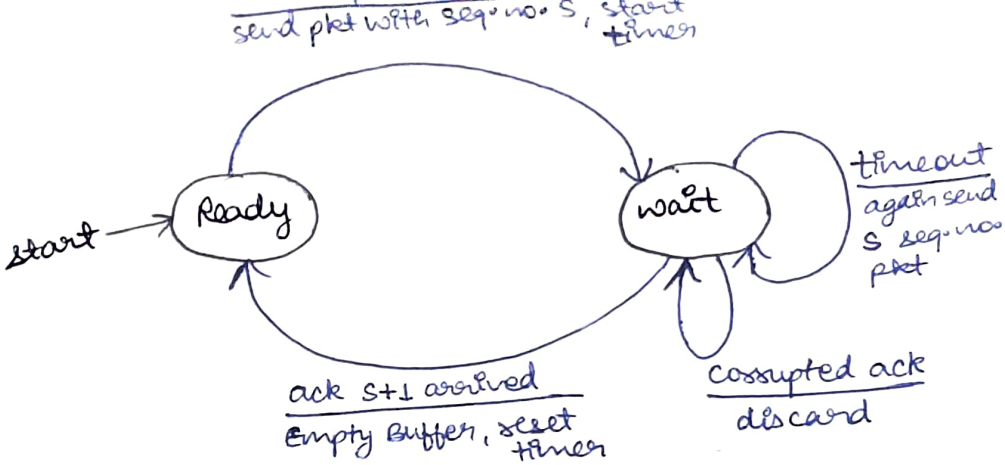


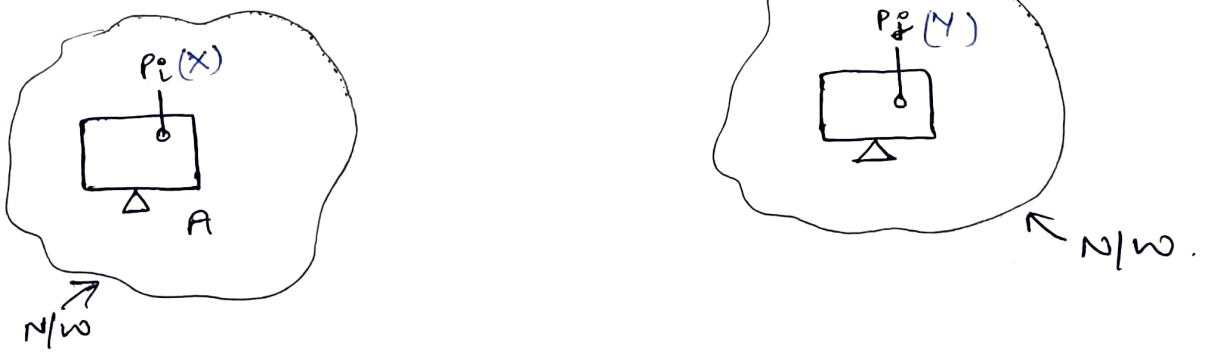
Sender

If pkt received by receiver is corrupted, then R is not incremented, it sends acknowledgment with same serial no.

Varun Kumar Sharma (Post Mid sem) -

IP Addressing -

To make two processes (OS) communicate with each other, we require port addresses. Processes know only about port addresses.



X, Y → port no. of P_i & P_j · A, B → IP Add. of both PCs

X	Y	Data

→ segment (Done by Transport layer)

X	Y	Data	A	B

→ Datagram (Done by Network layer)

Trailer	X	Y	Data	A	B	M_A	M_{R_1}

→ Frame (Done by Data Link layer)

↑ for error correction

↑ mac Add. of A

↑ mac Add. of R_1 (Next hop)

[0-255].[0-255].[0-255].[0-255] → Dotted decimal

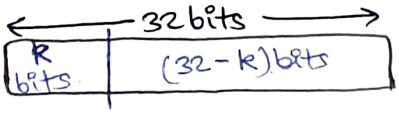
Size = 32 bits (8 bits per octet)

IP Add° contains two things -

① NET ID.

② Host ID.

With the help of NET ID, network of PC can be known.
With the help of Host ID, host of PC can be known.
NET ID & host ID are implicit inside IP Address.
Using NET ID, packet reaches up to default router of network. From there, default router uses Host ID to find the destination PC in the N/w.

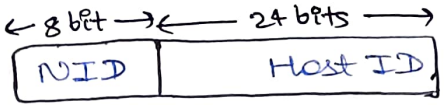


2^k → NET ID (NID)

2^{n-k} → Host ID. (n = 32)

2^k no° of networks.

2^{n-k} no° of PCs inside each network.



Static allocation of NID & Host ID.
$2^8 = 256$ Networks only.
$2^{24} = 60$ million PCs per N/w.

Static allocation is not fruitful.
∴ Classful addressing was introduced.

Classful Address -

A | 8 | 24

B | 16 | 16

C | 24 | 8

D X)(later on)

E X

A : 0
B : 10
C : 110
D : 1110
E : 1111

Code Prefixes

→ Unicast / Broadcast

→ Multicast

→ special security purposes.

Range of classes -

A : | 0 | 7 bits | 24 bits |

2^7 : Networks.

2^{24} : Size of each network.

(Range : 0 to 127)

0 | 000 ... 0 → 0
0 | 000 ... 1 → 1
⋮
0 | 111 ... 1 → 127

B : | 10 | | 16 bits |

10 | 000 000 | 00000000 → 128.0
10 | 000 000 | 00000001 → 128.1
⋮
128.255
129.0
129.1
⋮
129.255
⋮
10 | 111111 | 11111111 → 191.255

$2^6 \times 2^8 = 2^{14}$ = N/W. size

2^{16} PC per N/W.

Class C :

110	21 bits	8 bits

110	00000	00000000	00000000 → 192.0.0
110	00000	00000000	00000001 → 192.0.1
110	11111	11111111	11111111 → 223.255.255

$2^5 \times 2^8 \times 2^8 = 2^{21}$ bits = N/W size

2^8 PCs per N/W.

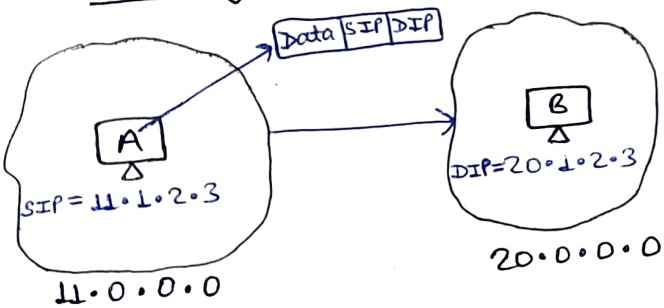
Practical values –

Class A ⇒ NID = 2^7, host = $2^{24} - 2$

Class B ⇒ NID = 2^{14}, host = $2^{16} - 2$

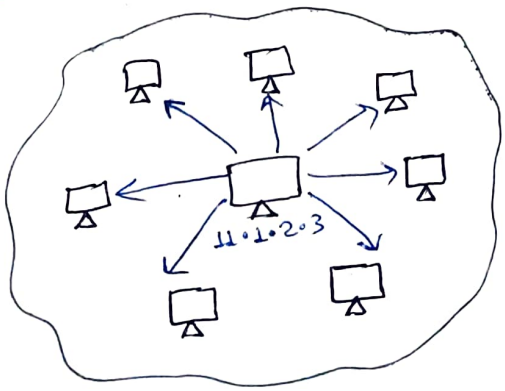
Class C ⇒ NID = 2^{21}, host = $2^8 - 2$.

Unicasting – → one-to-one communication.



Broadcasting – one-to-many communication

Limited Broadcast / Directed Broadcast



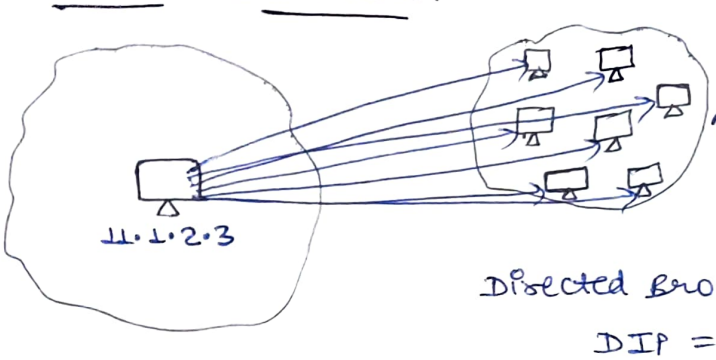
Limited Broadcast

Source IP = 11.1.2.3

Limited Broadcast Address (LBA) = 255.255.255.255

Destination IP = LBA

Directed Broadcast



SIP = 11.1.2.3

Directed Broadcast Addressing (DBA)

DIP = DBA = 20.255.255.255.

Eg:

	NID	DBA	LBA
1.2.3.4	1.0.0.0	1.255.255.255	255.255.255.255
130.1.2.3	130.1.0.0	130.1.255.255	—— 11 ——
220.15.1.10	220.15.1.0	220.15.1.255	—— 11 ——

In practical size of host -

-2 was there bcoz first IP is always given to the network itself and last IP is always given to the broadcast address (DBA)

Subnetting -

1 company ⟨ Dev. / HR / Testing ⟩ Three different dept. Each should not be able to access other's data.

Thus, divide the N/W into smaller N/W.

Subnetting : Divide a bigger N/W into smaller sub-N/W.

Without subnetting -
1. Find N/W boundary
2. Reach destination host.

With subnetting -
1. Find N/W boundary
2. Find sub N/W boundary
3. Reach destination host.

Dividing Networks into subnets using Host ID part -

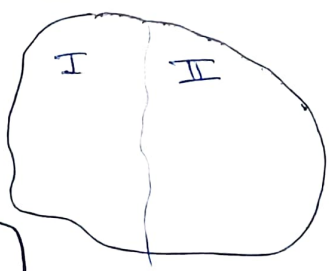
Subnet I → SID (Subnet ID) 200.1.2.0

200.1.2.0 ------

200.1.2.00000000

200.1.2.01111111

Range ⇒ 200.1.2.1 to 200.1.2.127



Subnet II → SID.

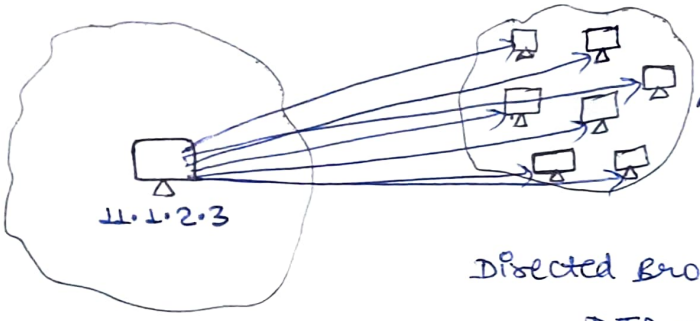
200.1.2.1 ------

200.1.2.10000001

200.1.2.11111111

Range ⇒ 200.1.2.128 to 200.1.2.

Directed Broadcast



SIP = 11.1.2.3

Directed Broadcast Addressing (DBA)

DIP = DBA = 20.255.255.255.

Eg:

	NID	DBA	LBA
1.2.3.4	1.0.0.0	1.255.255.255	255.255.255.255
130.1.2.3	130.1.0.0	130.1.255.255	—— 11 ——
220.15.1.10	220.15.1.0	220.15.1.255	—— 11 ——

In practical size of host -

-2 was there bcoz first IP is always given to the network itself and last IP is always given to the broadcast address (DBA)

Subnetting -

1 company ⟨ Dev. / HR / Testing } Three different dept. Each should not be able to access other's data.

Thus, divide the N/W into smaller N/W.

Subnetting : Divide a bigger N/W into smaller sub-N/W.

Without subnetting -
1. Find N/W boundary
2. Reach destination host.

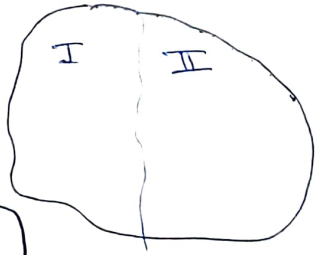
With subnetting -
1. Find N/W boundary
2. Find sub N/W boundary
3. Reach destination host.

Dividing Networks into subnets using Host ID part -

Subnet I → SID (Subnet ID) 200.1.2.0

200.1.2.0 -------
200.1.2.00000000
200.1.2.01111111

Range ⇒ 200.1.2.1 to 200.1.2.127



Subnet II → SID.

200.1.2.1 -------
200.1.2.10000001
200.1.2.11111111

Range ⇒ 200.1.2.128 to 200.1.2.255

NID of subnet I ⇒ 200.1.2.0 → same as NID of whole N/W

DBA of subnet I ⇒ 200.1.2.127

NID of subnet II ⇒ 200.1.2.128

DBA of subnet II ⇒ 200.1.2.255 → same as DBA of whole N/W.

Now, what does 200.1.2.255 mean? Does it mean DBA of whole N/W or does it mean DBA of subnet II?

If packet is coming from outside network, then it will be broadcasted to complete N/W.

If packet is coming from within the N/W, then it will be broadcasted to subnet II

Outside N/Ws don't know about internal arrangement of our N/W, so broadcast.

Internal PC knows about internal arrangement, ∴ broadcast to subnet II only.

Subnet Mask –

All 1's in NID + SID part. All 0's in Host ID part.

Class A ⇒ 255.0.0.0

Class B ⇒ 255.255.0.0

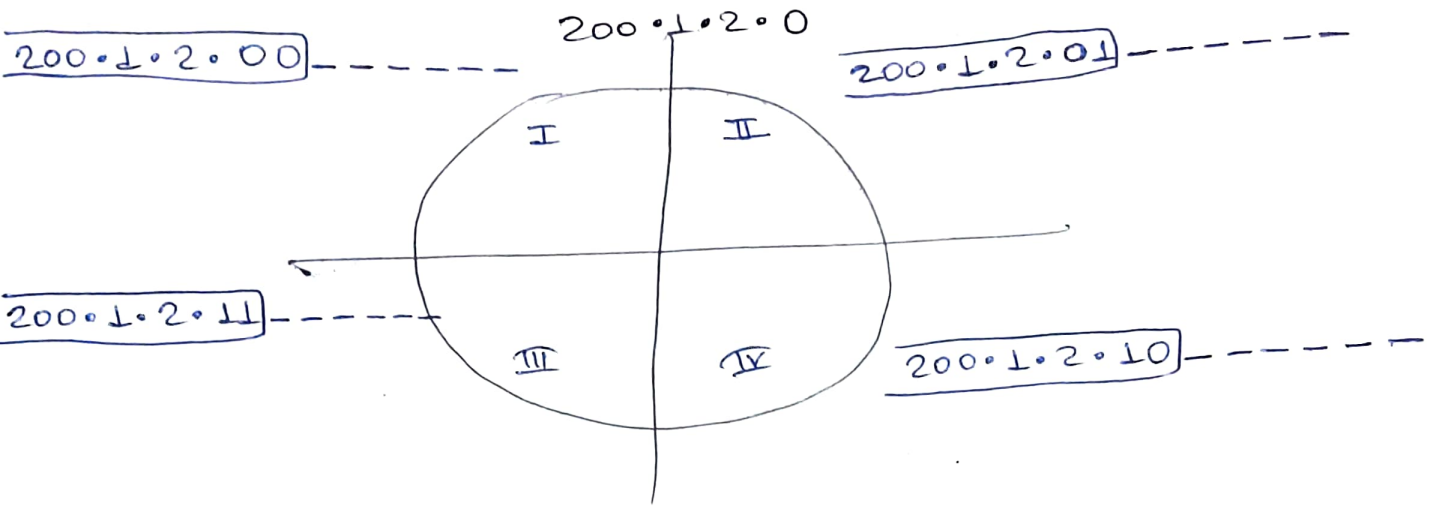
Class C ⇒ 255.255.255.0

In previous eg. of subnet –

Subnet mask of subnet I & II –

11111111.11111111.11111111.10000000

255.255.255.128.



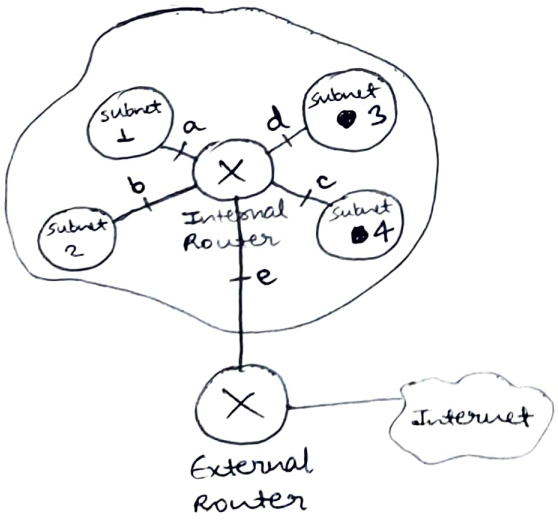
Range of subnet I -
200.1.2.0 to 200.1.2.63

Range of subnet II -
200.1.2.64 to 200.1.2.127

Range of subnet III -
200.1.2.128 to 200.1.2.191

Range of subnet IV -
200.1.2.192 to 200.1.[redacted]2.255

No subnetting ⇒ [redacted] - 2 [redacted] for host ID.
Subnetting into 2 ⇒ - 4 for host ID.
Subnetting into 4 ⇒ - 8 for host ID.



Subnet mask of each subnet is 255.255.255.192

If a packet from the Internet comes with destination IP = 200.1.2.20, then this packet will be sent to subnet 1.

What does the Internal Router do ?

Internal Router contains Routing Table

NID	Subnet Mask	Interface
200.1.2.0	255.255.255.192	a
200.1.2.64	"	b
200.1.2.128	"	c
200.1.2.192	"	d
0.0.0.0	0.0.0.0	e

← Routing Table

→ Default Entry.

Dest. IP = 200.1.2.20

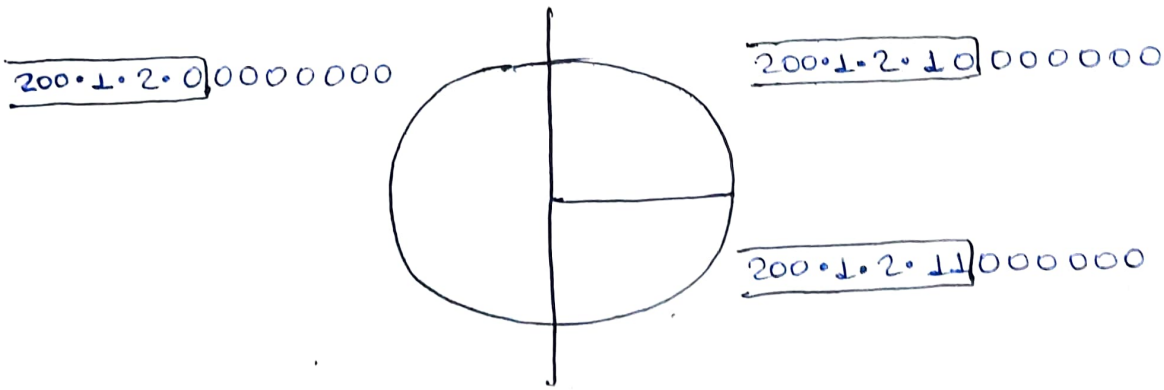
(Logical AND) SM = 255.255.255.192

200.1.2.0 → NID of 1st row. Goes to Interface a

If multiple entries match from routing table, then packet is sent to that subnet which has biggest/largest subnet mask (i.e. subnet mask with more no. of 1s)

If subnet masks are same, then multiple entries from routing table can never match.

If no entry matches, then packet is sent to default entry, i.e. back to External Router.



VLSM (variable length subnet masking).

SM = 255.255.255.192 → Class A.

NID + SID = 26.

8 + SID = 26 ⇒ SID = 18

Classless Addressing -

→ Scalability is an issue with classful Addressing.

→ wants 300 IP → takes class B → IP Address wastage.

Representation classless addresses -

a.b.c.d/n (CIDR → Classless Interdomain routing)

n → Block ID & NID

(no. of hosts = 2^{32-n})

Eg: 20.10.15.10/20

Block ID & NID = 20 bits.

Host ID = 32 - 20 bits

No. of hosts = 2^{12}

Rules for making CIDR block -

① All the IP addresses which will be given should be continuous.

② Size of each block should be in power of 2.

③ First IP Address will always be evenly divisible by size of block.

Eg: 100.1.2.32
100.1.2.33
⋮
100.1.2.47

} Block → Rule 1 satisfied, Rule 2 satisfied

size = 2^4

100.1.2.00100000

Last 4 bits are 0, ∴ it is evenly divisible by 2^4.

∴ valid CIDR block.

Eg: 20.10.30.32
20.10.30.33
⋮
20.10.30.63
} → Rule 1 & 2 satisfied.
Block size = 2^5

20.10.30.00100000
last 5 bits are zero.
∴ Evenly divisible by 2^5

Eg: 100.1.2.32
100.1.2.33
⋮
100.1.2.47
} valid CIDR.

Representating the complete CIDR.

100.1.2.39/28 or 100.1.2.40/28 or 100.1.2.41/28

39 → Any ID belonging to our block
28 → 32 − 4 = 28 (NID part)

100.1.2.39/28 → fix NID

100.1.2.0010 0111

100.1.2.00100000 → (100.1.2.32)
100.1.2.00100001 → (100.1.2.33)
100.1.2.00100010
⋮
100.1.2.00101111 → (100.1.2.47)

→ Thus we can know the complete CIDR block using its representation.

Subnetting in CIDR -

20.30.40.10/25 → NID

20.30.40.00 001010

20.30.40.00000000
——||—— 000001
⋮
——||—— 111111

(20.30.40.0 − 20.30.40.63)
20.30.40.0/26 → NID
20.30.40.63/26 → DBA.

(I) /26 | (II) /26

20.30.40.01 000000
——||—— 000001
⋮
——||—— 111111

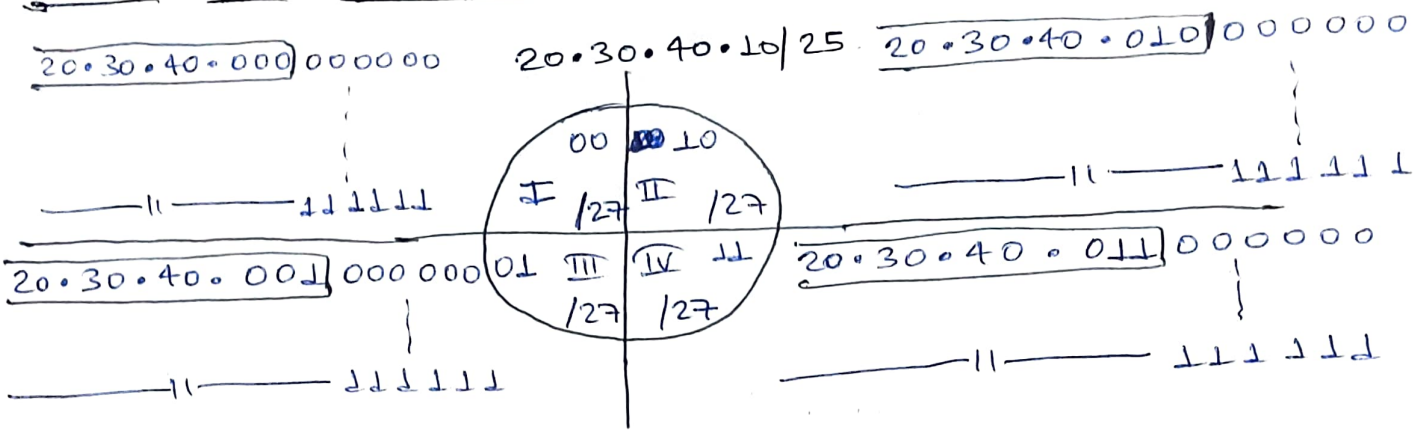
20.30.40.64/26 → NID
20.30.40.65/26
⋮
20.30.40.127/26 → DBA

Subnet mask of subnet I ⇒ 255.255.255.192/26

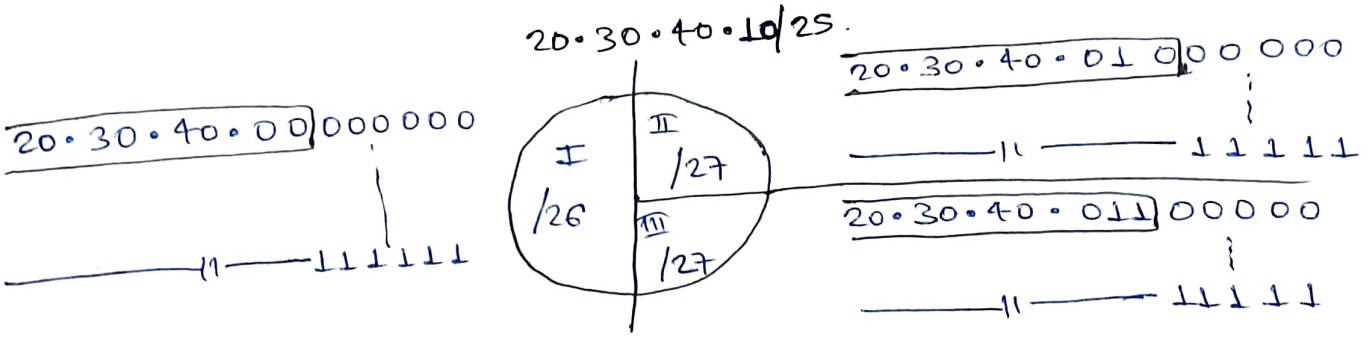
" " " II ⇒ 255.255.255.192/26

" " " complete N/W ⇒ 255.255.255.SM
(complete N/W)

Divide into 4 subnets -



Divide into 3 subnets -



IPv4 Header -

① Header length -

32 x 5 = 160 bits = 20 bytes.

with options, 20 + 40 bytes = 60 bytes

without options, 20 bytes

Header length ∈ [20 bytes, 60 bytes]

Bits given to mention header length = 4 bits (0 to 15)

Scaling factor Header length ∈ $\left[\frac{20}{4} \text{ bytes}, \frac{60}{4} \text{ bytes}\right]$

if header length = 20 bytes

Divide by 4 ⇒ 5 bytes ⇒ can represent in 4 bits.

If header length = 30 bytes

divide by 4 ⇒ $\frac{30}{4}$ ⇒ make header size multiple of 4.

Padding ⇒ Increase header size to make it multiple of 4 (scaling factor)

30 bytes —Padding→ 32 bytes.

We put these 2 extra bytes in options field.

② Version – (4 bits)

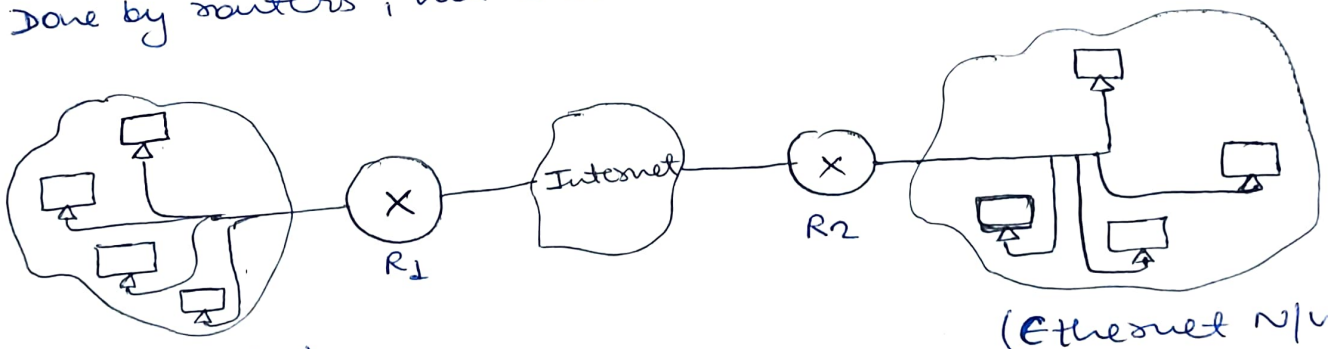
Specifies whether IP is of version 4 or version 6.

③ Identification – (16 bits)

Specifies the unique value to represent packet over the Internet.

Fragmentation –

- Done by routers, not end users.



(Token Ring N/W)

MTU = 4500 bits

(Ethernet N/W)

MTU = 1500 bits

R_2 divides the packet received into 3 packets bcoz maximum transferrable unit of right N/W is less.

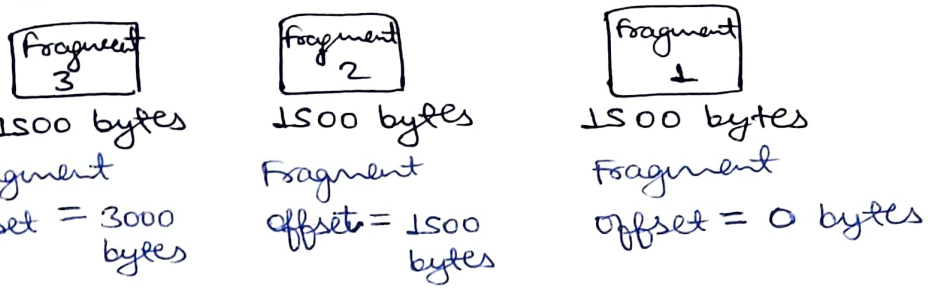
This is called fragmentation.

Identification no. of original pkt & fragmented pkts is same.

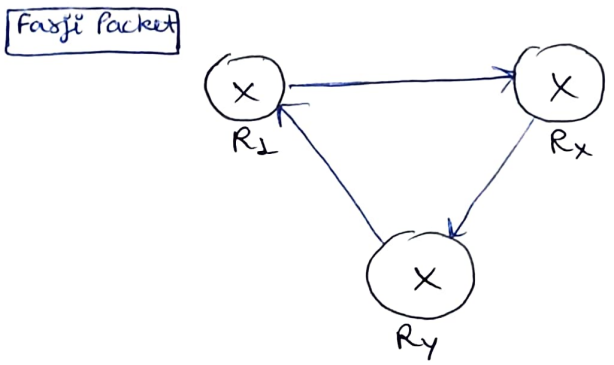
more fragment (MF) flag of all fragments except the last fragment is 1 indicating there are more fragments to come.

Don't fragment (DF) flag = 1 means that the sender does not want the router to fragment its pkt. So router will not fragment if DF = 1.

Fragment offset ⇒ How many bytes of data is there in fragments before it (cumulative).



Time to live (TTL) – (8 bits)

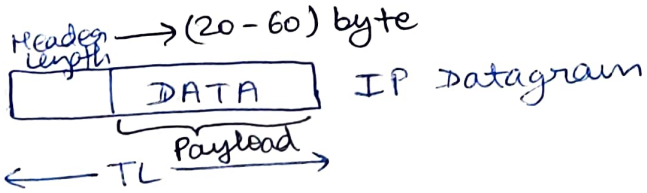


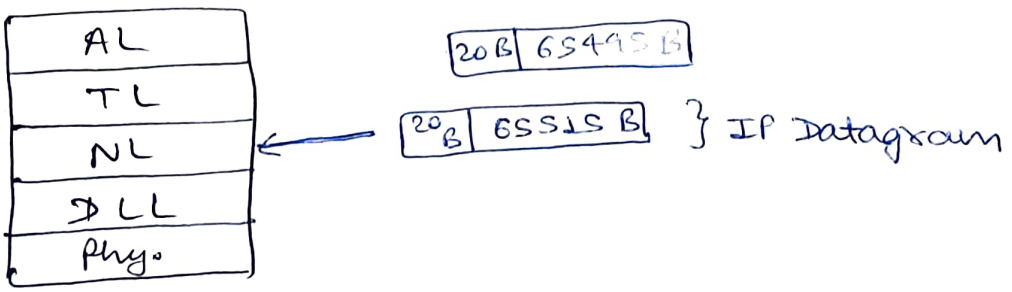
R_x is default of R_1
R_y is default of R_x
R_1 is default of R_y
fargi pkt keeps on going from R_1 to R_x to R_y to R_1.

∴ TTL specifies max. no. of hops that a pkt can make from one router to other.

Total length (TL) (16 Bits)

$2^{16} - 1 = 65535 = 65K$.





① Header Checksum –

→ To detect errors at each router while transmitting data.

(only for header / not for data)

② Protocol –

Priority – ICMP < IGMP < UDP < TCP.

If a router's buffer is full and a TCP type pkt comes to it, then it should remove ICMP type pkts from its buffer (if any) and fit in TCP type pkt.

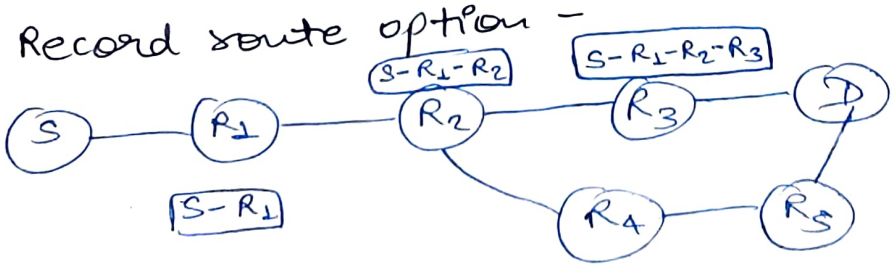
Router pe network layer tak hi hoti hai, to usko nahi pata ki pkt TCP hai ya UDP.

Isliye protocol field mein ek number store karte hain, jis se TCP type pkt hai ya UDP type, vo pata chal jae.

Value	Protocol
1	ICMP
2	IGMP
6	TCP → Highest priority.
17	UDP
89	OSPF

③ Options – (0 to 40 bits)

Record route option –



Records the route that the pkt travels to reach the destination.

As a user, we can't use this option.

This option can be used by N/W administrator.

9 IP can be stored in route. $\left[\frac{40}{4} - 1 = 9\right]$

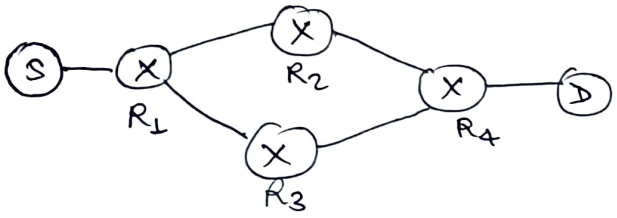
-1 for space btw. IP Addresses.

Source Routing option -

- Strict source routing
- Loose source routing.

Strict source routing - specify the complete path btw. source & destination, router is not given the choice to define path of pkt.
Available for N/w Administrator and ISP. Not at user level.

Loose source routing - Not specifying the complete path, but only few routers.

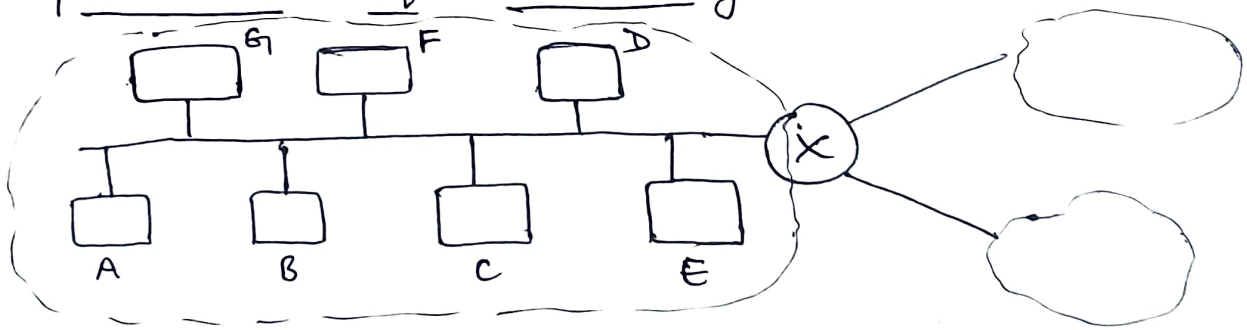


S - R_1 - R_2 - R_4 - D Strict

S - R_1 - D Loose.

Padding option - (To pehla padha, vahi hai)

Implementation of Broadcasting -



N-I

AL | m |

TL | m | X | Y |

NL | m | X | Y | I_A | 255.255.255.255 |

DLL | T | | M_A | FF:FF:FF:FF:FF:FF |

Phy.

Limited Broadcasting.

- Broadcasting over Internet is not allowed.

AL | m |

TL | m | x | y |

NL | m | x | y | I_A | DBA(N-II) |

DLL | | M_A | Mac of R_1 |

Phy.

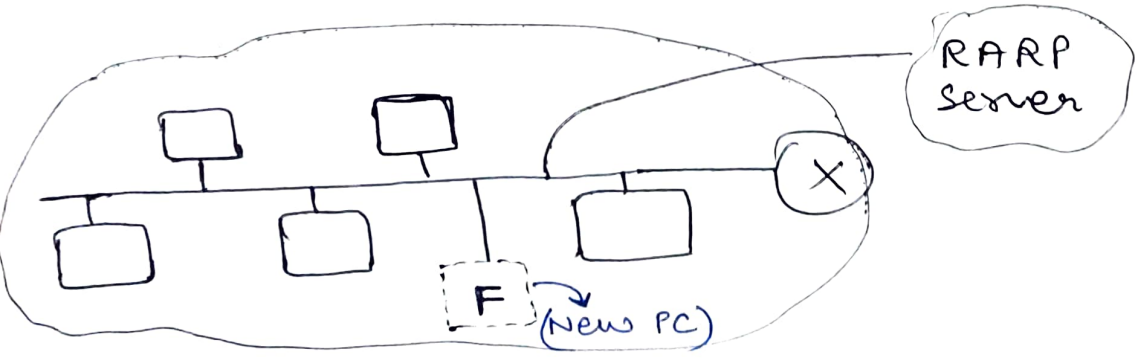
It is the responsibility of R_2 to convert DBA to LBA of N-II.

ARP request is broadcasted by default.
ARP request is unicasted by default.

Reverse ARP (RARP) -

ARP ⇒ IP hai, Mac nahi hai
RARP ⇒ Mac hai, IP nahi hai

ARP same network ke andar kaam karta hai.



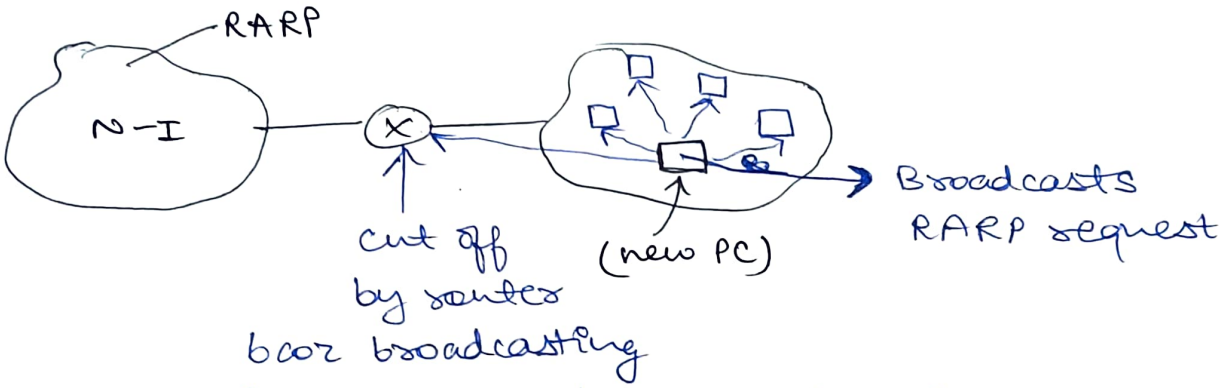
RARP request ⇒ | M_F | 0.0.0.0. | FF:FF:FF:FF:FF:FF |

RARP response ⇒ | M_F | I_F |

I_F of new PC provided by RARP server

RARP server maintains a table where it records IP provided to corresponding mac addresses in order to know which IP addresses are available for allocation.

Every network needs to have at least one RARP server. Because RARP request will be broadcasted but will be cut off by the router. Thus, need for separate RARP server.



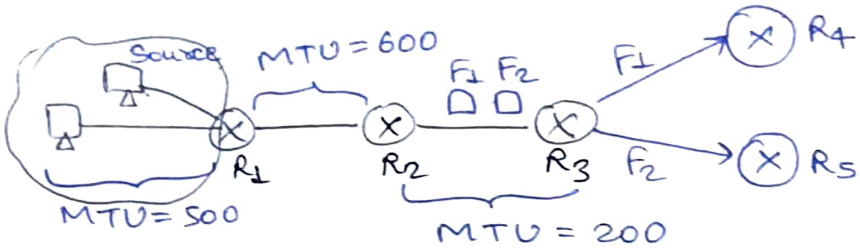
bcoz broadcasting is not allowed over Internet.

Different RARP server for each N/W ⇒ IP conflict problem.

Static IP problem ⇒ Ek baar jo mene IP dedi, uske baad mein change ni krta.

RARP ki problems hatane ke lie BOOTP aaya.

BOOTP ——"—— DHCP ——"——



Pkt from source is sent with DF (Don't fragment) = 1 so R_2 can't fragment it. R_2 drops the pkt and generates ICMP error msg.

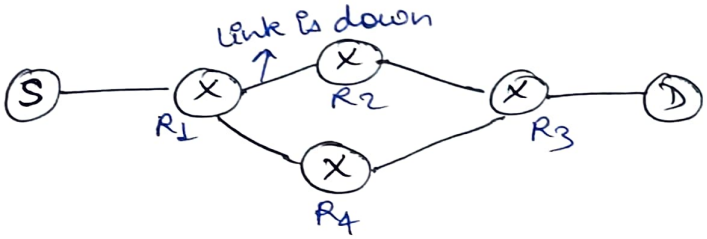
Now, let us say DF = 0 for pkt. Then R_2 fragments the pkt, let us say, F_1 & F_2.

Assume F_1 is lost at R_4. Then ICMP error msg is generated.

If F_2 is lost at R_5, then no ICMP error msg is generated.

Only if first fragment is lost, then ICMP is generated.

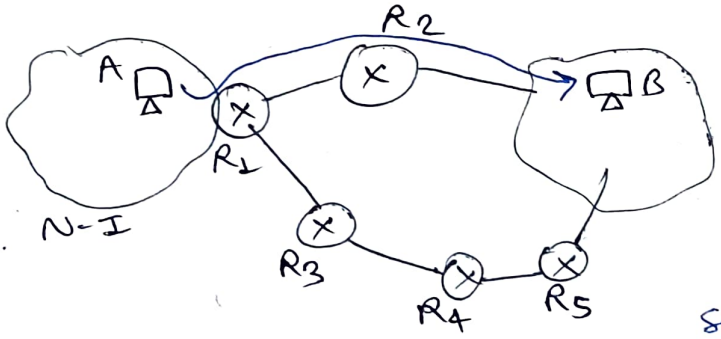
For all other fragments lost, no ICMP is generated.



S → R1 → R2 → R3 → D
(Source routing)

Pkt dropped at R_1, ICMP will sent (Parameter problem)

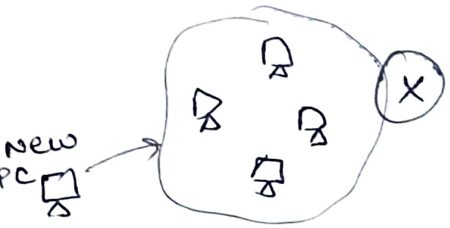
Source Redirect ICMP feedback msg -



If a pkt goes from R_3 wala path, then since it knows R_2 wala path is better, it will send source redirect ICMP feedback msg stating the same.

Also called Redirection.

ICMP request & reply -



A new device wants to access N/w but it doesn't have a default gateway IP so it will broadcast it within the N/w, default router will unicast its subnet & IP. Packet will be ICMP.

This is called Mask ICMP / router solicitation

ICMP router advertisement - If a new default router comes to the N/w, then it should broadcast its IP & subnet mask in order to let other devices know about its presence.

Routing Protocols -

→ Helps in maintaining & creating routing tables at router.

Routing → knowing the path through which pkt needs to be sent

(Switching → actually sending pkt to next hop given by routing)

Flooding → sending pkt via all possible paths.
↳ reliable. guarantees pkt transmission. But unnecessarily increases congestion.

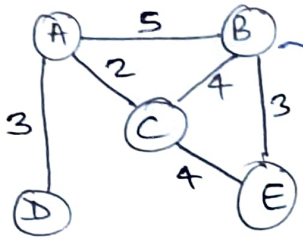
Routing can be of 2 types -

① Static - N/w Administrator maintains routing table.

② Dynamic - Router communicates among themselves to maintain routing table.

Dynamic routing is also called Unicast Routing protocols

Link State Routing (LSR) –



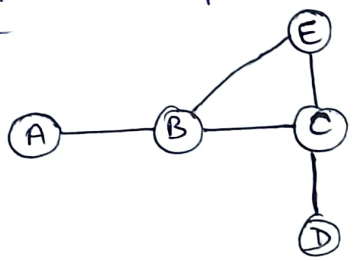
Uses concept of flooding.

→ Create a LSP pkt.

B	
Dest.	Distⁿ
A	5
C	4
E	3

LSP Packet.

B sends this LSP pkt to A, C, E.
A, C, E sends this pkt to their immediate neighbours.
Now, A will create its own LSP pkt. Send to C, D, B.
C, D, B will send to their immediate neighbour.
(A se C par aaya hai to C vapis A pe nahi bhejega).



D sends to C.
C sends to B, E.
B sends to A, E.
E sends to B.

Kya B vapis A ko bhejega?
Nahi, jo seq. no. process ho gaye hain, unko flood nahi karna hai. To jab E ne B ko bheja, tab B ko pata hai ki ye LSP pkt process karke flood kia ja chuka hai pehle hi.

B
Seq. no.
TTL
Dest.	Dist.

→ time baad vo LSP pkt ko drop kar dega. Iss se looping se bach jaenge.

With the help of these LSP pkts, each node knows the map of complete N/w. Then applies Dijsktra's to find shortest path.