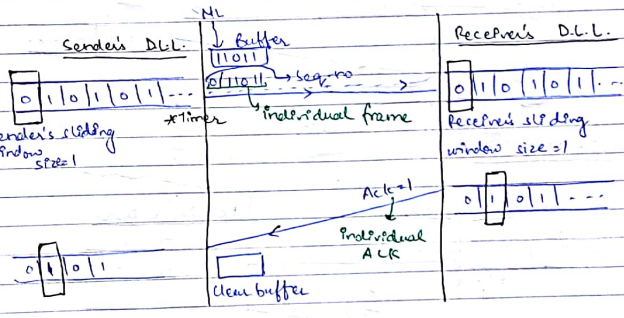
Error Control:

Error control policies
- Error correction policy — E.g. hamming code
- Error detection policy — E.g. parity, checksum, CRC

Data Link Layer:

* Flow control policies of Data link layer:
Stop & wait ARQ : Automatic Repeat Request

Case 1:



Rule of acceptance:

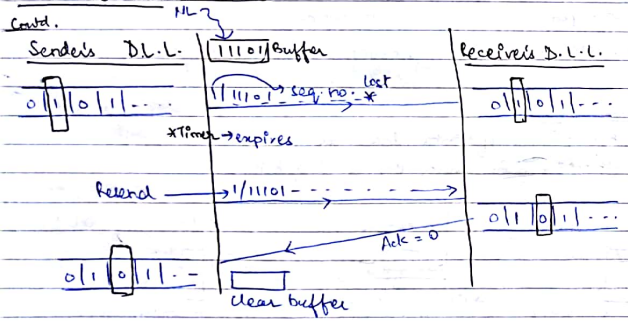
→ Once the data reaches to receiver's data link layer, then, the sequence number of the data is compared with receiver's sliding window number.

→ If there is a match, data will be accepted and receiver window will slide by 1-bit.
→ If there is a mismatch, data will not be accepted.

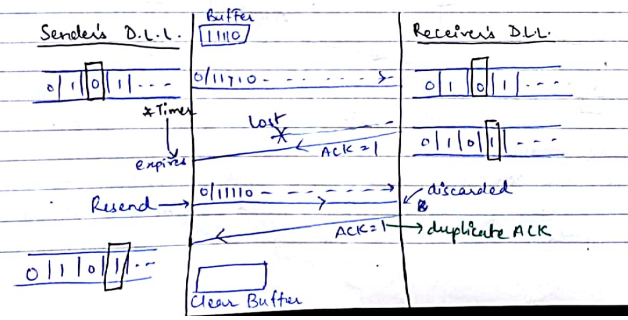
Rule of acknowledgement:
→ Once the acknowledgement reaches to sender, then, the ack no. will always be the sequence no. of the next expected data.
→ Then only, acknowledgement will be accepted and sender window will slide by 1-bit.

Case 2 - Data lost: ~~in trans~~ ML

Contd.



Case 3 - ACK lost:



→ In stop & wait ARQ, it supports only individual frame and individual Acknowledgements.
→ Stop and wait is a theoretical protocol without sliding windows, whereas stop and wait ARQ is a practical protocol with sliding windows.
→ In all sliding window protocols, maximum sender window size indicates number of frames that are transmitted in Round Trip Time (R.T.T.)
→ In all sliding window protocols, the (maximum sender window + maximum receiver window) will always be equal to distinct sequence number count.
→ In stop and wait ARQ, we are transmitting only one frame in Round Trip Time. So, bandwidth utilization is less.

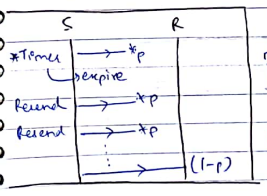
① BW = 10 Mbps; R.T.T. = 50 msec; frame size = 25 bits
↳ 1 sec = 10^7 bits; ↳ 50 msec = $50 * 10^{-6} * 10^7$
= 500 bits

No. of bits in RTT = 500 bits
No. of frames = $\frac{500}{25}$ = 20 frames
→ only 1 frame is transmitted in R.T.T.

% B.W. utilization = $\frac{1}{20} * 100 = 5\%$
(Stop & wait)

***

Q. Probability of frame being lost is "p", then mean no. of transmissions of a frame is?



k transmissions
$\underbrace{p * p * p * p \cdots p}_{(k-1)}\underbrace{(1-p)}_{k^{th}}$ time

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

mean
↳ Expectation(k) = $\sum_{k=1}^{\infty} k * p(k)$ (discrete) (or) $\int_{k=1}^{\infty} k \cdot p(k)$ (analog or continuous)

$E(k) = \sum_{k=1}^{\infty} k * p(k)$

$= \sum_{k=1}^{\infty} k * (p * p * p * p \cdots (k-1)) * (1-p)^1$

$\Rightarrow \sum_{k=1}^{\infty} k * p^{k-1} * (1-p) = (1-p) \sum_{k=1}^{\infty} (k * p^{k-1})$

$\Rightarrow (1-p)\sum_{k=1}^{\infty}\left(k * p^{((k-1))}\right)$

$= (1-p)\left[1 + 2p + 3p^2 + 4p^3 + \cdots\right]$ $\longrightarrow (1-x)^{-1} = 1 + x + x^2 + x^3 \cdots$

$\checkmark (1-x)^{-2} = 1 + 2x + 3x^2 + 4x^3 \cdots$

$= (1-p) * (1-p)^{-2}$

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

Mean no. of transmissions of a frame = $\frac{1}{1-p}$ (p → probability of frame being lost)

E.g. Probability of frame reaching safely is "Q" ; mean no. of transmissions of a frame?

$\Rightarrow \frac{1}{1-[1-Q]} = \frac{1}{Q}$

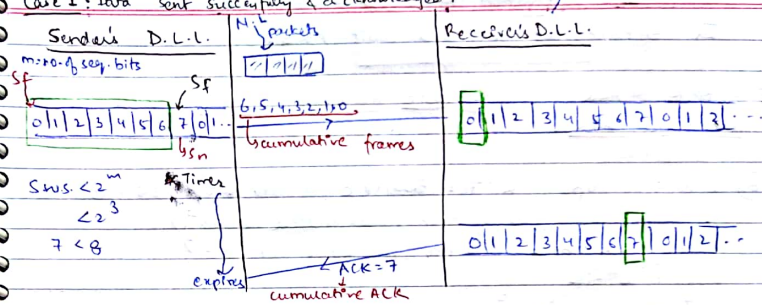
③ Probability of a frame reaching safely = 0.1 (one in 10 transmissions reaches safely)

Mean no. of transmissions of a frame = $\frac{1}{0.1} = 10$

Sequence bits		Sequence No.
1	→	0,1
2	→ 00	0,1,2,3
	01	
	10	
	11	
3	→	0,1,2,3,4,5,6,7

Go back N ARQ :

Case 1 : Data sent successfully & acknowledged.



→ Go back N ARQ supports cumulative frames and cumulative ACKs.

① 5 bit sequence number is used.

	S.W.S	R.W.S.
Go back N ARQ ⇒	31	1

S.W.S. $< 2^m$

$31 < 2^5$

② Max sender window size in Go back N ARQ = 7

No. of sequence bits = $\log_2(1+7)$

= 3 bits

$S.W.S < 2^m$

$S.W.S_{max} = 2^m - 1$

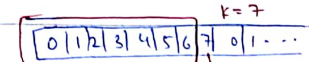
$\Rightarrow 2^m = 1 + SWS_{max} \Rightarrow$ **$m = \log_2(1 + SWS_{max})$**

③ Max sender window size in Go Back N ARQ = 8

No. of sequence bits = $\log_2(1+8)$

④ Max sequence number in Go Back N ARQ is "k". Max size of sender window is:

(a) k-1 (b) k (c) k+1 (d) None



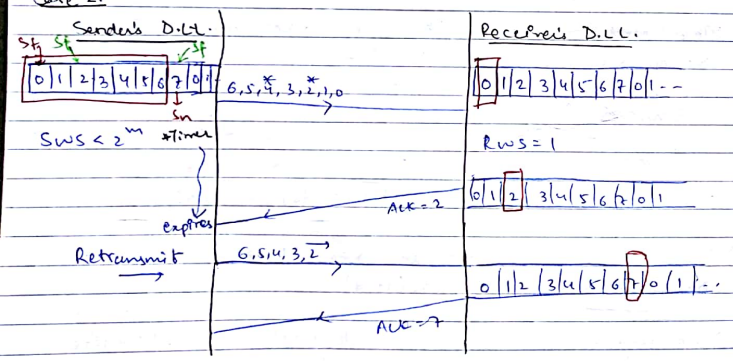
⑤ In Go Back N ARQ in the sender window condition:
S.W.S $< 2^m$; when m = 1.
It behaves as stop and wait ARQ.

⇒ Go Back N ARQ supports both Individual ACKs as well as cumulative ACKs.

⑥ B.W. = 10Mbps ; R.T.T. = 50msec ; Data size = 25 bits.
Window size = ? (No. of frames transmitted in R.T.T.)
No. of sequence bits in Go back N ARQ = ?
1 sec = 10^7 bits
R.T.T. ⇒ $50 \times 10^{-6} * 10^7$
= 500 bits → No. of data units = $\frac{500}{25}$ = 20 in R.T.T.
Window size = 20
No. of sequence bits in Go Back N ARQ = 5 bits

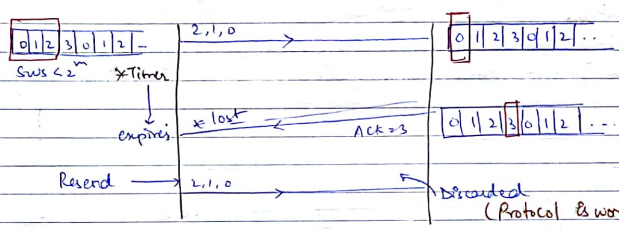
16/09/2017

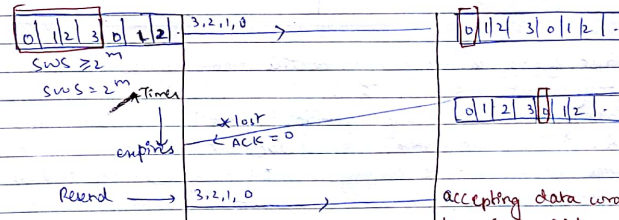
Case 2.



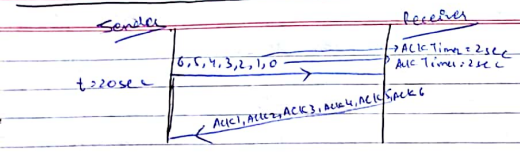
→ Both stop and wait ARQ and Go Back N ARQ will accept only in order frames because receiver's window size is 1.
→ There is a pipelining in Go Back N ARQ, so utilization is high.
→ For noisy channels, there are more no. of retransmissions, so overall utilization will decrease in Go Back N ARQ.
→ In Go Back N ARQ if a frame is lost, then that frame which is lost as well as all following frames should be retransmitted.

m=2



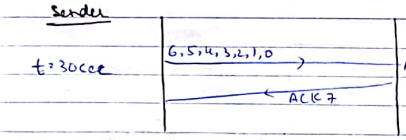


Case 1:
Individual ACKs GoBack N ARQ:



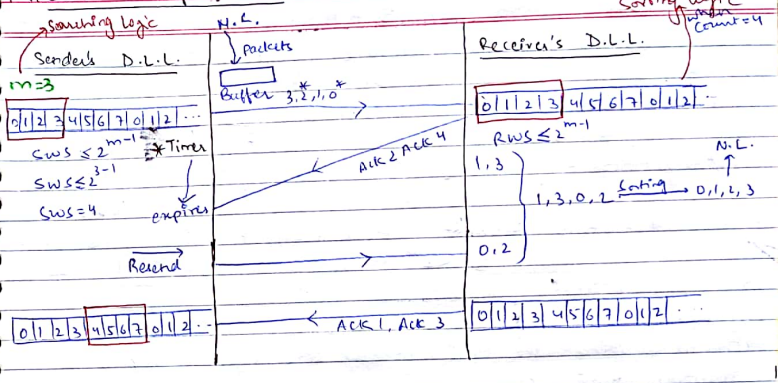
→ Once every frame is received, it will start the ACK timer.
→ This timer value should be small so that it will send ACK immediately.
→ So, during individual acknowledgements, for every frame that is received ACK timer is started.

Case 2: Cumulative ACKs



→ Once, first frame is reached to receiver, then receiver will immediately start ACK timer.
→ This timer should be large enough, so that all frames will be received and the cumulative acknowledgement will be transmitted.
→ This timer should not be so large in such a way the sender's timer expires before it.

Selective Repeat ARQ:



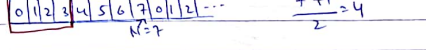
① 5 bit sequence no. is used

$SWS \le 2^{m-1}$

	S.W.S	R.W.S
$2^5 - 1$		
$= 16$	16	16

② Max^m sequence no. in selective repeat ARQ = "N"

Max size of sender window = $\frac{N+1}{2}$



$\frac{7+1}{2} = 4$

③ Maximum size of sender window = 4
No. of sequence bits required in Selective repeat ARQ = $\log_2(2*4) = 3$ bits

④ Max size of sender window = "Q"
No. of sequence bits required = $\log_2(2Q)$

$= \log_2 2 + \log_2 Q$

$= 1 + \log_2 Q$

(5) BW = 10Mbps = 10^7 bits

RTT = 50ms ; Data size = 5 bits ; window size - ?

No. of sequence bits in selective repeat ARQ?

In RTT = $50 * 10^{-6} * 10^7$ = 500 bits

Data units = $\frac{500}{5}$ = 100

Window size = 100 frames

	SWS	RWS
$2^8 = 256 = 8$ bits. ← $SWS \leq 2^{m-1}$ ← Selective Repeat	128	128
$SWS < 2^m$ ← Go Back N	127	1

→ For maintaining the same window size, selective repeat ARQ requires more sequence bits compared to Go Back N ARQ.

*Longest Mask Matching:

Whenever a packet comes to the router, and the router has identified multiple paths for the packet, then the path having more no. of 1s in the mask is preferred.

Error Control Policies of Data Link Layer:

Hamming Code : Error correcting code.

Data + Parity bits = codeword

Sender

Data = 1001 1010

$2^r \geq m + r + 1$

r = parity bits
m = message bits

r = 3 ✗

$2^3 \geq 8 + 3 + 1$

$8 \geq 12$ ✗

r = 4 $2^4 \geq 8 + 4 + 1$

$16 \geq 13$ ✓

* Parity bits placed in power of 2 positions.

1	2	3	4	5	6	7	8	9	10	11	12
P_1	P_2	1	P_4	0	0	1	P_8	1	0	1	0

*Even parity

P_1 ⇒ 1, 3, 5, 7, 9, 11 → 0, 1, 0, 1, 1, 1 → $P_1 = 0$

P_2 ⇒ 2, 3, 6, 7, 10, 11 → 1, 1, 0, 1, 0, 1 → $P_2 = 1$

P_4 ⇒ 4, 5, 6, 7, 12 → 1, 0, 0, 1, 0 → $P_4 = 1$

P_8 ⇒ 8, 9, 10, 11, 12 → 0, 1, 0, 1, 0 → $P_8 = 0$

Data = 1001 1010

↓

Sender hamming code

↓

Sender codeword = 0 1 1 1 0 0 1 0 1 0 1 0

noise

Hamming code

Receiver codeword = 0 1 1 1 0 0 1 0 1 1 1 0

$P_1 P_2 1 P_4 0 0 1 P_8 1 1 1 0$

($P_1 = 0$; $P_2 = 1$; $P_4 = 1$; $P_8 = 0$)

↑ Received parity bits.

Received Codeword: 0 1 1 1 0 0 1 0 1 1 1 0
1 2 3 4 5 6 7 8 9 10 11 12
P_1 P_2 P_4 0 0 1 P_8 1 1 0

P_1 ⇒ 1, 3, 5, 7, 9, 11
0, 1, 0, 1, 1, 1 ($P_1 = 0$) ✓

P_2 ⇒ 2, 3, 6, 7, 10, 11
0, 1, 0, 1, 1, 1 ($P_2 = 0$) ✗

P_4 ⇒ 4, 5, 6, 7, 12
1, 0, 0, 1, 0 ($P_4 = 1$) ✓

P_8 ⇒ 8, 9, 10, 11, 12
1, 1, 1, 1, 0 ($P_8 = 1$) ✗

Error bit ⇒ $P_2 + P_8 = 10^{th}$ bit (flip the bit)

* Drawbacks of Hamming Code & it will correct only single bit errors.
→ If noise modifies parity bits copy it will be immediately known to the receiver by comparing with reliable copy.

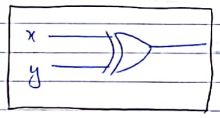
→ Codewords which are generated from a circuit are known as valid codewords.
→ Remaining all are known as invalid codewords.

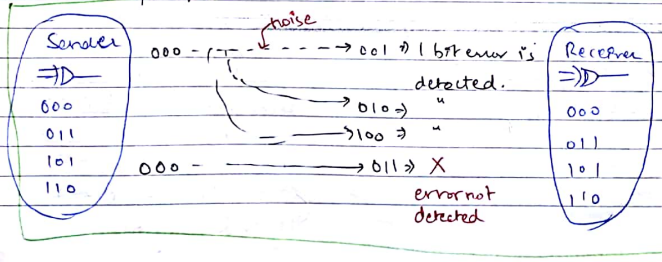
* Error Detecting polices:

Parity scheme:

x	y	o/p
0	0	0
0	1	1
1	0	1
1	1	0

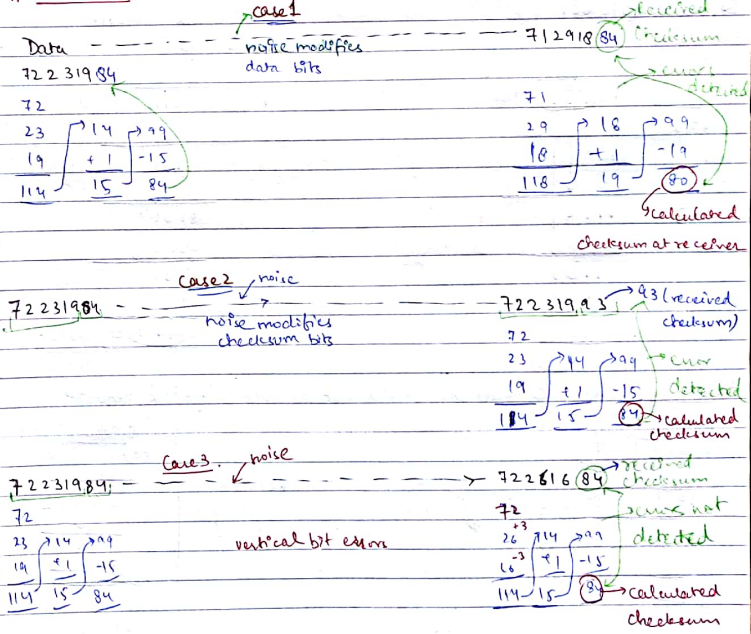
X-OR





→ A valid codeword, if it is converted into an invalid codeword by noise, then, errors can be detected.
→ A valid codeword, if it is converted into another valid codeword then errors cannot be detected.
→ Drawback of parity scheme is that it will detect only odd no. of error bits.
* ⇒ To detect, D errors, the minimum Hamming distance = D+1, between two codewords.
(OR) ⇒ To detect (D-1) errors, the minimum Hamming distance = D.

Checksum: (Data + checksum = codeword)



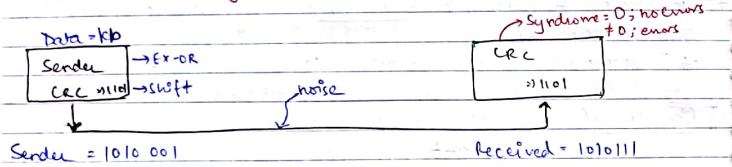
Vertical bit errors:

→ If noise modifies the data in such a way that the vertically placed bits cancel each other, then the calculated checksum will be equal to received checksum, then such errors cannot be detected.

→ Such errors are known as vertical bit errors.

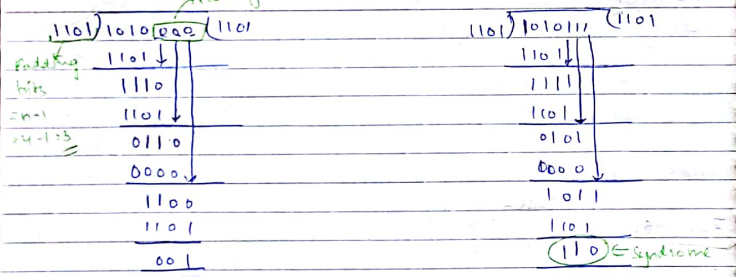
Cyclic Redundancy Code: (CRC)

CRC generator polynomial
CRC = 1101 → x^3+x^2+1

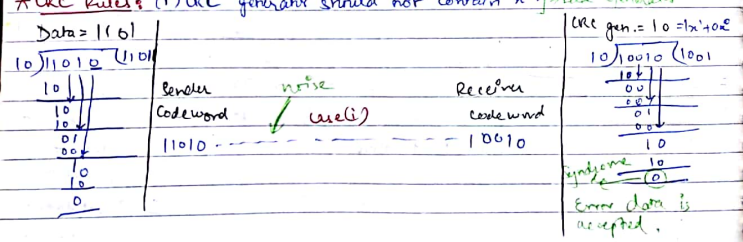


Sender = 1010 001 codeword

Received = 1010111 Codeword

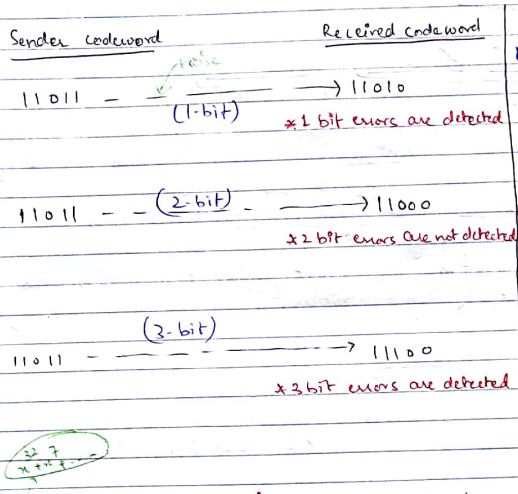


*CRC Rules: (1) CRC generator should not contain 'x'. → Bad Generator



(ii) If (x+1) is generator, it can detect odd no. of errors.

CRC generator = $1+x \Rightarrow 1\cdot x^1 + 1\cdot x^0$



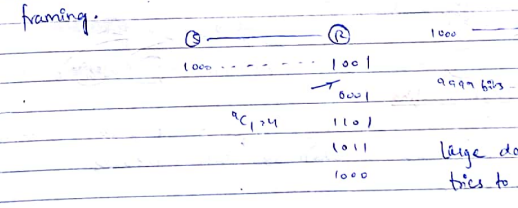
$x^{32}+x^7+\dots+1$

*CRC32 is a standard for detecting all types of errors, i.e, vertical, odd, even types of errors at the cost of taking a good generator.

17/09/2017

FRAMING:

→ Dividing large amount of data into small parts, so that error detection schemes can detect the errors easily is known as framing.



(i) Character Count:

N.L. → 4312585942105

Sender's D.LL. | 5 | 4 | 3 | 1 | 2 | 5 | | 3 | 8 | 9 | 4 | | 4 | 2 | 1 | 0 | 5 |

(count) Data (count) Data (count) Data

→ In character count technique, count value indicates the size of the frame.

→ If noise modifies the data, error detection techniques will detect the errors easily.

→ But if noise modifies the count value, both sender and receiver are out of synchronization.

(ii) Character Stuffing / Byte stuffing:

N.L. → AB

S's D.LL. → FLAG AB FLAG - - - - - - - - - - → R's D.L.L. → N.L. → AB : FLAG AB FLAG

N.L. → A FLAG K

X S's D.LL → FLAG A FLAG K FLAG - - - - - - - - - - → R's D.L.L. → N.L. → A : FLAG A FLAG K FLAG

N.L. → A FLAG K

S's D.L.L → FLAG A ESC FLAG K FLAG - - - - - - → R's D.LL. → N.L. → A FLAG K : FLAG A ESC FLAG K FLAG

(After character stuffing)

N.L. → A ESC FLAG K

S's D.L.L. → A ESC ESC ESC FLAG K FLAG - - - - → R's D.LL. → N.L. → A ESC FLAG K : A ESC ESC ESC FLAG K FLAG

(Drawback of character stuffing:

→ In character stuffing, if more flags occur in the data, then more ESCAPES should by added, so, overhead size increases

* So bit stuffing is used.

(iii) Bit Stuffing:

N.L. → A FLAG B

S's D.L.L. → FLAG A ESC FLAG B FLAG

01111110 01000001 01111110 01000010 01111110.

A = 65 = 01000001

B = 66 = 01000010

FLAG = 01111110

↓ After bit stuffing

01111110 01000001 011111010 01000010 01111110

↳ stuffed bit ("0" after 5 1's)

② N.L. → 01111110 11110

FLAG → 01111110 ; Data at D.L.L. after bit stuffing?

After bit stuffing - FLAG 01111101010 11111000 FLAG

= 01111110 01111101010 11111000 01111110

③ N.L. → 01111101111110

FLAG → 01111110

Data at D.L.L. after bit ~~stuf~~ stuffing?

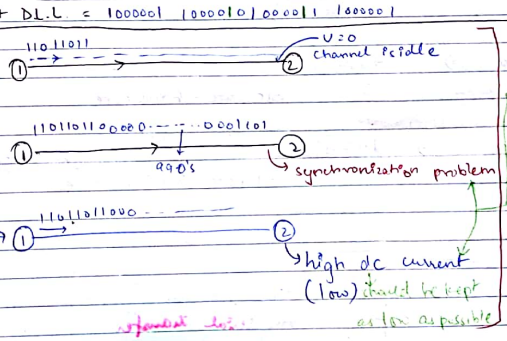
= 01111110 01111101011111000 01111110

④ FLAG = 1000001

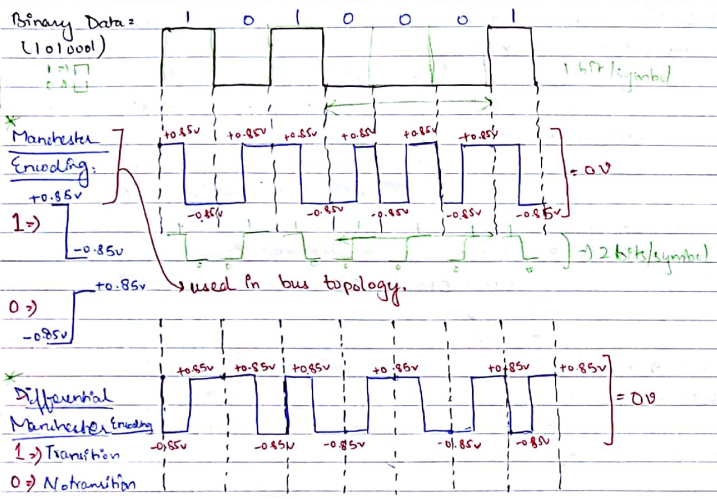
Data at N.L → 1000001 00001

Data at DL.L = 1000001 1000010 00001 1 1000001

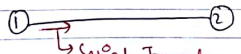
1 → +5V, 0 → 0V



#ENCODING



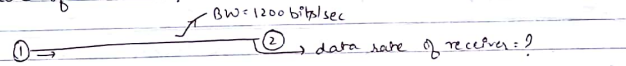
	Bit rate (bits/sec)	Baud rate (symbols/sec)
Binary Data ⇒	7 bits/sec	7 symbols/sec
	Bit rate = Baud rate	
Manchester Encoding ⇒	14 bits/sec	7 symbols/sec
	Bit rate = 2 * Baud rate	



Serial Transfer
- Synchronous serial transfer
- Asynchronous serial transfer

(i) Synchronous Serial Transfer:

Q. In synchronous serial data transfer, 3 eight bit sync. characters are included in 30 eight bit synchronous characters and the bandwidth of the channel is 1200 bits/s, then, what is the data rate of receiver?



3 eight bit sync. characters included → 30 eight bit info characters

3 * 8 = 24 sync bits → 30 * 8 = 240 data bits

$\frac{120 \text{ sync. bits}}{} = \frac{24 * 1200}{240}$ ← 1200 bits

Data rate of receiver = (1200 − 120) bits/sec
= 1080 bits/sec
$\Rightarrow \frac{1080}{8}$ char/sec = 135 characters/sec

→ In synchronous serial transfer extra bits are added for group of characters.
→ These extra bits are not taken by the receiver.
→ These bits are only to alert the receiver that the data is coming.

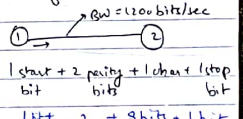
(ii) Asynchronous Serial Transfer:

Q. In asynchronous serial transfer, one start bit, 2 parity bits, 1 stop bit are added for a character, bandwidth of the channel is 1200 bits/sec, then, what is the data rate of the receiver?

Data rate of receiver = 1200 bits/sec

$\Rightarrow \frac{1200}{1+8+2+1} = \frac{1200}{100}$

= 100 char/sec

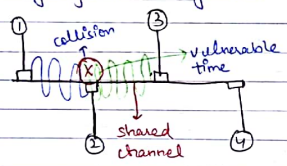


→ In asynchronous serial transfer, extra bits are treated as a part of data.
→ These bits are taken as a part of data by the receiver.

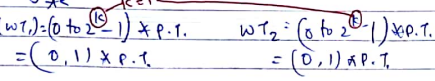
D.L.L.
- → L.L.C. Sublayer (Logical Link Control Sublayer)
  - ⇒ flow control, error control, connection establishment
- → MAC Sublayer (Medium Access Control Sublayer)
  - → to interact with NIC
    - ↳ to prevent or to overcome
      - → collisions
      - ↳ reduce collisions, ↑ throughput ↑ses

* MAC Sublayer Protocols:

(1) **Pure Aloha**: Any system having the data can transmit immediately.



→ Two or more systems transmit the data at the same time, then there is a possibility of collision.
→ The time at which collision occurs is known as vulnerable time.
→ **Backoff Time**: (Exponential Backoff algorithm)



WT_1	WT_2
0	0
P.T.	P.T.
0	P.T.
P.T.	0

Waiting Time $(WT_1) = (0 \text{ to } 2^K - 1) * P.T.$
$= (0, 1) * P.T.$

$WT_2 = (0 \text{ to } 2^K - 1) * P.T.$
$= (0, 1) * P.T.$

① Systems 1 and 2 have transmitted their data for the first time, collided and waited for some random amount of time by applying exponential back off, then, what is the probability that system 1 will retransmit before system 2?

Total possibilities = 4
Favourable (WT_1 should be less than WT_2)
$= \frac{1}{4}$

WT_1	WT_2
0	0
0	P.T.
P.T.	0
P.T.	P.T.

• In the above problem what is the probability that both systems will retransmit at the same time?

Probability $= \frac{2}{4} = \frac{1}{2}$ $(WT_1 = WT_2)$

* Data collided second time:

WT_1	WT_2
0	0
P.T.	P.T.
2P.T.	2P.T.
3P.T.	3P.T.

Total 16 possibilities

$WT_1 = (0, 1, 2, 3) * P.T.$ $WT_2 = (0, 1, 2, 3) * P.T.$

② Systems 1 and 2 have transmitted their data for second time, collided and waited for some random amount of time. Then what is the probability that system 1 will retransmit before system 2?

Total possibilities = 16
favourable = 6
Probability $= \frac{6}{16} = \frac{3}{8}$

WT_1	WT_2
0	P.T., 2P.T. 3P.T.
1PT	2PT, 3PT
2PT	3P.T.

• In the above problem what is the probability that both systems will retransmit at the same time? $(WT_1 = WT_2)$

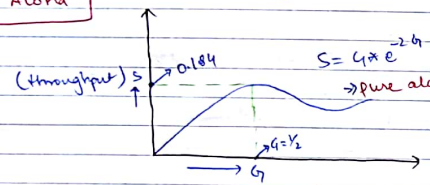
Probability $= \frac{4}{16} = \frac{1}{4}$

→ As the k-value increases, the chance of collision decreases and the chance of data reaching safely increases.

Throughput:

→ The rate at which user transmits the data and the data should reach reach safely is known as throughput.

*** Pure Aloha**



$S = G \times e^{-2G}$

$\frac{dS}{dG} = 0$ ⇒ $G \times (-2) \times e^{-2G} + 1 \times e^{-2G} = 0$

⇒ $e^{-2G}[-2G + 1] = 0$

⇒ $G = \frac{1}{2}$, S_{max}

$S_{max} = \frac{1}{2} \times e^{-2 \times \frac{1}{2}} = \frac{1}{2e}$

$= 0.184$ (18.4%) → out of 100 frames transfritted, only 18.4 frames will be transmitted safely at max.

Q. BW = 50 Mbps. Max throughput of pure aloha.

$S = 18.4\% \text{ of } B.W$

$= \frac{18.4}{100} \times 50 = 9.2 \text{ Mbps}$

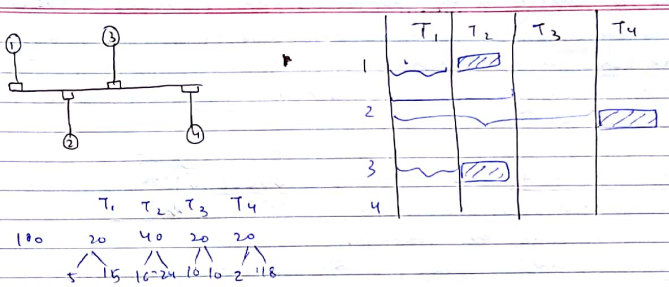
• Throughput of pure aloha if $G = 1$.

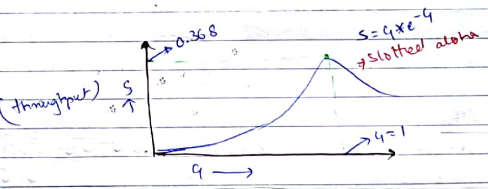
$S = G \times e^{-2G} = 1 \cdot e^{-2}$ ⇒ $\frac{1}{e^2} \times 50$

$= \frac{100}{e^2}\% = 13.5\%$

$= 6.77 \text{ Mbps}$

(ii) *** Slotted Aloha** → a station can transmit only at the start of time slots.





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

$\frac{dS}{dG} = 0$ ⇒ $G \times (-1) \times e^{-G} + 1 \times e^{-G} = 0$

$e^{-G}[-G + 1] = 0$

$G = 1$; S_{max}

$S_{max} = 1 \times e^{-1} = \frac{1}{e}$

$= 0.368$ (36.8%) → out of 100 frames transmitted, a maximum of 36.8 frames will be transmitted safely.

→ The throughput of slotted aloha is double of that of pure aloha.

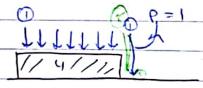
(iii) Carrier Sense Multiple Access (CSMA):

→ When the energy of the channel is low, the channel is idle.

→ When the energy of the channel is moderate, the channel is busy with some other system.

When the energy of the channel is high, there is a collision in the network.

* 1-persistent CSMA:

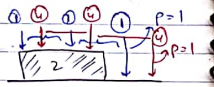


→ In 1-persistent CSMA, stations will continuously sense the channel.

→ Once the channel is idle, it will transmit immediately with $p = 1$.

→ If two or more systems find the channel idle at the same time, then, there is a possibility of collision.

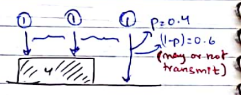
* Non-persistent CSMA:



→ In non-persistent CSMA, if the channel is busy, then, system will wait for a random amount of time and again senses the channel.

→ In this, the possibility of collisions is less because the stations find the channel idleness at different times.

* P-persistent CSMA:



→ In p-persistent CSMA, once the channel is idle, it may transmit with probability "p" or it may not transmit with probability "(1-p)".

E.g. There are 5 stations in a slot. Probability of transmitting the data is 0.4. Only 1 station should transmit to overcome collisions. Then what is the probability that only one station should transmit in the given slot.

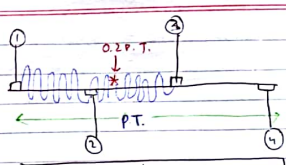
⇒ $1 * (0.4)^1 * (1-0.4)^{5-1} + 1 * (0.4)^1 * (1-0.4)^{5-1} + \dots$ — 5 stations.

⇒ $5 * (0.4)^1 * (1-0.4)^{5-1}$ ⇒ $^5C_1 * (0.4)^1 * (1-0.4)^{5-1}$

⇒ $\boxed{^nC_1 * p^1 * (1-p)^{n-1}}$

(iv) CSMA/CD [Carrier Sense Multiple Access / Collision Detection]

Every computer
- → one program to transmit data
- → one program to check/sense for collisions.



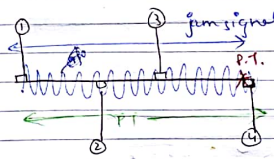
Collision Occurence	Collision Detection
0.2 * P.T.	0.4 * P.T.
0.1 * P.T.	0.2 * P.T.
0.4 * P.T.	0.8 * P.T.
0 * P.T.	0 * P.T.
P.T.	2 * P.T.

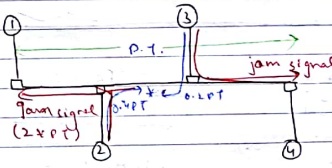
→ Range of collision occurence = 0 to P.T.

→ Range of collision detection = 0 to 2*P.T.

→ If collision is not detected at less than 2 P.T. or at 2 P.T., then the station has acquired the channel or captured the channel or get control over the channel.

→ Then, from that time onwards, the station can start transmitting the original data. (At first, the station sends a jam signal)





→ whenever collision occurs, then the nearest stations will send a jam signal upto a time of 2*P.T.
Then, the exponential back off algorithm is applied.

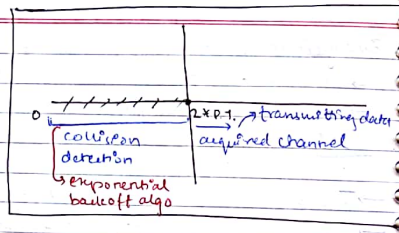
→ The purpose of jam signal is to inform to unknown stations about the collision.

E.g. In CSMA/CD (of ethernet. B.W. = 10Mbps, l = 200m, v = $2 * 10^8$ m/sec. Calculate min. frame size to acquire the channel?

T.T. = 2 * P.T. ⇒ $T.T. = \frac{2 * 200}{2 * 10^8} = 2 * 10^{-6} = \boxed{2\mu s}$

$\because$ T.T. = $\frac{\text{frame size}}{\text{B.W.}}$

frame size = B.W. $\times$ T.T.
= $10^7 \times 2 \times 10^{-6}$
= 20 bits.



(2) In CSMA/CD (or) Ethernet; B.W = 100 Mbps; l = 2300 m; v = 2.3×10^8 m/sec. Calculate max frame size to detect a collision?

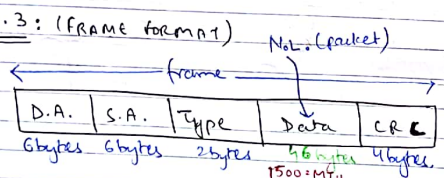
T.T. = 2P.T. $\Rightarrow$ frame size = $2 \times \frac{l}{v}$ B.W.

$\Rightarrow$ frame size = $2 \times \frac{2300}{2.3 \times 10^8} \times 10^8 = 2 \times 10^3$ bits

$= 2 \times \frac{2300}{2.3 \times 10^8} \times 10^8 = 2000$ bits

IEEE 802.3 : (FRAME FORMAT)

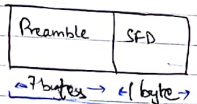
(LAN (Bus Topology))



→ Data Link layer will have header as well as trailer.

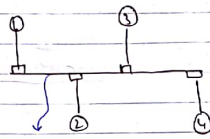
→ Parallelly calculation of CRC and the data transfer are done at the same time

→ By the time data transfer is completed, CRC calculation is also done. So, CRC can be placed at the end (trailer)



starting frame delimiter

→ Preamble and SFD are used for synchronization b/w sender & receiver or to alert the receiver that the data is coming.



10base5
↳ Standard Ethernet

T.T. = 2×P.T.

Frame size = $2 \times \left(\frac{l}{v}\right)$ B.W.

64 Bytes.

→ The minimum frame size in IEEE 802.3 is 64 bytes (or) the minimum data or payload value in IEEE 802.3 is 46 bytes to acquire a channel or to support a protocol [CSMA/CD].

→ If the data that is coming from network layer is less than 46 bytes, then, upto 46 bytes padding bits are added, so that the frame size becomes 64 bytes.

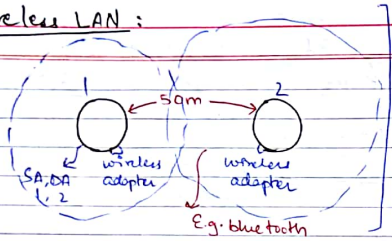
→ The maximum data or the payload value in IEEE 802.3 is 1500 bytes (or) MTU ⇒ Maximum Transferable Unit

The maximum frame size in IEEE 802.3 is 1518 bytes.

→ This is done to give fair and equal chance to all the systems in the network.

(or) not to assign any priorities to a particular station.

Wireless LAN:



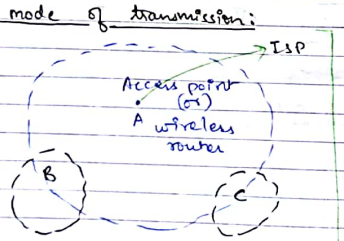
Ad-hoc mode of transmission.

Client to Client Model

→ In the wired LAN or bus topology all nodes will have the single shared channel.
whereas in wireless LAN, every node has its own shared channel.
→ In the wireless LAN, every node can transmit the data to every other node; when they are visible to each other.
→ The effect of noise is more in wireless compared to wired LAN. So, loss of data is more in wireless LAN.
→ So, more number of retransmissions are possible in wireless.

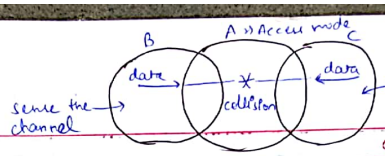
* Infrastructure mode of transmission:

Client Server Model



BSS ⇒ Basic Service Set

→ In infrastructure mode of transmission every node will get the service of access point or access node, when the nodes are in the range of access point.

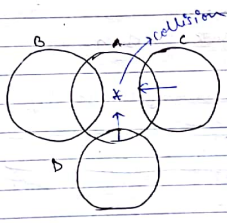


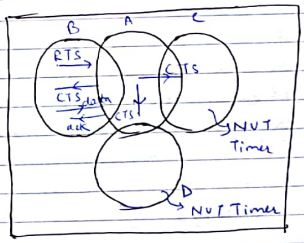
CSMA/CD doesn't work.

→ When the nodes B, A, C are in the planar (straight line) region, where A is the access node as in the above diagram.
→ When nodes B and C transmit the data to node A, then there is a collision at node A.
→ This collision energy will not reach B and C, so, they cannot be detected by B and C. This problem is known as hidden node problem.
→ So CSMA/CD cannot be applied because of hidden node problem.

Carrier Sense Multiple Access / Collision Avoidance [CSMA/CA]

RTS ⇒ Request to send (control signal)
CTS ⇒ Clear to send (control signal)



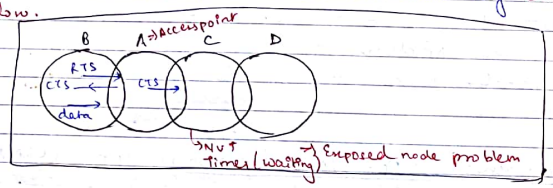


→ When two or more systems transmit RTS at the same time, then there is a collision at access point.

→ Not getting the CTS is a confirmation that there is a collision at node "A".

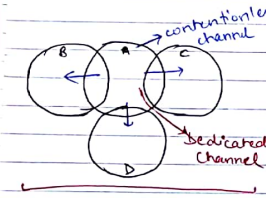
→ Then nodes "C" & "D" will apply exponential backoff algo.

→ Whenever a system tries to acquire the channel, if it fails, then all these failures are indicated by contention window.

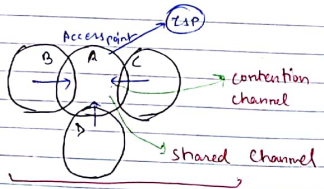


→ At the cost of solving, hidden node problem, there might be a chance of exposed node problem (visible node problem) i.e., even when the node is visible it is unable to transmit the data during that time.
This problem is known as exposed node problem.

Wireless LAN
↳ IEEE 802.11
↳ two functions — DCF (Distributed coordination fn) — four way handshaking — f(RTS, CTS, data, ack)
PCF (Point coordination fn) — f(Data, ack)



PCF



DCF

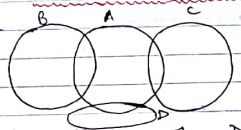
Access node to node → PCF [∵ access node owns the channel, so it's dedicated]

Node to Access node → DCF

→ When a access node wants to transmit the data to a node, then it uses PCF.

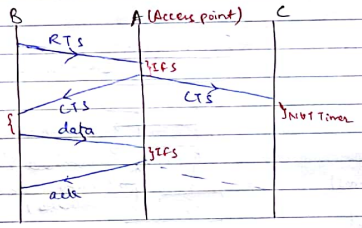
→ When a node wants to transmit the data to a access node, it uses DCF.

Node to Access Node Transfer

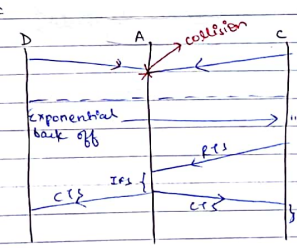


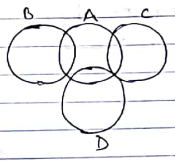
Timing diagram of DCF

IFS → Inter frame space
DIFS → Distributed IFS.



DCF

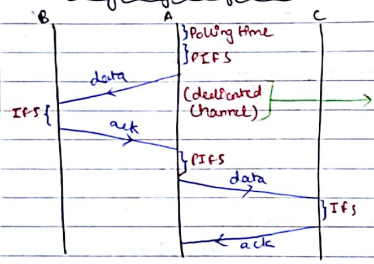


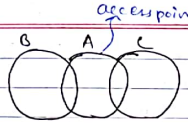


→ When a node wants to transmit the data to access node and parallely the access node wants to transfer the data to a node, then access node will win the race for transfering the data. Priority → Access node > Node

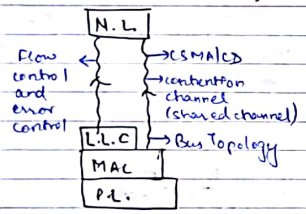
Access node to Node

Timing diagram of PCF Func^{n}

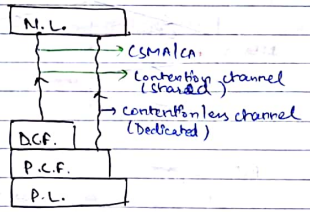




* IEEE 802.3 (Receiver side)
[Wired LAN]



* IEEE 802.11
[Wireless LAN]



Networking Devices:

(i) Repeater, Hub.

(ii) Bridge, Router, Gateway.

* Bridge (LAN): [Not a broadcast domain separator by default]

→ Bridge is a LAN device.

→ Operation is based on MAC address

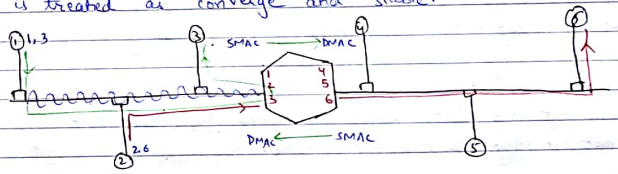
→ Bridge is used for connecting similar LAN networks.

→ Initially, the bridge table of a bridge is empty.

→ The operations of a bridge are:

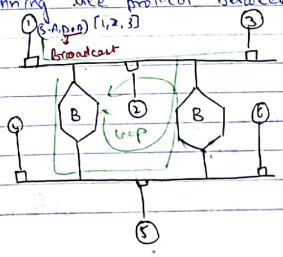
① Learning ② Forwarding ③ Blocking

→ Once bridge knows the complete information of the table, it is treated as converge and stable.

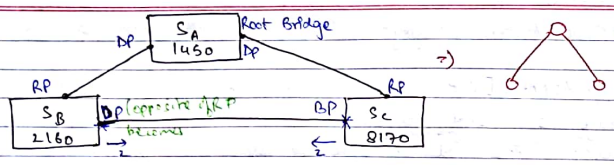


→ Between 2 similar LANs, we connect more than one bridge, to support fault tolerance.

→ When more than one bridge is connected, there is a possibility of getting cycle or loop between the bridges.
So, a graph should be converted into a tree by applying Spanning tree protocol between the bridges.



#Spanning Tree Protocol: [IEEE 802.1C]



⇒

① **Root Bridge:**

→ Out of all bridges, the bridge which is having the least MAC address will become root bridge.

② **Root Port:**

→ Root port is a port which is having the least cost path from non-root bridge to root bridge for sending the data.

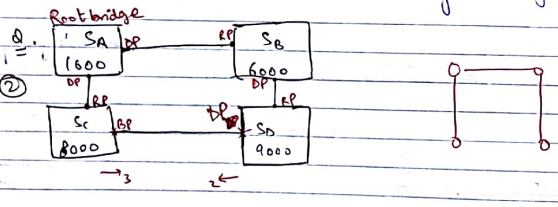
③ **Designated Port:**

→ Designated port is a port which is having the least cost path from root bridge to non root bridge.

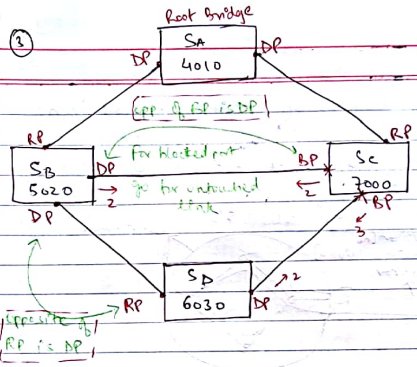
④ **Blocked Port:**

→ Blocked port is a port which is having the highest cost path.

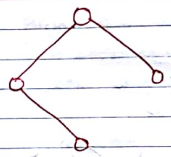
Q.

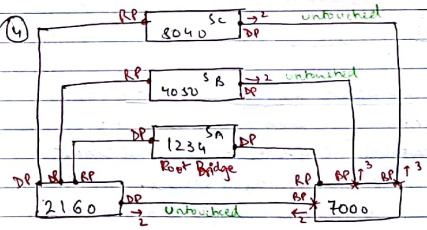


* For finding RPs, go for untouched ~~ports~~ links.
* Opposite of BP is DP.
* Opposite of RP is DP.
* If path length is same, go for lower MAC address [for RP & BP]

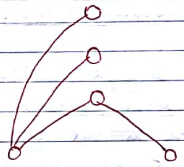


⇒





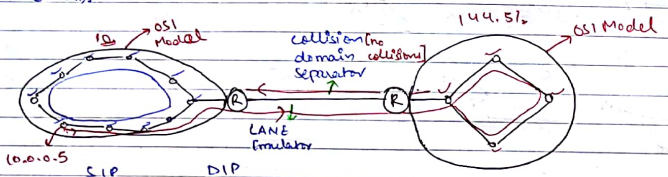
⇒



* Router (WAN) :

→ Router is a WAN devices and its operation is based upon IP address.

→ Router is used for connecting different networks or dissimilar networks.



	SIP	DIP
D	10.0.0.5	255.255.255.255
D	10.0.0.5	144.51.255.255

→ By default router is a collision domain separator and a broadcast domain separator.

→ Router is not a multi protocol converter because it cannot convert one model of packets into another model.

→ Gateway is a multi protocol converter because it can convert one model of packet to another model of packet.

W-B.

Chap3. Qu Two dimensional parity check :

Data :) 1010 1110 1111 1000

W-B. Chp-3

Q.5. $l = 3000km$

Propagation speed = 6 μsec/km

B.W = 1.544 Mbps

Frame size = 64 bytes

1km = 6μsec

3000km = 18 ~~msec~~ millisec

P.T. = ~~36 millisec~~ 18 millisec

R.T.T. = 36 millisec

→ 1sec = $1.544 * 10^6$ bits

→ 36 millisec = $36 * 10^{-3} * 1.544 * 10^6$ bits

No. of bits in R.T.T. = (36×1544) bits

Window size = No. of frames in RTT

$\Rightarrow \frac{\text{No. of bits in RTT}}{\text{frame size}} = \frac{36 \times 1544}{64 \times 8}$

$\Rightarrow 108.56 \approx 109$

Q.6. m=7

SWS	RWS
$< 2^m$	
< 128 (127)	D

Q.7. average transmission rate =

$$\text{Throughput} = \frac{\text{Data size}}{\text{T.T.} + 2 * \text{P.T.}}$$

$$\text{T.T.} = \frac{\text{Data size}}{\text{B.W.}} = \frac{100}{10^5} = 10^{-3}\ s$$

R.T.T. = 2sec

$$\text{Throughput} = \frac{100}{10^{-3} + 2sec} = \frac{100}{2.001} = 49.97\ bps$$

Q.8. $\text{LU} = \frac{\text{Throughput}}{\text{B.W.}} = \frac{49.97}{10^5} = 0.0005$

(9) Data size = 53 bytes — Go Back N ARQ

RTT = 60 millisec; B.W = 155 Mbps

1 sec = 155×10^6 bits

60 millisec = $(60 \times 10^{-3}) \times 155 \times 10^6$ bits

No. of bits in RTT = $(60 \times 155 \times 10^3)$ bits

Window size = $\frac{60 \times 155 \times 10^3}{53 \times 8} = 21,933$

Sequence bits = 15

Q.10 A ⇒ False

*True only when for same window size.

R ⇒ false

Q.15. S — data — P; data+ack; data+ack

piggybacking with full duplex.

A — TT = 53×50 = 2650 ms; PT = 200 msec; 450 msec — B

Q.16. Throughput = $\frac{\text{Data size}}{\text{Total time}} = \frac{5 \times 10000 \text{ Bytes}}{450 \text{ msec}}$

$\Rightarrow \frac{5000}{450} \times 10^6$ Bytes/sec = 11.11×10^6 Bps

17. $G(x) = x^4 + x + 1 = 10011$

$M(x) = x^7 + x^6 + x^4 + x^2 + x = 11010110$

$\Rightarrow 110101100110$ (13 12 11 10 9 8 7 6 5 4 3 2 1 0)

$x^{11} + x^{10} + x^8 + x^6 + x^5 + x^2 + x$

10011) 11010110 0000 (11000010

⋮

0110

18. $P(x) = 1011110001111$

$T(x) = 1101011001110$

7 bit errors.

19. ~~For 1 sec → R bits~~ (M) ——— (N) ← L-km →

~~RTT =~~

P.d = (1/t) sec/km

1 km : t sec

"L" km : (Lt) sec

P.d = Lt sec

B.W = R bits/sec ⇒ 1 sec = R bits

(2Lt) sec = (2LtR) bits

No. of bits in R.T.T.

Window size = $\frac{\text{No. of bits in R.T.T}}{\text{Frame size}} = \frac{2LtR}{K}$

No. of sequence bits = $\log_2(1 + B)$

$\Rightarrow \log_2\left(1 + \frac{2LtR}{K}\right) = \log_2\left(\frac{K + 2LtR}{K}\right)$

(23)

message signal (low freq sig) → Modulator (carrier sig (high freq)) → modulated sig → demodulator (carrier sig) → message signal

Nyquist rate

$f_s \geq 2f_m$

$f_s \geq 2(f_2 - 0)$

$f_s \geq 2f_2$

twice of max

$-f_2$ 0 f_2

image message

(1) $v = n\lambda$

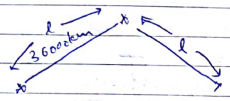
$n = 3\,kHz$; $3\times10^8 = 3\times10^3 * \lambda$; $\lambda = 100\,km$ ✗ Not possible → height of tower

(2) $n = 3MHz$; $3\times10^8 = 3\times10^6 * \lambda$; $\lambda = 100\,m$ possible → height of tower

(3) $n = 3\,GHz$; $3\times10^8 = 3\times10^9 * \lambda$; $\lambda = 0.1\,m$ → height of tower

(28) $P.T. = 2*\left(\frac{1}{v}\right) = \frac{2*36000*10^3\,m}{3*10^8\,m/s}$

$P.T. = 24*10^{-2}\,sec$



L.U. = 50% → window size

$\Rightarrow 50 = \frac{Nx}{TT+2PT} * 100\% \Rightarrow \frac{50}{100} = \frac{TT}{TT+2PT} \Rightarrow TT+2PT = 200TT \Rightarrow 2*PT = 199TT$

$\Rightarrow 2*24*10^{-2} = 199 * \frac{\text{Frame size}}{B.W.} \Rightarrow 2*24*10^{-2} = 199 * \frac{x}{10^7\,bits/sec}$

$\text{Frame size} = \frac{2*24*10^{-2}*10^7}{199} = 24.1*10^3\,bits$

$= 3.01\,KBytes.$

(30)

	3 bits	$SWS \le 2^{m-1}$	$SWS < 2^m$
	Stop & wait	SR	
W.S.	1	4	7
Buffer size	low	moderate	high

(35) $\frac{S}{N} = 20\,dB = 10\log_{10}\left(\frac{S}{N}\right)$

$= 10^2 = 100$

Max data rate $= B\log_2(1+S/N)$

$= 3*10^3 * \log_2(1+100)$

$= 19.975*10^3\,bits/s$

(36) $BW = 4\,kbps$

$P.d = 20\,ms$; $\%\,LU = 50\%$

$T.T. = 2PT$; $\text{Frame size} = 2*p.d.$ / $B.W.$

$x = \frac{}{4*10^3\,bps} = 2*20*10^{-3}\,sec \Rightarrow x = 160\,bits$

(37) Minimum Hamming dist. = 4

To correct "d" errors, the min^m hamming distance is $(2d+1)$.

$2d+1 = 4 \Rightarrow d = 1.5$

$d = 1$

1.0, 1.1, 1.9 $\Rightarrow 1$; 2.0, 2.1, 2.9 $\Rightarrow 2$

S: 0000, 0111 → 0001, 0111 (1 bit ✓) R

(38) $\left(\frac{S}{N}\right) = 7\,dB = 10\log_{10}(S/N)$

$S/N = 10^{0.7}$

$= 5.011$

Max data rate $= B\log_2(1+S/N)$

$= 400*\log_2(1+5.011)$

$= 1035.04 \approx 1035\,bps$

(T3) 10 packets. WS = 4

S: 4,3,2,1 ; 8,7,6,5* ; 8,7,6,5 ; 9,8*,7,6 ; 10,9,8 → R

CSN, x = 19

SR

S: 4,3,2,1 ; 8,7,6,5* ; 10,9*,5 ; 9 → R

y = 12

x + y = 19 + 12 = 31

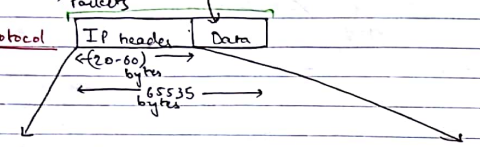
19/09/2017

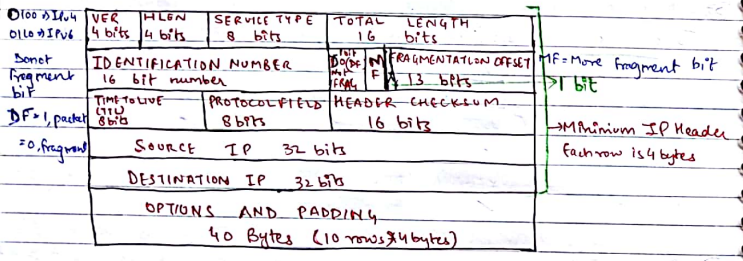
Network Layer:

T.L.

Packets

IP Protocol





→ Starting 4 bits of the IP packet decide whether the packet is IPv4 or IPv6.

→ Header Length is going to indicate the size of the header, that is available in the packet.

0000
0001
0010
0100
→ Don't care

E.g. HLEN = 1001 = 9
Size of header = 9 rows
= 9 × 4
= 36 bytes.

0101 5 rows * 4 bytes = 20 bytes
0110 6 rows * 4 bytes = 24 bytes
⋮ ⋮
1111 15 rows * 4 bytes = 60 bytes

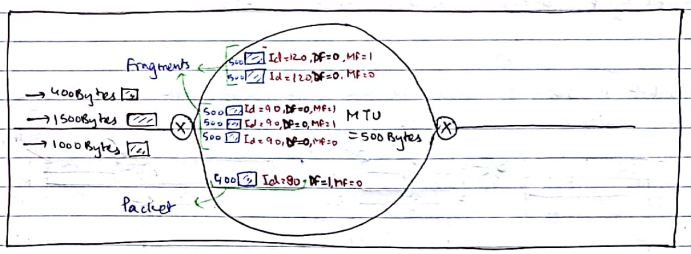
→ Service Type is going to indicate the type of service that is provided to the packet by the router.

→ Total length indicates the size of the packet.

E.g. Total length = 0000000011111111 ; HLEN = 1001
Size of packet = 255 ; Header size = 9 * 4 = 36 bytes.

Header + Data = Packet size
36 + x = 255
x = 255 - 36
x = 219 bytes

→ If both, total length and header length are given, we can calculate size of the data or payload value.

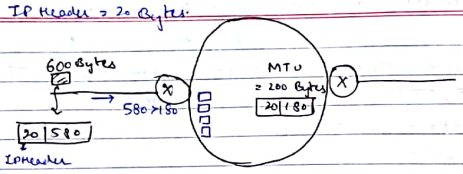


→ Fragments belonging to same packet will be given the same identification number. So that the destination can easily combine fragments belonging to same packet.

→ For all intermediate fragments starting from 1st MF = 1. Specially for the last fragment, MF = 0

→ Fragment offset indicates the size of the fragment and the position of the fragment in the packet.

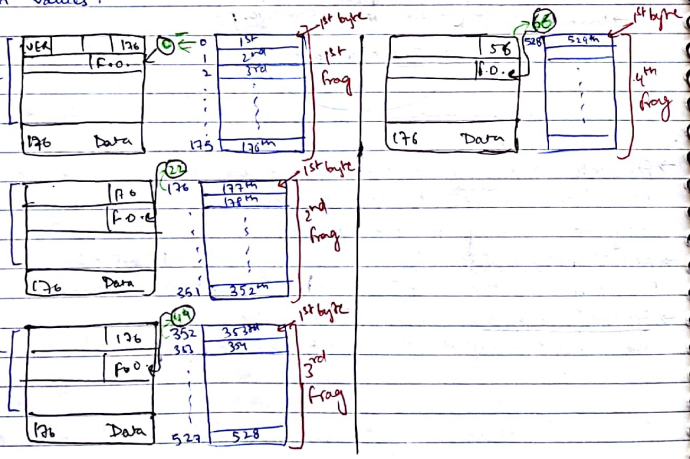
IP Header = 20 Bytes.



22×8 = 176 ✓
23×8 = 184 ✗

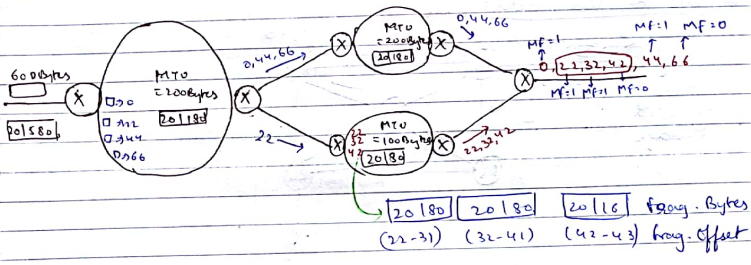
	1st Frag	2nd Frag	3rd Frag	4th Frag
Id	70	70	70	70
DF	should be 0	0	0	0
MF	dissatisfy type 1	1	1	0
Fragment Bytes	20|176	20|176	20|176	20|156 (54+4)
Fragment offset	(0-21)	(22-43)	(44-65)	(66-72)

→ Packet Header is given to all fragments with some change in values.



→ Total Length in Fragment indicates no. of bytes of data in the fragment.

→ Fragment offset * 8 will indicate the starting address of the fragment



→ A fragment can further be fragmented during the travelling path, when a smaller MTU is available in the LAN.

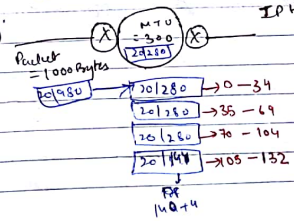
→ Identification number will be same throughout the journey from source to destination for a packet/fragments.

→ Initially for a packet DF = 1.
Once, it is converted into fragment DF = 0. and it will be same upto the destination before it is becoming a packet.

→ In the travel path, MF value will change from source to destination.

→ Fragment Offset will change from source to destination.

Q. 2)

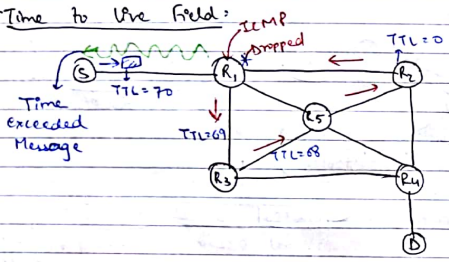


5. The fragment offsets are given as 0, 50, 100, 150 & IP Header = 20 bytes. All fragments are of equal size. Then calculate the packet size.

50 × 8 = 400 → fragment size (data)

400 × 4 = 1600 Bytes → Packet size = 1600 + 20 (IP header) = 1620 Bytes.

#Time to live field:



→ The purpose of TTL is to identify if any loop will exist for the packet or not.

→ Whenever the packet is forward in the wrong direction there might be a chance that the packet will be in the loop.

- Then at one point of time TTL value will become 0.
- Then the next upcoming packet will drop the packet.
- Then ICMP will take source IP address from the dropped packet and informs to source by sending Time exceeded message.

Protocol field is going to indicate the type of application of which the packet belongs to.

IP Protocol is a connectionless, unreliable, best effort delivery protocol.

→ Does not provide any error control.

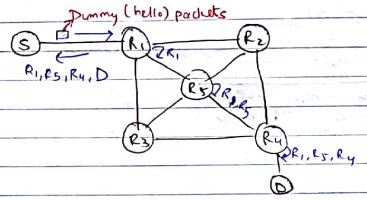
→ Checksum is provided only for the header because for the data, it is already provided in the TCP Protocol in the transport layer.

→ Checksum is provided only for the header so that the processing time is less, so that the packet will be forwarded fast.

#Options And Padding Field:

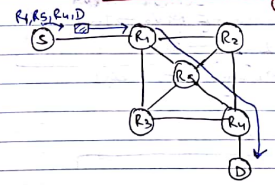
(i) Record Route Option:

→ Record route option is used to record the path (packet) it has traversed from source to destination.

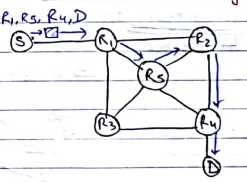


(ii) Source Routing Option:

* Strict Source Routing



* Loose Source Routing

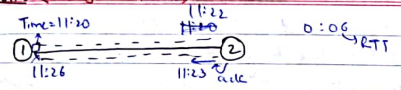


→ In strict ~~root~~ source routing, packets are strictly following the path that is specified by the source.

→ Whereas in Loose Source Routing, along with the path that is mentioned by the source, some other paths, can be visited.

→ By default, packets are routed by routers, but it is not a necessary condition, because packets can be forwarded by host also.

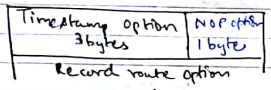
(iii) Timestamp Option: (3 Bytes Option)



→ Timestamp option is used for calculating RTT b/w two end systems.

(iv) NOP Option: (1 byte option)

→ NOP option is used to fill the gaps between options.



(v) EOP Option (End of option):

→ EOP option is used as a separator between header & data.

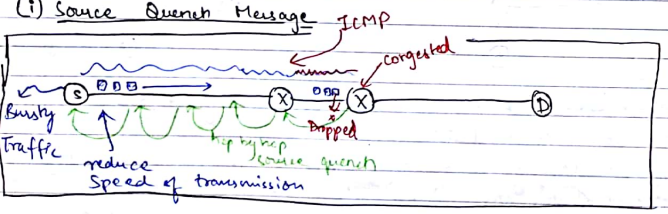
ICMP Protocol:

↳ Internet Control Message Protocol

↳ Reporting errors & Management queries

ICMPv4 → IPv4
ICMPv6 → IPv6
(i) Neighbor solicitation Msg
(ii) Router solicitation Msg

(i) Source Quench Message



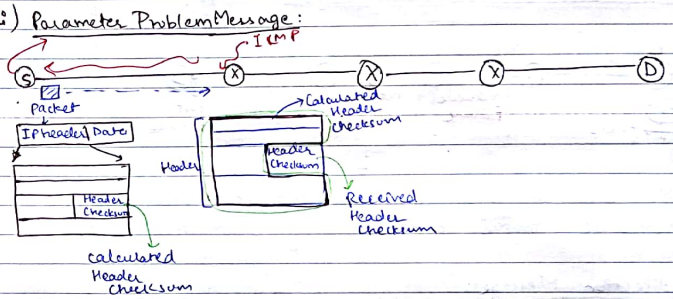
→ Whenever a router is congested, then some packets will be dropped by the router, then ICMP will take source IP from the dropped packet and inform the source by sending Source quench message.

→ Then source will reduce the speed of transmission, then the congested router will be free from congestion.

→ If the congested router is far away from the source, then ICMP will send hop by hop source quench message.

→ Then every router via that path reduces the speed of transmission.

(ii) Parameter Problem Message:

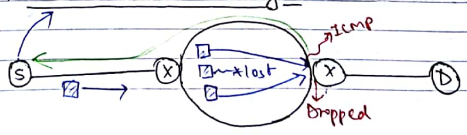


→ Once the data reaches a router, then the calculated header checksum will ~~become the~~ be equal to received header checksum then only the packet will be accepted by the router.

→ If noise modifies the header bits, then the calculated header checksum will not be equal to received header checksum, then the packet will be dropped.

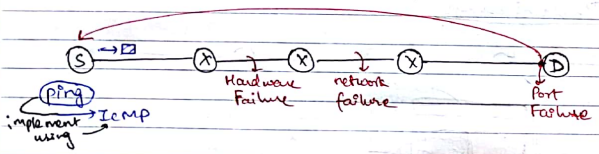
→ ICMP will take the source IP address from the dropped packet and informs to source by sending Parameter problem Message.

(iii) Time Exceeded Message:



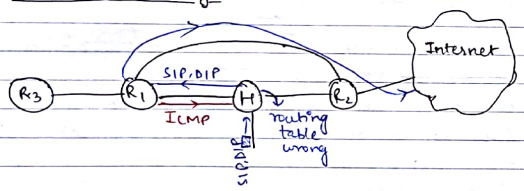
→ When some fragments are lost in the network, then the router will drop the holding fragments.
→ Then, the ICMP protocol will take the source IP address from the dropped packet and informs to source by sending time exceeded message.

(iv) Destination Unreachable Message:



→ ICMP error messages are transmitted not only by the intermediate routers but also by destination host.

(v) Redirection Message:

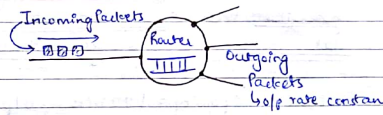


→ If packets are forwarded in a wrong direction and later it is redirected in the correct direction, then ICMP will send Redirection Message to update the routing table with the correct entries.
→ In this scenario, the data is not lost, it is simply redirected.

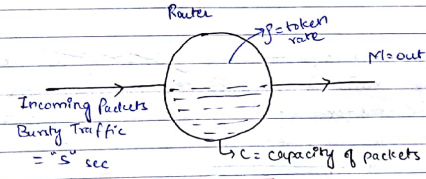
* ICMP message is itself an ~~path~~ IP packet.
* In the IP packet, if the protocol field is 1, then it is treated as the ICMP packet.

Traffic Shaping:

(i) Steady State Traffic
(ii) Bursty Traffic



→ Constant flow of data is known as Steady State Traffic.
→ When there is a sudden increase in the traffic along with the steady state traffic, then it is treated as Bursty Traffic.
→ Whether the input traffic is a steady state or bursty traffic, if the output rate is maintained as constant, then, the router has achieved Traffic Shaping.



$C + \rho s = Ms$

→ When bursty traffic reached to router, if initial capacity is less, then more tokens are generated, then more amount of data is forwarded but total data is always constant.

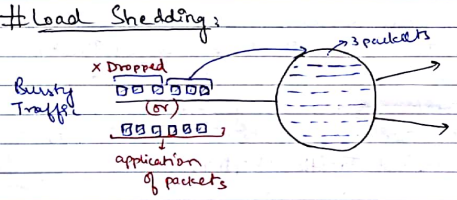
→ When the initial capacity is high, then less tokens are generated, then less amount of data is forwarded, but total data is always constant.

① Initial Capacity = 1 MB bits ; o/p rate = 8 Mbps ; Token rate = 6 Mbps.
C + Ps = Ms | Bursty Traffic time = ?

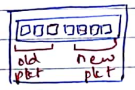
$1 + 6s = 8s \Rightarrow s(8-6) = 1 \Rightarrow \boxed{s = \frac{1}{2}s : 0.5s}$

*Can handle the bursty traffic for a time of 0.5sec..

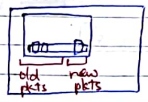
Load Shedding:



→ Load shedding is a way of losing packets, when the packets cannot be handled by routers.
→ Applications like FTP, preference is given to old packets.
→ Applications like multimedia, preference is given to new packets.



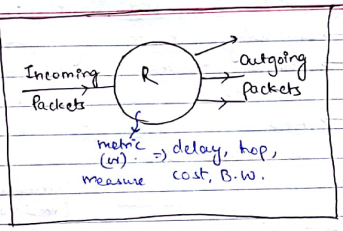
FTP



Multimedia

* Milk & Inline Concept

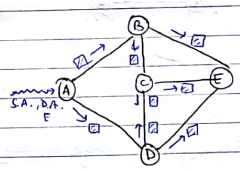
ROUTING ALGORITHMS:



(i) Static Algorithm:
→ does not consider the load on network.
→ Non-Adaptive Algorithms
→ Eg. Flooding

(ii) Dynamic Algorithm:
→ Consider the load on network.
→ Adaptive Algorithms
→ E.g. Distance Vector Routing, Link state Routing, Path Vector Routing

Flooding [Oct 2017]
→ Flooding is defined as whenever a packet comes to router, it is simply forwarded in all directions except the point of origin.



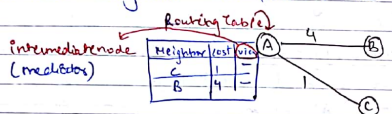
① Calculate all possible paths from A to B using flooding algo using hop as a metric.

ABE → 2 hops	ADE → 2 hops
ABCE → 3 hops	ADCE → 3 hops
ABCDE → 4 hops	ADBCE → 4 hops

→ The advantage of flooding is, it is used to find out unknown destination.
It is used in military applications.
→ Drawback of flooding is that it creates redundant packets which may lead to congestion of a router.

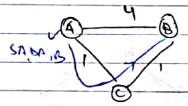
Distance Vector Routing: [Bellman Ford Algo] (Distributed Algo)

→ Initially, the routing table of a router is empty.
→ Every router will be knowing the information of directly connected routers without applying any routing algorithm.
→ And the delay to the neighbor is known as the estimated cost.



(I) (II)

(III)

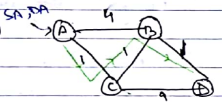


C	1	-
B	2	C

A → B unused link at that time.

→ Whenever, the new links are added, then the path of the packet will change even if the destination address is same.

(IV.)



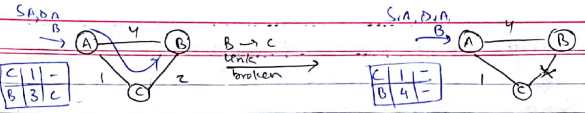
A →

C	1	-
B	4	E
D	3	C

C → D → 2
B

→ In Distance vector routing every router will be knowing the information of the entire network only with the help of neighboring routers.

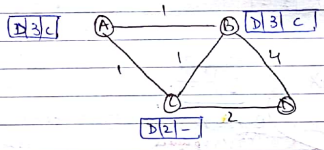
NOTE:
⇒ The target of the algorithm is that the estimated costs should converge to final cost. (shortest path).



C	1	-
B	3	C

C	1	-
B	4	-

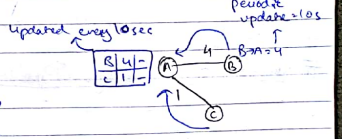
⇒ The path of a ~~path~~ packet is changed when the link is broken, even if the destination is same.
→ It is also known as distributed algorithm because every router will update the information about a route.
→ It is also known as Iterative algorithm because the output of ~~it~~ vector table is given as input for other routers to update the route.



→ Distance Vector is known as the slow convergence algorithm because whenever the link is added, it will O(n) time to be known to all routers about that node. n → no. of routers
→ It is also known as asynchronous algorithm because neighbors will give their vector tables at different time instances.

Periodic Timer:

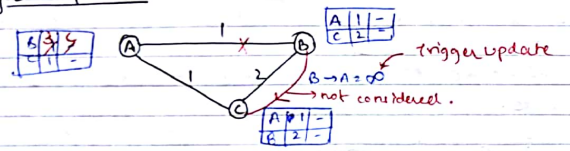
→ The routers in the routing table will be updated periodically, i.e., the neighboring nodes will send periodic update messages to the other nodes.
→ Periodic update message will be helpful to know the status of a router or link whether it is alive or dead.



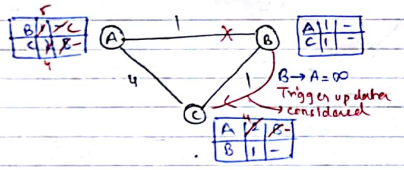
→ A periodic update maybe updated or maynot be updated.
→ A periodic update will be updated when the existing entry in the routing table is larger than the periodic update.
→ Whenever, there is no change in the topology periodic updates will be transmitted.

Trigger Update:

(I)

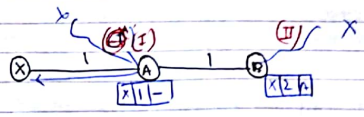


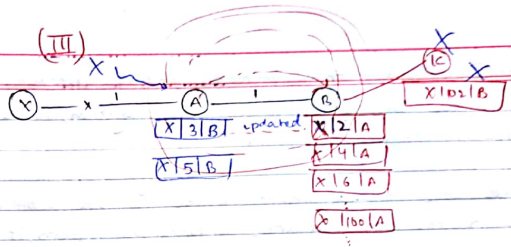
(II)



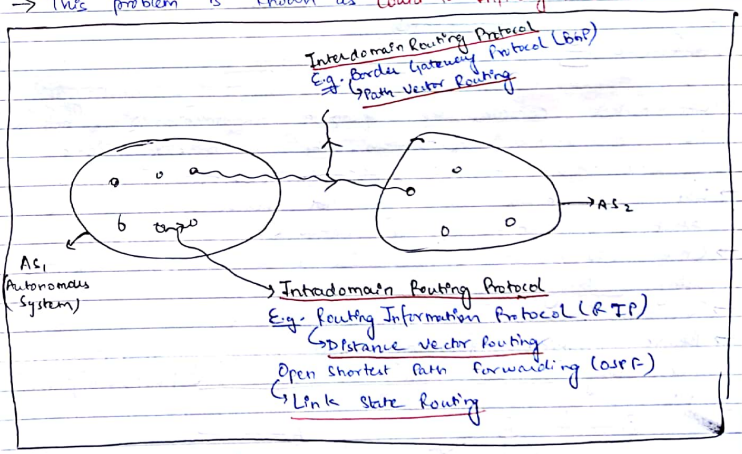
→ Trigger update is immediate and instant.
→ Other node may be updated or may not be updated.
→ Whenever there is a change in the topology, a trigger update will be transmitted.

Count to Infinity Problem:





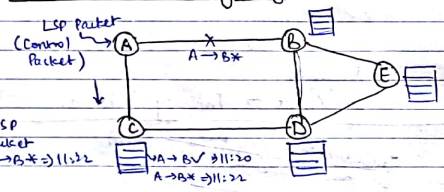
→ Whenever the link is broken, the neighboring routers are giving a false information, that they know how to reach the broken link and it is updated as per the rules of Distance Vector Routing.
→ The routers are updated with wrong values and finally the network will collapse.
→ This problem is known as Count to Infinity Problem.



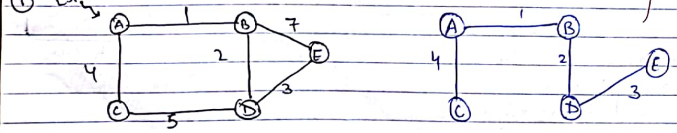
* Kaamesh wari Chebolu
DVR] Youtube
LSR] Earn money online

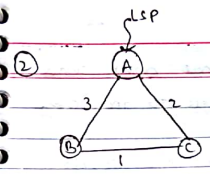
→ If the packets are routed from a router in one autonomous system to a router in same autonomous system, it is known as Intradomain Routing Protocol.
→ E.g. RIP, OSPF.
→ If the packets are routed from router in one autonomous system to a router in another autonomous systems, it is known as Interdomain Routing Protocol.
→ E.g. BGP.

Link State Routing Algo: [Distributed Algorithm]

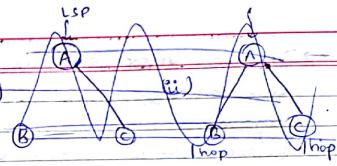


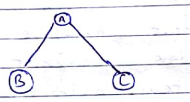
→ LSP packet contains the complete information of the network, i.e., the number of routers, the numbers of links, up and down links, LANs that are connected.
→ LSP packet should be generated periodically with the latest information of the network.
→ Once LSP packet is generated, it is given to all routers using flooding algorithm.
→ So, every router will update with the latest information.
→ Before applying flooding, graph should be converted into tree using shortest path tree algorithm [Djikstra's Algorithm]

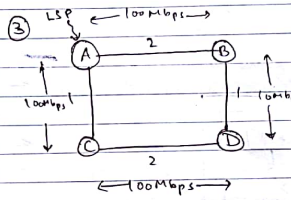




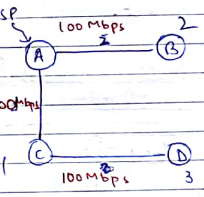
Metric:
(i) delay
(ii) hop.





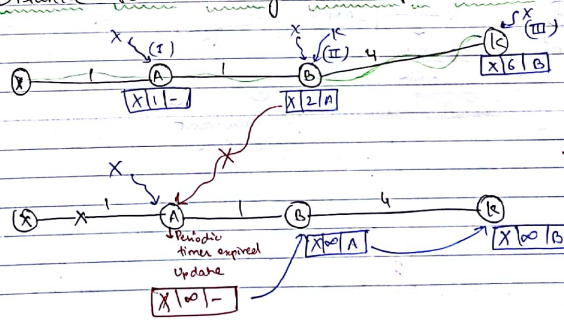


Metrics:
(i) Delay
(ii) Hop
(iii) B.W.



(4) Using Link State
→ Using Link ~~Stating~~ State Routing there is no count to infinity problem, whereas in Distance Vector Routing, there is a count to infinity problem.

Distance Vector Routing with Split Horizon:

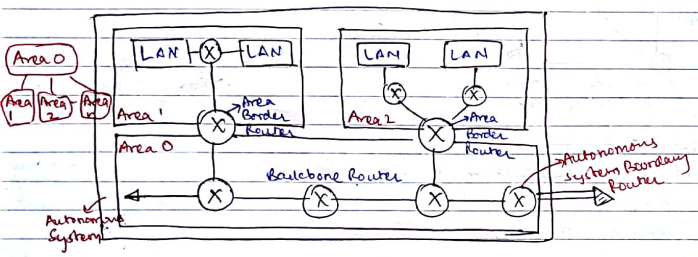


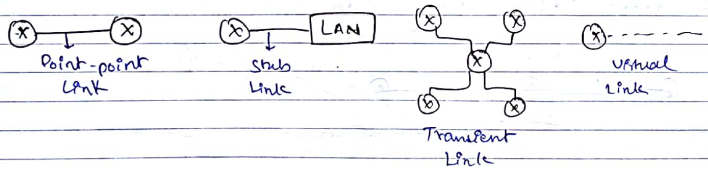
→ Distance Vector with split horizon says that don't send the route learned from a neighbor back to the same neighbor.

→ Distance Vector Routing with split horizon, there is no count to infinity problem.

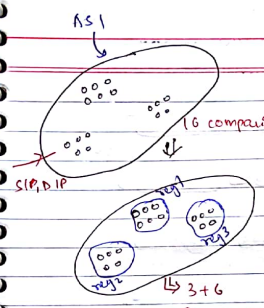
→ Link State Routing algorithm is a fast convergence algorithm because whenever the link is broken, it will be known to all routers immediately with the help of LSP packet.

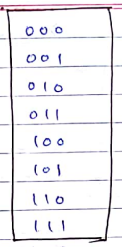
Inter domain Routing Protocol

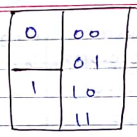




Hierarchical Routing







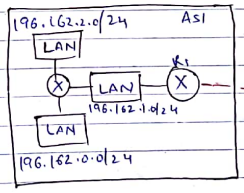
→ Area border router is used for connecting Area 0 with other areas.

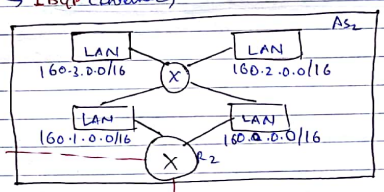
→ Out of all areas, Area 0 is known as backbone area.

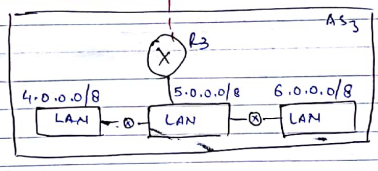
→ Whenever the link is broken, the data will be diverted via virtual link.

Path Vector Routing:

→ BGP → E-BGP (External)
→ IBGP (Internal)







* R1, R2, R3 all are autonomous system boundary routers.

R1

Network	Path
196.162.0.0/24	AS1
196.162.1.0/24	AS1
196.162.2.0/24	AS1
160.0.0.0/16	AS1→AS2
160.1.0.0/16	AS1→AS2
160.2.0.0/16	AS1→AS2
160.3.0.0/16	AS1→AS2
4.0.0.0/8	AS1→AS2→AS3
5.0.0.0/8	AS1→AS2→AS3
6.0.0.0/8	AS1→AS2→AS3

196.162.0.0/24
0.0 ⇒ 00000000.00000000 ⋮ 00000000.11111111 } 2^8

196.162.1.0/24
1.0 ⇒ 00000001.00000000 ⋮ 00000001.11111111 } 2^8

196.162.2.0/24
2.0 ⇒ 00000010.00000000 ⋮ 00000010.11111111 } 2^8

Network	Path
196.162.0.0/22	AS1
160.0.0.0/14	AS1→AS2
4.0.0.0/6	AS1→AS2→AS3

$= 3 \times 2^8 \approx 4 \times 2^8 = 2^{10} = 2^{32-22}$

↳ Rule of supernetting
No. of n/ws should be in power of 2 only

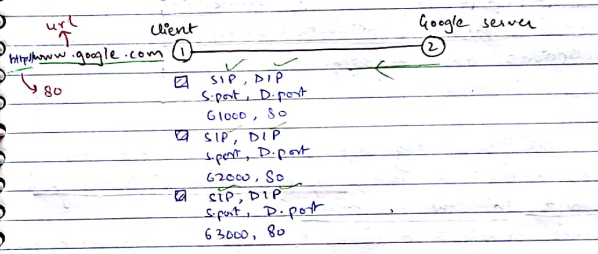
TRANSPORT LAYER

↳ Process-Process Delivery (end-end delivery)

* Port address:
16 bit address (0 to $2^{16}-1$)
(0 to 65535) port addresses

ftp → 21
http → 80

0 to 1023 ⇒ predefined ports, universal ports, fixed ports
1024 to 49151 ⇒ Registered ports.
49152 to 65535 ⇒ Dynamic ports, Ephemeral ports.



→ Predefined ports are the ports which are used for some predefined applications like http, ftp,
→ Registered ports are the ports which are used by different companies to test the networking softwares.
→ Dynamic ports are the ports which are used to distinguish different process in the network.

21/09/2017

TCP Protocol: (Segment) A.L

Segment → group of bytes

flow control (end to end)

TCP Header	Data

segment size ↑ anysize extended

← (20-60) bytes →

← 65535 bytes →

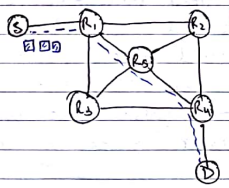
Source Port (16 bits)	Destination Port (16 bits)
Sequence Number (32 bits)	
Ack No. (32 bits) [0 to $2^{32}-1$] Range	
HLEN (4 bits) | S N | A C K | F I N | R S T | S H | U R G | x x x x x x | Window Size (16 bits)	
Checksum (16 bits)	Urgent Pointer (16 bits)
Options and Padding (40 bytes)	

advertised window size

→ In the Data link layer, sequence nos. are assigned for every frame, whereas in TCP, sequence number is assigned for every byte in the segment.

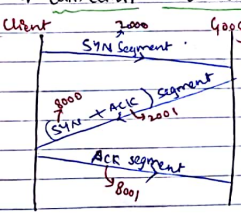
→ Initial sequence number in case of TCP is a random number within the range of [0 to ~~65535~~ $2^{32}-1$]

(i) Connection establishment
(ii) Data Transfer
(iii) Connection Release



TCP → full duplex

* Connection Established



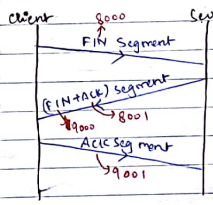
⇒ For complete connection establishment 3-way handshake is required.

⇒ SYN segment doesn't carry any data but it consumes one sequence number.

* HLEN:

0000
0001
0011
0100 Don't care
0101 ⇒ 5 × 4 bytes = 20 bytes
0110
0111
⋮
1111 ⇒ 15 × 4 bytes = 60 bytes

* Connection Release:

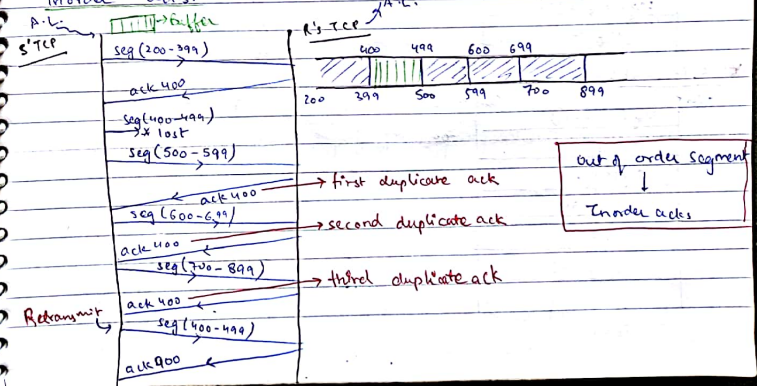


⇒ For complete connection release, 3-way handshake is required.

⇒ During connection establishment and connection release, control segments are transmitted, whereas during data transfer phase data segments are transmitted.

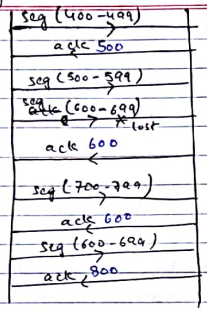
Flow Control policies of TCP:

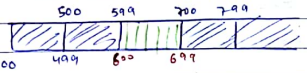
→ TCP can accept out of order segments but always sends Inorder acks.



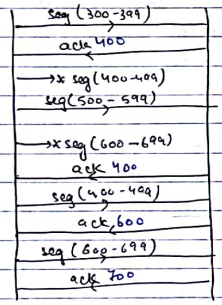
Out of order Segment → Inorder acks

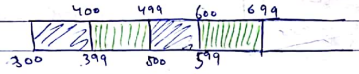
Q. ①





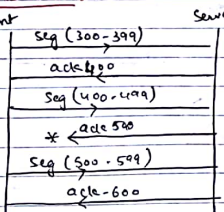
②

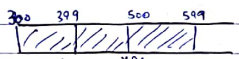




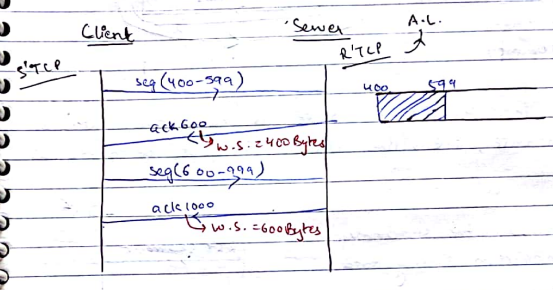
* If first segment is lost, then the ack from connection establishment is used.

*Ack is lost:

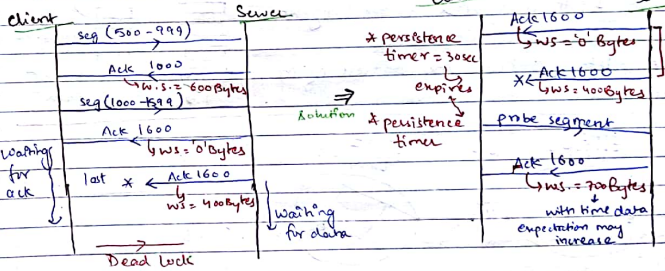




→ Whenever the acks are lost, the next upcoming ack will nullify previously lost acks.



⇒ Window Size (W.S.) is used for synchronization between sender and receiver.

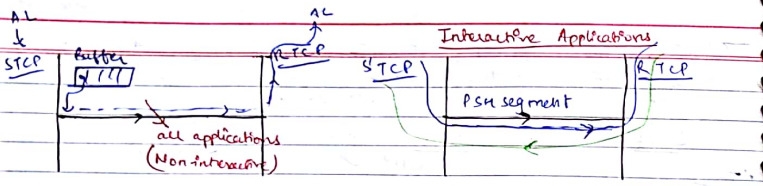


→ Whenever the ack segment containing window size zero reaches to client and if the next ack is lost, then sender is waiting for ack and receiver is waiting for data, this condition is known as deadlock.

→ The problem of deadlock is resolved using persistence timer, this ~~time~~ timer is started only when the window size is zero.

IEEE 802.3 → Bus Topology

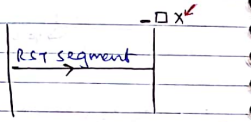
* PSH bit : IEEE 802.5 → Ring Topology



→ If PSH = 1, it indicates that it is an interactive data, so response is immediate, i.e., the data will not be buffered.

* RST bit :

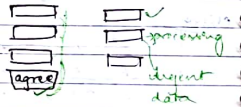
→ If RST = 1, it is used for suddenly closing the connection during data transfer phase.

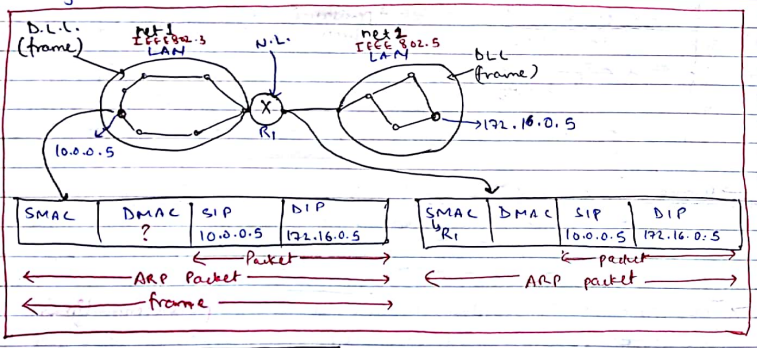


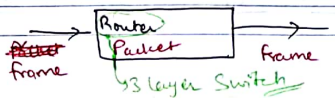
* URG bit :

→ If URG = 1, it is treated as the urgent data.

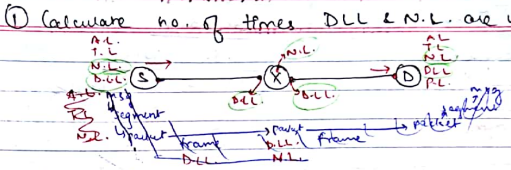
→ Urgent pointer contains the address of the urgent data.



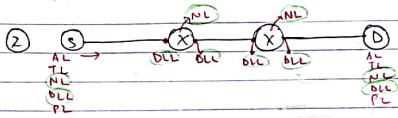


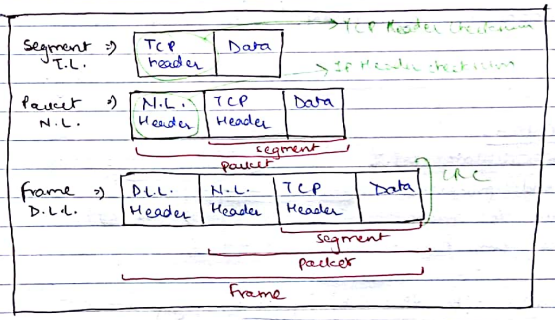


① Calculate no. of times DLL & N.L. are visited from S to D?

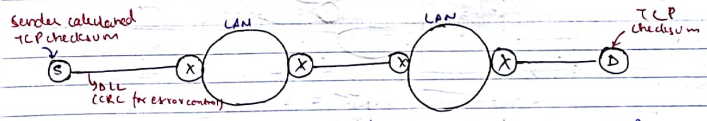


②





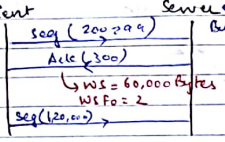
⇒ TCP provides checksum because of potential errors occurring inside the routers.



⇒ TCP header checksum is provided only at the source & destination.

Options & Padding:

i) Window Scaling factor Option:



Standard

W.S. (2^16) * Wsfo (2^14) Bytes

RFC [Request for comment]

→ Segment is divided into small parts according to the allowable capacity of a packet.
This is known as segmentation.

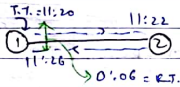
→ The packet is divided into small parts according to the allowable capacity of a LAN. This is known as fragmentation.

→ MTU is kept small so that CRC can detect errors easily.

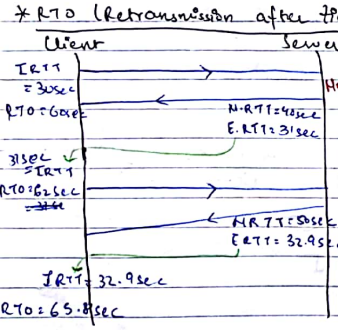
ii) Timestamp Option:

→ It is used to calculate R.T.T. between two end processes.

→ End process time is greater than end to end system time.



* RTO (Retransmission after time out) Timer:



Initial RTT

NewRTT → IRTT = 30sec Estimate RTT

NRTT = 40sec

α = 0.9 scaling factor

E.R.T.T = α * IRTT + (1 − α) * NRTT

= 0.9 * 30 + 0.1 * 40

E.R.T.T. = 27 + 4 = 31 sec

IRTT = 31sec

NRTT = 50sec

ERTT = α * IRTT + (1 − α) * NRTT

= 32.9 sec

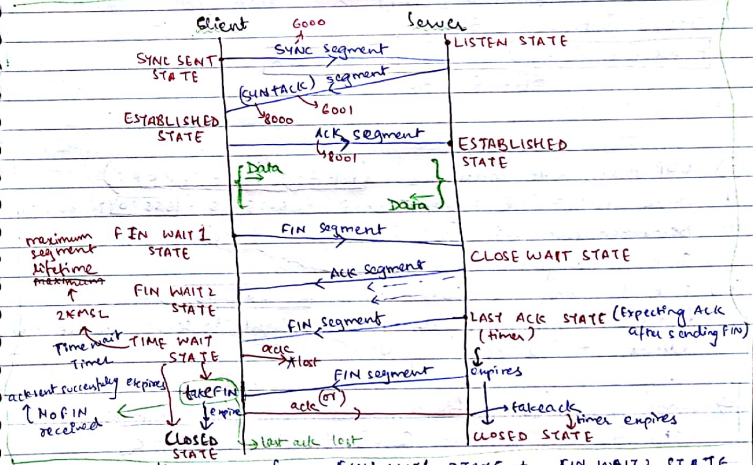
→ RTO timer is a dynamic timer which depends on the load of the network.

Q. for what value of "α", Estimated RTT will be the average of IRTT & NRTT.

α = 0.5 ⇒ ERTT = 0.5 * IRTT + (1 − 0.5) * NRTT

= (IRTT + NRTT) / 2

State Transitions of TCP:



→ Client will move from FIN WAIT1 STATE to FIN WAIT2 STATE if it gets only ACK from the server.

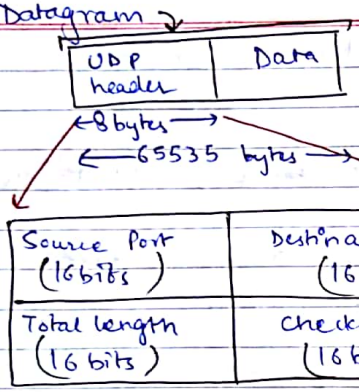
→ After sending FIN segment client expects (FIN+ACK) segment from server.

→ During FIN WAIT1 & FIN WAIT2 STATE, client is not sending any data to the server but it receives data from the server.

→ Last event should take place at client.

UDP:

↳ Datagram



*Port address needed for process to process transmission

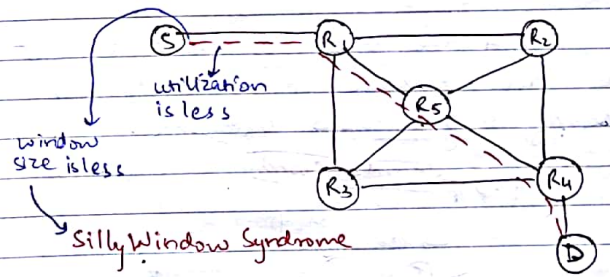
(1) Total length = 0000000011111111
Size of datagram = 511 Bytes
Header + Payload = Size of datagram
Payload = 511 − 8 = 503 Bytes.

(2) UDP Header is given as (FFF0 0050 FFFE FFFE)
(i) Source port = FFF0 = (FFFF − 000F) = 65535 − 15 = 65520 (Dynamic port)
(ii) Destination port = 0050 = 80 (fixed port) [Http]
(iii) Size of datagram = FFFE = 65534 Bytes
(iv) Size of data value = 65534 − 8 = 65526 Bytes
(v) Is the datagram travelling from client to server or vice versa.
Client to server.

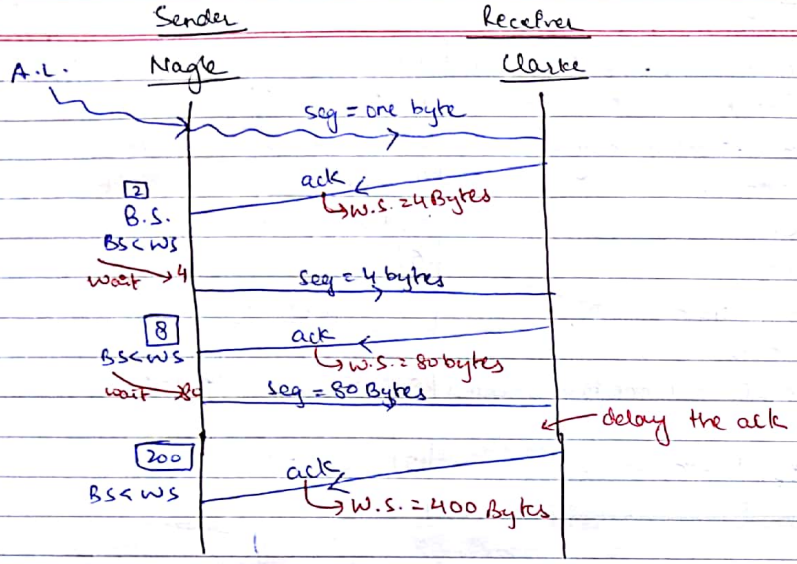
⇒ When source port = fixed, destination port = dynamic port
Data is moving from Server to client. S → C
⇒ When source port = dynamic port, destination port = fixed port
Data is moving from client to server. C → S

TCP	UDP
(1) Dynamic Header. (20-80 bytes)	(1) Fixed header (8 bytes).
(2) Flow Control.	(2) Has no flow control.
(3) Checksum is mandatory.	(3) Checksum is optional.
(4) Error Control.	(4) Has no error control.
(5) Reliable.	(5) Unreliable.
(6) Long messages.	(6) Short messages.
(7) Does not support multicasting or broadcasting.	(7) Supports multicasting & broadcasting.
(8) TCP with IP is connection oriented protocol.	(8) UDP with IP is a connection less protocol.
(9) Http, ftp, Telnet, SMTP	(9) DNS, TFTP, SNMP

TCP
↳ full duplex
↳ path is connected
applications ⇒ transmitting small amount of data
↳ TCP ↳ slowly



Solution: → Sender side
→ Nagle suggested that send the first byte as it is, start buffering the remaining data.
→ Once the ack reaches to client, compare the buffer size with the window size.



→ When the buffer size is less than window size, then, sender has to wait until the buffer data is equal to window size and then transmit the data.

→ receiver side

⇒ Clarke suggested that delay the acknowledgement so that parallely window side will increase along with buffer data, so, the problem of silly window syndrome.

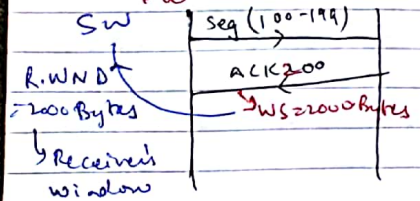
Congestion Control Policies of TCP:

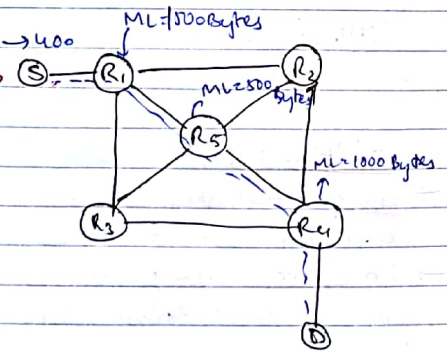
→ Cwnd << Rwnd

→ S.W.

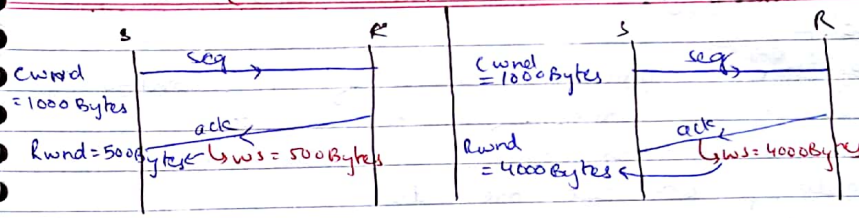
CWND = 500 Bytes (Congestion window)

Data Transfer





SW = min (Rwnd, Cwnd)

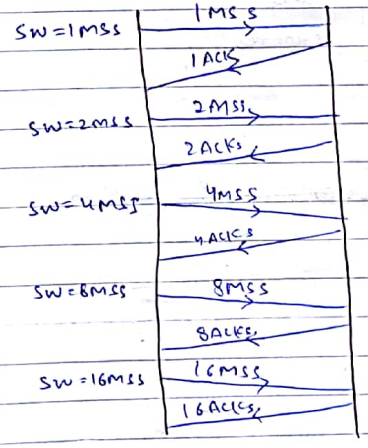


If Rwnd << Cwnd
S.W. = Rwnd ⇒ flow control policies of TCP

If Cwnd << Rwnd
S.W. = Cwnd ⇒ Congestion policies of TCP

* Congestion policies of TCP
(i) Slow start (Exponential Backoff algo)
(ii) Congestion Avoidance (Additive Increase algo)
(iii) Congestion detection (Multiplicative Decrease algo)

* Slow Start Algo:



MSS = Max segment size
1 MSS = 100 Bytes
Cwnd = 3200 Bytes
SS threshold = 1600 Bytes
↳ slow start threshold

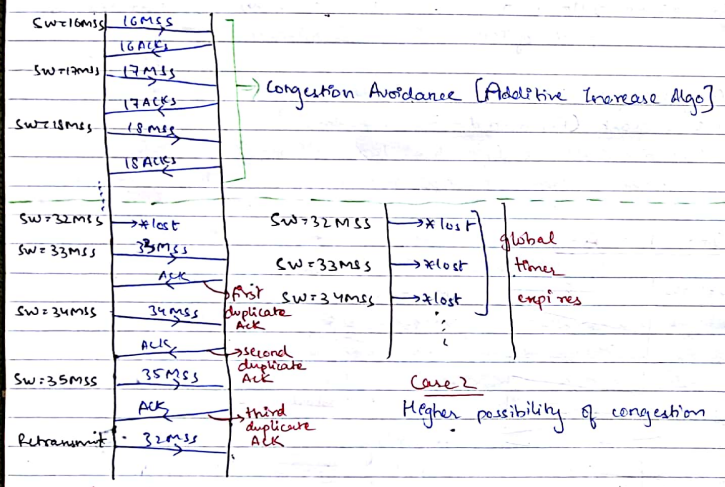
Initially, SW = 2^0 MSS
After 1 RTT, SW = 2^1 MSS
After 2 RTT, SW = 2^2 MSS
After 3 RTT, SW = 2^3 MSS
⋮
After "n" RTT, SW = 2^n MSS.

→ In slow start algorithm, data is transmitted in the form of MSS.

→ In slow start algorithm, the increase of sender window size is based on the number of acknowledgements.

→ In slow start algorithm, sender window size increases exponentially up slow start threshold.

* Congestion Avoidance :



Case 1

Less possibility of congestion.

Case 2

Higher possibility of congestion

* Congestion Detection

Data is lost,
After 3 duplicate Acks,
If it is accepted
(Weak possibility of congestion)
⇒ S.W. = $\frac{1}{2}$ * (Present window)
↳ Congestion Avoidance is applied

Data is lost continuously, until global timer expires
(Strong possibility of congestion)
⇒ S.W. = 1MSS
threshold = $\frac{1}{2}$ * (Present window)
↳ slow start algorithm

E.g. SW = 100 Bytes ← slow start Algo [CWND << RWND]

200
~~400~~ 400
800
1600
1700 ← CA
1800 –
1900
2000 → * lost
(weak possibility of congestion)

SW = 1000 Bytes ← CA
1100
1200
1300
1400 → * lost
global timer expires

Threshold = 700 Bytes
SW = 100 Bytes = 1 MSS ← slow start
200
400
500 ← C.A.
600

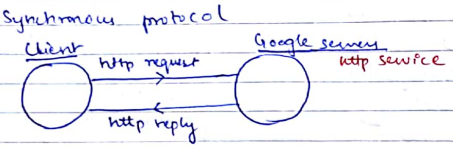
APPLICATION LAYER

* http protocol

(hypertext transfer protocol)

(i) Synchronous protocol

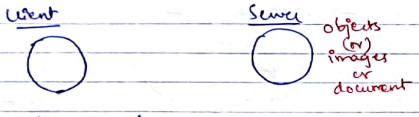
(ii)



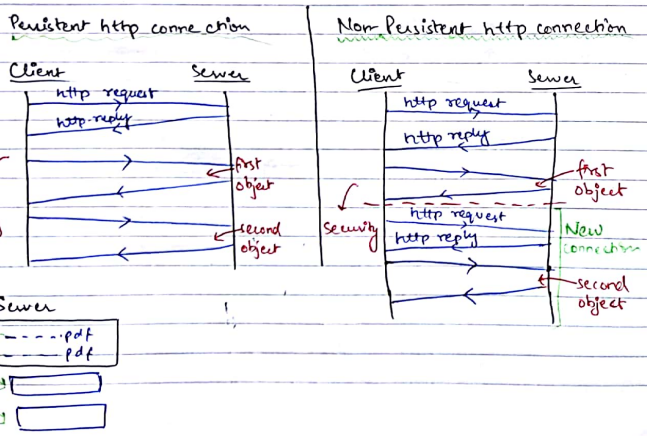
http://www.google.com/
↳ URL

(iii) Port = 80

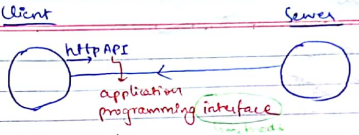
(iv)



http connections
- (i) Persistent http connection
- (ii) Non Persistent http connection



(v)



http methods
get(), head()
post(), connect()
put(), option() → Record Route Option

→ get() method is used to retrieve the document.
→ put() method is used to modify the document.
→ post() method is used to place the modified document in the server.
→ head() method is used to retrieve the information about the document.
→ When connect() method is used, data will go via a secure channel and that too in an encrypted form.

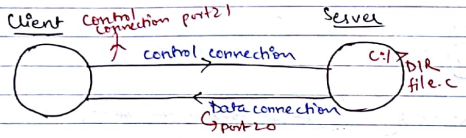
http —connect()→ https
↳ TCP as T.L.

→ option() method is used to trace the path from source to destination

(vi) Stateless Protocol

→ HTTP is a stateless protocol because it doesn't retain any information about the server, in the client browser.
→ Cookie is a piece of code that is transmitted from a server or a mediating agency to the client system.
→ The advantage of cookies is:
① Faster Response ② Authorization

* FTP (File Transfer Protocol)
  - Downloading a file
  - two separate connections
    (i) control connection (port 21)
    (ii) data connection (port 20)



→ Sender sends control commands via a control connection to reach the desired path
→ When the file is about to download, a separate data connection will be established on port 20.
→ When the file is completely downloaded data connection will be closed but control connection will be there only to download some other files

FTP	TELNET
(1) Downloading a large file.	(1) Chat operations (or) Exchange of words
(2) Securely transmitting data.	(2) Remotely connecting the systems.
(3) Port 20 & Port 21	(3) Port 23
(4) Two separate connections required	(4) One common connection sufficient

Similarity
↓ ↓
TCP as Transport Layer

FTP — HTTP
- Synchronous Protocols
- Not store and forward techniques

FTP
- FTP (File Transfer Protocol) ← antivirus experts
  - → authorized users
  - → reliable data
  - → TCP as T.L. ← so relies on TCP
  - → Application layer protocol, so having no internal flow control
  - → Payment for: (i) Internet traffic/access (ii) Purchasing genuine software
- TFTP (Trivial FTP)
  - → anonymous users
  - → unreliable data
  - → UDP as T.L. ← so, can rely on UDP
  - → has internal flow control in the A.L. itself.
  - → Payment for: (i) Internet traffic (ii) Antivirus software

* SMTP (Simple Mail Transfer Protocol)
(i) Text based protocol
→ SMTP is a text based protocol but we can send ~~multiple~~ graphical data using MIME extension.
MIME = Multimedia Internet Mail Extension
↳ Internet Browser

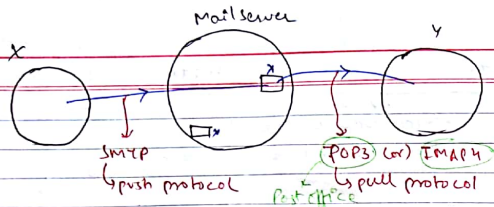
(ii) Port 25
↳ TCP as T.L.

(iii) Base 64 Encoding

a-z	0-25
A-Z	26-51
0-9	52-61
+, /	62, 63

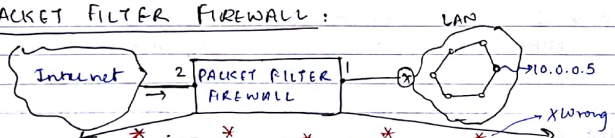
A B
↓ ↓
65 66
↓ ↓

(iv)



→ SMTP is a push protocol because it is used for sending the mail into mail server.
→ POP3 or IMAP4 are known as pull protocols because they are used for retrieving the mails from mail server.
→ SMTP combined with POP3 or IMAP4 is known as client to client protocol with the mediation done by the mail server.
→ SMTP combined with POP3 or IMAP4 is a asynchronous protocol because their clocks need not be synchronised.
→ SMTP combined with POP3 or IMAP4 is a store and forward technique with respect to the mediating server.
→ IMAP4 is more secure than POP3 because it will scan for viruses before the file gets downloaded.
→ Mails can be kept in hierarchy in case of IMAP4 whereas in POP3 all mails are equal.

PACKET FILTER FIREWALL :



	Interface	SIP	S.port	DIP	D.port
Blocked n/w ←	(i) 2	144.16.0.0	*	*	*
Blocked I system →	(ii) 2	*	*	10.0.0.5	*
	(iii) 2	*	*	*	23
	(iv) 2	*	80	*	*

→ Packet filter firewall is a firewall which blocks or forwards the data by observing the transport layer and network layer headers of the content.

Table Rows:

(i) Packets coming from a source IP, i.e., 144.16.0.0 are blocked by the firewall.
(ii) Packets destinated to 10.0.0.5 are blocked because this computer is used for internal LAN only.
(iii) Packets destinated to port 23 are blocked, ie, TELNET service is blocked.
(iv) Access to http service is blocked. Most of the websites use http, so internet is blocked.

⇒ If a virus is placed in the application data then, packet filter firewall cannot detect it.

W.B Pg 16

0.8. $^nC_1 * p^1 \times (1-p)^{n-1}$

$0.8 = n * (0.2) * (0.8)^{n-1}$

$(0.8)^1 = (0.8)^{n-1} \Rightarrow \boxed{n=2}$

19. Length of each time slot = 15 usec

Total throughput = $\frac{\text{Data size}}{\text{Time slot}} = \frac{100 \text{ bits}}{15 * 4S} = \frac{20}{3}$ Mbps

N stations → $\frac{20}{2}$ Mbps | 1 station → $\frac{2}{3}$ Mbps

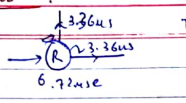
$\boxed{N = 10}$

CSMA/CD for ethernet

20: TT = ~~2PT~~ ⇒ This includes the jamming signal.

Chapter - 5.

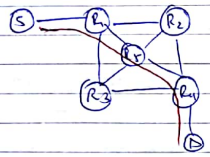
6.

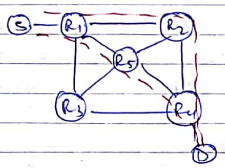


$TT = \frac{\text{Data size}}{B.W.} = \frac{84 \times 8}{100} = 6.72 \text{ msec}$

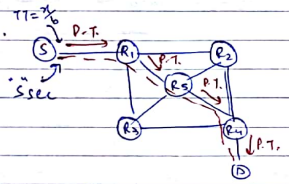
13. Connection
↳ only single connection

Session
↳ Multiple connections at different instances of time





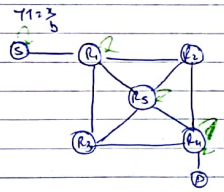
14.



α = "d" sec/hop $\Rightarrow kd$

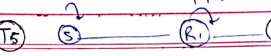
Total = $s + \frac{x}{b} + kd$

15.



Total time = $\frac{x}{b} + kd + (k-1)\frac{p}{b}$

16. X.25 ~~Model~~ ⇒ 3 layer protocol
↳ N.L.
D.L.L.
P.L.



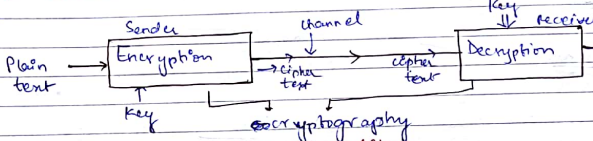
TT = 17.18 msec

TT = 6877

Total = 17.18 + 9 × 6 × 17.18 = 944.9

22/09/2017

BASIC SECURITY



Cryptography

(i) Confidentiality
(ii) Authentication.

→ Cryptography is a science or art of converting one form of data into other form for providing security to data.
→ Providing secrecy to the data → Confidentiality
→ Proving user's identity or the integrity of the user → Authentication

Cryptography

Symmetric Key
E.g. Deffie Hellman Key Exchange

Asymmetric Key.
E.g. RSA Algo

→ Symmetric key Cryptography → If same key is used for encryption and decryption.
→ Asymmetric key Cryptography: If different key is used for encryption and decryption.

Key
- Public Key → If the key is transmitted on the channel.
- Private Key → If the key is kept as secret and later used for encryption or decryption.

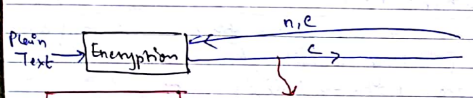
Key features of cryptography:

(1) Prime numbers
(2) Random numbers
(3) Key → Public key
        → Private key
(4) Time stamp

✱ RSA Algorithm:

Sender

Plain text = P



$C = P^e \bmod n$

confidentiality

Receiver

P, q are prime numbers
$n = p \times q$
$\phi(n) = (p-1) \times (q-1)$
$(d \times e) \bmod \phi(n) = 1$

C → Decryption → P

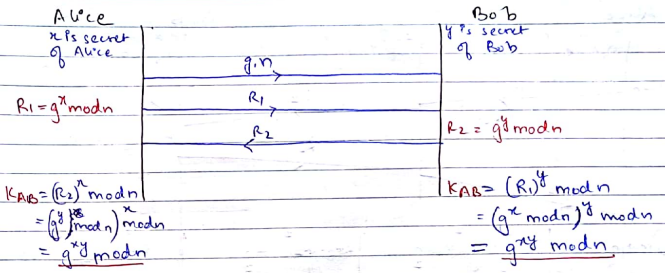
$P = C^d \bmod n$

⇒ Also known as Public key Cryptography.
⇒ e = public key.

→ In RSA algorithm, sender is encrypting with receiver's public key and receiver is decrypting with its own private key.

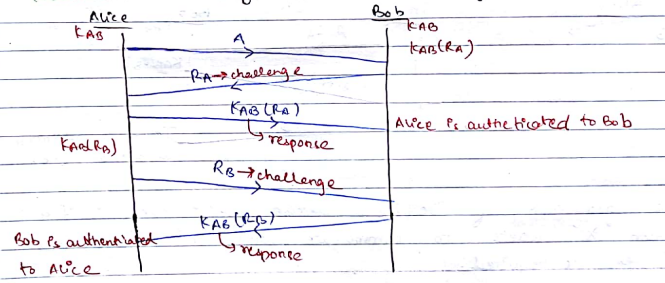
→ It is used to provide confidentiality.
→ In the design of RSA algorithm, only ~~one~~ one system is involved.

✱ Deffie Hellman Algorithm



→ In the design of Deffie Hellman key, two systems are involved.
→ It is also known as the Symmetric Key Cryptography.

✱ Authentication using Deffie Hellman Key Exchange:



* Man in middle attack [Deffie Hellmann]

* Authentication using Boolean reflection attack:

→ The drawback of Deffie Hellman key exchange is if a person 'X' wants to communicate with 'n' people, then 'n' keys are required.

→ Remembering all the key values or maintaining the database of all key values is difficult. So, the burden of key values is taken care by third party or public authority.

$X_{KRZ} \rightarrow {}^{Z}_{F_{XZ}}$

$X_{EXY} \rightarrow {}^{Y}_{K_{XY}}$

(e) $X_{K_{XW}} \rightarrow {}^{W}_{F_{XW}}$

* Mutual Authentication using public key cryptography: (RSA)

4. $p=7 \quad q=11$

$e=7$

$n = p \times q = 7 \times 11 = 77$

$\phi(n) = (p-1) \times (q-1) = 6 \times 10 = 60$

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

$(d \times 7) \bmod \phi(n) = 1$

$(43 \times 7) \bmod 60 = 1$

$301 \bmod 60 = 1$ ✓ (d)

5. $C = P^e \bmod n$

$= 9^7 \bmod 77$

$= 37$

6. $g = 7 \quad n = 23$

$R_1 = g^x \bmod n = 7^3 \bmod 23$

$= 21$

7. $y = 5 \quad R_2 = g^y \bmod n$

$= 7^5 \bmod 23 = 17$

8. $k = g^{xy} \bmod n$

$= 7^{15} \bmod 23 = (7^5 \bmod 23)^3$

$= 17^3 \bmod 23 = 14$

9. $n = 47 \quad g = 3$

$x = 8 \quad ; R_1 = g^x \bmod n$

$= 3^8 \bmod 47 = 28$

10. $y = 10 \quad R_2 = g^y \bmod n$

$= 3^{10} \bmod 47 = 17$

11. $k = g^{xy} \bmod n$

$= 3^{80} \bmod 47 = (3^{10} \bmod 47)^8$

$= (17 \bmod 47)^8 = 17^8 \bmod 47 = (17^2 \bmod 47)^4$

$= 7^4 \bmod 47 = 4$

$17^2 \bmod 47 = 7$

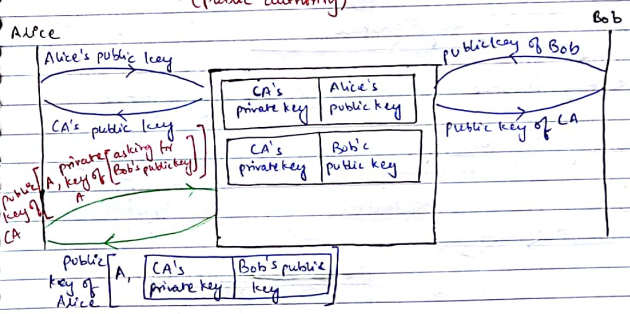
* Authentication using RSA Algorithm:

→ It is better than authentication using Deffie Hellman Key Exchange in terms of security.

→ Mutual authentication using Deffie Hellman key is better than mutual authentication using RSA in terms of speed.

CERTIFICATION AUTHORITY → RSA

(Public authority)



→ If sender is encrypting with it's own private key and receiver is decrypting with sender's public key, it is used to provide authentication.

→ If sender is encrypting with receiver's public key and receiver is decrypting with its own private key, it is used to provide confidentiality.

→ If sender is encrypting with ~~its own~~ receiver's private key.

↳ This statement is wrong.

→ Sender is encrypting with its own publickey. [It is possible, but it is of no use]

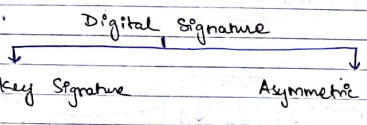
Digital Signature:

* Authentication of Data

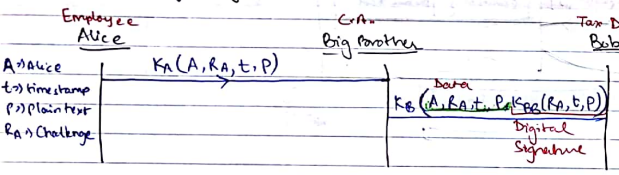
→ In the case of hand written signatures both data and signature cannot be separated.

→ Whereas in the case of digital signature both signature and data can be separated.

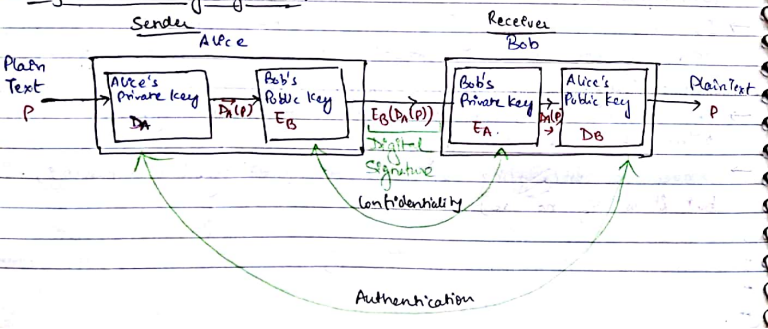
→ In case of hand written signatures, for all types of data, same signature is used, whereas in digital signature, for every individual data, a separate signature is created.



* Symmetric Key Signature:



* Asymmetric Key Signature:



→ In symmetric key signature, then entire algorithm is based on the mediating agency (or) Big brother. So, if he does something ~~at~~ wrong, the entire algorithm will go wrong.

→ Asymmetric key signature is better than symmetric key signature in terms of security.

→ Symmetric key signature is better than asymmetric key signature in terms of speed.

* Message Digest:

Plain text P = 156789001931091067

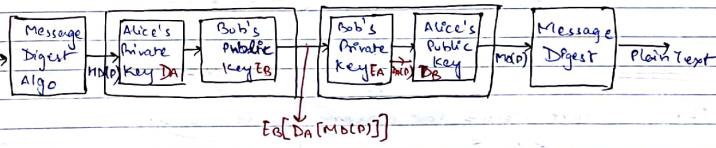
MD(P) = P mod n [n = 100]

= 67

167 ✓
267 ✓
367 ✓

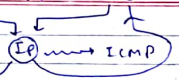
(1) Given "P", anyone can calculate MD(P) easily.

(2) Given MD(P), no one can calculate P' such that [P' = P] where MD(P) = ~~MD~~ MD(P')

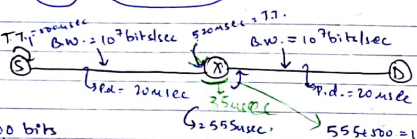


WB chapter 6

9. TCP → T.L. ← UDP



12.



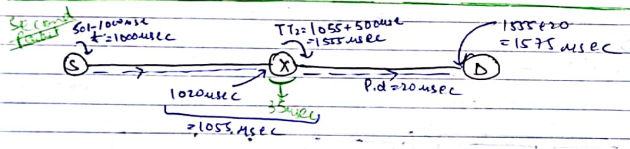
10000 bits
↳ 5000 bits (two packets each)

For 1st packet

$TT_1 = \frac{5000}{10^7} = 500\ \mu sec$

35 μsec = for storing it in queue

TT_2 (at Router) = 555 μsec + 500 μsec = 1055 μsec.

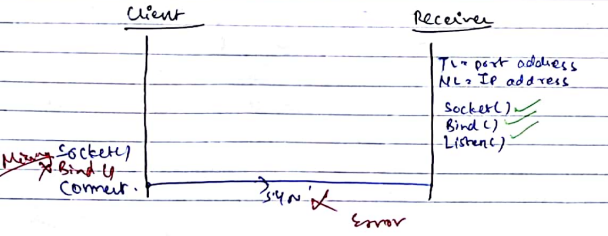


Chapter-7

12. Socket address = Port address + IP address

T.L. N.L.

Socket() =
Bind() =



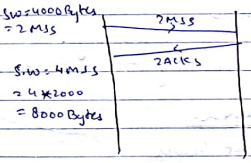
⇒ Socket() functions will create a logical path between the transport layer and the network layer.
⇒ Bind() function will join IP address and port address.
⇒ Connect() system call is a request from client to server for connection establishment.

27. BW = 40 Gbps
↳ 1 sec = 40×10^9 bits
= 5×10^9 bytes

1 seq no ⇒ 1 byte [In TCP]
2^{32} seq no ⇒ 2^{32} bytes
↑ wrap around

$\frac{1 \times 2^{32}}{5 \times 10^9} = 859$ millisec

28. 1 MSS = 2000 Bytes



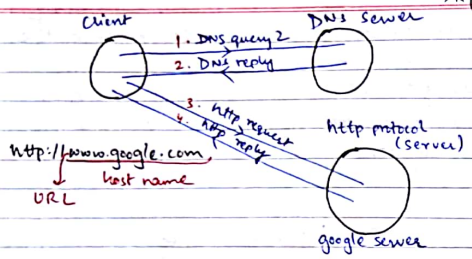
29. C+ Ps = ms

30. ≈ 12 min 50 sec

31.

DNS : Domain Name Service: ⇒ Port = 53

↳ Application Layer Protocol



→ DNS sever is used for mapping host names to IP addresses or vice versa.

→ It is a database containing IP addresses of all computers in the internet.

Design of DNS sever:

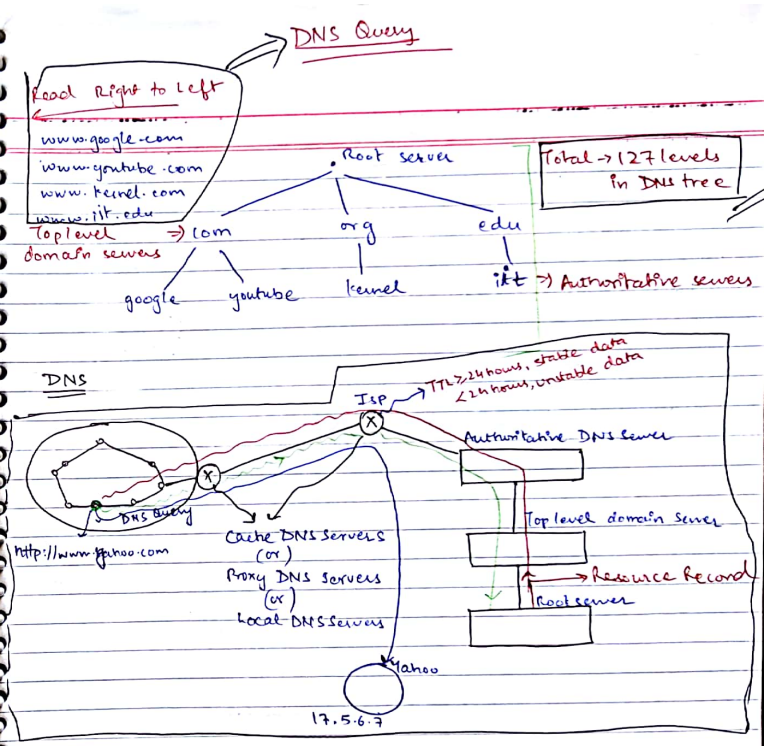
(i) DNS servers placed in hierarchy.

↳ searching time will be less. & response fast.

DNS Servers

→ Root sever
→ Top level domain sever
→ Authoritative Sever
→ Local DNS Server

(ii) DNS severs placed in different geographical areas of world.



* DNS uses UDP as T.L. protocol
* DNS query size < 512 Bytes ⇒ UDP as T.L. protocol
* DNS query size > 512 Bytes ⇒ TCP as T.L. protocol