Date | 09-06-25
Page |

WEEK - 2

ADDRESS :

A unique no. or an ID which is assigned to the host or interface of network.

SUBNET :

A portion of network that shares a particular subnet address.

ex: 192 . 168 . 32 . 5

11000000 . 10101000 . 00100000 . 00000101

SUBNET MASK :

A 32-bit combination used to describe which portion of an address refers to subnet & which refers to host.

1 Octet = 8 bits

255 . 255 . 255 . 0

11111111 . 11111111 . 11111111 . 00000000

(8) (16) (24)

1 → ON
0 → OFF

1 = network bits , 0 = host bits

NOTE:

If all binary bits are one, the decimal equivalent would be 255 as shown :

1 1 1 1 1 1 1 1

255 64 32 16 8 4 2 1 [255]

When not all of bits are set to 1

0 1 0 0 0 0 0 1

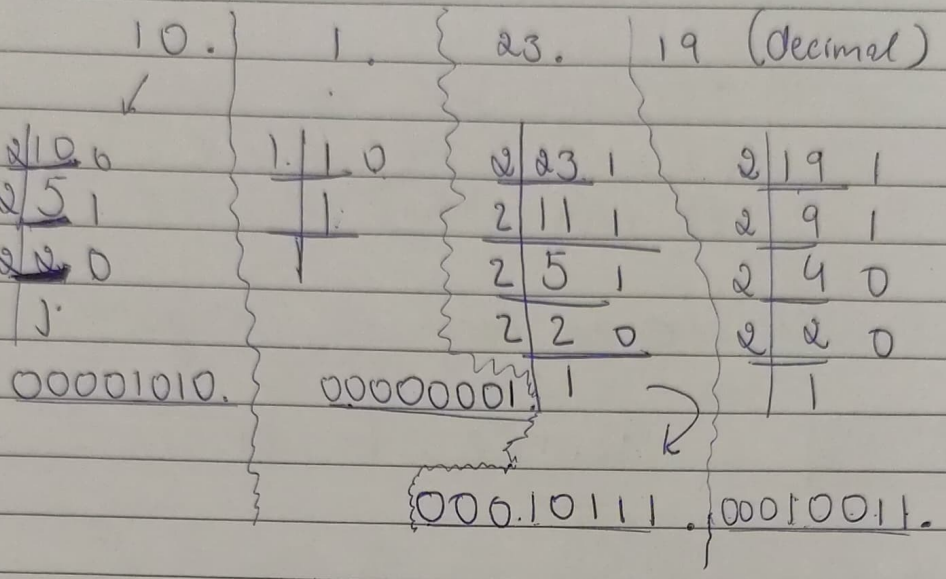
0 64 0 0 0 0 0 1 [0+64+0+0+0+0+0+1]

=65

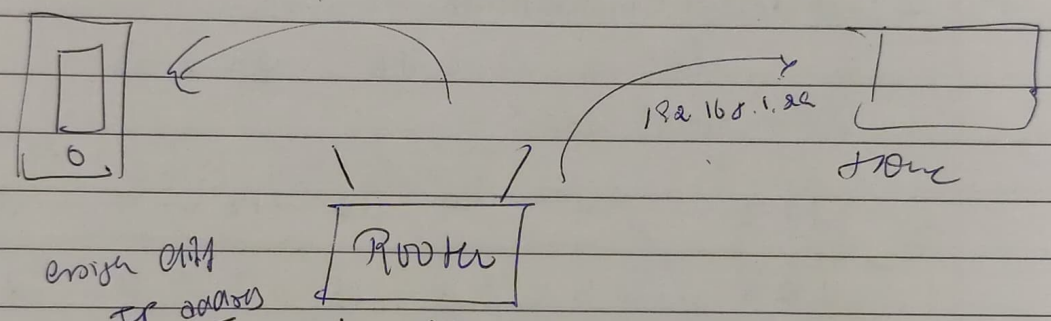
4.3 Billion IP Addresses

Date
Page

In below example, the IP address represent both binary & decimal :



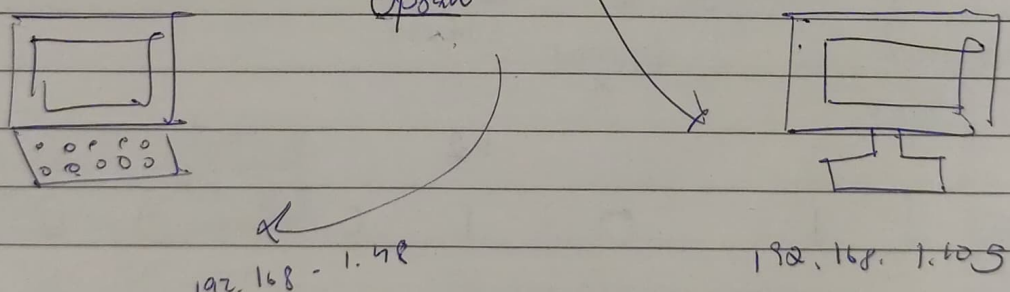
—



[NAT]
↓
to Convert Private IP to Public IP

assign this IP address

acts like 'Oprah'



The 2 IP addresses that are reserved

192.168.1.0 192.168.1.255

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

Date |
Page |

11000000.10101000.00000001.00010101

(192)

1 1 0 0 0 0 0 0

128 64 32 16 8 4 2 1

128 + 64 = 192

By doing other, we get,

DECIMAL 192.168.1.21

IP Address in MATRIX

172.16.34.3

172	44	44	12	12	4	0	0
128	64	32	16	8	4	2	1
	can't be Subtracted ×	can be Subtracted ✓	×	4	0	X	X

~~True~~ (can be Subtracted)

44

12

BINARY [1 0 1 0 1 1 0 0]

by doing other, we get

10101100.00010000.00100010.00000011

If we want to create the 17 networks from our existing networks then, another method comes in

128 64 32 (16) (8) (4) (2) (1)

256 128 64 32 16 8 4 2

As to create 17 network, we need atleast 32 & need 5 bits to create 17 networks
[16, 8, 4, 2, 1] as circle

So, if host bits are 00000000 out of this, we will keep 5 bits, the result is [11111000]

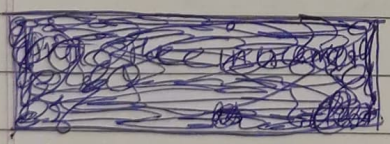
new subnet mask

to convert into decimal

1 1 1 1 1 0 0 0
128 64 32 16 8 4 2 1

128 + 64 + 32 + 16 + 8 = 248

decimal bit type



If the IP address is 192.168.1.0 (Broadcast address)

as the last no. 1 lies at 8, so the ranges will be,

Create the network ← 192.168.1.0 → 192.168.1.7

192.168.1.8 → 192.168.1.15 [as 0 is still exist, so]

192.168.1.17 → 192.168.1.25

192.168.1.26 [illegible] → 192.168.1. 192.168.1.254

Date |
Page |

TO KNOW HOW THE HOST BITS [WE NEED TO HACK] ①

② Save the host bits

Nosferatu

③ Find the increment

④ Create your network

the network → 10.1.1.0/24

255.255.255.0

(Binary conversion) 11111111.11111111.11111111.00000000

We need [40] hosts or IP address on each network

~~THROUGH THIS CHART~~

		1	1	1	1	1	1
128	64	(32)	(16)	(8)	(4)	(2)	(1)
256	128	<u>64</u>	32	16	8	4	2

We need [6] host bit to hack

Flip OR REVERSE THE hosts bits

8 16 24 25 26

11111111.11111111.11111111.00ØØØØØØ ←

③ ~~To find the~~ Increment

the last network bit.

1 1

128 (64) 32 16 8 4 2 1

We increment by 64

10.1.1.0 ⟶ 10.1.1.63
10.1.1.63 ⟶ 10.1.1.127
10.1.1.127 ⟶ 10.1.1.191
10.1.1.191 ⟶ 10.1.1.255

[255.255.255.192]

To find the /24, we add the network bits

128 + 64 + 32 + 2 = (26)

11111111.11111111.11111111.00000000

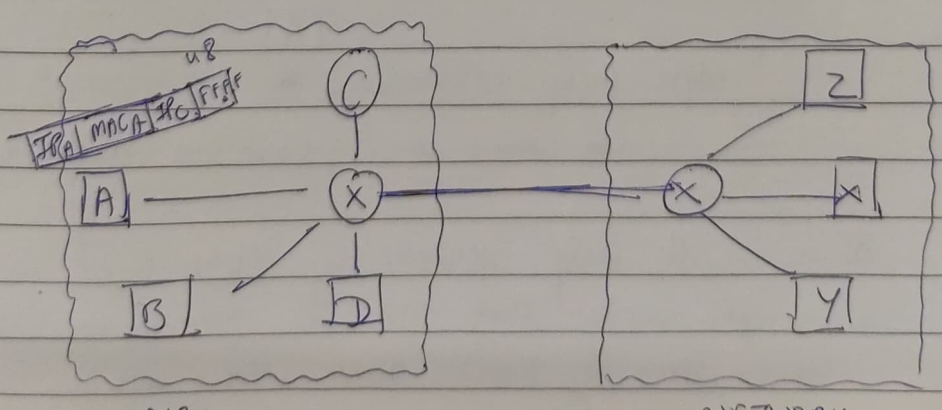
255.255.255.192/26

① 10.1.1.0/26 ② 10.1.1.64
③ 10.1.1.128

Address Resolution Protocol: [ARP] LAYER-3

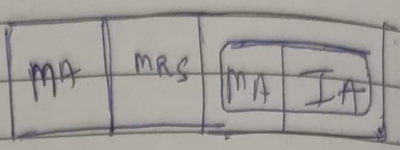
(Host-H) (Router-R)
H → H
H → R

IP → MAC LOGICAL → PHYSICAL

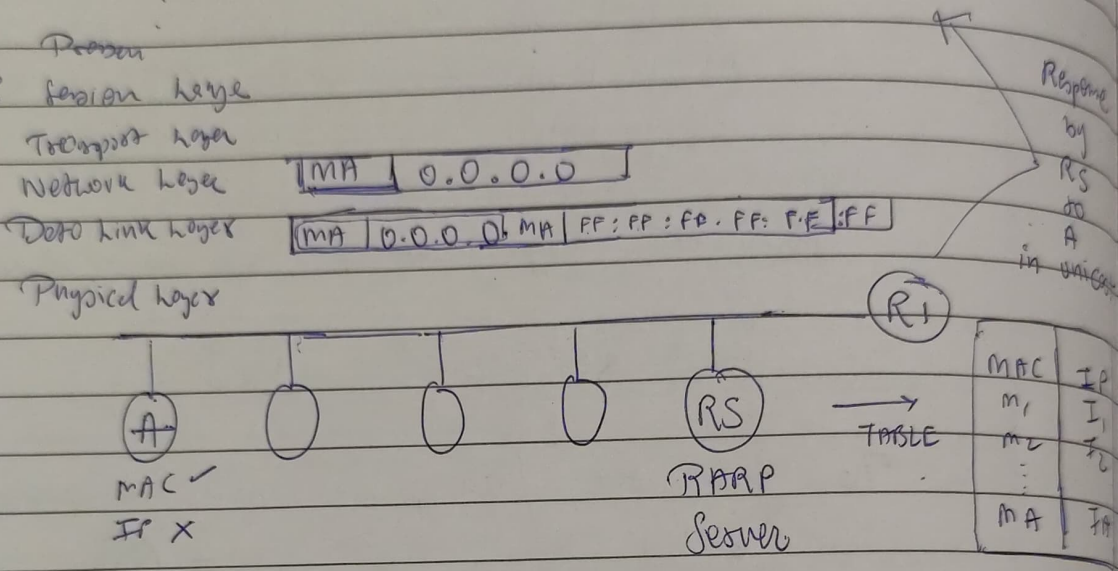


RARP: [Reverse ARP]

MAC → IP
(known) [Unknown]



EXAMPLE:



- Suppose A has MAC address not IP address, so it will send packet through [Network Layer] as shown
- After the request will send to [DATA LINK LAYER] through [NETWORK LAYER]
- As [A] don't know the MAC address of RARP server it will broadcast the network (FF:FF:FF:FF:FF:FF)

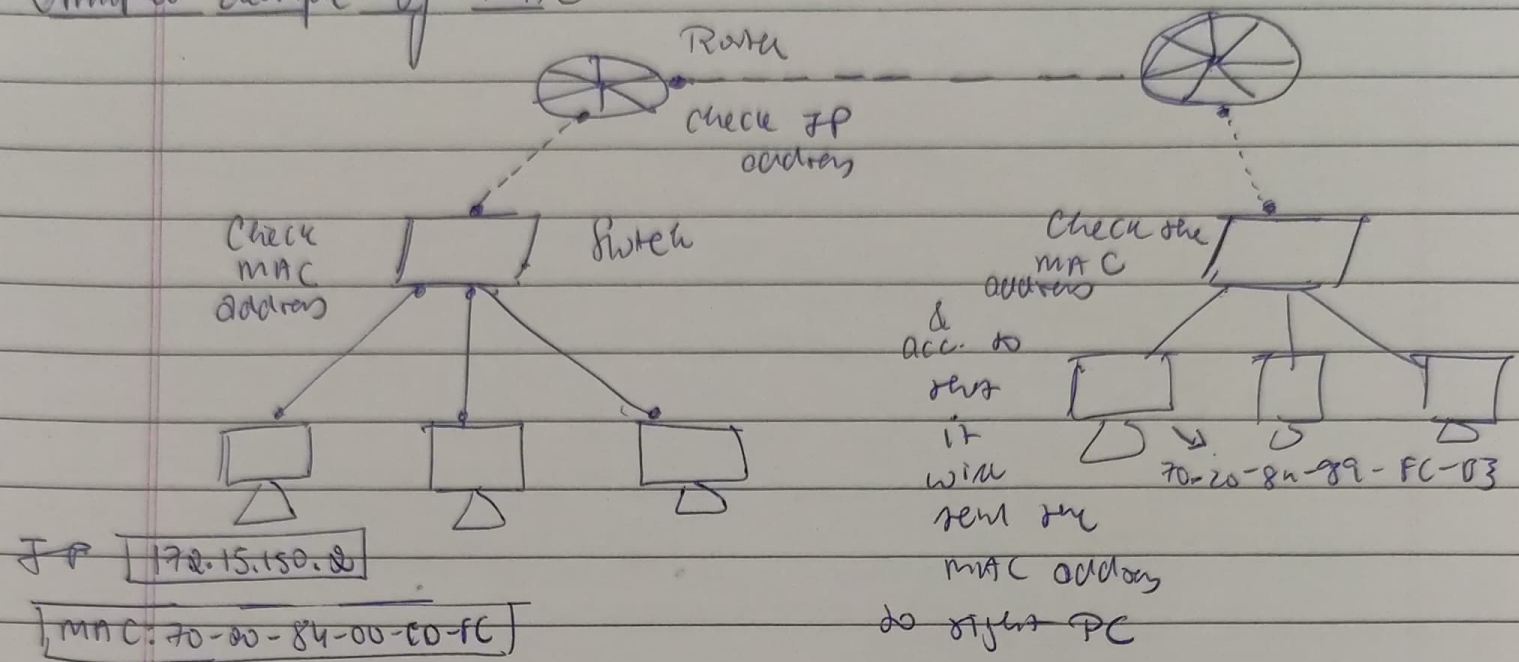
[A]

After doing broadcast, it will check each host which are present in network

- Answer will only generate by RARP server, as it has table, after verify that MA is present in table, we also get to know about the IA

(See above the arrow diagram)

Another Example of MAC



MAC → media access Control

[Every node in LANs identified with help of MAC address]

IP address → location of person

MAC " → Name of the person