Date :- 25/02/2019

Network Layer :-

IPv4 :- → 4 Bytes = 32 bits

→ classful Addressing.

A, B, C, D, E

→ 10101111.1000 1001.0000 0001.0000 1010
↓ Binary format

172.10.1.50
↓ Dotted Decimal format

class A :-

→ Ip address range from 0 → 127

N/w ID ← 8 bits — 24 bits → Host ID.

00000000 → 0
0 000001 → 1
|
0 1111110 → 126
0 1111 111 → 127

Ex :- 123.10.1.34

→ 2^{32} = 4 billion ← Address spaces.

→ 2^{24} → Hostes (devices) connected.

→ Mask → N/w ID + HOST ID.

255.0.0.0
N/w ID — Host ID.

class B :-

16 bits N/w ID — 16 bits Host ID.

→ First 2 bits are fixed.

10 000000 → 128
10 000001 → 129
|
10 111110 → 190
10 111111 → 191

→ Ex :- 168.10.1.13

→ N/w Mask :- 255.25
N/w ID

class C:- ← N/w ID 24bits → ~ 8bits Host ID.

110 00000 → 192

fixed

110 00001 → 193

110 11110 → 222

110 11111 → 223

→ Ex:- 195.16.1.3

→ N/w mask → 255.255.255.0

N/w ID | Host ID.

→ For small organizations like colleges class 'C'.

~~For~~ large organizations class 'A' used.

class D:-

→ Used for special purpose.

→ It is reserved for ↓.

1110 0000 → 224

1110 0001 → 225

1110 1110 → 238

1110 1111 → 239

class E:-

1111 0000 → 240

1111 0001 → 241

1111 1110 → 254

1111 1111 → 255

~~Transport layer~~

→ Device to device communication occur in Network layer.

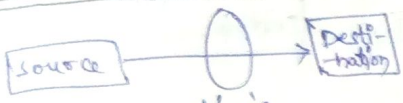
→ Packets present.

Transport layer :-

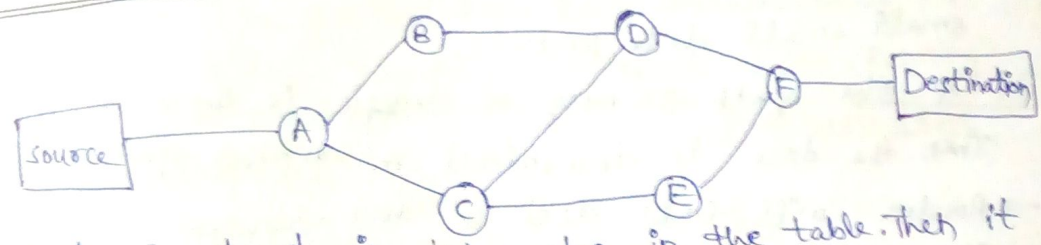
→ process to process communication.

→ segments will be there.

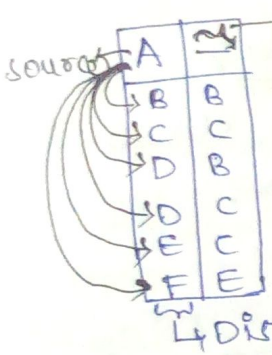
Routing Algorithms :-



store and forward :-



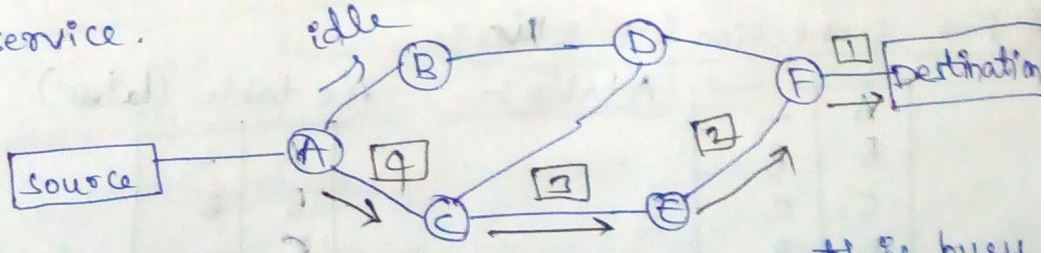
→ Data is stored in 'A' router in the table. Then it will forward it to neighbouring routers.



→ Network consists of
* Connection oriented service.
* Connection less service.

→ Before sending the data packet, the path is established b/w source & destination. → Virtual path.

→ There is no virtual path in connection-less service.



→ In connection less service, If the path is busy then it will go for idle path for trans

Implementation of Connectionless service :-

→ Data frame is very large then it divided into small small data packets.

→ If the path is busy or traffic is there on path then the data is transmitted in another path.

→ Router will take decision and forward the data any path.

→ Datagram :-

Implementation of connection oriented service :-

C's table :-

A	A
B	A
C	—
D	D
E	E
F	E

E's table :-

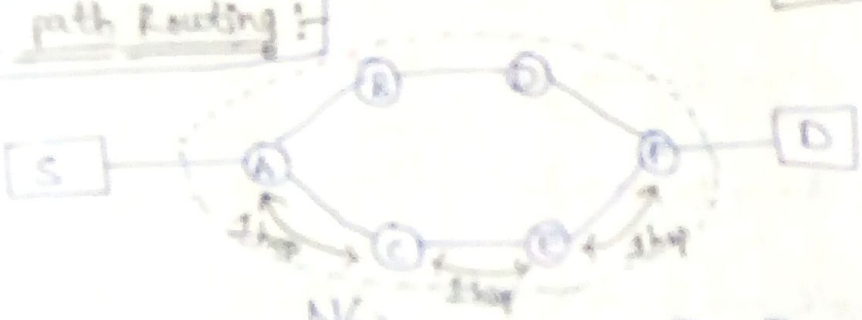
A	C
B	C
C	C
D	C
E	—
F	F

Routing Algorithms :-

1) Non Adaptive Routing Algorithms → static routing
2) Adaptive Routing Algorithms. → Dynamic routing (Run time)

Date : 10/08/2019

Shortest path Routing :-



→ (A) - (C) → 1 hop
(A) - (E) → 2 hops
(A) - (F) → 3 hops

→ Before sending the data, the distance of path is calculated from source to destination. It is known as Non Adaptive Routing Algorithms.

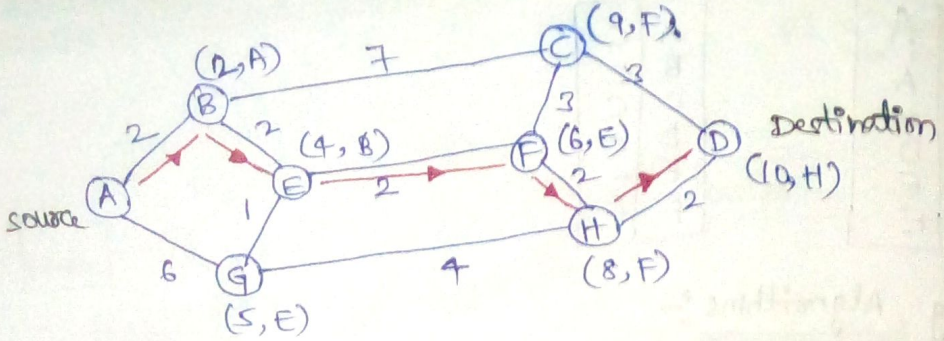
→ Each router has two processes.

* Forwarding process :-
→ Based on the decision, data is transmitted to the next neighbouring router.

* Fill and update process :-
→ Fill the information about the from and to the data packet accessed and transmitted and update in routing table.

Shortest path Routing :-

→ It is non-adaptive Routing algorithm.
→ It finds shortest path before transmitting the data packet



shortest path:-

A → B → E → F → H → D → 10

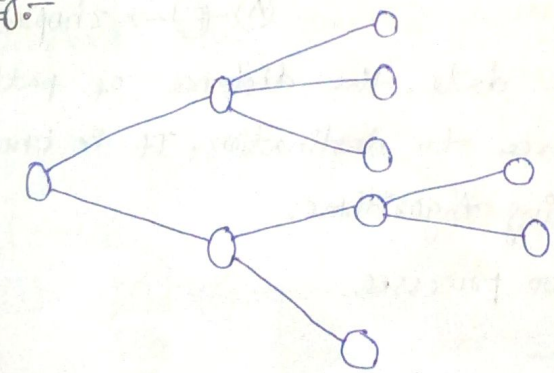
→ A → B → E → G → H → D → 11
2 2 1 4 2

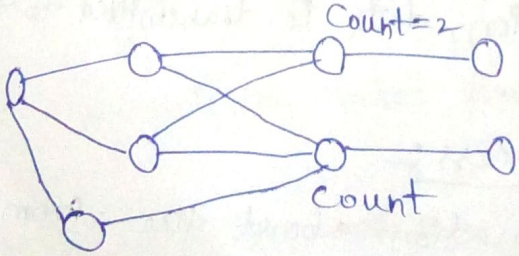
A → B → C → D → 12
2 7 3

A → G → H → D → 12
6 4 2

A → B → E → F → C → D → 12
2 2 2 3 3

Flooding:-





→ Each router will have the count value.

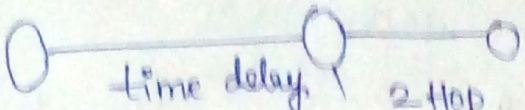
→ If router receives two packets, then count value is decremented by one.

→ If count=3, then decremented by two.

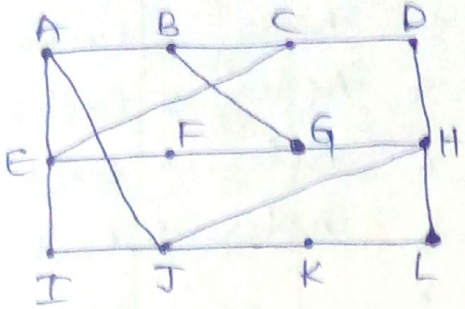
It is called Flooding.

Distance Vector Routing :

→ ECHO Packet :



* The time required to send echo packet from one router to another is called as "Time delay.

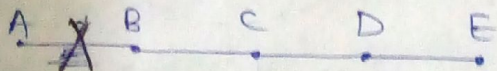


	A	I	H	K	J	
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	[illegible]	[illegible]
K	24	22	22	0	6	K
L	29	33	9	9	15	K
Time delay →	JA 8	JI 10	JH 12	JK 6		

→ new routing table for J R

→ The count to infinity problem occurs in Distance Vector Routing Algorithm.

Date: 06/03



Initial

After 1 Exchange

hop → no. of links

A → hop → zero

B → hop → 1

* metric (or) time delay

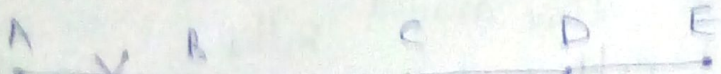
1 2 . . After 2 Exchanges

1 2 3 . After 3 Exchanges

1 2 3 4 After 4 Exchanges

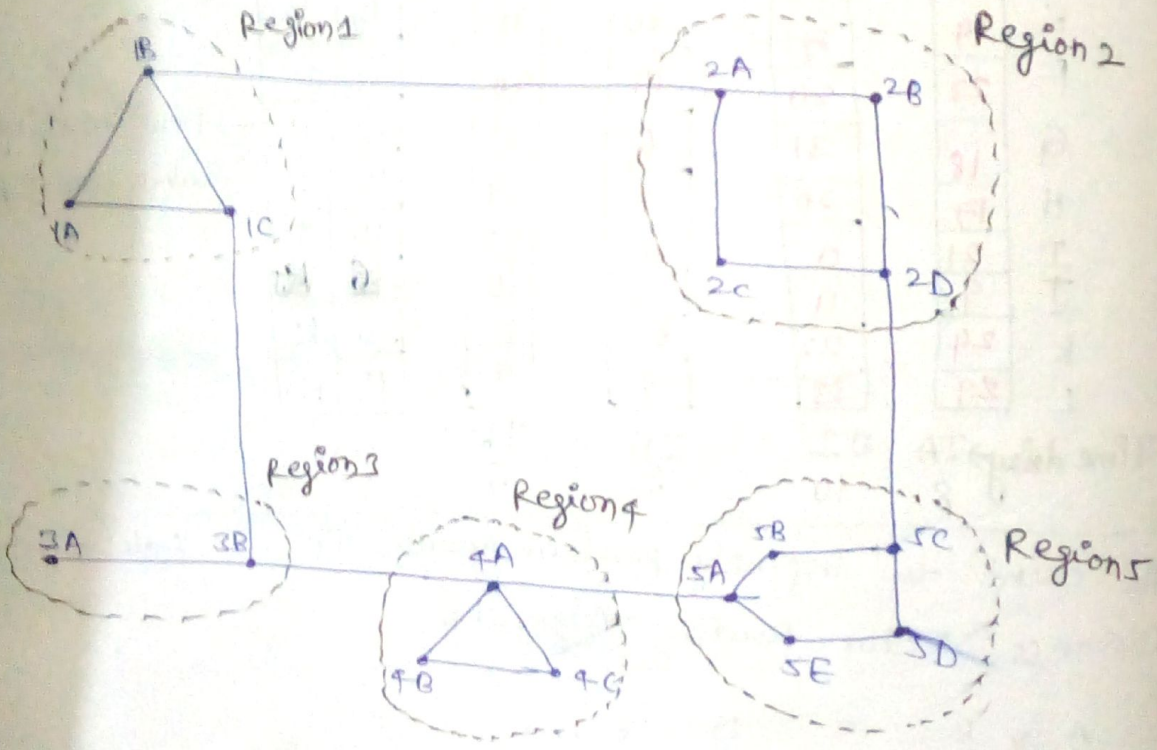
→ If suddenly, A Router is down, then entire va

Drawback is if A router is down, i.e., A router to B connection fails.



	1	2	3	4
Initial				
After 1 Exchange	3	2	3	4
After 2 Exchanges	3	4	3	4
After 3 Exchanges	5	4	5	4
After 4 Exchanges	5	6	5	6
After 5 Exchanges	7	6	7	6
	7	8	7	8

Hierarchical Routing :-



→ Every router connected to all its neighbouring routers.

→ Instead of sending the data packet to router to router. It will be like region to region.

1A	—	—
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

Large memory storage is required.

1A	—	—
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

Memory saving occurs.
* Less memory space required

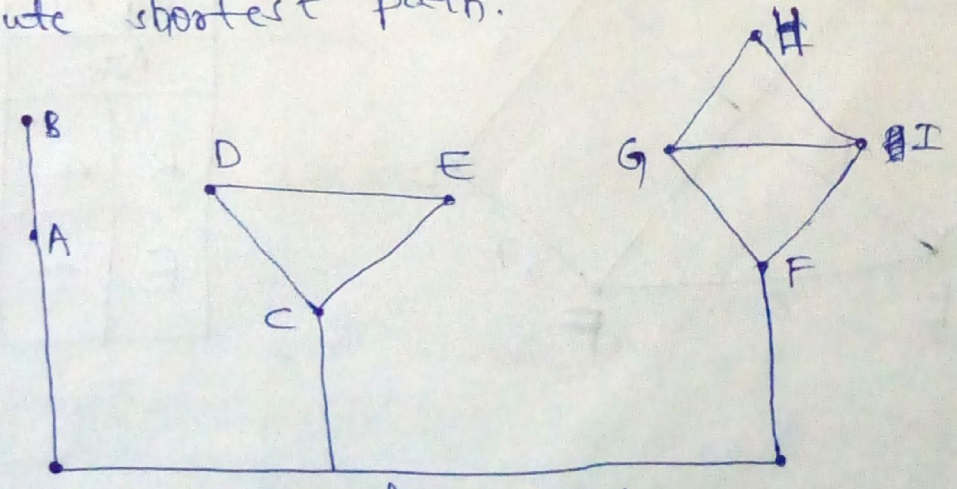
Hierarchial Routing table for 1A Router

Link State Routing:-

Date:-07|03|2

1) To discover all neighbors and learn network addresses of neighbours.
2) Set distance metric (or) Cost of neighbours
3) Construct packets
4) Send packets (or) Receive packets from all of neighbours
5) Compute shortest path.

(1)

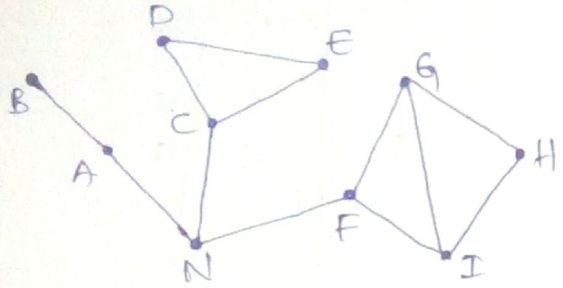


Broadcast LAN

→ Bus topology.

→ If we send Hello packet from A router to C router then F router region also get data packet which is useless Of type. With this "topology size" increases in Broadcast LAN.

→ We can avoid this by creating artificial node which has characteristics same as a router.



N → Artificial node

Graph model

→ A router has to communicate with C router, then path is A → N → C. chooses shortest path.

→ Router first send Hello packet to neighbouring routers and those will respond back.

② Set Distance metric (or) Cost of neighbours:-

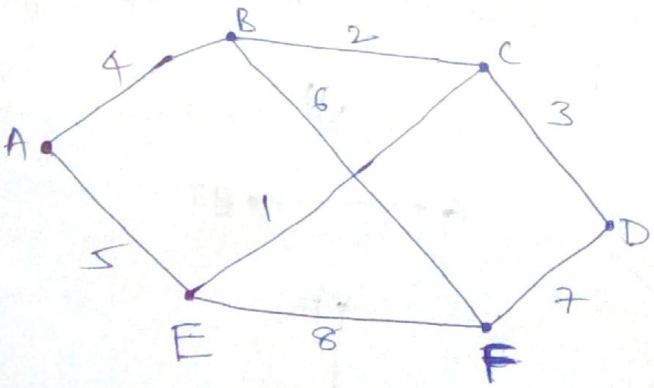
100Mbps

→ Time delay is 10 times more b/w router to router.
→ N/w traffic is high.
→ Cost is high.

1Gbps

→ Time delay is less b/w router to router.
→ N/w traffic is low.
→ Cost is low.

③ Building link state packets:-



A router data packet

A	
Seq no.	
Age	
B	4
E	5

B	
Seq no	
Age	
A	4
C	2
F	6

C	
Seq no	
Age	
B	2
D	3
E	1

D	
Seq no	
Age	
C	3
F	7

E	
Seq no	
Age	
A	5
C	1
F	8

F	
Seq no	
Age	
E	8
D	7
B	6

④ Distribute packets to all of its neighbours :-

→ We are using Flooding algorithm here.

→ The purpose of seq. no. is to remove the duplicate packets.

→ Every time seq no. is incremented by one.

→ Age indicates the how long time data packet is active in a network.

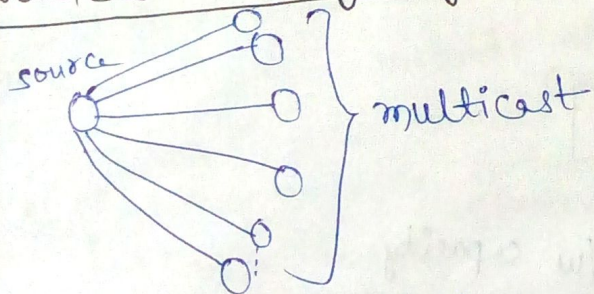
→ Age is decremented by one everytime when data packet transmitted from router to router. A → B.

→ Age → 0 then data packet is inactive.

⑤ Compute shortest path :-

→ shortest path is choosen everytime for the transmission.

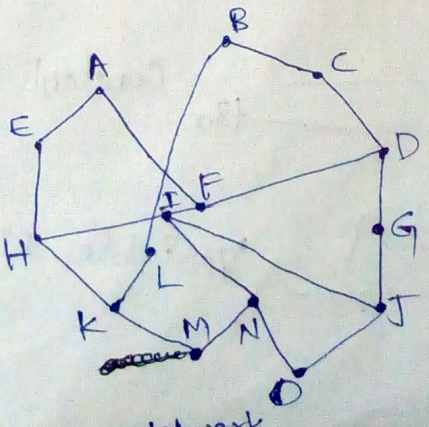
Broadcast Routing Algorithm :-

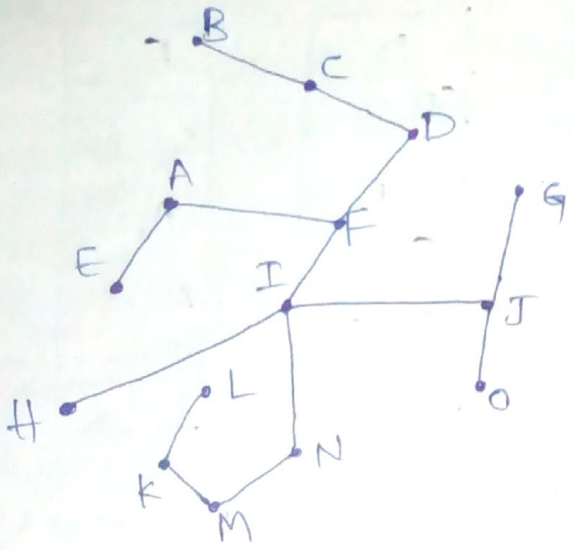


Multidestination

→ Source router has to contain multidestination routers information.

Reverse path forwarding :-

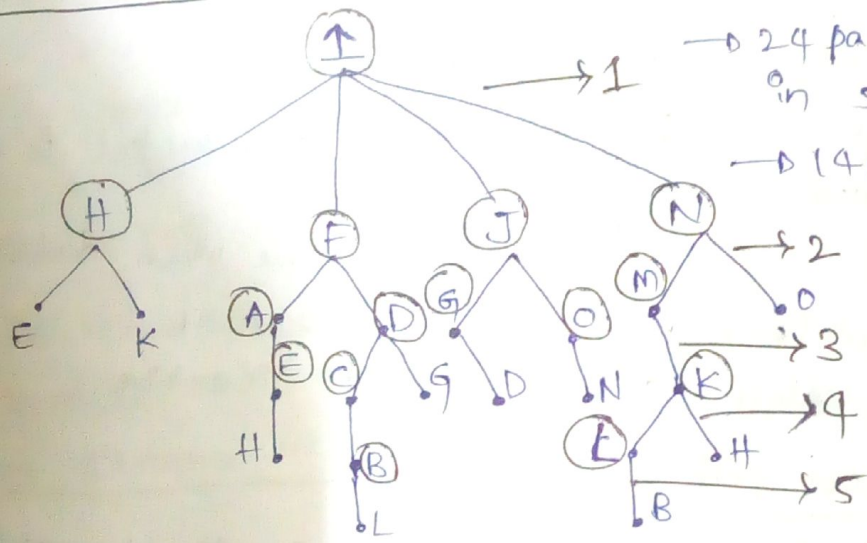




There is no loop.

sink tree

Neighbour routers:-



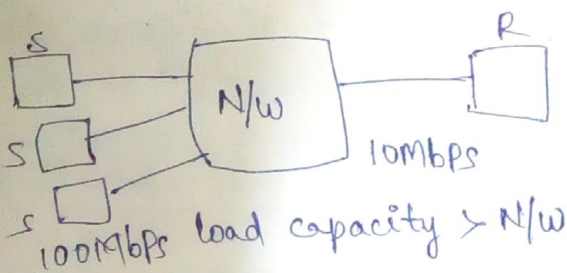
→ Total 5 hops

→ 24 packets forwarded in 5 hops.

→ 14 packets forwarded in 4 hops.

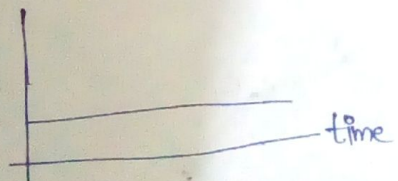
Date:- 11/03/2019

Congestion control :- → Dataflow in a N/w.

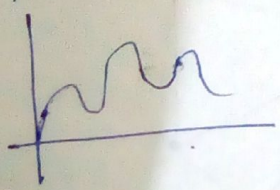


load capacity > N/w capacity

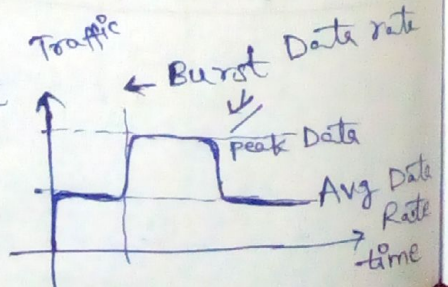
→ There is a traffic in the network.

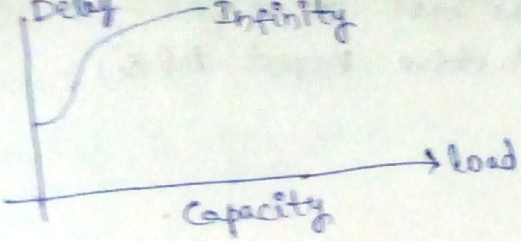


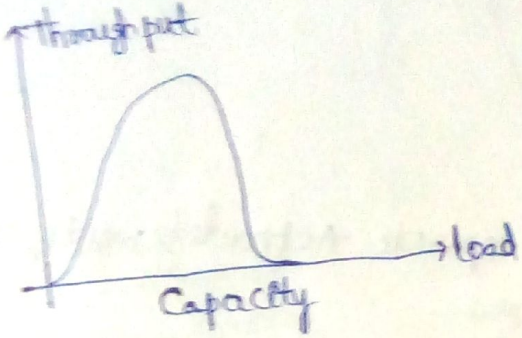
Constant Bit Rate



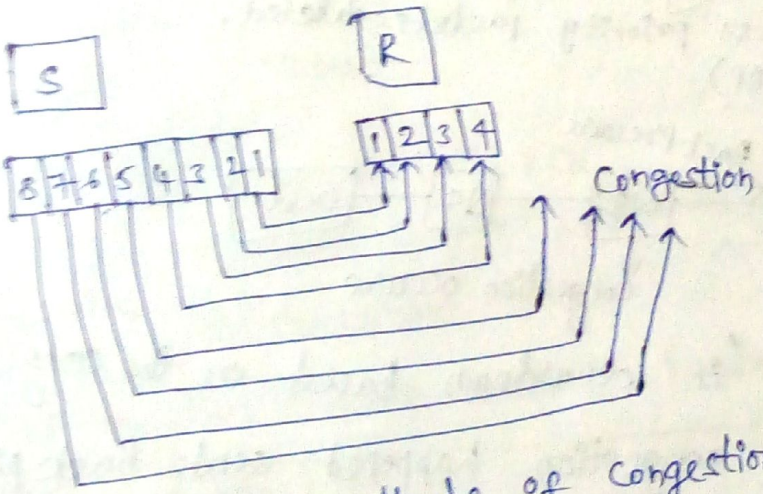
Variable Bit rate







load > Capacity → No. of packet Network can handle.
↓
no. of packets sent to network

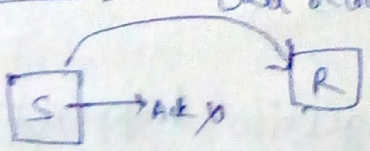


→ There are two methods of congestion control.

1) Open loop Congestion control. (Prevention)
   - 1) Retransmit policy
   - 2) Window policy
   - 3) Acknowledgement policy
   - 4) Discarding policy
2) Closed loop Congestion control. (Removal)
   - 1) Back Pressure
   - 2) Choke packet
   - 3) Implicit signaling
   - 4) Explicit signaling → Forwarding / Backwarding

Admission Control

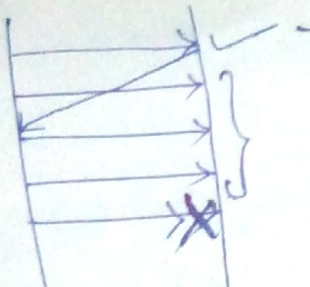
1) Retransmit policy:-



→ N/w traffic occurs.

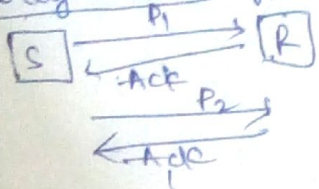
→ To avoid this, Good Retransmit (TCP) policy/method is foll...

2) Window Policy:-



→ choose the best (selective Repeat ARQ)

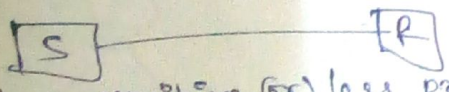
3) Acknowledgement Policy :-



→ Instead of sending separate Acknowledgements, cumulative Acknowledgement is better.

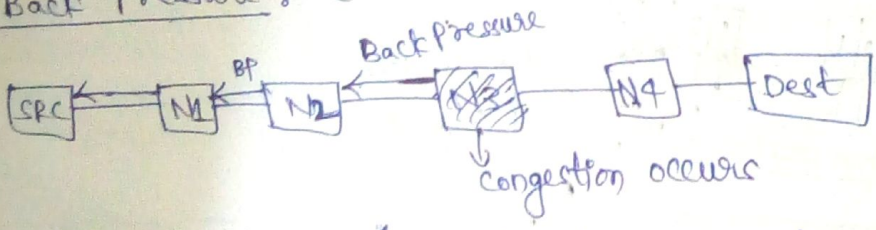
4) Discarding policy :-

→ Some packets can be discarded based on the policy from sender side or receiver side.



→ Less sensitive (or) less priority packets deleted.

1) Back Pressure :- (BP)

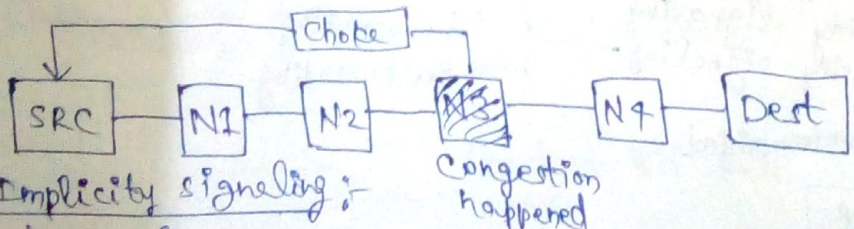


→ Data transmission is slowdown based on the congestion.

→ Node in which congestion happened sends Back pressure to the back.

→ Drawback is BP transmitted in reverse direction from congestion happened node to source.

2) Choke packet :-



1) Implicity signaling :-

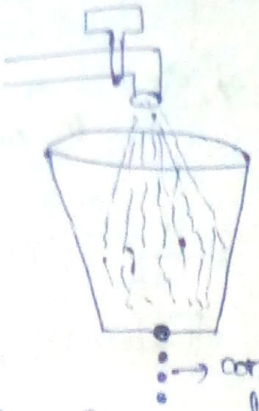
→ There is no communication b/w congestion happened node to source or other devices.

4) Explicit signaling :- Forwarding & backwarding

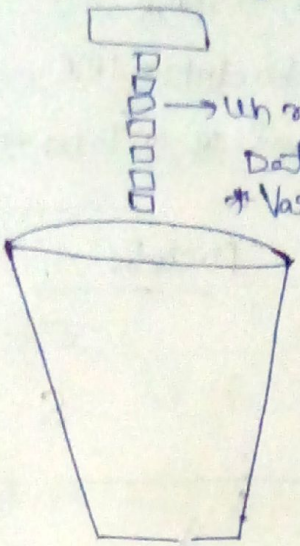
→ Source & destination receives the congestion happened node information and stop the transmission.

→ There are two N/w traffic shaping Algorithms / N/w flow control Algorithms.

1) Leaky Bucket :-



→ constant leaking



→ Un regulated Data packet
* Variable Data Rate

→ There is overflow (Data packets may lost).

→ Regulated Data packet
* Fixed Data Rate

2) Token Bucket :-

For every Δt time one Token is added.

→ Tokens

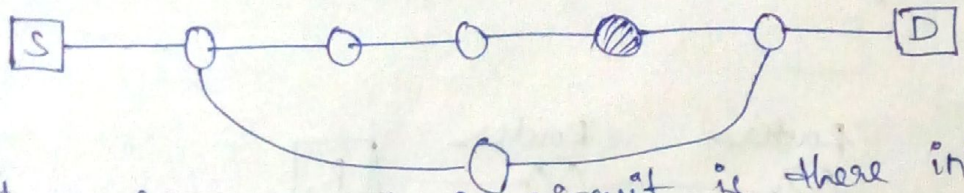
→ Variable Data Rate.
(not constant Data Rate)

→ There is no overflow.

→ Based on the Tokens packets loaded into bucket.

Date :- 13/03/2019

Addmission Control :-



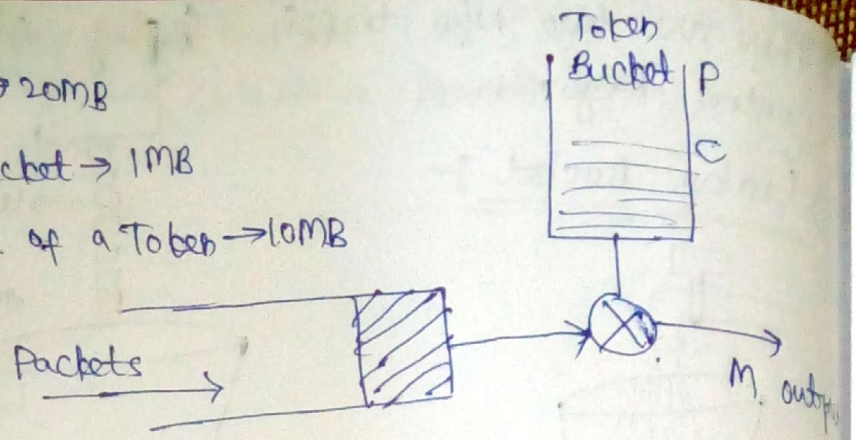
→ There is no virtual circuit is there initially and New virtual circuit (temporary) is created for transmission from source to Destination.

→ For a host machine that uses a Token Bucket algorithm for congestion control. A token bucket has a capacity of 1MB and max output rate is 20MB/sec.

M = Max burst rate → 20MB

C = Capacity of the bucket → 1MB

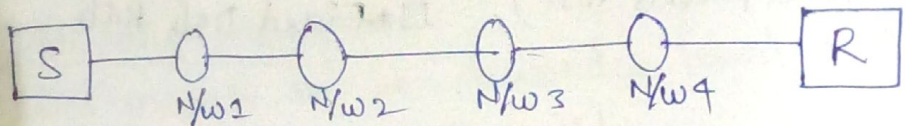
P = Rate of arrival of a Token → 10MB



$$S = \frac{C}{(M-P)} = \frac{1}{20-10} = \frac{1}{10} = 0.1 \text{ sec}$$

Date :- 14/03/2019

InterNetworking :-



→ One network communicate with another network.

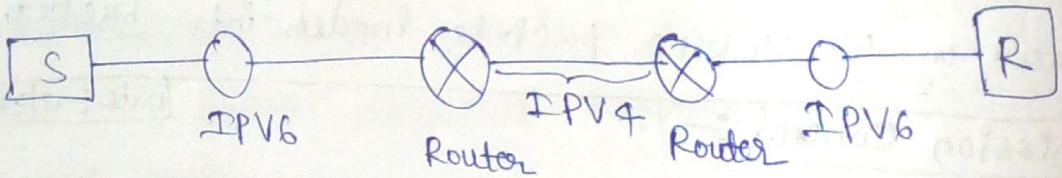
→ N/w may be wired / wireless.

→ Data rate

1) Ethernet N/w → 1500 Bytes → 65,535 Bytes sometimes

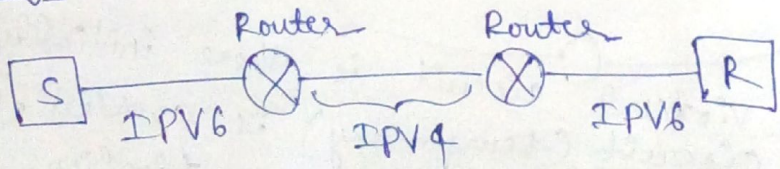
2) Wireless N/w → ≤ 2300 Bytes

→ IPV4 network devices & IPV6 network devices (Internet protocol Version)



→ N/w's may be same or different in Internetworking.

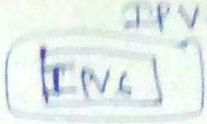
Tunneling :-



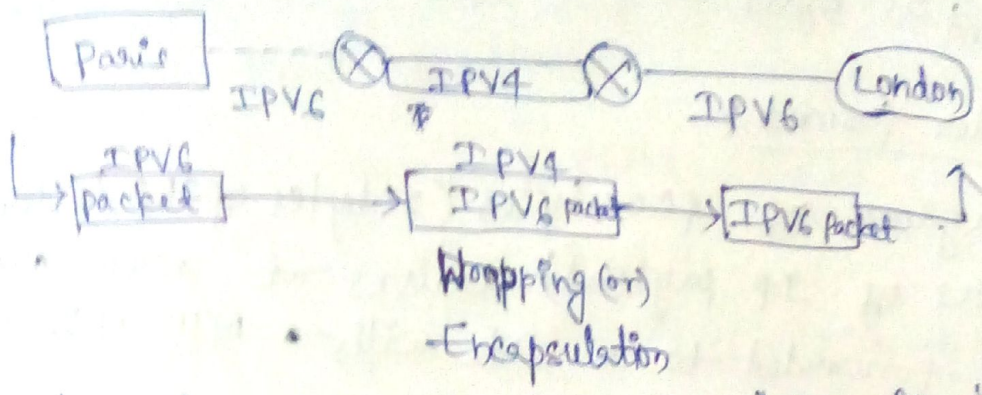
→ IPV4, IPV6 formats are different.

→ Source & Receiver deals with IPV6. But the routers deals with IPV4 packet.

→ Router has to send IPV4 packet but it is receiving IPV6. Then IPV6 packet is dropped into IPV4 packet.



→ And again second router open the IPV4 packet and IPV6 packet is transmitted further.

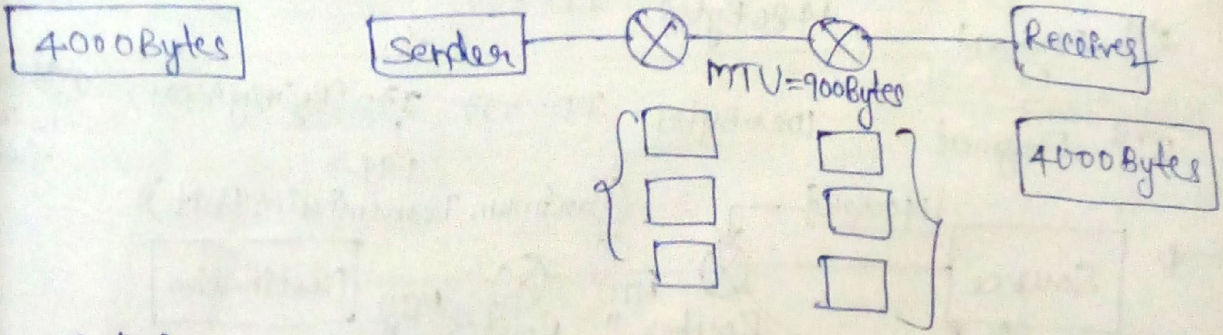


→ This phenomenon is known as "Tunneling".

→ Ex: Car - Truck → Transportation.

→ The main purpose of Tunneling is Conversion of data packets from one format to another format.

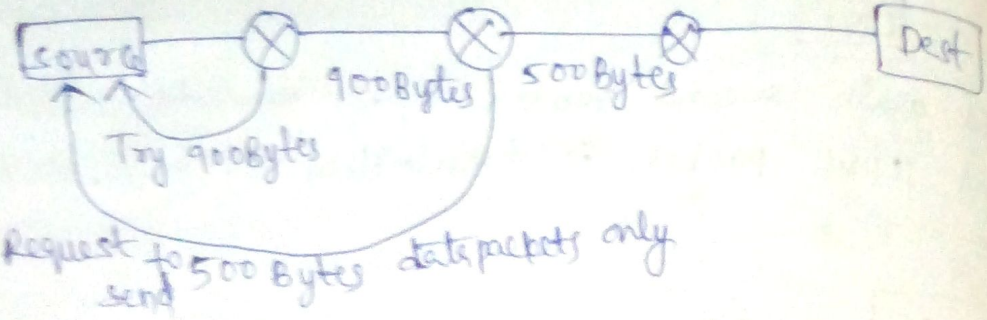
Packet Fragmentation :-



→ Division of data packet into small small fragments is known as "Data Fragmentation".

→ If the network capacity doesn't carry the data packet, then only fragmentation occurs.

→ Again receiver receives the data packet as sent by sender (i.e, single data packet)



IPV4 Header format :-

→ A datagram of 4000 bytes (20bytes of IP header + 3980 bytes of IP payload) arrives at router and must be forwarded to a link with MTU of 1500 bytes.

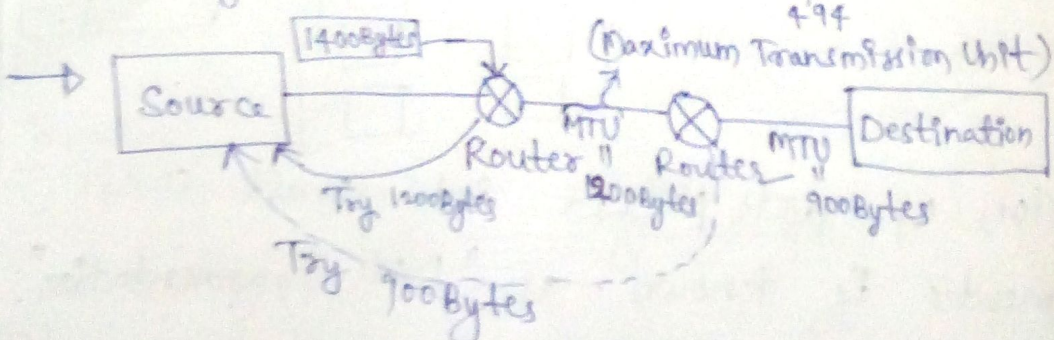
Ans:

3980 bytes

1480 + 20 Bytes header

Each offset capacity = 8bytes

Fragment	Bytes	ID	Offset	Flag
1st Fragment	1480Bytes of data	ID=777	0 (Beginning of Byte) 184	1 (MTU was fragmented)
2nd Fragment	1480Bytes	ID=777	185 (Beginning offset) 369	1
3rd Fragment	1020Bytes	ID=777	370 (Beginning offset) 494	0 (No more fragments)



* In only two fragments, data is transmitted.

↳ Internet service provides (or) Autonomus system connects two different networks for communication with each other.

IPV4 Header formatting :-

IPV4
↓
32 bits

2^{32} = 4 billion devices can be connected.

→ But not sufficient for present world.

So we move to IPV6.

↓

32 × 4 = 128 bits

↓

Hexadecimal digits

→ IPV4 is represented in dotted decimal / binary format.

Ex: 172.10.1.25 } IPV4

0000:0000:1234:—:—:—:—:— } IPV6

Total 8 fields. 8 × 16 = 128 bits

4 Hexadecimal digits

4 × 4 = 16.

2^{128} addresses available in IPV6.

→ Maximum length in IPV4 is 2^{32} = 65,535 Bytes.

Date :- 26/03/2019

IPV4 header :-

32 bits

VER (4)	HLEN (4)	Services (8)	Total length (16)
Identification (16)		Flags 3	Fragmentation offset (13)
TTL (8)	Protocol 8	Header checksum (16)	
Source IP Address 32 bits			
Destination IP Address 32 bits			
Option			

VERSION :- IPV4 (or) IPV6

HLEN :- Min 20 Bytes (Variable header length)

Max header length → 60 Bytes.

→ If value is 5 (5 × 4 = 20 Bytes)

Max header length → 15 × 4 Bytes = 60 Bytes

rvices:-

- First 3 bits indicates the priority.
- lowest priority data is discarded when conge... happens.

- Next 4 bits indicates the Type of service (TOS).

Code	TOS
0000	Normal
0001	
0010	→ Minimize cost
0011	→ Maximize throughput

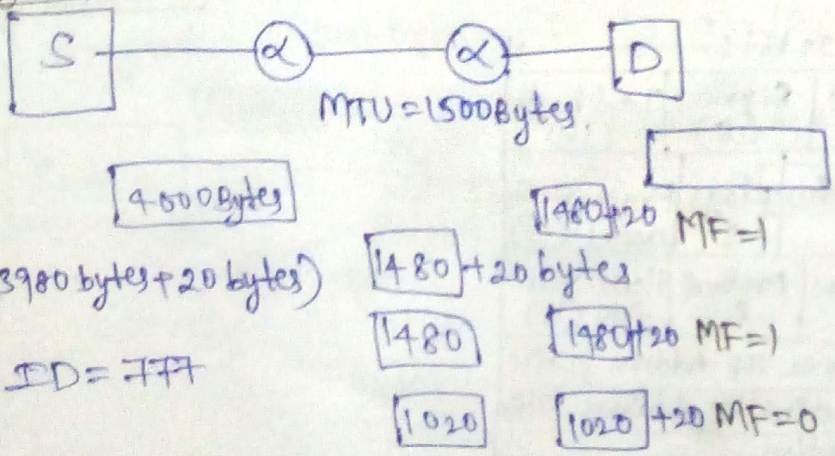
Total length:-

- $2^{16} - 1 = 65,535$ Bytes
- Data length = Total length − Header length

Identification:-

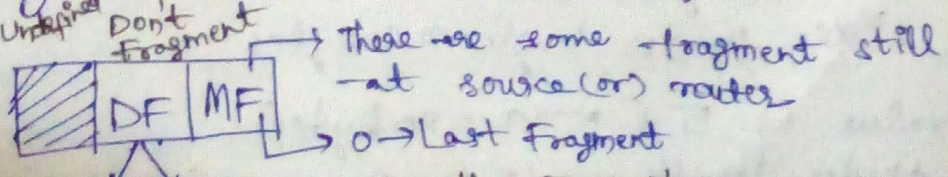
- Source IP address & Identification is unique.
- For identifying the fragments ID No. is required.

Fragmentation:-



Flags:- → There are 3 flags.

→ flag bits required in the fragmentation.



Fragment offset :-

→ $\frac{1480}{8} = 185$

0 (Beginning offset)

184 (Last offset)

Time to live :- (TTL) (Time stamp)

→ Time stamp value decremented by one.

→ whenever the TTL reaches to zero (0), router will discard the datagram.

Protocol :-

Transport	TCP SCTP UDP
N/w Layer	ICMP IP RARP ARP

code	protocol
1	ICMP
6	TCP
17	UDP

IPV4	Data

Option :- (40Bytes)

→ For Testing & Debugging purposes.

→ IPV5 is designed only for Testing purpose.

IPV6 :-

→ In IPV4, 2^{32} devices will be connected with the n/w.

" 4 billion

IPV4

* 2^{32} = 4 billion.

* 168 . 10 . 1 . 23

↓ 8bits ↓ 8bits 8bits 8bits

IPV6

* 128 bits. 2^{128} devices

* FBD1 : 0000 : 0000 : 0000 : 1230 : 0012 : 0003 : 4BAD

↓ 4 Hexa decimal ↓ 4 4 4 4 4 4 4

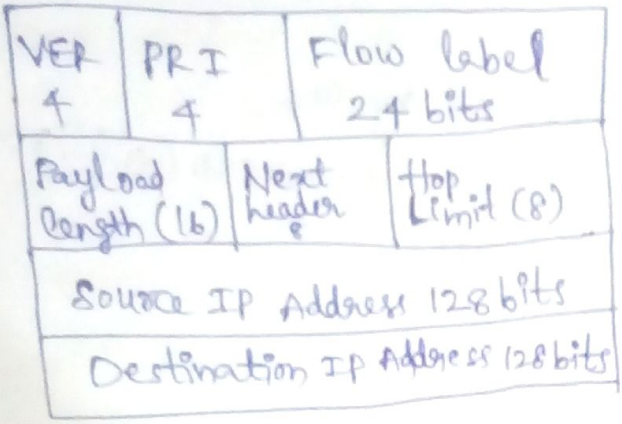
16bits 16 16 16 16 16 16 16

→ 32 Hexadecimal digits = 128 bits.

→ Omitting left zeros :-

FBD1:0:0:0:1230:12:3:4BAO.

IPV6 Header :-



VERSION :- IPV6

PRIORITY :- If congestion happens, lowest priority Datagrams discarded first.

→ IPV6 Fixed header length → 40 Bytes

Flow label :- Data flow
(or)
flow of packets in the network.

Payload length :- (Original data length)

→ $2^{16} = 65,356$ Bytes

Next header :-

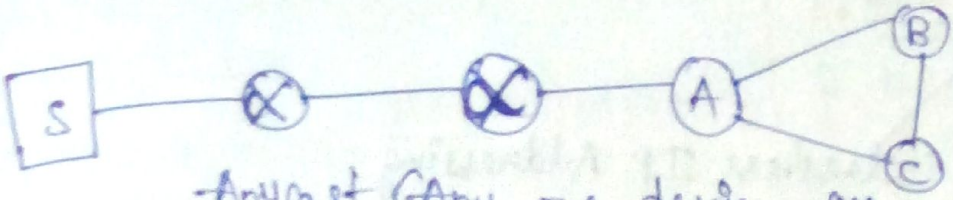
→ Protocol

→ Next header is added in a IPV6.

Code	Next header
1	ICMP
6	TCP
17	UDP

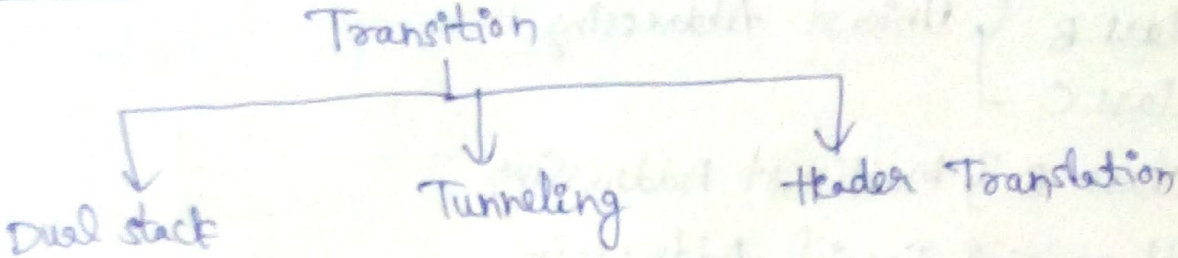
Hop limit :- (Time to live) same as

→ Time stamp.

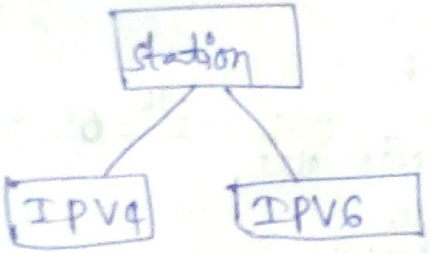


Anycast (Any one device will receive data)

Transition:-

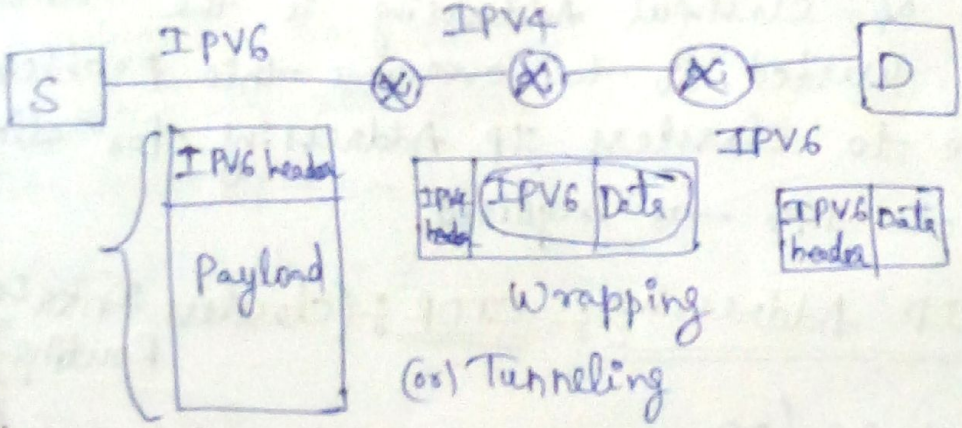


Dual stack:-

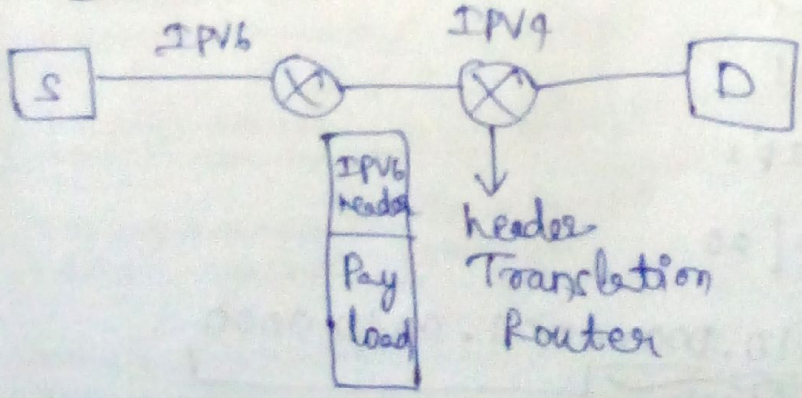


→ consists of both IPV4 & IPV6 protocols.

Tunneling:-



Header Translation:-



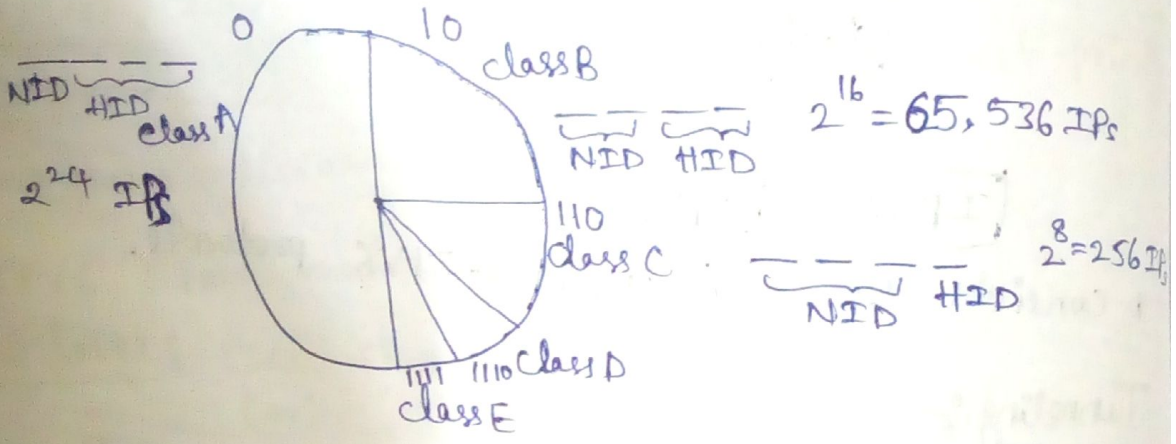
→ Header changes from IPV6 to IPV4.

Addressing :-

classful classless IP Addressing

class A
class B
class C
} Unicast Addressing

class D } Multicast Addressing
class E } Special Addressing



→ Drawback of classful Addressing is the remaining IPs are wasted. So to overcome this problem we move to classless IP Addressing for allocation of exact IPs as required.

// Classless IP Addressing :- CIDR :- (classless Inter Domain Routing)

→ 168.10.1.32 / 20

NID = 20 bits
HID = 12 bits
2^{12} = 4096 IP's

10.20.30.0 / 20

→ SM → 168.10.0000 0001.0010 0000

20 bits HID 12 bits

NID (or) 255.255.224.0

1 HID → 168.10.0.1

2 HID → 168.10.0.2

Lastmost → 168.10.15.254 (1111 / 1111111 0)
HID

Broadcast → 168.10.15.255

A	B	C	D	E
0 ↓ 127	128 ↓ 191	192 ↓ 223	224 ↓ 239	240 ↓ 255

Subnetting :-

→ classful :-

200.1.2.0

NID HID

Class C

$2^8 = 256$

SID

0 0 0 0 0 0 0 0 → 200.1.2.0

0 0 0 0 0 0 0 1 → 200.1.2.1

0 1 1 1 1 1 1 0 → 200.1.2.126

0 1 1 1 1 1 1 1 → 200.1.2.127

S_1	S_2
0 to 127	128 to 255

1 0 0 0 0 0 0 0 → 200.1.2.128

1 0 0 0 0 0 0 1 → 200.1.2.129 → 1 HID

1 1 1 1 1 1 1 0 → 200.1.2.254 → Last HID

1 1 1 1 1 1 1 1 → 200.1.2.255 → Broadcast Ip

→ Subnet ID → 200.1.2.0

1 HID → 200.1.2.1

Last → 200.1.2.126

Broadcast Ip → 200.1.2.127

→ $2^7 - 2 = 126$ Ip's

→ $2^7 - 2 = 126$ Ip's

→ 252 Ip's

→ Subnet Mask (SM) → 255.255.255.128 (NID)

→ 200.1.2.0.

SID

00 0 0 0 0 0 0 0 → 200.1.2.0

00 0 0 0 0 0 0 1 → 200.1.2.1

00 1 1 1 1 1 0 → 200.1.2.62

00 1 1 1 1 1 1 → 200.1.2.63

01 0 1 0 0 0 0 0 0 → 200.1.2.64

01 0 1 0 0 0 0 1 → 200.1.2.65

01 1 1 1 1 1 0 → 200.1.2.126

01 1 1 1 1 1 1 → 200.1.2.127

S_1	S_4
0 to 63	192-255
S_2	S_3
64 to 127	128 to 191

11 1 0 0 0 0 0 0 → 200.12.192

11 1 0 0 0 0 0 1 → 200.1.2.193

11 1 1 1 1 1 0 → 200.1.2.254

11 1 1 1 1 1 1 → 200.1.2.255

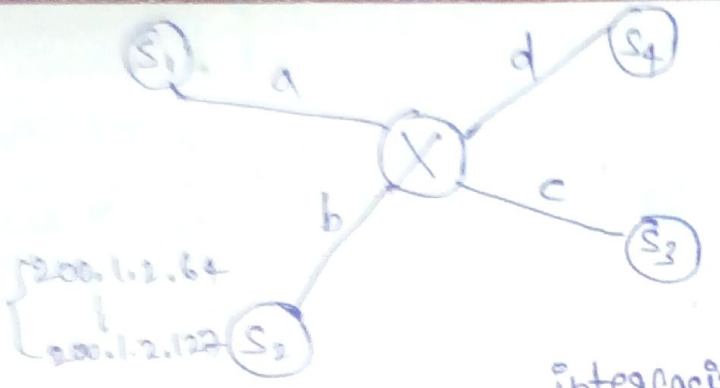
10.0 0 0 0 0 0 0 → 200.1.2.128

10 0 0 0 0 0 1 → 200.1.2.129

10 1 1 1 1 1 0 → 200.1.2.190

10 1 1 1 1 1 1 → 200.1.2.191

→



NID	SM	interfacing
200.1.2.0	255.255.255.192	a
200.1.2.64	"	b
200.1.2.128	"	c
200.1.2.192	"	d

→ 200.1.2.108 IP

SM → 11111111 11111111 11111111 11000000

IP → 11001000 00000001 00000010 01101100

200. 1. 2. 01000000

64

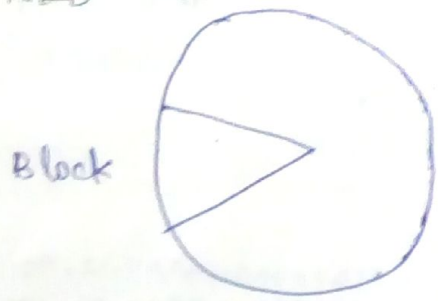
SID → 200.1.2.64

- CIDR classless subnetting :-

Date: 27/03/20

→ 10.3.2.0/23 } NID → 23, HID → 9

NID HID



1) The all Ip addresses should be contiguous.
2) Block size should be the power of 2 i.e. 2^n.
3) The first Ip address should be devisible by Block size.

→ classless SM → 255.255.254.0

10.3.00000010.00000000

SID → 10.3.1.0. 10.00000001

H.11111110

1st Host → 10.3.1.1

⋮

Last → 10.3.3.254

Broadcast → 10.3.3.255

→ 10.3.2.0/29

NID → 29

HID → 3

SM → 255.255.255.248 (11111000)

(24 bits)

SID → 10.3.2.248

1st Host → 10.3.2.249 → 11111001

10.3.2.250 → 010

10.3.2.251 → 011

10.3.2.252 → 100

10.3.2.253 → 101

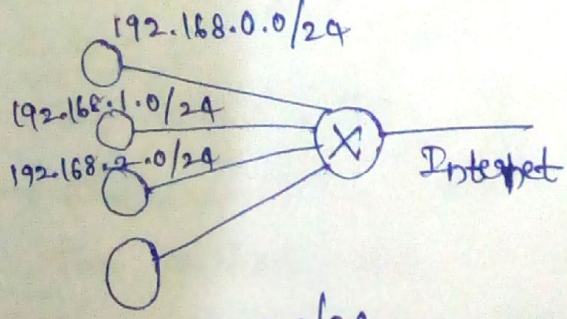
10.3.2.254 → 110

Last → 10.3.2.254

Broadcast → 10.3.2.255 → 111

IP

Supernetting :-



→ 192.168.3.0/24

→ SM → 255.255.255.0

SID_1 → 11000000.10101000.00000000.00000000.

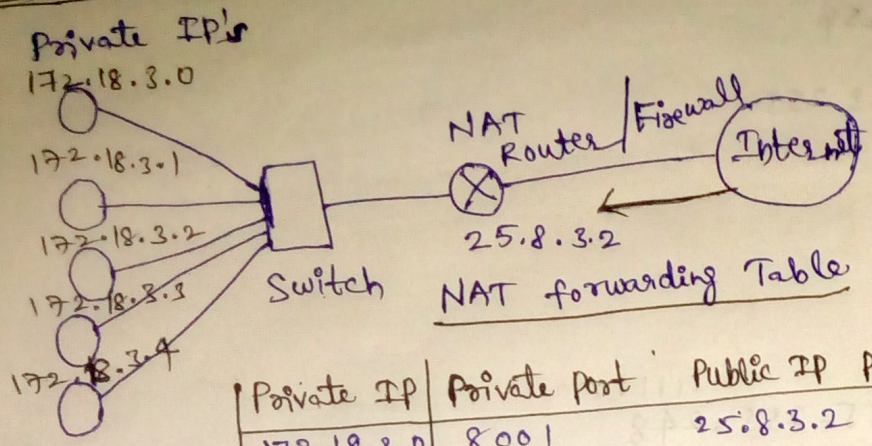
SID_2 → 11000000.10101000.00000001.00000000.

SID_3 → 11000000.10101000.00000010.00000000.

SID_4 → 11000000.10101000.00000011.00000000.

Supernet ID → 192.168.0.0 (AND operation)

N/w address Translation :- (NAT)



NAT forwarding Table

Private IP	Private port	Public IP	Public port	Protocol
172.18.3.0	8001	25.8.3.2	80	TCP
172.18.3.1	8002	25.8.3.2	80	TCP
172.18.3.2	8003	25.8.3.2	80	TCP
172.18.3.3	8004	25.8.3.2	80	TCP

→ Only public IP is visible to the N/w & the private IP's are not visible to the N/w.
→ Each system will generate separate port no.'s.
→ IANA → Internet Assigned Numbers Authority.
↳ produce port numbers.
→ There are 3 different port numbers.

$2^{16} \rightarrow 65,536$

* Well knowing
  0 – 1023
* Registered
  1024 – 49,151
* 49,152 → 65,535.

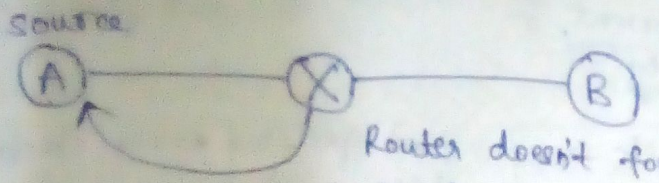
ICMP :- → Reports the error but doesn't correct error.
[Internet Control Message Protocol]

- Error reporting
  - ↳ Destination Unreachable
  - ↳ Source Quench
  - ↳ Time exceeded
  - ↳ Parameter Problem
  - ↳ Redirection
- Query message
  - ↳ echo request & reply
  - ↳ Time stamp request & reply.

Error reporting

Date: 28/03/2019

1) Destination Unreachable:

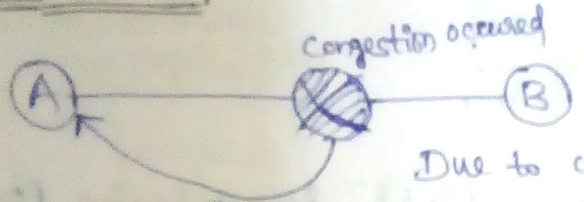


Router doesn't found route to B.

Destination Unreachable

Packet discarded.

2) Source Quench:

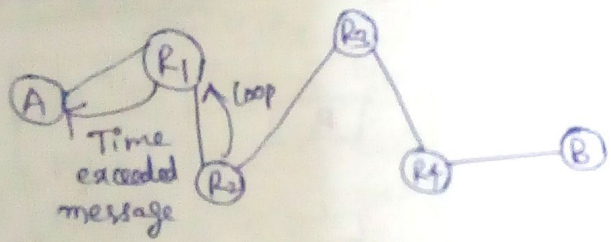


Due to congestion packet discarded from the router.

Source Quench

3) Time exceeded:

→ There are two codes.
1) Code '0' 2) Code '1'

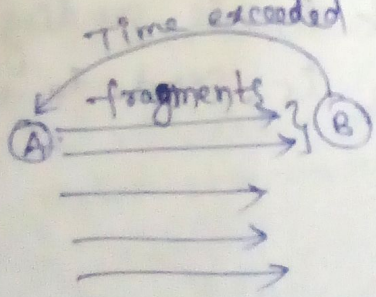


Code '0':

→ In code '0', TTL is an integer.

→ Whenever TTL (loops occurs) reaches to zero, the router will discard the packet and it will send the Time exceeded message to the source device.

Code '1':



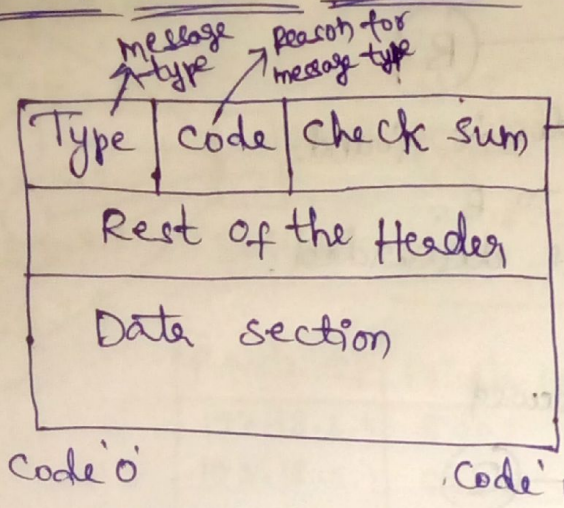
→ only two fragments are received at destination in set time.

→ After time over, Received fragments are discarded.

(i.e., it has to receive 5 fragments but received only 2 fragments)

4) Parameter problem :-

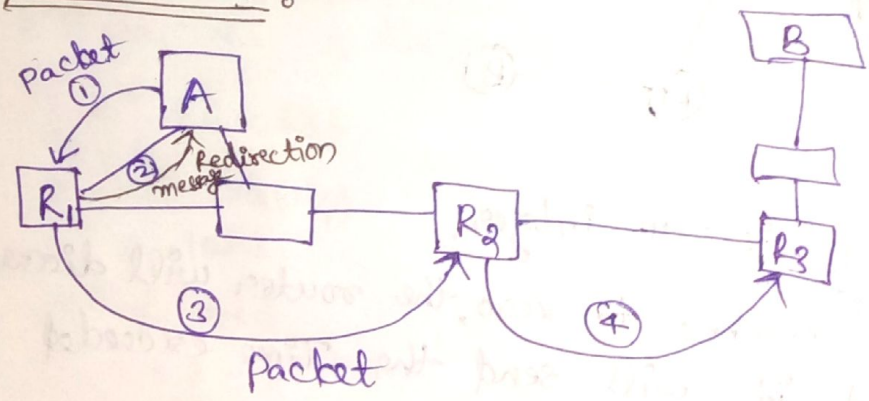
→ ICMP packet format :-



→ find specified error checking bits packet.

→ If one of the field is missing (or) error occured in one of the field of ICMP packet formet then parameter problem message is send by code '0' & code '1' respectively to source device.

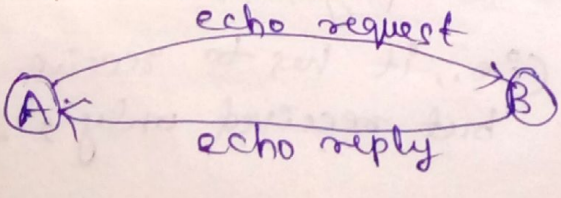
5) Redirection :-



→ A have path to send data directly to the R_2 router but it sending to R_1 & R_1 is sending to R_2. hence Redirection message is send to the A.

Query messages :-

① Echo request and reply :-

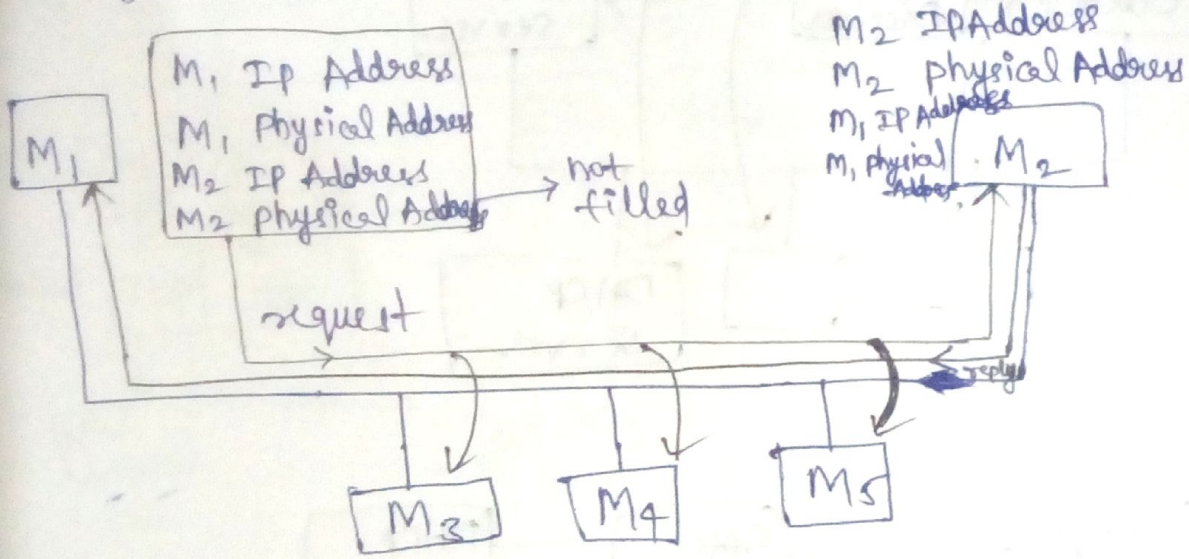


② Time stamp request & reply :-

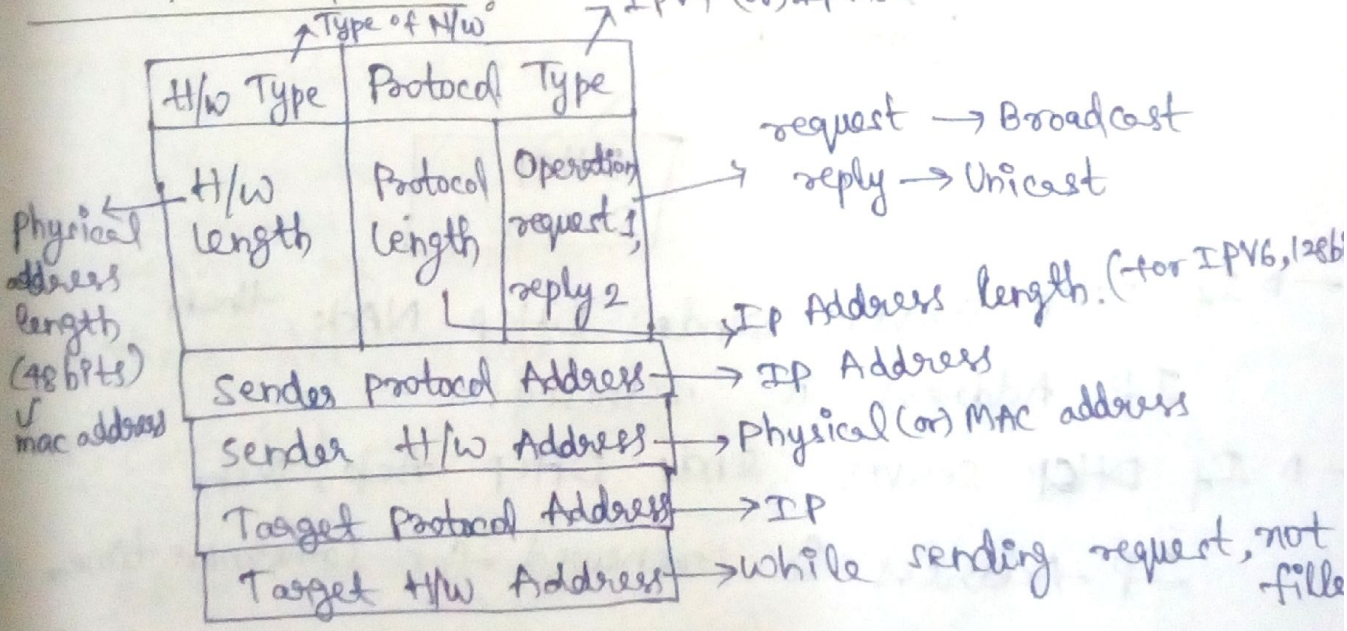
→ Round trip time is determined.

Address Resolution protocol :- (ARP)

→ Logical Address is mapped with physical Address.

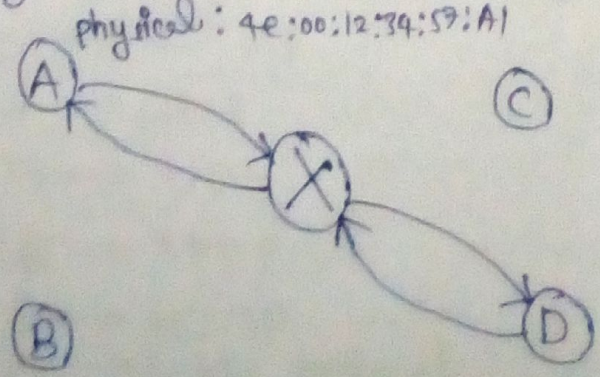


ARP packet format :-



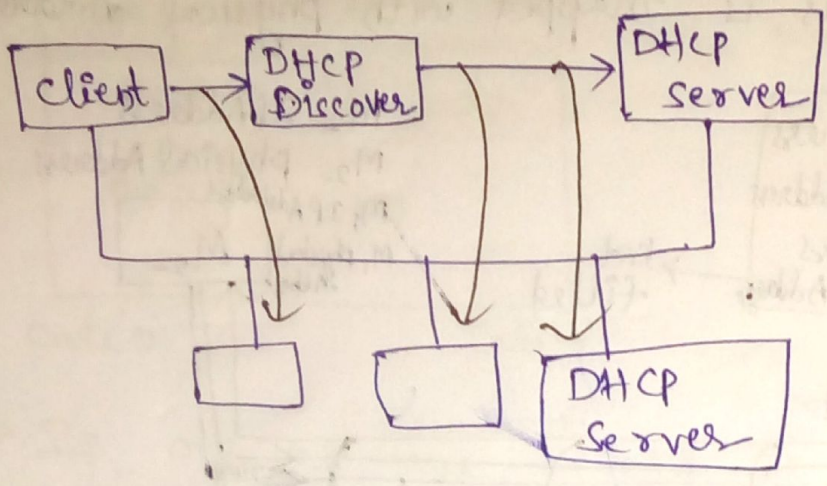
RARP :- (Reverse Address Resolution protocol)

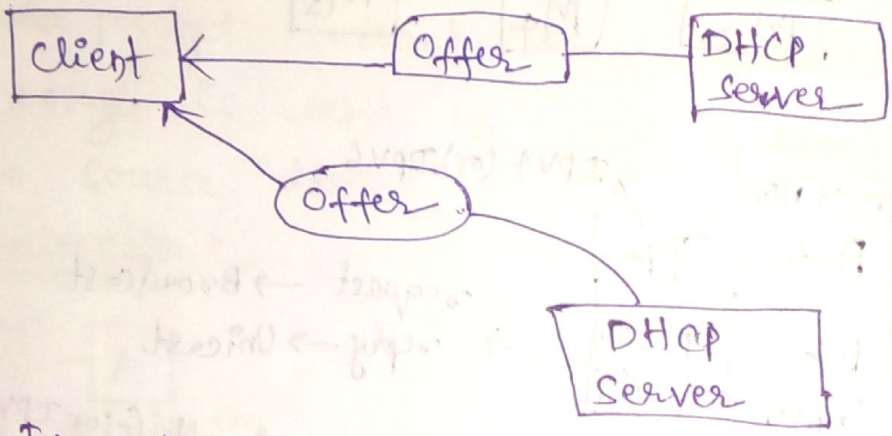
→ physical Address → Logical Address Mapping.



DHCP :- [Dynamic Host Configuration Protocol]
↓
device.

→ IPs added automatically (Dynamically).

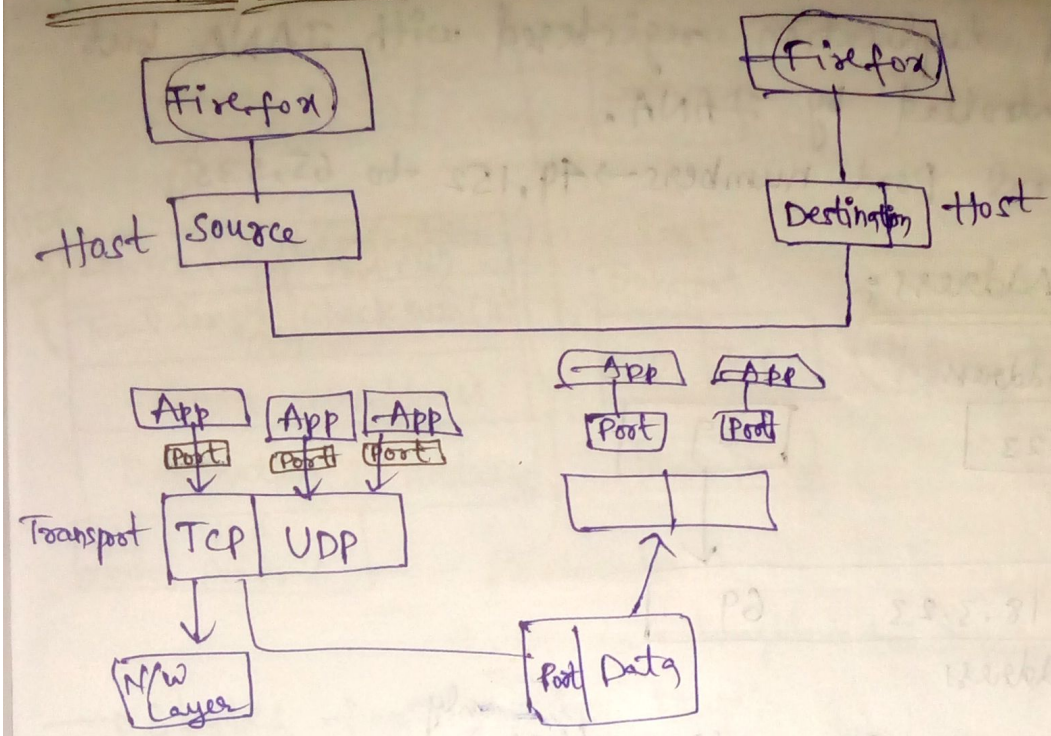




→ If DHCP server sends DHCP NAck, then IP Address is released.

→ If DHCP server sends DHCP Ack, then IP Address is renewed for some more time.

Transport Layer :- → process-process communication.



→ Each process (Application) has port no.

→ Port Address → 16 bit Address.

2^{16} = 0 to 65,535

↳ standard port no.'s
↳ given by IANA.

→ There are 3 types.

1) Controlled and Assigned by IANA → 0-1023
Well known port numbers.

Port No.

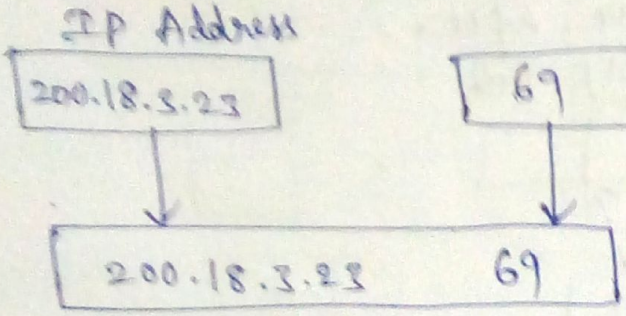
* 20 → FTP Data connection
21 → FTP control connection
23 → Telnet
1 → ICMP
6 → TCP
17 → UDP
25 → SMTP
80 → HTTP
513 → login protocol

2) Registered port numbers → 1024 to 49,151

* To avoid duplication registered with IANA but not controlled by IANA.

3) Ephemeral port numbers → 49,152 to 65,535

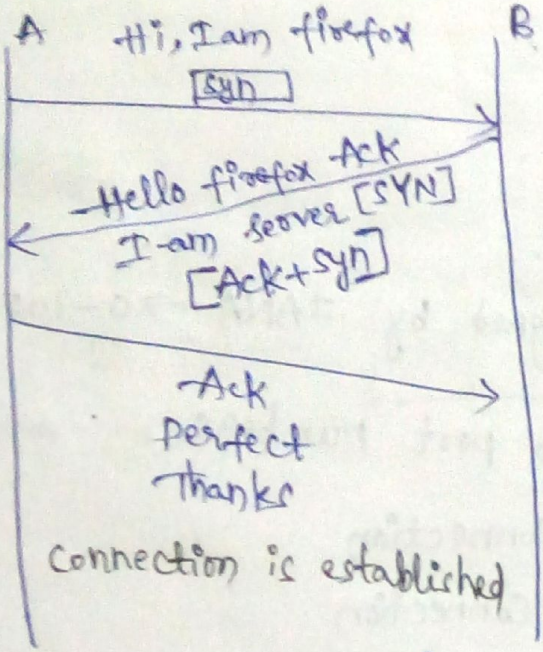
Socket Address:



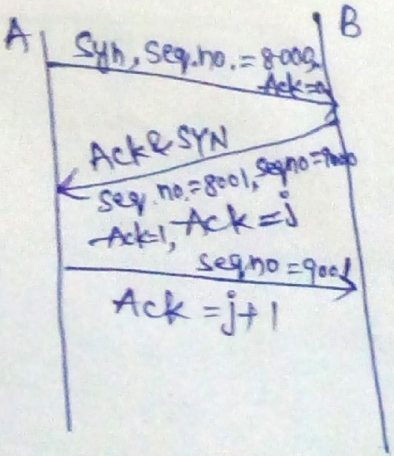
Socket Address

→ If socket is active, then only transmission occurs.

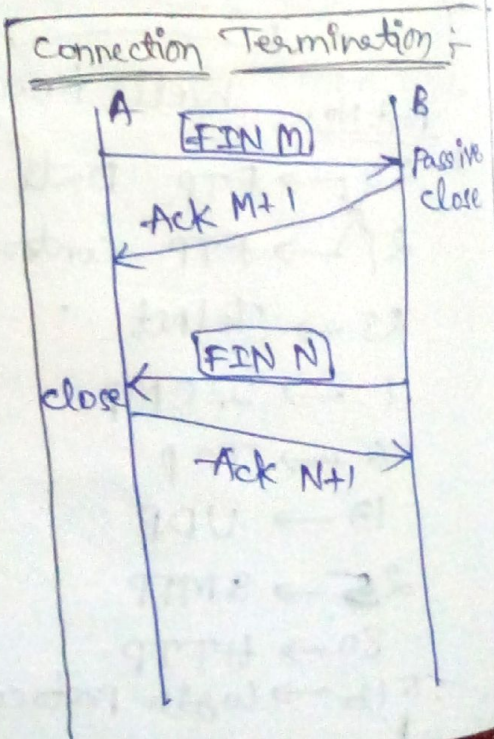
Connection Establish:



full duplex

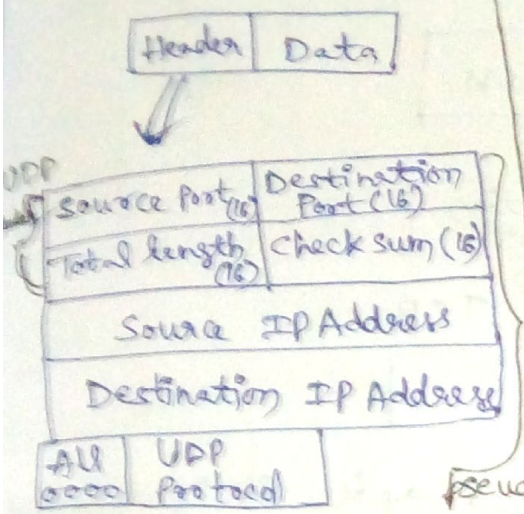


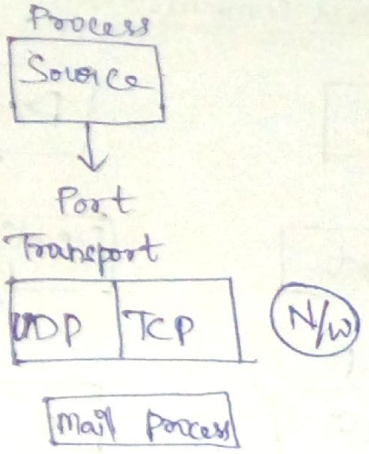
Connection Termination:

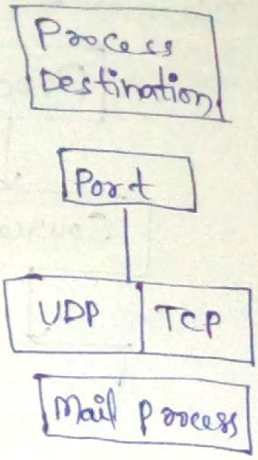


Date :- 01/09/2019

UDP message format :-







Total length = 2

* UDP data length = total length - Header length

→ 8 bytes for Header length.

→ UDP Header is Fixed.

→ Error checking sum.

~~Connection~~ UnReliable

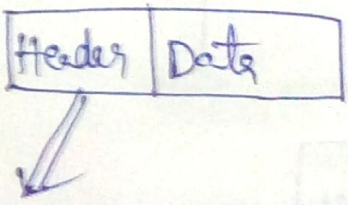
Connectionless

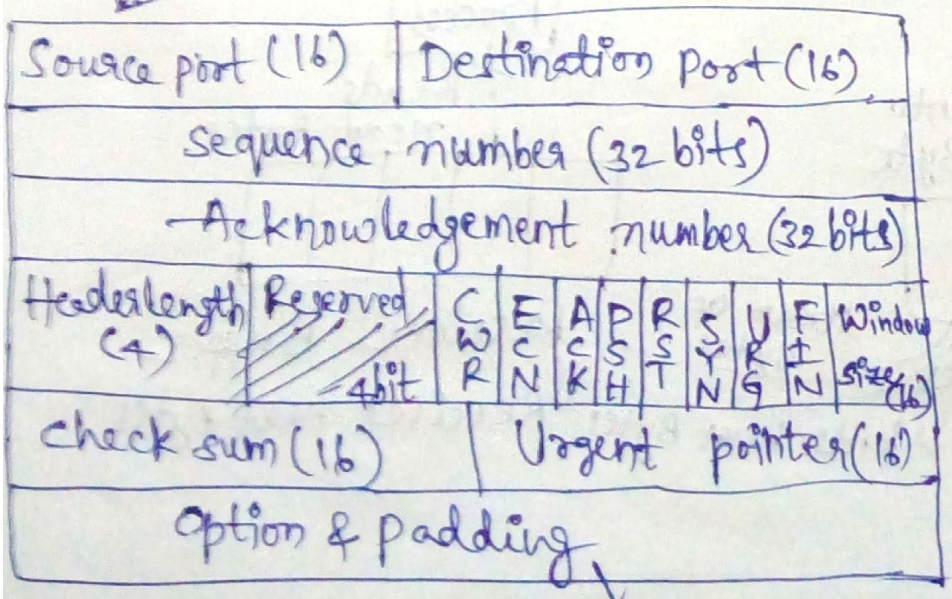
→ No connection is established before sending data.

No Ack

Seq number

TCP Header :-

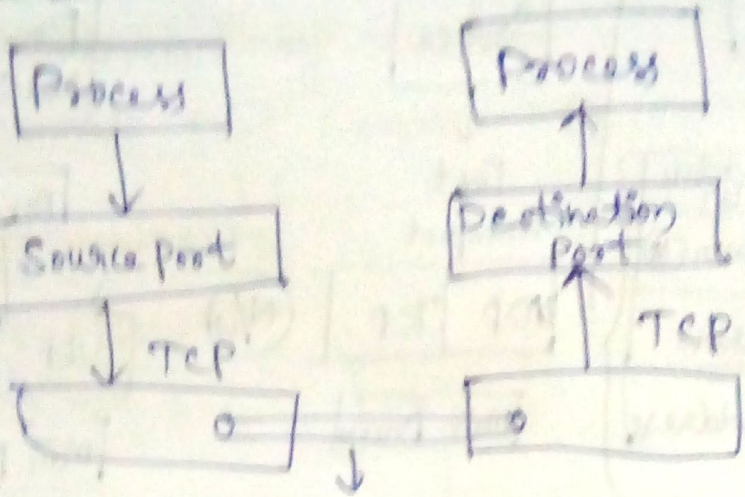




max is 40 Bytes

Service :-

1) Process to process communication



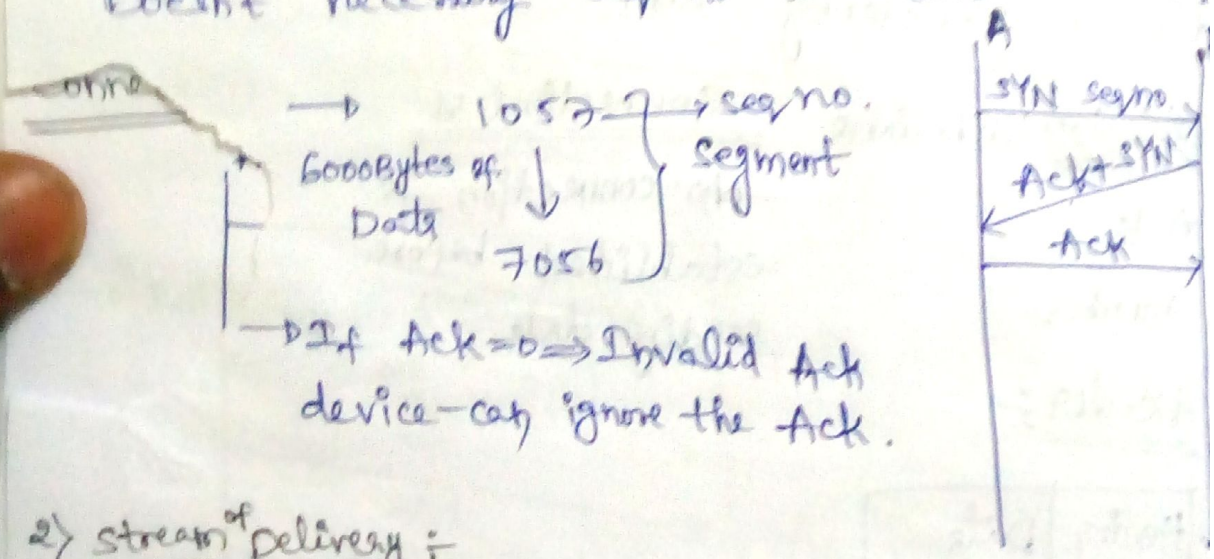
Data is transferred after it is data

Sequence number :-

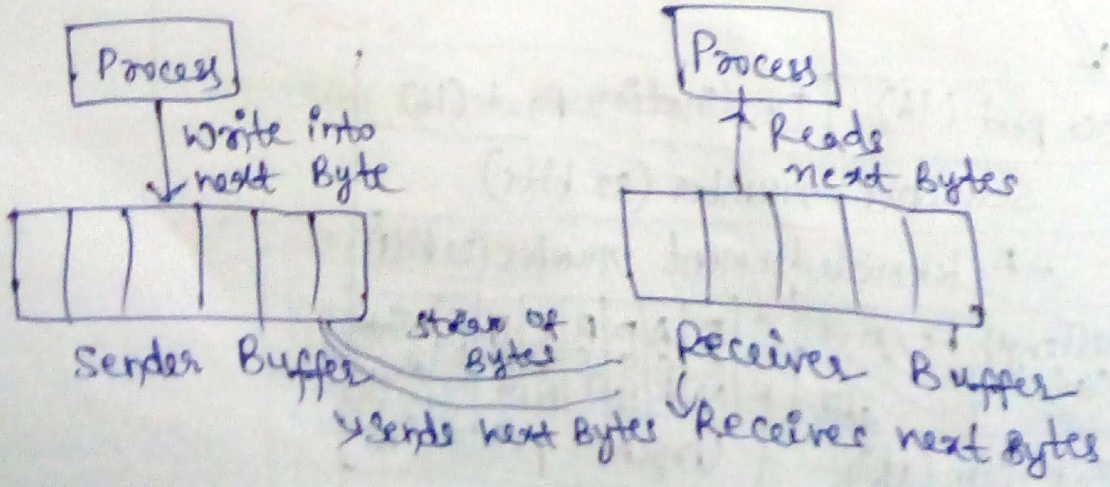
→ Seq number range ⇒ 0 to $2^{32}-1$

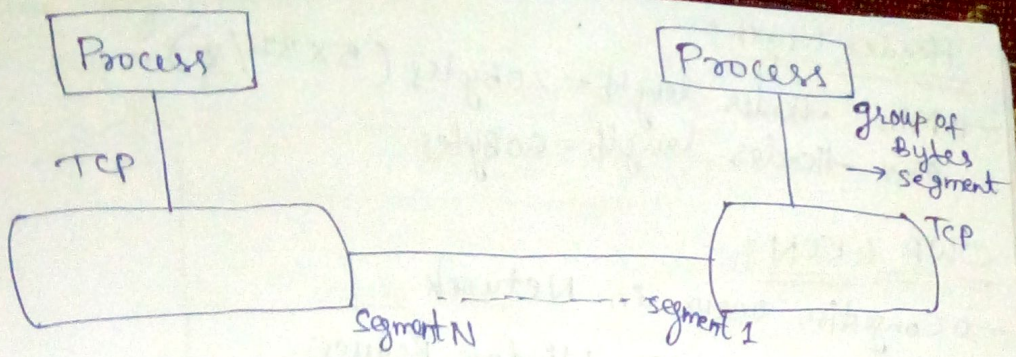
Random number

Doesn't necessary seq no to start with 0.



2) Stream of Delivery :-





Ex: Source want to Data transmit 5000 Bytes, in five segments and each segment size 1000 Bytes.

* Segment 1 ⟶ seq no ⇒ (1000 to 11000)

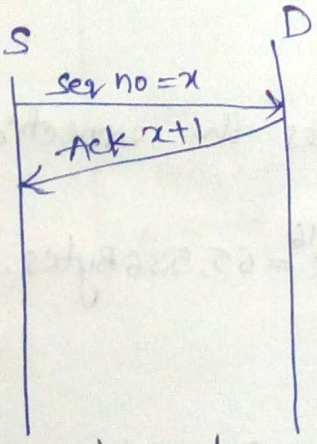
⟵ Ack = 1100

→ Ack flag is set to 1 after the first SYN seq no. received at destination and again Ack (valid) sent back to source side.

→ Push flag is set to 1 when the data transmission takes place.

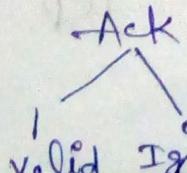
* segment 2 ⟶ seq no ⇒ 1100 (1001 to 12000)

segment 5 ⟶ seq no ⇒ 14001 (14001 to 15000)

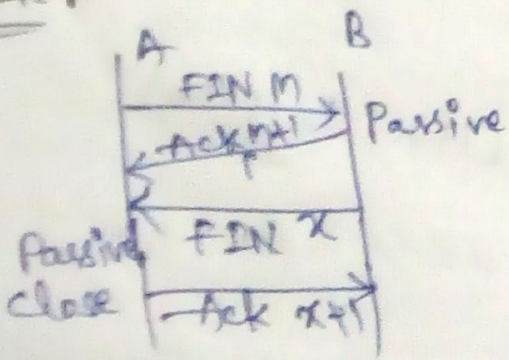


Acknowledgment number :

→ except '0' any number in b/w range 0 to 2^{32}.



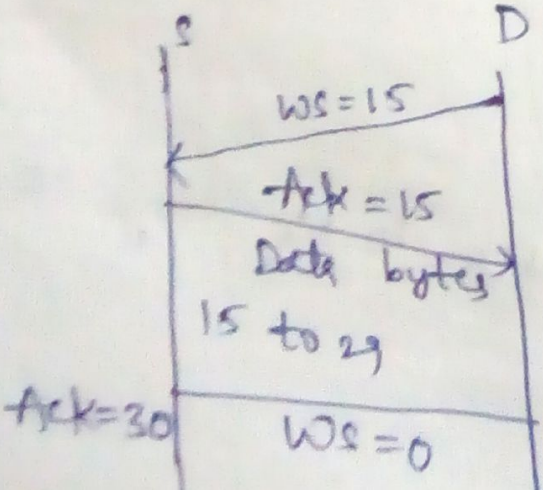
FIN:



→ If FIN bit = 1 ⟹ Terminates the connection.

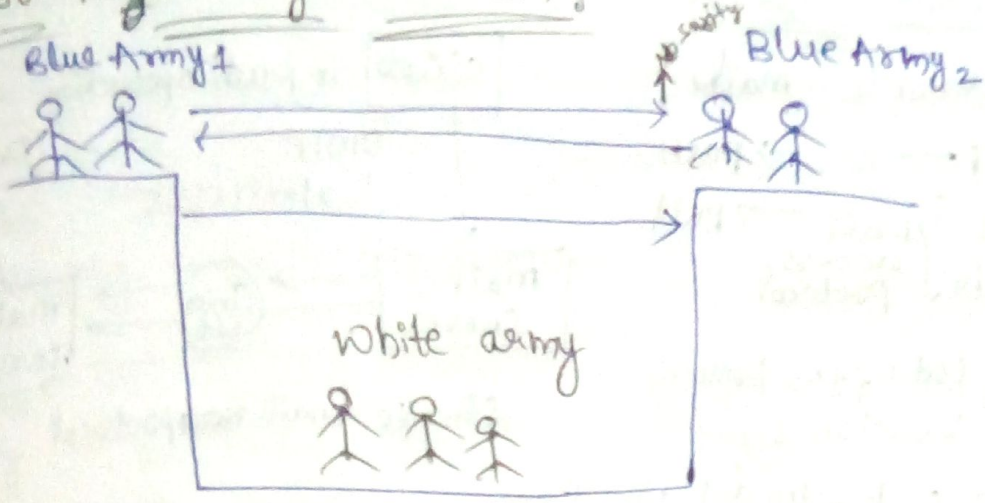
Window size:

→ The max. window size ⟹ 2^{16} = 65,536 Bytes.



→ Data flow control is another service in TCP.

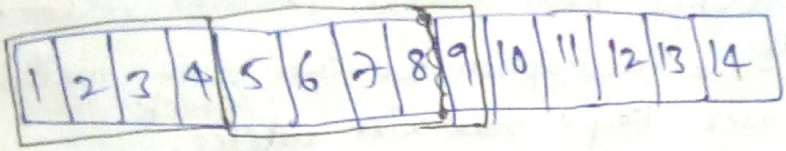
Two ~~Army~~ Army Problem:

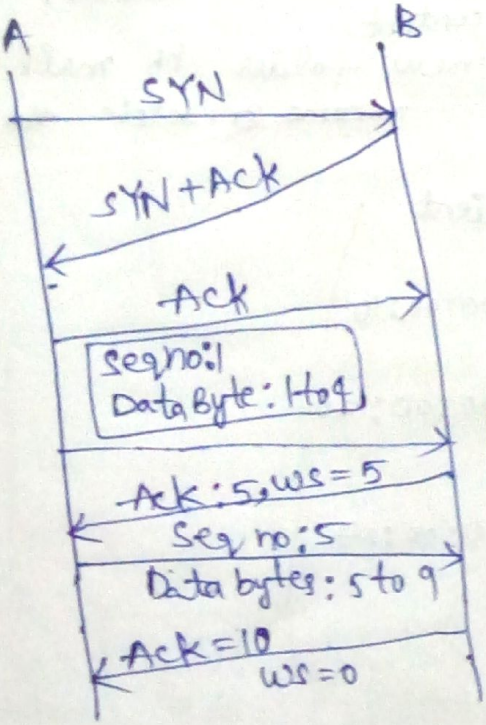


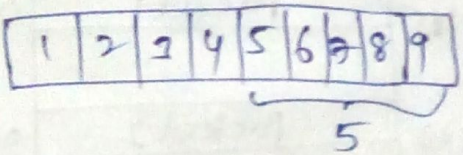
→ Synchronize:-

Blue Army 1 —send message 3rd April→ Blue army 2

Sliding window:-

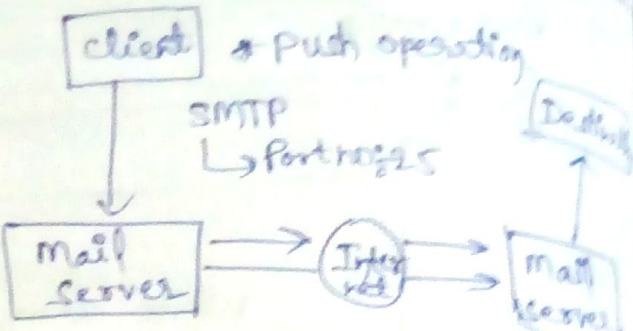






Application Layer :-

- Electronic mail
- SMTP → push
- POP3 } mail access protocol → pull
- IMAP4 }
- POP2 (Post office protocol Version 3)
  - 2 modes are there.
    - keep mode
    - delete mode.

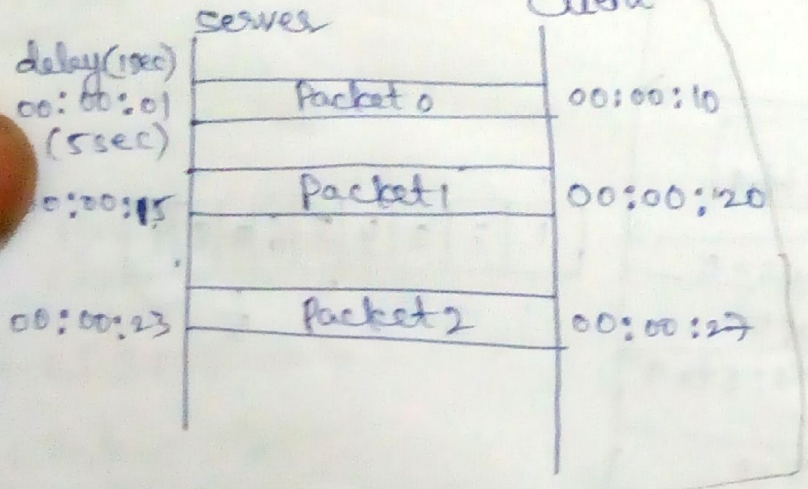


Simple mail protocol

- only two operations in POP3.
- Drawback :- * there is no search option
  * User cannot create folder in mail server
  * Before opening/downloading, user cannot see mail header
- IMAP4 :- (Internet mail access protocol version 4)
- Before (downloading) the description, we can see mail header
- If necessary user keeps otherwise deletes.
- User can search mail content.
- User can create the new folder in mail server and rename (or) delete the folder.

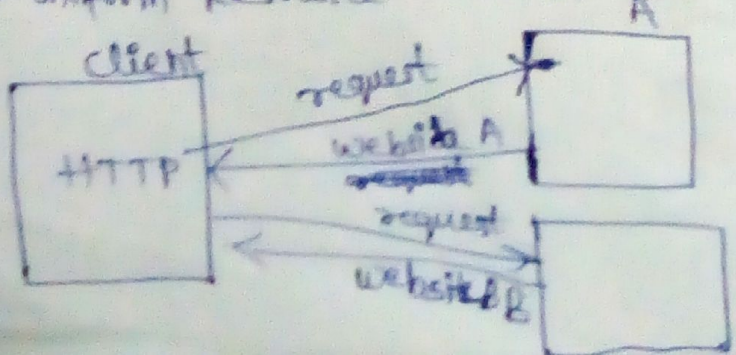
RTP :-

↳ Audio/Video



HTTP :-

- WWW → World wide web
- URL → Uniform Resource Locator.



FTP :- (File Transfer Protocol)

Port 20 — Control Connection
Port 21 — Data connection → For sending the data

↳ For sending control information.

Telnet :- port no :- 23

→ By using TCP we can send the data from own to remote system.

SSH :

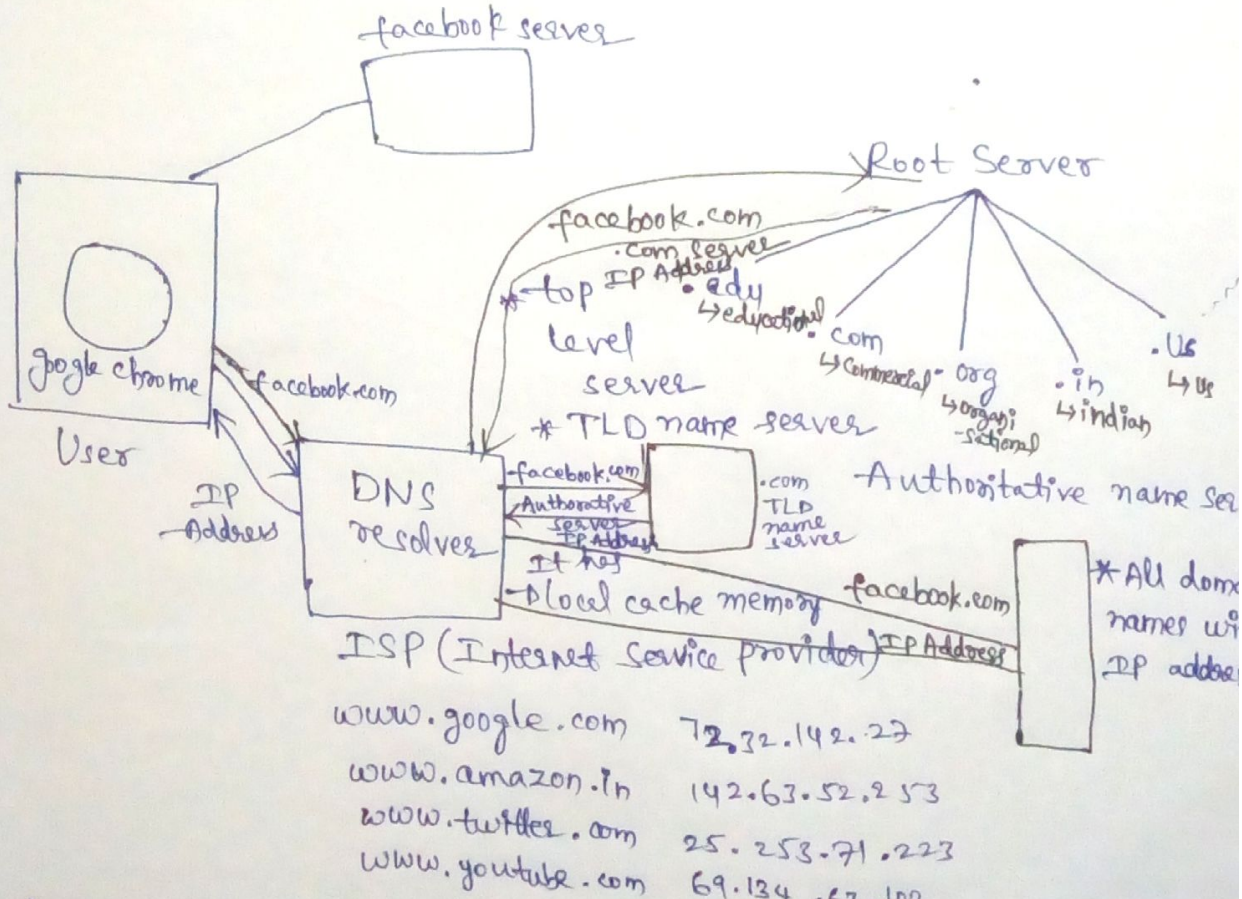
→ Secure shell protocol

DNS :- [Domain Name System]

→ Domain Name to the IP Address mapping.

www.facebook.com — same — changes (mapping) — Ex : 72.142.25.25

mapping



→ If facebook.com is not there in DNS resolver, then it requests Root server.

TLD name server

. com TLD name server

.in TLD name server