IPv4 Addresses :-

An IPv4 address is 32 bit address that uniquely and universally define the connection of a device.

IPv4 address are unique. They are unique in the sense that each address defines one and only one, connection to the Internet. Two device on Internet can never have the same address at the same time.

Address Space

A protocol such as IPv4 that defines addresses has a address space. An address space is total Number of addresses used by the protocol.
If a protocol uses N bits to define an address the address space is 2^N because each bit can have two different value (0 or 1). and N bits can have 2^N values.

IPv4 uses 32-bit Addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than 4 billion). This mean that theoritically, if there were no restrictions, more than 4 billions devices could be connected to the Internet.

Notations :-

There are two Notations to show an IPv4 address : 1) Binary Notation
2) Dotted decimal Notation.

1) Binary Notation :- In Binary Notation, the IPv4 address is displayed as 32bit. Each octet is often referred to as byte. It is common to hear IPv4 address referred to as 32-bit addresses or 4 byte address

The following example of IPV4 address in Binary notation.

01110101	10010101	00011101	00000010

Dotted-Decimal Notation :-

To make the IPV4 address more compact and easier to read, Internet address are usually written in decimal form with a decimal point (dot) separating the byte.

The following is dotted-decimal Notation

$$\boxed{117.149.29.2}$$

IPV4 address is both Binary and dotted decimal Notation. Note that each byte (octet) is 8bit each number is dotted decimal notation is a value ranging from 0 to 255.

10000000 00001011 00000011 00011111

128.11.3.31

Dotted decimal notation and Binary Notation for an IPV4 address

Example -

① Change the following IPV4 address from Binary notation to dotted-decimal Notation

a) 10000001 00001011 00001011 11101111

b) 11000001 10000011 00011011 11111111

Solution - We Replace each group of 8 bit with equivalent decimal Number and add dots for Separation

a) 129.11.11.239
b) 193.131.27.255

Q2. changes the following IPV4 address from Dotted-decimal Notation to binary Notation.

a) 111.56.45.78
b) 221.34.7.82

Solution - We Replace each decimal Number with its Binary equivalent.

a) 01101111 00111000 00101101 01001110

b) 11011101 00100010 00000111 01010010

Q3. Find the Error, if any in following IPV4 address.

a) 111.56.045.78
b) 221.34.7.8.20
c) 75.45.301.14
d) 11100010.23.14.67

Solution -

a) There must be no leading zero (045)
b) There can be no more than 4 Number in a IPV4 address.
c) Each number needs to less than or equal to 255 (301 is outside this)
d) A Mixture of Binary Notation and dotted decimal Notation is not allowed.

Classfull Addressing:- In this the address space is divided into 5 classes
- Class A
- Class B
- Class C
- Class D
- Class E

Each class occupies some part of Address space

> Note: In classful addressing, the Address space is divided into five classes: A, B, C, D and E

We can find the class of an address when given the address in binary Notation or dotted-decimal notation. If the address is given in binary notation. The first few bits can immediately tell us the class of the address.

- If the decimal dotted notation, the first byte defines the class.

IMP How to find class of an address:-

Binary Notation First Byte		IMP Dotted Decimal First Byte	
Class A	0...	Class A	0 – 127
Class B	10..	Class B	128 – 191
Class C	110...	Class C	192 - 223
Class D	1110..	Class D	224 - 239
Class E	1111....	Class E	240 - 255

Ques ① Find the class of each address: (Decimal Dotted

First byte

a) 227.12.14.87 – Class D

b) 193.14.56.22 – Class C

c) 14.23.120.8 – Class A

d) 252.5.15.111 – Class E

Ques 2 Find the class of each address

Binary Notation

a) 00000001 00 . . . . → Class A

b) 11000001 100 . . . . → Class C

c) 10100111 110 . . . . → Class B

d) 11110011 100 . . → Class E

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

Ques 3 Find the class of each address

a) 00000001 00001011 00001011 11101111

b) 11000001 10000011 00011011 11111111

c) 14.23.128.8

d) 252.5.15.111

Ans:- a) The first bit 0. This is class A address

b) The first 2 bits are 1, third bit is 0. This class C address

c) The first byte is 14 (between 0 and 127); the class A

d) The first byte is 252 (between 240–255). the Class E

NETID and HOSTID

Only class A, B or C is divided into Netid and hostid. The concept does not apply to class D+E

IMP

Class	Byte 1	Byte 2	Byte 3	Byte 4
Class A	Netid	Hostid	Hostid	Hostid
Class B	Netid	Netid	Hostid	Hostid
Class C	Netid	Netid	Netid	Hostid

~~Netmask~~:-

* In class A, one byte defines Netid and three bytes define the hostid.

* In class B two bytes define the Netid and two bytes define the hostid.

* In class C, three bytes define Netid and one byte define the hostid.

Mask :- It helps to find netid and Hostid. Mask (also called default Mask), a 32 bit made of contiguous 1s followed by contiguous 0s. The mask for classes A, B, and C shown below. The concept does not apply to classes D and E.

Classless Interdomain Routing

Class	Binary (Mask)	Decimal	CIDR
Class A	11111111 00000000 00000000 00000000	255.0.0.0	/8
Class B	11111111 11111111 00000000 00000000	255.255.0.0	/16
Class C	11111111 11111111 11111111 00000000	255.255.255.0	/24

Default Mask for Classful Addressing.

n bits = 1   A = n = 8
(32 - n) = 0   B = n = 16
C = n = 24

The last column shown in table the mask in the form /n where n can be 8, 16, 24 in classful addressing. This notation is also called slash Notation or Classless Interdomain Routing (CIDR) notation.

Class D address used for Multi Cast purpose
Class E address used for Reserved for future used.

Classless Addressing :- In this variable length blocks are used that belongs to no class.

Restriction :-

To simplify the handling of addresses, the Internet authorities impose three Restriction on classless address blocks.

a) The address in a block must be contiguous, one after another.

b) The Number of address in block must be a power of 2 (1, 2, 4 ... 8)

c) The first address must be evenly divisible by the Number of addresses.

Example - Show a block of addresses, in both binary and dotted decimal notation granted to a small business that needs 16 addresses.

A block of 16 address granted to a small organization

	Block
First →	205.16.37.32
	205.16.37.33
	205.16.37.34
	⋮
Last →	205.16.37.47

a) Decimal

Block
11001101 00010000 00100101 00100000
11001101 00010000 00100101 00100001
11001101 00010000 00100101 00100010
⋮
11001101 00010000 00100101 00101111

16 Addresses

b) Binary

· $16 = 2^4$ clear

We can see the Restriction are applied to this Block. The address are contiguous. The No. of address are power of 2 ($16 = 2^4$). The first address is divisible by 16. The first Address when converted to a decimal Number is 3,440,387,360 which when divided by 16 result is 215,024,210.

Mask - Can take any values from 0 to 32.
Mask is 32 bit number in which the n leftmost bits are 1s, and (32-n) rightmost bits are 0's

↳ Slash / CIDR Notation
$x.y.z.t/\underline{n}$ — defines the mask

Note:-

In IPV4 addressing, a block of addresses can be defined as $x.y.z.t/\underline{n}$
in which x, y, z, t defines one of the addresses and /n defines the mask

Imp point:-

i) First address in block can be found by setting the 32-n right most bits in Binary notation of the address to 0's,

ii) Last address can be found by setting (32-n) the Rightmost (32-n) bits to 1's.

iii) Number of address in the block in the difference between the last and first address. It can be found using the formula 2^{32-n}

Example - One of the address is $205.16.37.39/\underbrace{28}_{n}$. What is the first address in the block.

Solution - Binary Representation of given address is 11001101 00010000 00100101 00100111.

① First Address → 11001101 00010000 00100101 00100000
→ 205.16.37.32

00101111
32 16 8 4 2 1

② Last Address 205.16.37.47

32
15
47

③ No. of address → $2^{32-n} = 2^{32-28} = 2^4 = 16$.

One More Way of Extracting Block Information

i) No. of address, $N = 2^{32-n}$

ii) First address = (Address) AND (MASK)

iii) Last address = (Address) OR (Complement of Mask)

Example – For address 205.16.37.39/$\underset{n}{28}$. Find

1) First Address

2) Last Address

3) No. of Address $= 2^{32-28} = 2^4 = 16$ Address.

1) First Address :-

Address:	11001101	00010000	00100101	00100111
Mask:	11111111	11111111	11111111	1111 $\underset{4}{0000}$
First Address:-	11001101	00010000	00100101	00100000

AND = 1+1=1

2) Last Address

Mask	11001101	00010000	00100101	00100111
Complement	00000000	00000000	00000000	00001111
	11001101	00010000	00100101	00101111

OR → 0+0=0
1+0=1
0+1=1
1+1=1

A Comparison

Table B.1 shows how systems represent the decimal numbers 0 through 15. As you can see, decimal 13 is equivalent to binary 1101, which is equivalent to hexadecimal D.

Table B.1 *Comparison of three systems*

Decimal	*Binary*	*Hexadecimal*
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

B.4 BASE 256: IP ADDRESSES

One numbering system that is used in the Internet is base 256. IPv4 addresses use this base to represent an address in dotted decimal notation. When we define an IPv4 address as 131.32.7.8, we are using a base-256 number. In this base, we could have used 256 unique symbols, but remembering that many symbols and their values is burdensome. The designers of the IPv4 address decided to use decimal numbers 0 to 255 as symbols and to distinguish between the symbols, a *dot* is used. The dot is used to separate the symbols; it marks the boundary between the positions. For example, the IPv4 address 131.32.7.8 is made of the four symbols 8, 7, 32, and 131 at positions 0, 1, 2, and 3, respectively.

> **IPv4 addresses use the base-256 numbering system.**
> **The symbols in IPv4 are decimal numbers between 0 and 255; the separator is a dot.**

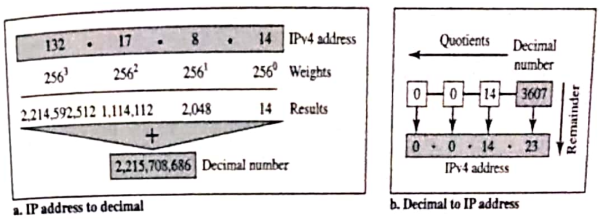
Weights

In base 256, each weight equals 256 raised to the power of its position. The weight of the symbol at position 0 is 256^0 (1); the weight of the symbol at position 1 is 256^1 (256); and so on.

Conversion

Now let us see how we can convert hexadecimal to decimal and decimal to hexadecimal. Figure B.4 show the two processes.

Figure B.4 *IPv4 address to decimal transformation*



To convert an IPv4 address to decimal, we use the weights. We multiply each symbol by its weight and add all the weighted results. The figure shows how the IPv4 address 131.32.7.8 is transformed to its decimal equivalent.

We use the same trick we used for changing decimal to binary to transform a decimal to an IPv4 address. The only difference is that we divide the number by 256 instead of 2. However, we need to remember that the IPv4 address has four positions. This means that when we are dealing with an IPv4 address, we must stop after we have found four values. Figure B.4 shows an example for an IPv4 address.

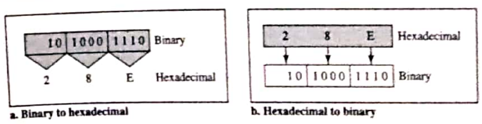
B.5 OTHER CONVERSIONS

There are other transformations such as base 2 to base 16 or base 16 to base 256. It is easy to use base 10 as the intermediate system. In other words, to change a number from binary to hexadecimal we first change the binary to decimal and then change the decimal to hexadecimal. We discuss some easy methods for common transformations.

Binary and Hexadecimal

There is a simple way to convert binary to hexadecimal and vice versa as shown in Figure B.5.

Figure B.5 *Transformation from binary to hexadecimal*



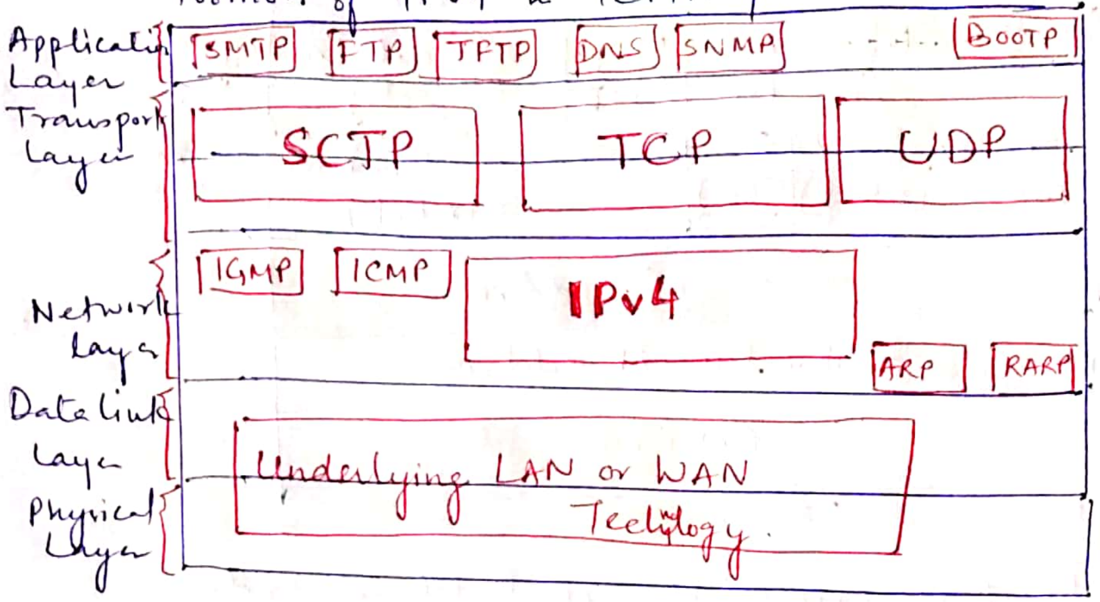
To change a number from binary to hexadecimal, we group the binary digits from the right by fours. Then we convert each 4-bit group to its hexadecimal equivalent.

Date - 24/3/2020 ①

IPv4 Protocol : IP Packet / Datagram Format

The Internet Protocol Version 4 (IPv4) is the delivery Mechanism used by the TCP/IP protocols. Used by TCP/IP protocol at N/W layer
↳ Unreliable and Connectionless.

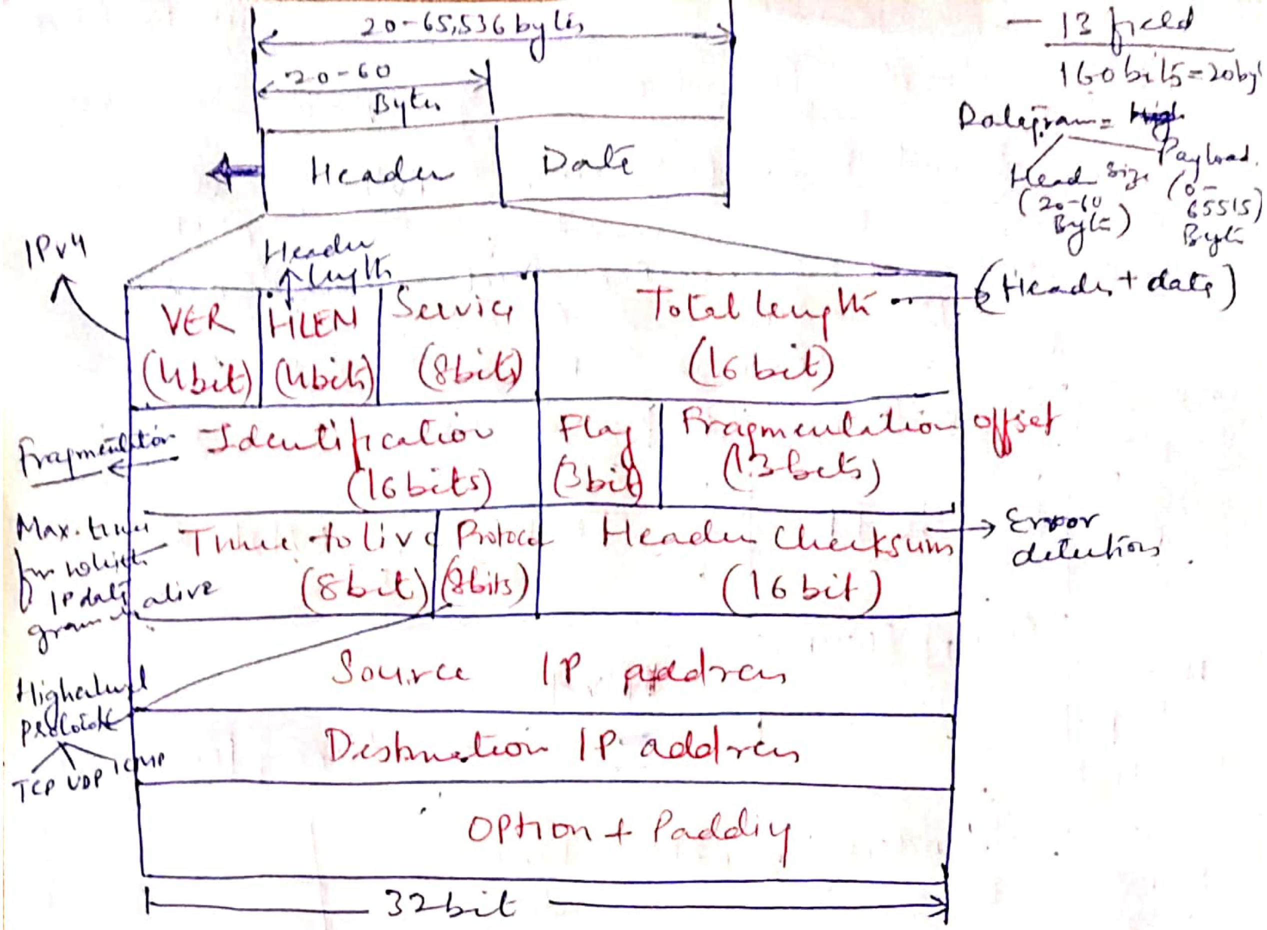
Position of IPv4 in TCP/IP protocol suite



IPv4 is an Unreliable and Connectionless datagram protocol - a best effort delivery service. The term best effort means that IPv4 provides no error control or flow control (except for error detection on the header). IPv4 assumes the unreliability of the underlying layer and does its best to get a transmission through to its destination, but with no guarantees.

Datagram :-

- Packet in the IPv4 layer are called Datagrams shows the IPv4 datagram format.
- A datagram is a variable length packet Consist of two part : header and Data
- The header is 20 to 60 bytes in length and Contains information essential to routing and delivery



A brief description of each field in order

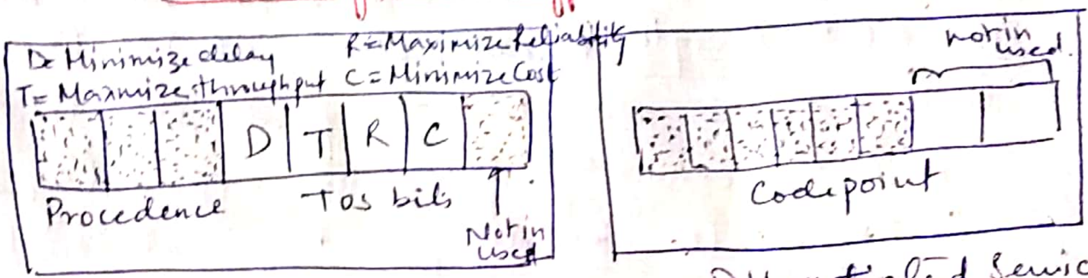
1) Version (VER):- This 4bit field defines the Version of the IPv4 protocol. Currently the version is 4. However, version 6 (or IPng) may totally Replace Version 4 in the future. This field tells the IPv4 S/W running in the Processing machine that the datagram has the format of version4.

2) Header Length (HLEN):- This 4 bit field defines the total length of the datagram header in 4byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes).

* When there are no options, the header length is 20 byte, and the value of this field is 5 (5x4=20).
* When the option field is at its Maximum Size, the value of this field is 15 (15x4 = 60).

3) Service - IETF has changed the Interpretation and name of this field 8bit field. This field previously called Service type, is now Called differentiated Service

Service type or differentiated Service



Differentiated Service

1. Service Type :-

In this Interpretation, the first 3 bits are called Precedence bits. The next 4 bit are called type of Service (TOS bits) and last bit is not used.

a) Precedence is a 3bit subfield ranging from 0 (000 in binary) to 7 (111 in binary). The precedence defines the priority of Datagram in issues such as Congestion. If a router is Congested and needs to discard some datagram, those datagram with lowest procedence are discarded first.

b) TOS bits - is a 4bit subfield with each bit having a special meaning. Although a bit can be either 0 or 1, one and only one of the bit can have value of 1 in each datagram. Since, we can have five with only 1 bit set a time, we can have five different types of Service (TOS.)

Types of Services

Tos bit	Description
0000	Normal (default)
0001	Minimize cost - C
0010	Maximize Reliability - R
0100	Maximize throughput - T
1000	Minimize Delay - D

2) Differentiated Service:-

In this Interpretation, the first 6 bit make up the codepoint subfield. and last 2 bit are not used.

4) Total length:- This is a 16 bit field that define the total length (header plus data) of the IPv4 datagram in bytes. To find the length of Data comming from the upper layer, Subtract the header from the total length. The header length can be found by Multiplying the value in HLEN field by 4.

$$\boxed{\text{length of data} = \text{total length} - \text{header Length}}$$

The total length field define the total length of the datagram including the header.

Since field length 16 bit, total length of IPv4 datagram is limitted to 65535 ($2^{16}-1$) bytes. of which 20 to 60 bytes are the header and the rest is Data from upper layer.

5) Identification:- This field is used in Fragmentation

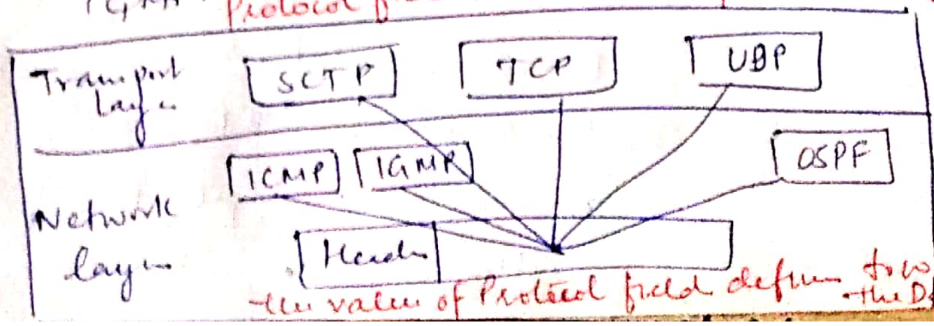
6) Flags - This field used in Fragmentation

7) Fragmentation offset - This field is used in Fragmentation

8) Time to live - A Datagram has a limitted lifetime in its travel through an Internet.

9) Protocol - This 8 bit field define the Higher-Level protocol that uses the Service of IPv4 layer. Higher level protocol such TCP, UDP, ICMP and IGMP.

Protocol field and Encapsulated Data



The value of Protocol field define to which Protocol the Data belongs

The value of this field for each higher level protocol :

Value	Protocol
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

Protocol value

10) Checksum The checksum concept and its calculation

11) Source address :- The 32bit field defines the IPv4 address of the Source. This field must Remain unchanged during the time the IPv4 datagram travel from the Source host to destination host

12) Destination address :- This 32bit field defines the IPv4 address of the destination. This field must remain unchanged during the time IPv4 datagram travels from the source host to Destination host.

Question ① An IPv4 packet has arrived with the first 8 010000 10. The Receiver discard the packet. why?

Answer - There is error in this packet. The 4 leftmost bits (0100) show version, which is correct. The Next 4 bits (0010) show invalid header length (2×4 = 8) the Minimum number of byte in the header must be 20 - the packet has been corrupted in Transmission.

Question ② In IPv4 packet the value of HLEN is 1000 in binary. How many bytes of options are being carried by this packet.?

Solution - The HLEN value is 8, which means the total Number of bytes in header 8 × 4 or 32 bytes. The first 20 bytes are base header, the Next 12 bytes are the option.

26/3/2020 ①

IPv6 Addresses:- $2^{128} \rightarrow 128\text{bit} \rightarrow 16$ bytes

It is of 128bits or 16 bytes. length is ④ times the length address of IPv4.

Notations

1) Dotted Decimal:- It is used for IPv4 compatibility.

221.14.65.11.105.45.170.34.12.234.18.0.14.0.115.225 [16 digit]

IMP

ii) Colon Hexadecimal:- It is used to make the address more readable. In this notation on the 128 bits are divided into 8 sections, each of 2 bytes in length. [Two bytes in Hexadecimal required 4 Hexadecimal digits].

FDEC : BA98 : 7654 : 3210 : ADBF : BBFF : 2922 : FFFF [8]

Abbreviation :- [Zero Compression]

It is a Technique to reduce the length of IPV6 address. It is done by omitting / removing the leading Zeros of a section.

[Note: Only leading zero can be dropped] u can dropped trailing zero. [This is known as Zero Compression]

Suppose an IP address is given below - IPV6

FDEC : 0074 : 0000 : 0000 : 0000 : BOFF : 0000 : FFF0

omitting these zero

FDEC : 74 : 0 : 0 : 0 : BOFF : 0 : FFF0 } This address is called as Abbreviated address.

Now you can more abreviated this address by using colon. In this combination of zero will be remove.

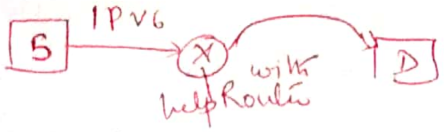
FDEC : 74 : : BOFF : 0 : FFF0

↳ GAP.

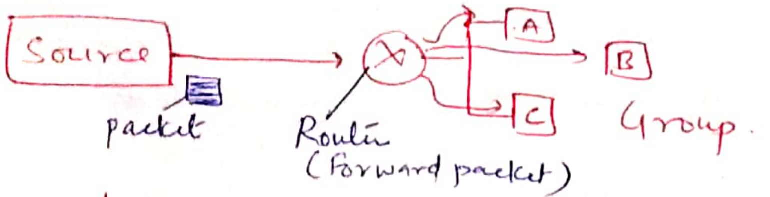
Abbreviated mean to make IPV6 address small.

Types of Address space in IPV6:-

i) Unicast Address: (single) It defines single Interface or Computer. The packet sent to a unicast address will be routed to the intended PC or Recipient.

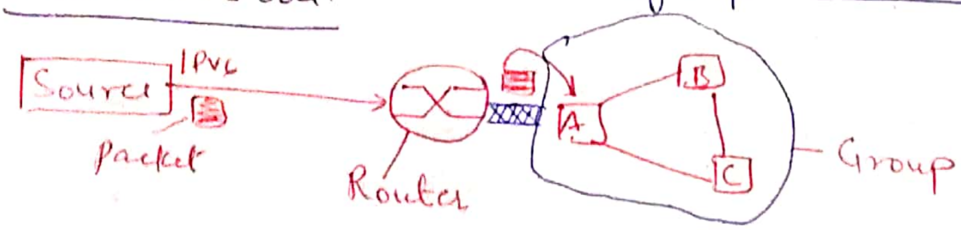


ii) Multicast Address:- These are used to define a group of computer / hosts. In this, each member of group receives the packet



In Multicast each member of group received the packet.

iii) Anycast Addresses:- Defines group of nodes or Computer that all share a single address. A packet with anycast address is delivered to only one member of the group which is the most reachable one.

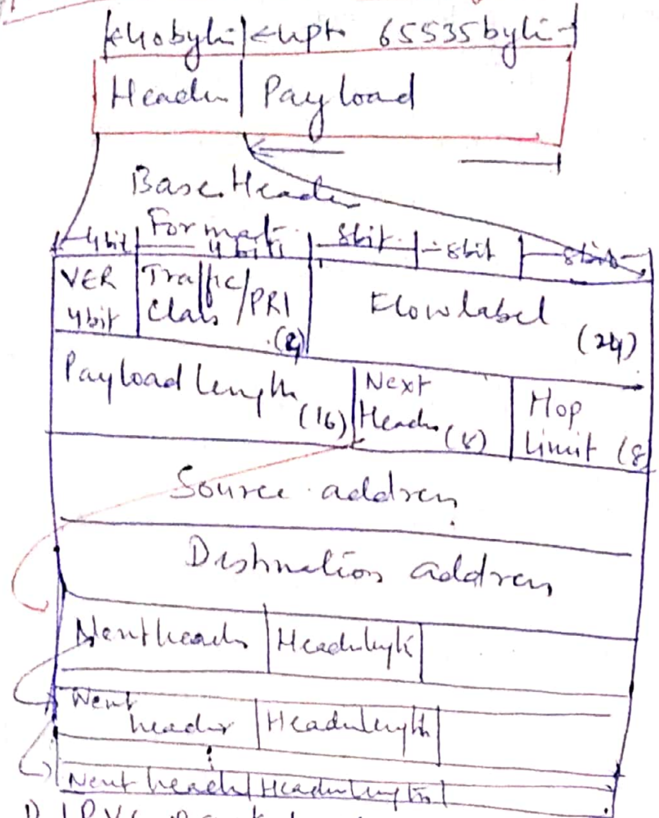


A is near to Router

(iv) Broadcasting and Multicasting IPV6 does not define Broadcasting and considered it as an special Case of Multicasting (special case consider)

(v) Reserved Addresses:- Reserved address are starts with 8 0's.

IPV6 Protocol: Packet format / Datagram format



* VER – IPV6 → (6) (0110)
* Traffic class – Distinguish different Payload
* Flow label:- Provide special handling for Particular data flow
* Payload length – length of IP datagram – Base header.
* Next header:- It define the header of the Base header. For Example code – 2 – ICMP, 6 – TCP, 17 – UDP
* Hop limit – Time to live.

1) IPV6 packet shown above. Each packet is composed of a mandatory Base header followed by the payload. The payload consist of two parts: optional extention header and data from an upper layer.

2) The Base header occupies 40 bytes, whereas the extension header and data from upper layer contain upto 65535 bytes of information.

Base header Base header has Eight fields. These field are as follows:

1) **Version** – This is 4 bit field address define the version number of the IP. For IPV6, the value is 6

2) **Priority / Traffic class**:- This 4 bit priority field define the priority of the packet with respect to traffic congestion.

3) **Flow label** – Flow label is 3 byte (24 bit) field that is ~~defined~~ designed to provide special handling for particular flow of Data.

4) **Payload length**: This 2 byte payload length field defines the length of the IP datagram excluding Base header.

Payload length = length of IP datagram – Base header

5) Next header :- The Next header is an 8bit field defining the header that follows the Base header in the datagram. The Next header is either one of the optional extension header used by IP or the header of an encapsulated packet such as UDP or Next header.

6) Hop limit This is 8bit hop limit field serve the same purpose as the TTL field in IPV4. Time to live

7) Source Address The source Address field is a 16 byte (128bits) Internet Address that identifies the original source of the datagram.

8) Destination Address :- The destination address field is a 16 byte (128bit) Internet address that usually identifies the final destination of the datagram. However if source routing is used, this field contain the address of the Next router.

Next header codes for IPV6

Code	Next header
0	Hop-by Hop option.
2	ICMP
6	TCP
17	UDP
43	Source Routing
44	Fragmentation
50	Encrypted Security Payload
51	Authentication
59	NULL (No next header)
60	Destination option

Advantages / Changes in IPV6 :-

- The Next generation IP or IPV6 has some advantages over IPV4 that can be summarized

(a) Larger address space : An IPV6 address is 128 bits long. Compared to with the 32 bit Address of IPV4. This has huge (2^{96}) increase in address space.

$2^{128} - 2^{32} = 2^{96}$ time more address than IPV4.

(b) Better header format : IPV6 uses a New header format in which options are separated from the Base header and inserted, when needed between Base header and upper layer data

Options are separated from Base head

(c) New options - IPV6 has new option to allow for additional functionalities.

(d) Allowance for Extension :- IPV6 is designed to allow the Extension of the protocol if required by new technologies or application.

(e) Support for resource allocation IPV6 the type of service field (TOS field) has been removed, but mechanism of (called flow label) has been added to enable the source to request special handling of packet. This Mechanism can be used to support traffic such as Real time audio and vedio.

Support for resource allocation — Traffic Class / Flow label — special Handling of the packet.

(f) Support for more security

The encryption and authentication option in IPV6 provide confidentiality and Integrity of the packet.

Difference Between IPV4 and IPV6

IPV4	IPV6
1) Length of Address 32bit	1) Length of Address 128bit
2) Represent in Decimal notation	2) Represented in Hexadecimal Notation
3) IPsec Support optional	3) IPsec support Inbuilt is more secure
4) Packet flow indication —NONE	4) Packetflow —Yes → with the help of flow label
5) Checksum field —Yes	5) Checksum field —NONE
6) Option field—YES	6) Option field —NONE; does have IPV6 Extension header.
7) Address (IP) to MAC → (ARP)	7) Replaced By - NDP Neighbour Discovery Protocol
8) Broadcast Message—YES	8) Broadcast - special type of Multicast Address
9) Total no. of address $= 2^{32}$	9) Total No. of address $= 2^{128}$

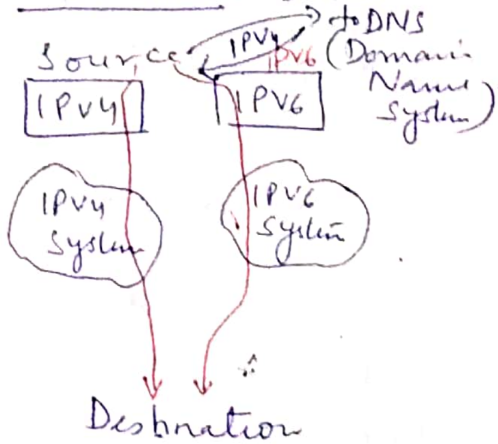
Scanned with CamScanner

Transition from IPV4 to IPV6 :- There are three transition strategies.

① Dual Stack

Tunneling

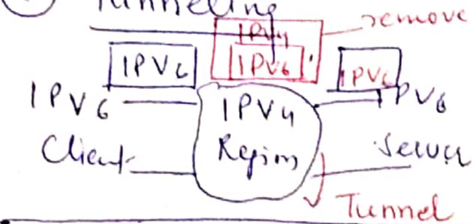
Header Translation



⇒ To determine which version to use when sending a Packet to destination the source host queries the DNS.

* If DNS return an IPV4 address then the source host sends an IPV4 packet. If the DNS return IPV6 address, the Source host send an IPV6 packet

② Tunneling

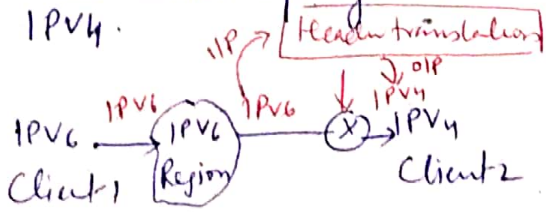


IPV6 Packet is encapsulated in IPV4 Packet when its Enter in IPV4 Region

⇒ ① Tunneling is a strategy used when two computer using IPV6 want to communication with each other and packet must pass through a region that uses IPV4. To pass through this Region, the packet must have IPV4 address. So the IPV6 packet is Encapsulated in IPV4 Packet when it enter the Region & it decapsulate when exist the Region.

③ Header Translation

When most of system are on IPV6 but some still uses IPV4.



Header translation is necessary when the majority of the Internet has moved to IPV6 but some system still use IPV4. The sender wants to use IPV6, but the reciever does not understand IPV6. Tunneling does not work in this situation because the packet must be in IPV4 format to be understood by the Reciever. In this case the header format must be totally changed through header translation the header of IPV6 packet is converted to an IPV4 packet.

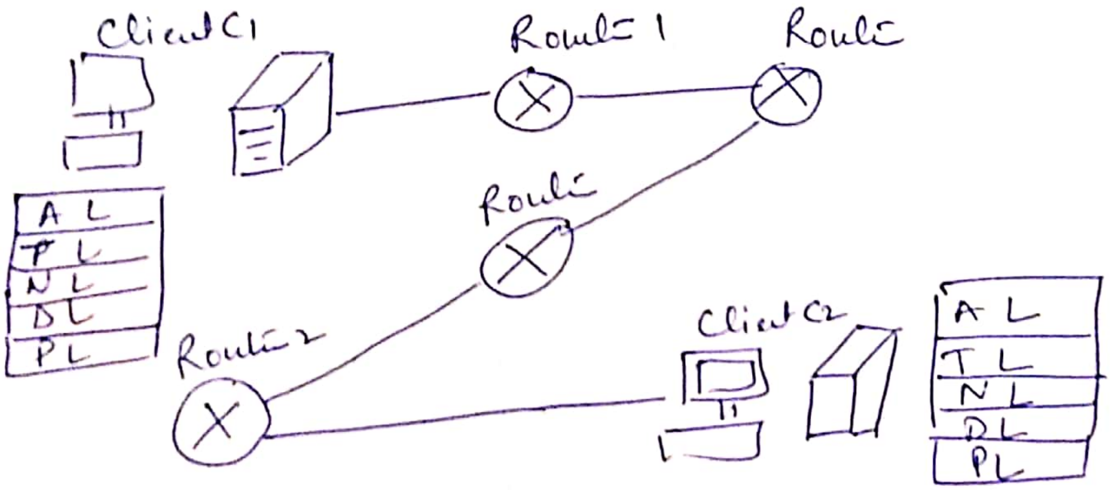
UNIT-4 (NETWORK LAYER)

NETWORK LAYER DESIGN ISSUES:-

Network layer is third layer in TCP/IP model. It provide service for delivering the packet from source to destination.

- For delivery the packets from source to destination it provide the routing mechanism. it decide the path that a packet has to traverse.

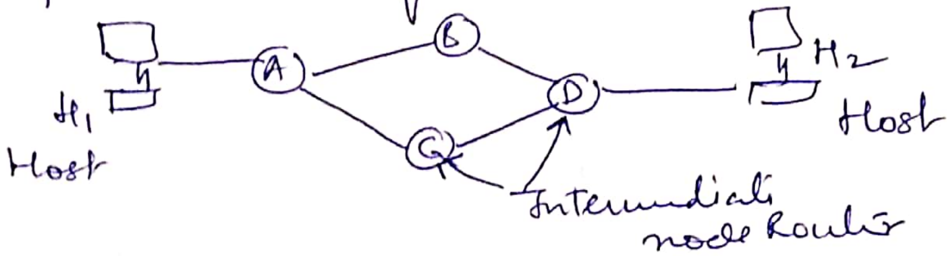
Suppose we have two client C1 and C2 and both are in different Networks. There can be many No. of routers on path C1 and C2.



Now suppose C1 want to communicate with C2 and wants to send some data to C2. First of all the N/W layer of client C1 takes segment from Transport layer of C1 & encapsulate the segment in the Datagram and passes it to the nearby route 1. The N/W layer of route 1 take the datagram and passes to nearby route. The nearby router takes the datagram and passes it further and so on. Finally the datagram is recieved by the Network layer of client C2. Here, the segment is extracted from the datagram and is delivered to the transport layer of Client C2.

Network layer Design Issues:-

* Store and forward Packet switching
* Services Provided to the Transport layer.
* Implementation of Connectionless Service
* Implementation of Connection-Oriented



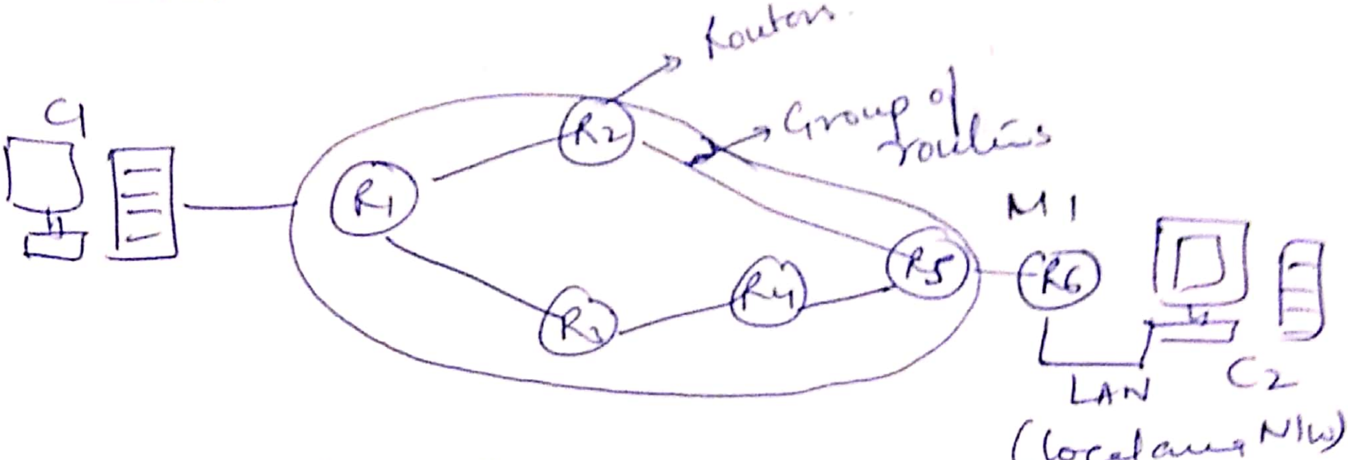
a) Store and forward Packet Switching.
Station A packet is arrive and forwarded to B or C.

b) Services provided to Transport layer.
Network layer provided Service to Transport layer
The Service need to be Carefully designed with following goals.
a) Service must be independent from the subnet Technology
b) The Transport layer should be shielded from the number, type and topology of the subnet.
c) The Network address should be uniform across the Network.

c) Implementation of Connectionless Service.
In Connectionless Service as it name indicate No connection setup is needed. Packet are sent directly to the Network and routed to the other client. In Connectionless services text packet are called datagram (Such as Message or Telegram) and the subnet/network is called datagram subnet.
For Example: There are two clients C1 and C2 C1 wants to send a Message to C2. Message

divided into packets such as M1, M2, M3, M4.
Network layer of client C1 sends each packet to its nearby router R1.

Every router has its internal routing table which tells which path is to be selected among all available.



Every router has Routing table contain two entries (i) Destination router and path entry.
Path Entry tells router name via which packet can reach the Destination.

Router R1 table.

Destination Entry	Path Entry
R1	–
R2	R2
R3	R3
R4	R3
R5	R2
R6	R2

Client C1 starts sending the packet to R1 (nearby router). When packet arrives at R1, R1 checks its internal table for Next suitable router. Packets have to reach at R6 for being available to client C2. In Routing table there is a path to R6 through R2. R1 sends M1, M2 and M? to R2.

R_2 table

Destination Entry	Path Entry
R_1	R1
R_2	—
R_3	R_3
R_4	R_4
R_5	R_5
R_6	R_5

R2 table

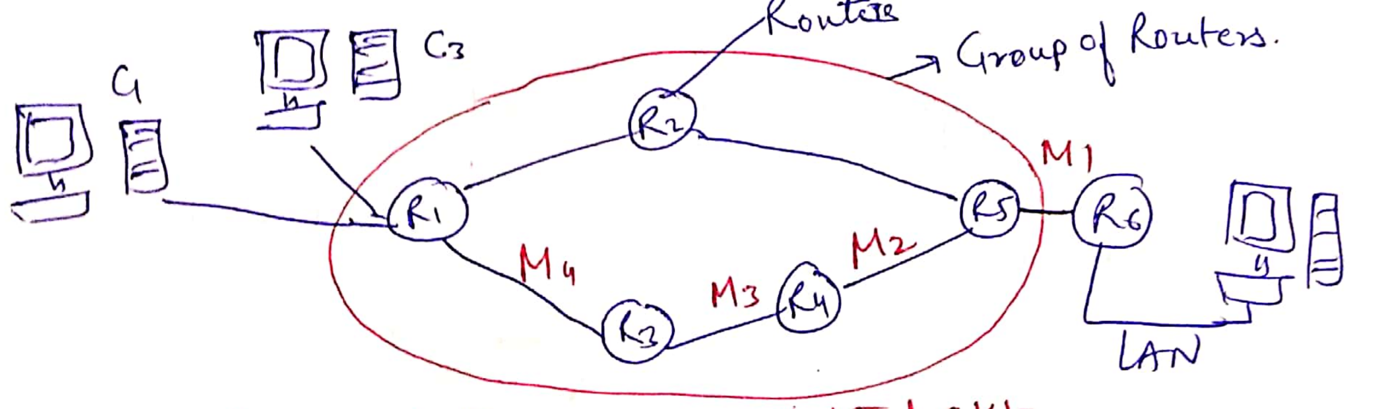
Destination Entry	Path Entry
R1	R1
R2	—
R3	R3
R4	R4
R5	R5
R6	R5

d) Implementation of Connection Oriented Service

In connectionless service, packet can be sent from source to destination through different routing since a particular route is not specified bef transmission takes place.

- But in case of connection oriented service a source has to send some data packet to destination It must create a virtual ckt. Virtual ckt can be created with connection establishment.

When a connection is established, a route from source to destination is choosen and route is used for flowing all traffic. When connection is released, the virtual ckt is also terminated.



Routing datagrams in Virtual ckt (Connection oriented Service).

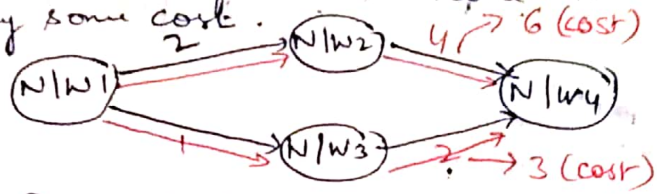
Routing Protocols :- (Unicast Communication)

It means Communication b/w one sender and one reciever. [one to - one Communication] so it is called Unicast Protocols. We study different Protocol RIP(DV), OSPF (LS), BGP (PV)

Basic Terms :-

cost of metric is Minimum.

i) **Cost or Metric** - It is assigned for passing the N/w Our packet travel from one node to other node in N/w. it pay some cost.



2) **Static vs Dynamic Routing Tables :-**

Static

1) It has mannual Entries.

Dynamic

2) But in Dynamic table entries are automatically update.

↳ It gets updated when there is ~~only~~ any Change due to route is destroy or cost is decrease.

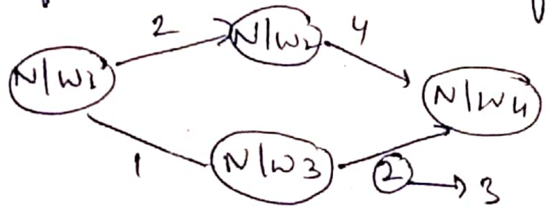
* Now a day Dynamic Routing table are used.

3) **Routing Protocol** :- If we have multiple no. of route we can select ophmized route is selected. Routing protocol Choose minimize route from N/w.

Routing Protocol → Rule, Procedures

1) Change is route information. is inform to other routers.

2) Combine information received from other routers.



Routing protocol divided into two

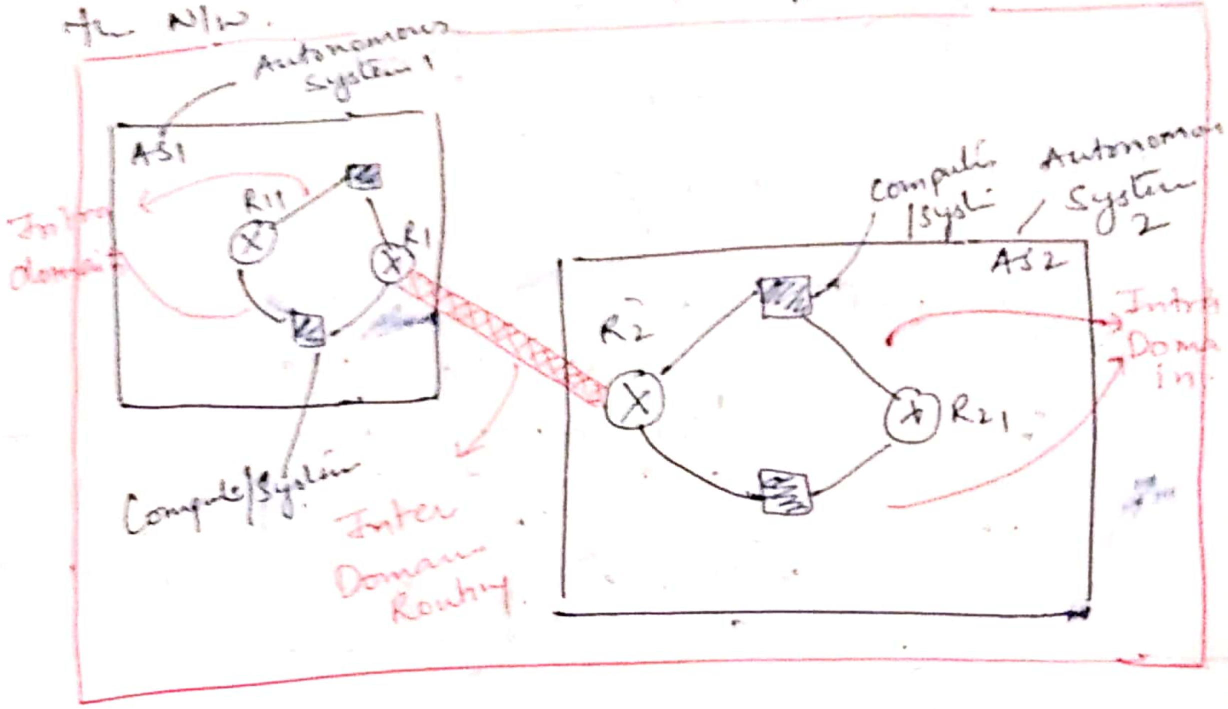
Interior (Intra) Between

Exterior (Inter) outside

Intra- and Inter Domain Routing :-

Internet is divided into Autonomous System (AS) which is a group of N/W and Routers under authority of single administration.

Suppose consider Example of college have different department IT, CSE, EC, EEE are under single authority by Principal. Single user can use resources is difficult to manage the N/W.



Address Maping: ARP and RARP

Address Resolution Protocol ARP

ARP associates an IP address with Physical address

logical Address $\xrightarrow[(ARP)]{Mapping}$ Physical Address

IMP: As 'IP' uses the Service of Data link layer it needs to know the physical address of the Next Hop ⟹ ARP (Address Resolution Protocol)

Mapping of IP address into MAC Address

Static Mapping | Dynamic Mapping (ARP, RARP)

Static Mapping

1) Table is created with logical + physical Address

This table is stored in each machine on the N/W. For Example IP address of another ~~address~~ machine but not its physical address, can Look it up in the table.

Some limitation

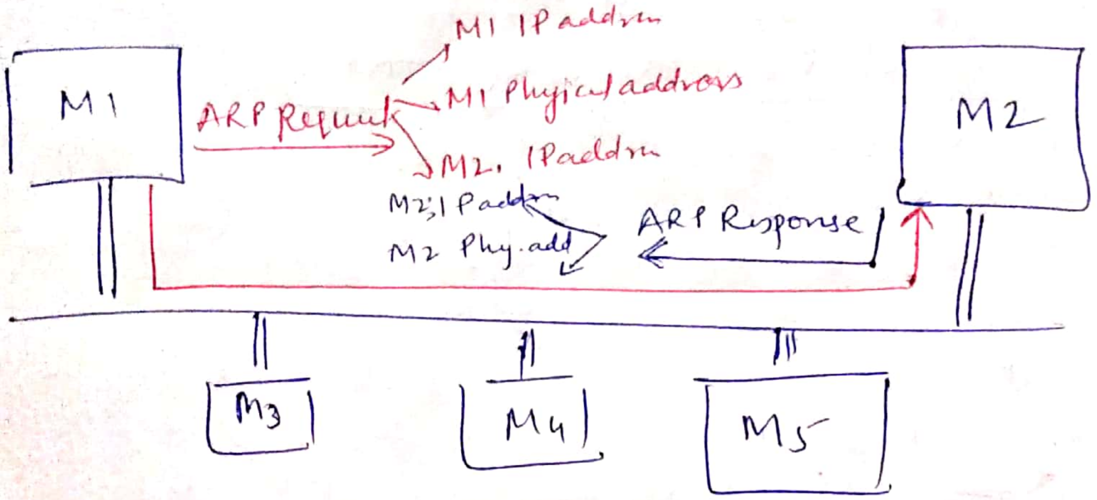
1) A machine could change its NIC resulting a new physical Address
2) Some LAN, such as LocalTALK, the Physical address changes Every time the Computer is turned on.
3. A mobile computer can move from one physical N/W to another, resulting in change its physical Address

Dynamic Mapping

1). Each time a m/c knows the logical address of another m/c, It can use a protocol to find physical Address.

ARP Protocol :-

ARP accept a logical address from the IP protocol maps the address to the Corresponding physical address and pass it to the data link layer.

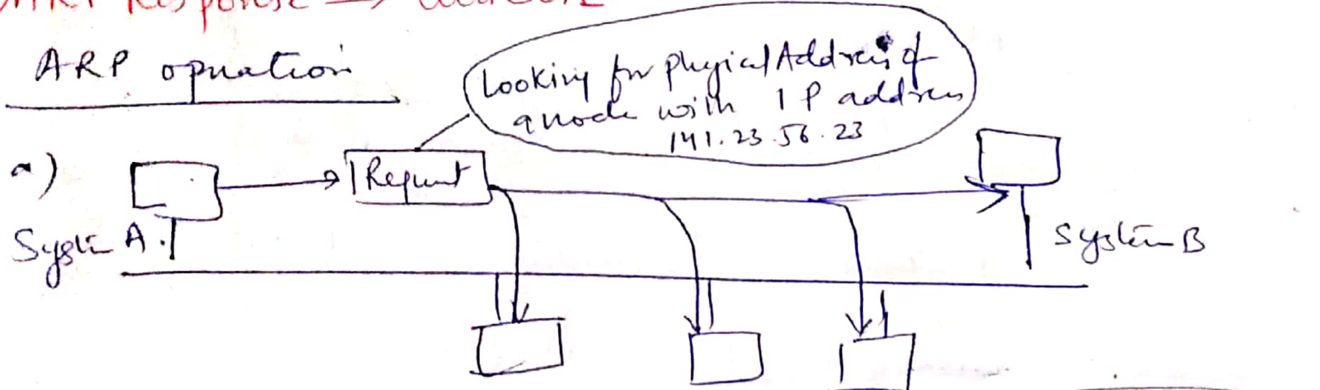


ARP protocol working

* ARP Request → Broadcast
* ARP Response → Unicast

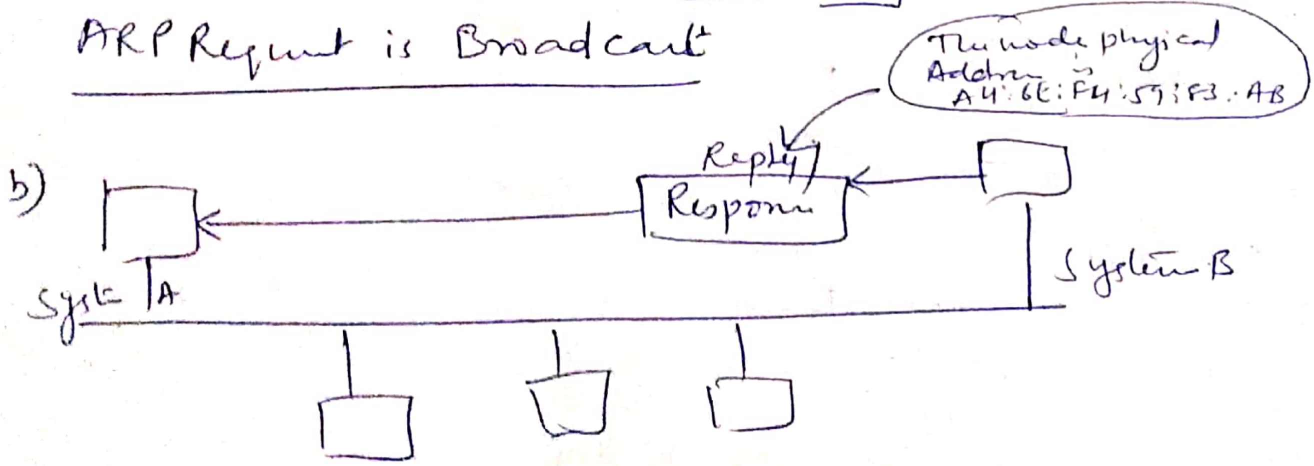
ARP operation

a)



ARP Request is Broadcast

b)



ARP Reply is Unicast

ARP operation :- Anytime a HOST or a Router needs to find a physical address of another another Host or Router on its N/W, it sends an ARP Query packet. The packet include the physical and IP address of the sender and the IP address of the Receiver. Query is broadcast to the Network. When the intended recipient recognizes its IP address it sends back an ARP Response packet which contain both IP and physical Address.

ARP packet format :-

8 bit	8 bit	16 bit
Hardware type		Protocol type
H/w length	Protocol length	Operation Request 1, Reply 2
Sender H/w address (For Example, 6 byte for Ethernet)		
Sender Protocol Address (For Example, 4 byte for IP)		
Target H/w address [Not filled] (For Example, 6 byte for Ethernet)		
Target Protocol Address (For Example, 4 byte for IP)		

32 bits

- Type of N/W
- Length of physical address
- Length of logical Address

- IPV4 / IPV6 ARP run
- Request(1) Reply(2)
- Physical address of Sender
- Logical IP address of Sender
- In a Request
- Recipient IP / logical Address

The field are as follows (9)

1) H/w type :- This is 16 bit field defining the type of the N/W on which ARP is ~~running~~ running. For Example Ethernet is given type 1.

2) Protocol type :- This is 16 bit field defining the protocol For Example, the value of this field IPV4 protocol is (0800_{16}). IPV4 / IPV6 ARP run on system.

3) Hardware length This is 8 bit field defining the length of Physical address in byte. For Example for Ethernet the value is 6.

4) Protocol length :- This is 8 bit field defining the length of the logical address in byte. For Example IPV4 protocol value is 4
IPV6 protocol is value 6

5) operation :- This 16 bit defining the type of packet. Two packet types are defined : ARP Request (1)
ARP Reply (2)

6) Sender H/w address :- This is Variable length field defining the physical address of the Sender.

7) Sender Protocol Address :- This is variable length field defining the logical address (for Example IP) address of the Sender

8) Target H/W address - This is variable length field defining the physical address of the target For Example : Ethernet the field is 6 byte long.

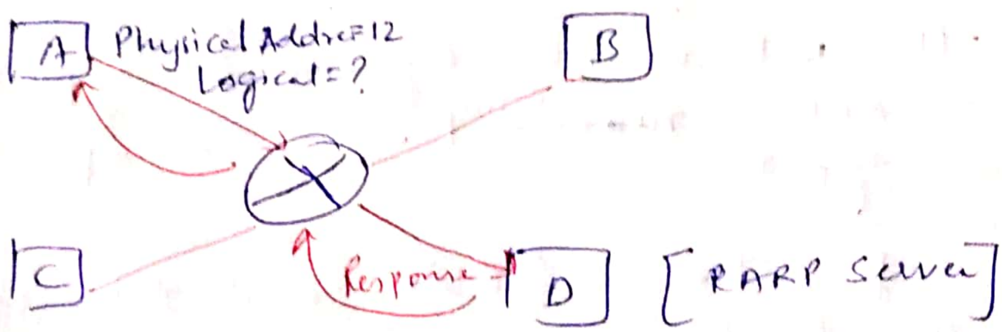
9) Target Protocol Address :-

This is variable length field defining the logical (For Example IP) address of the target. For IPv4 protocol this field is 4 byte long.

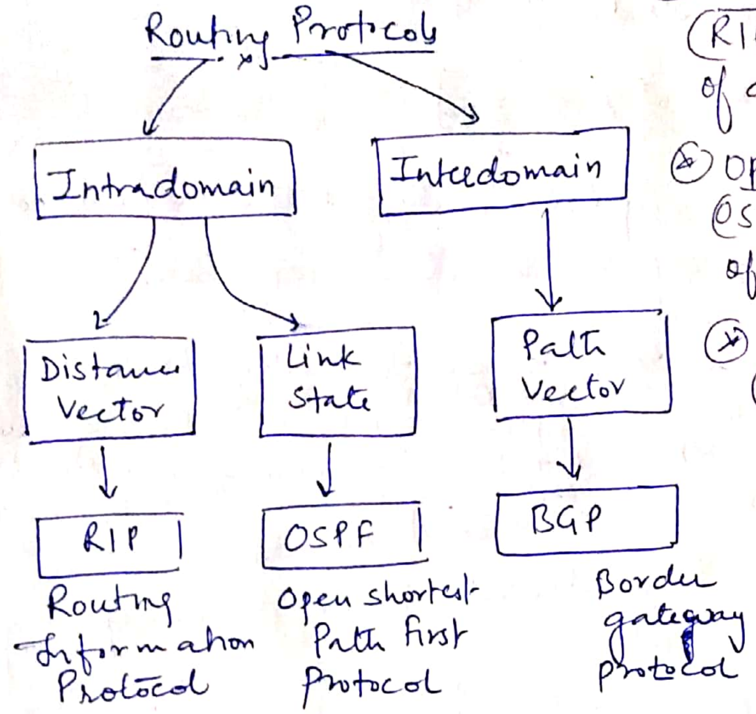
Reverse Address Resolution Protocol (RARP)

'RARP maps a physical address to a logical / IP address

$$\text{Physical Address} \xrightarrow[\text{RARP}]{\text{Mapping}} \text{Logical / IP address}$$



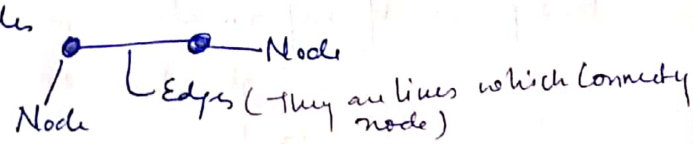
Routing Protocol



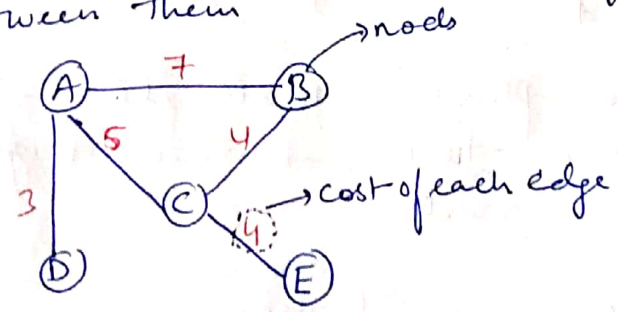
* Routing Information Protocol (RIP) is implementation of distance vector Protocol.
* Open Shortest Path first (OSPF) is implementation of link state protocol
* Border Gateway Protocol (BGP) is implementation of path vector Protocol.

Distance Vector Routing :- It sees as AS (Autonomous System) with all the routers and N/w as a Graph

Graph { sets of nodes / Edges



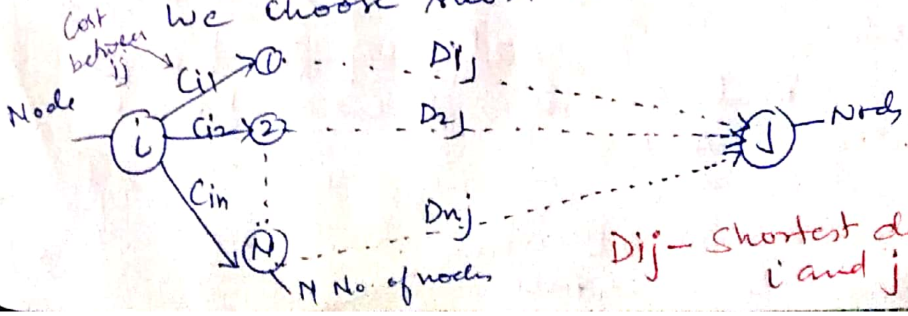
Bellman-Ford Algorithm :- Used to find shortest path between nodes in a graph given distance between them



For Example we want to communicate (A) to (E) we check the path 1) Path 1 - $A \xrightarrow{7} B \xrightarrow{4} C \xrightarrow{4} E$ cost is (15)

Path 2 - $A \xrightarrow{5} C \xrightarrow{4} E$ cost (9)

We choose short distance path bet A to E.



$D_{ij} = \text{Minimum} \begin{cases} (C_{i1} + D_{1j}) \\ (C_{i2} + D_{2j}) \\ \vdots \\ (C_{in} + D_{nj}) \end{cases}$

D_{ij} – Shortest distance between i and j.

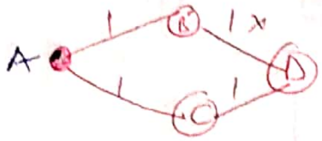
Distance Vector Routing Algorithm

1) Cost is normally Hop counts → No of N/w crossed.

[A→C = 2]

2) Each Router needs to update routing table asynchronously [when it recieves information from neighbour]

3) After update, result is sent to all the Neighbour.



B to D link is damage.
Suppose B→D link is damage then A table is shared with B & C
Now C get knows that B→C link has damage.

4) Each router should keep atleast three pieces of information for each route.

- Destination N/w → D
- Cost → 2
- Next hop → B

5) Two piece of information is received via update.

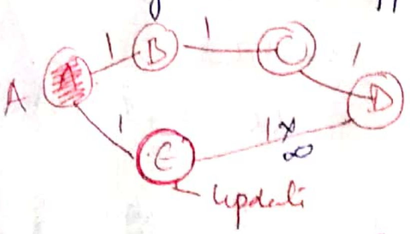
- Destination N/w
- cost.

When a record arrives the router searches for the destination address in the routing table.

1) if the entry is found.

- a) if the record cost plus 1 is smaller than corresponding cost in table, it means neighbour have found a better route.

b) If the Next hop is same, it mean some changes has happened in some part of the N/w.



A table (A→B→C→D)

destination	cost	Next hop
discarded ~~D~~	~~3~~	~~B~~
D	2	E
D	∞	E ✓

Now update via Router E, cost is 2 and Next hop E make entry in A table.
Then Compare the two entry which has less cost keep in table, and which have high cost is Discarded shown by table.

Scanned with CamScanner

Distance Vector Routing

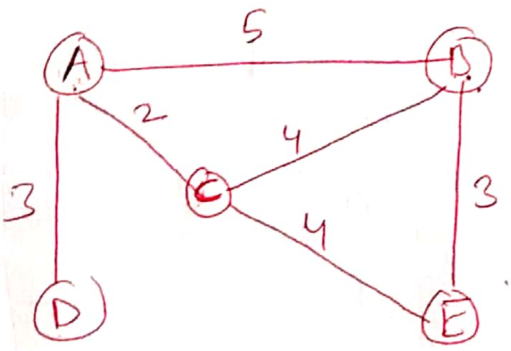
In distance vector Routing, the least cost route btn any two nodes is the route with Minimum distance. In this protocol as name implies each node maintain a vector (table) of minimum distance to every node.

* The table at each node also guides the packet to the desired node by showing the Next stop in the route (Next hop routing)

* We can thinks of nodes as the cities in a area and lines as the road connecting them. A table can show a tourist the minimum distance btn cities.

Show, a system five nodes with their Corresponding tables.

Distance Vector Routing table



A's table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

B's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

E's table

To	cost	Next
A	6	C
B	3	—
C	4	—
D	9	C
E	0	—

D's table

To	cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

C's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

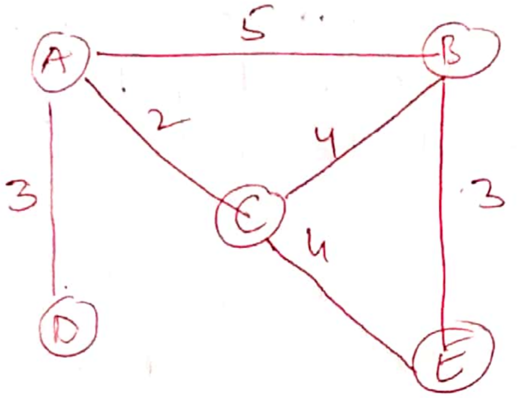
The table for Node A show how we can reach any node from this node. For Example our least cost to reach node E is 6. The route passes through C

The distance for any entry that is not a Neighbor is marked as Infinite (unreachable).

Initialization of tables in Distance Vector Routing

A's table

To	Cost	Next
A	0	–
B	5	–
C	2	–
D	3	–
E	∞	



B's table

To	Cost	Next
A	5	–
B	0	–
C	4	–
D	∞	
E	3	–

D's table

To	Cost	Next
A	3	–
B	∞	
C	∞	
D	0	–
E	∞	–

C's table

To	Cost	Next
A	2	–
B	4	–
C	0	–
D	∞	
E	4	–

E's table

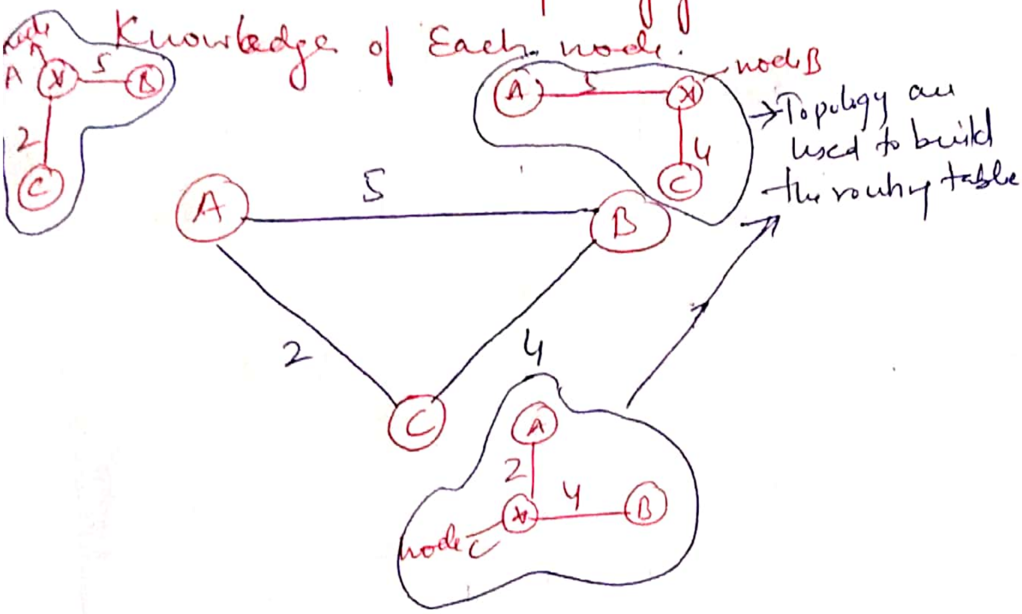
To	Cost	Next
A	∞	
B	3	–
C	4	–
D	∞	
E	0	–

In Distance Vector Routing each node shares its routing table with its immediate Neighbors periodically and when there is a change.

Link State Routing :-

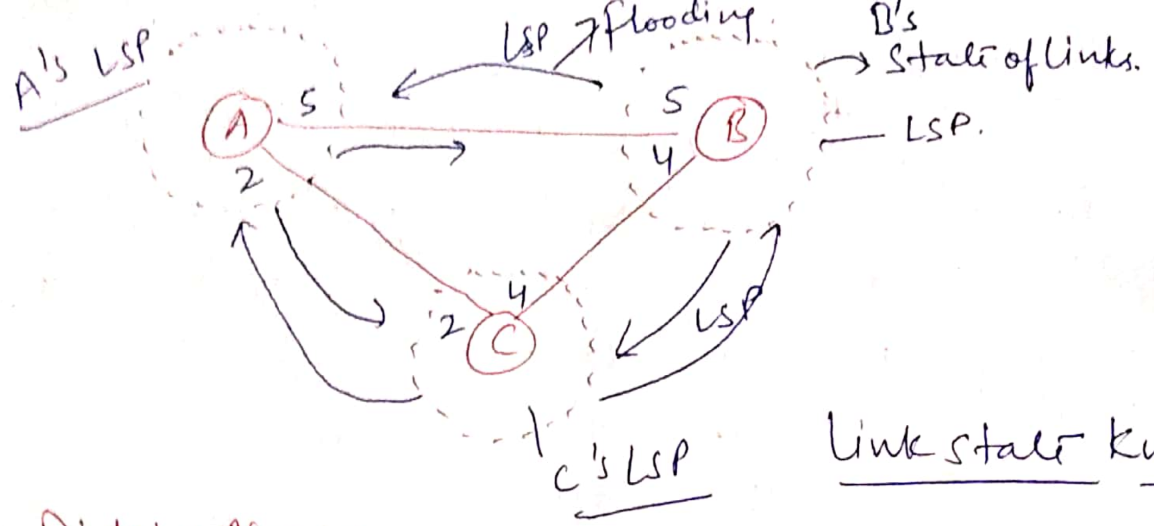
In link state routing, if each node in the domain has the entire topology of the Domain the list of nodes and links, how they are connected including of type, cost (metric) and condition of the links (up or down). Then the node can use Dijkstra's Algo to build a routing table.

In this whole Topology can be compiled from partial knowledge of Each node.



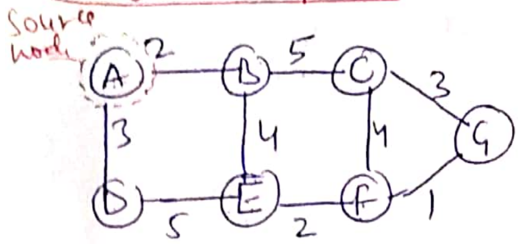
Building Routing table :- following action are required to ensure that each node has the Routing table :

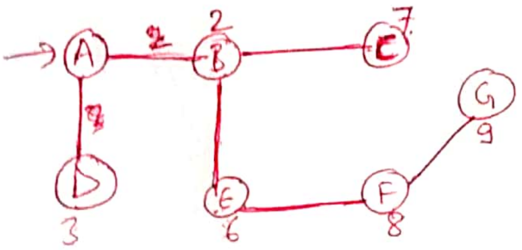
1) creation of the states of links by each node, called the Link State packet (LSP)
2) Dissemination of LSP's to every other router called Flooding
3) Formation of Shortest path tree for each node.
4) Calculation of a routing table based on the Shortest path tree.



Link state knowledge

Dijkstra Algo :-



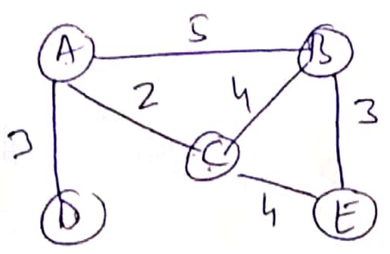


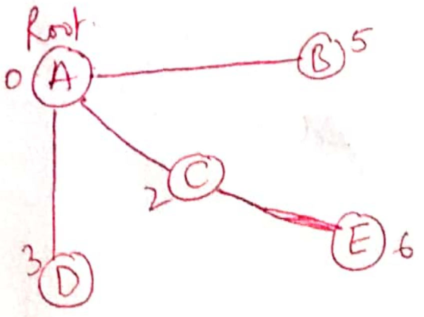
Calculation of Routing table :-

Routing table for Node A :-

Destination	Cost	Next Router
A	0	—
B	2	—
C	7	B
D	3	—
E	6	B
F	8	B
G	9	B

Example ②

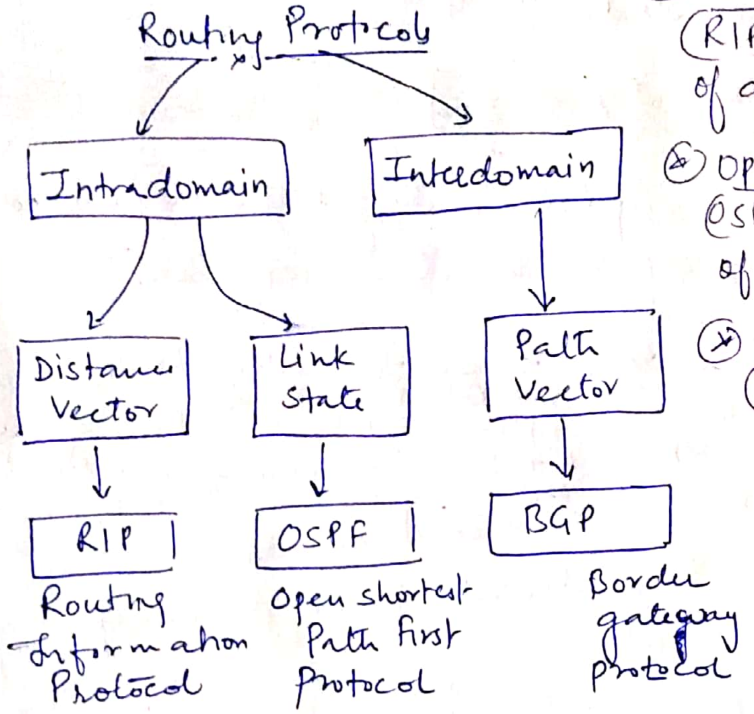




Calculation of Routing table from Shortest Path Tree

Destination	Cost	Next Router
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

Routing Protocol.



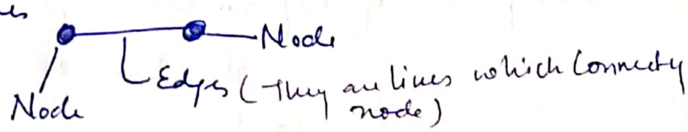
(*) Routing Information Protocol (RIP) is implementation of distance vector Protocol.

(*) Open Shortest Path first (OSPF) is implementation of link state protocol

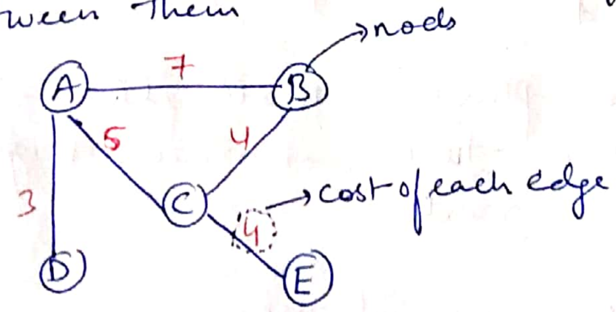
(*) Border Gateway Protocol (BGP) is implementation of path vector Protocol.

Distance Vector Routing :- It sees as AS (Autonomous System) with all the routers and N/w as a Graph

Graph { sets of nodes / Edges



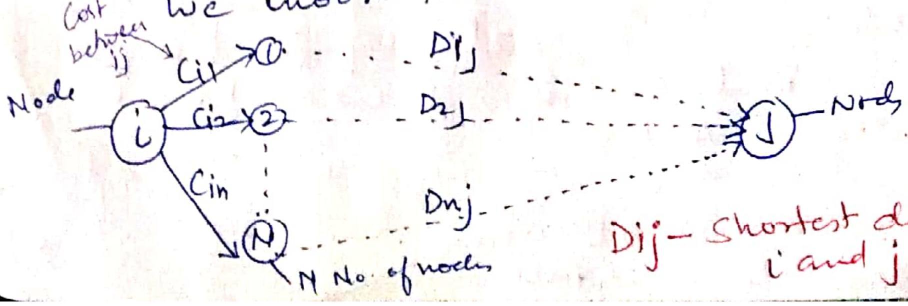
Bellman-Ford Algorithm :- Used to find shortest path between nodes in a graph given distance between them



For Example we want to communicate (A) to (E) we check the path 1) Path 1 - $A \xrightarrow{7} B \xrightarrow{4} C \xrightarrow{4} E$ cost is (15)

Path 2 - $A \xrightarrow{5} C \xrightarrow{4} E$ cost (9)

We choose short distance path bet^n A to E.

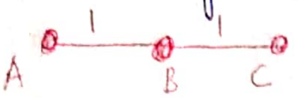


$$D_{ij} = \text{Minimum} \begin{cases} (C_{i1} + D_{1j}) \\ (C_{i2} + D_{2j}) \\ \vdots \\ C_{in} + D_{nj} \end{cases}$$

D_{ij} – Shortest distance between i and j.

Distance Vector Routing Algorithm

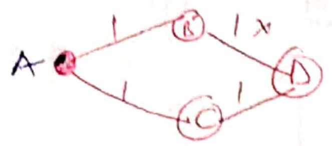
1) Cost is normally Hop counts → No of N/w crossed.



[A→C=2]

2) Each Router needs to update routing table asynchronously [when it recieves information from neighbour]

3) After update, result is sent to all the Neighbour.



B to D link is damage.
Suppose B→D link is damage then A table is shared with B & C
Now C get knows that B→C link is damage.

4) Each router should keep atleast three pieces of information for each route.

- Destination N/w → D | 2 | B
- Cost
- Next hop

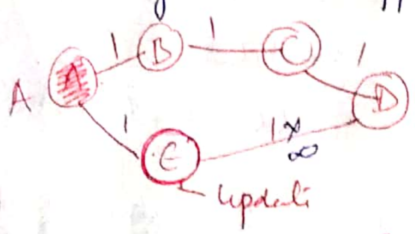
5) Two piece of information is received via update.

- Destination N/w
- cost.

When a record arrives the router searches for the destination address in the routing table.

1) if the entry is found.
   a) if the record cost plus 1 is smaller than corresponding cost in table, it means neighbour have found a better route.

   b) If the Next hop is same, it mean some changes has happened in some part of the N/w.



A table (A→B→C→D)

destination	Cost	Next hop
~~D~~	~~3~~	~~B~~
D	2	E
D	∞	E ✓

discarded

Now update via Router E, Cost is 2 and Next hop E make entry in A table.
Then Compare the two entry which has less cost keep in table and which have high cost is Discarded shown by table.

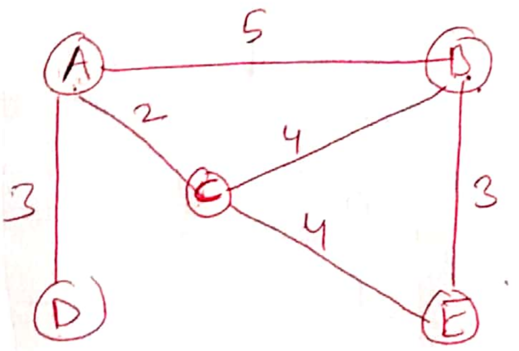
Distance Vector Routing

In distance vector Routing, the least cost route betn any two nodes is the route with Minimum distance. In this protocol as name implies each node maintain a vector (table) of minimum distance to every node.

* The table at each node also guides the packet to the desired node by showing the Next stop in the route (Next hop routing)
* We can think of nodes as the cities in a area and lines as the road connecting them. A table can show a tourist the minimum distance betn cities.

Show a system five nodes with their corresponding tables.

Distance Vector Routing table



A's table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

B's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

E's table

To	cost	Next
A	6	C
B	3	—
C	4	—
D	9	C
E	0	—

D's table

To	cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

C's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

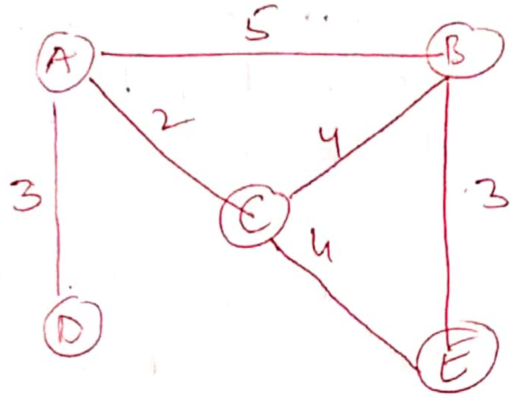
The table for Node A show how we can reach any node from this node. For Example our least cost to reach node E is 6. The route passes through C

The distance for any entry that is not a Neighbor is marked as Infinit (unreachable).

Initialization of tables in Distance Vector Routing

A's table

To	Cost	Next
A	0	–
B	5	–
C	2	–
D	3	–
E	∞	



B's table

To	Cost	Next
A	5	–
B	0	–
C	4	–
D	∞	
E	3	–

D's table

To	Cost	Next
A	3	–
B	∞	
C	∞	
D	0	–
E	∞	–

C's table

To	Cost	Next
A	2	–
B	4	–
C	0	–
D	∞	
E	4	–

E's table

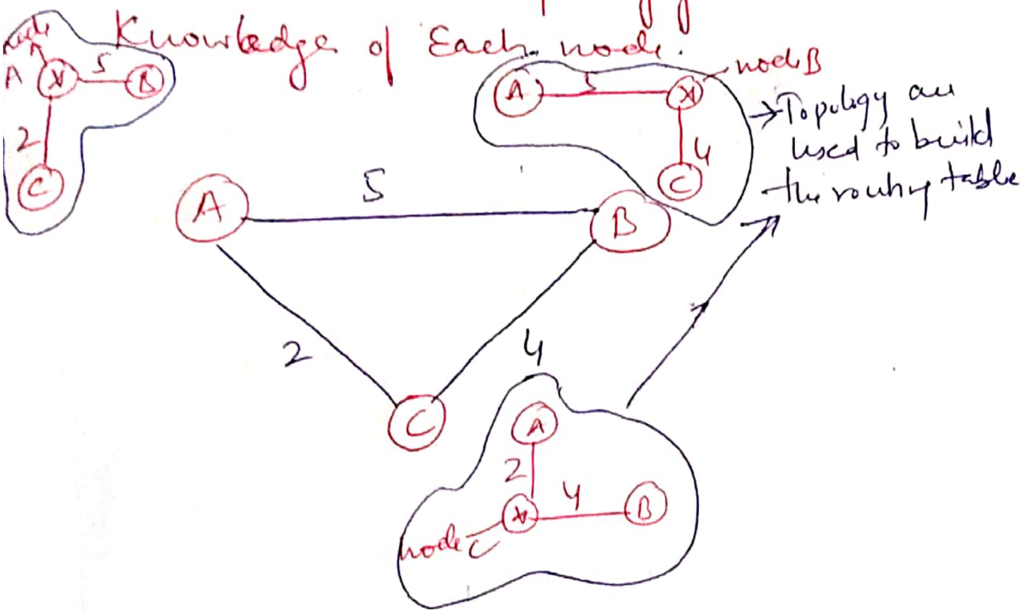
To	Cost	Next
A	∞	
B	3	–
C	4	–
D	∞	
E	0	–

In Distance Vector Routing each node shares its routing table with its immediate Neighbors periodically and when there is a change.

Link State Routing:-

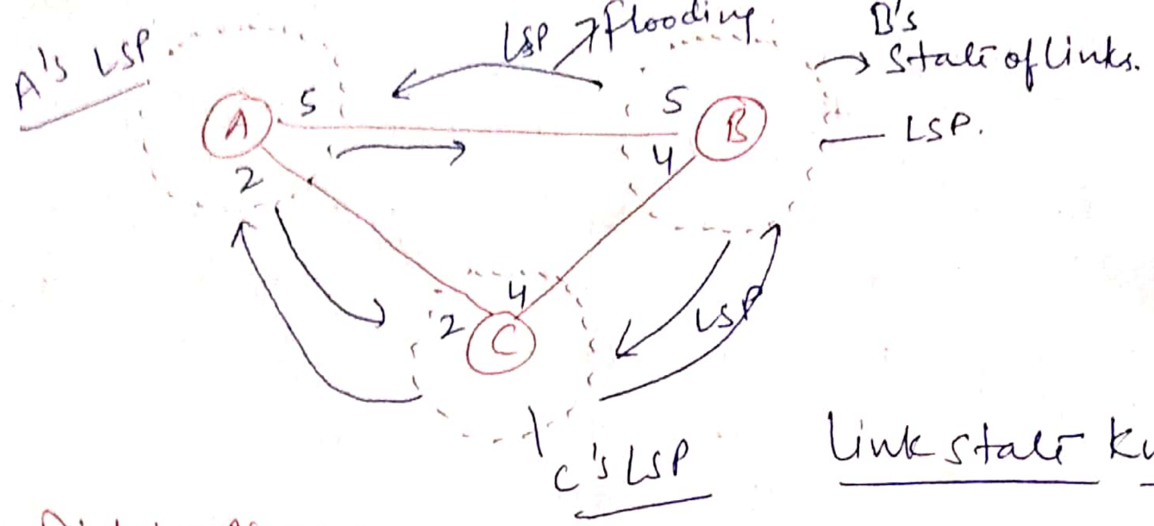
In link state routing, if each node in the domain has the entire topology of the Domain the list of nodes and links, how they are connected including of type, cost (metric) and condition of the links (up or down). Then the node can use Dijkstra's Algo to build a routing table.

In this whole Topology can be compiled from partial knowledge of each node.



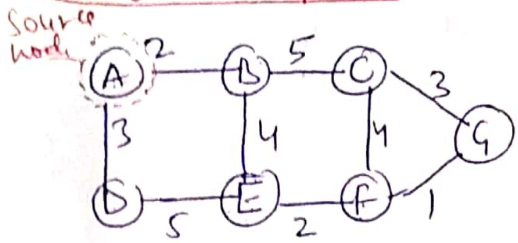
Building Routing table:- following action are required to ensure that each node has the Routing table:

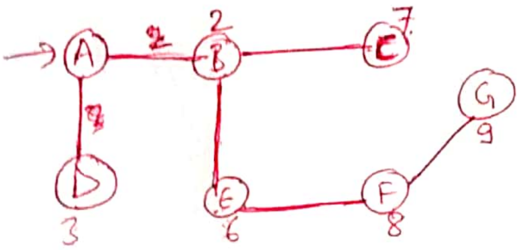
1) creation of the states of links by each node, called the Link state packet (LSP)
2) Dissemination of LSP's to every other router called Flooding
3) Formation of Shortest path tree for each node.
4) Calculation of a routing table based on the Shortest path tree.



Link state knowledge

Dijkstra Algo :-



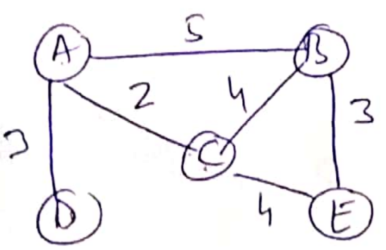


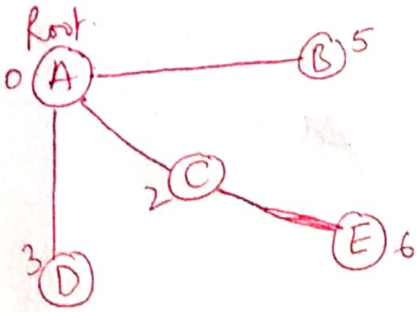
Calculation of Routing table :-

Routing table for Node A :-

Destination	Cost	Next Router
A	0	—
B	2	—
C	7	B
D	3	—
E	6	B
F	8	B
G	9	B

Example ②





Calculation of Routing table from Shortest Path Tree

Destination	Cost	Next Router
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

OSPF - OPEN SHORTEST PATH FIRST :-

The open shortest path first (OSPF) protocol is an (within) Intradomain routing protocol based on link state Routing. Its domain is also an Autonomous system.

OSPF Areas :-

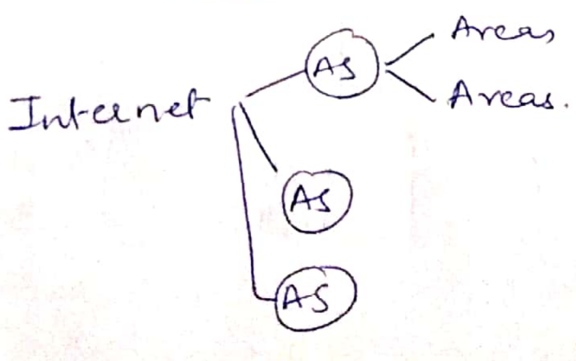
* OSPF divides an autonomous system into areas. An Area is a collection of Networks, hosts and routers all different areas. All Networks inside an area must be connected.

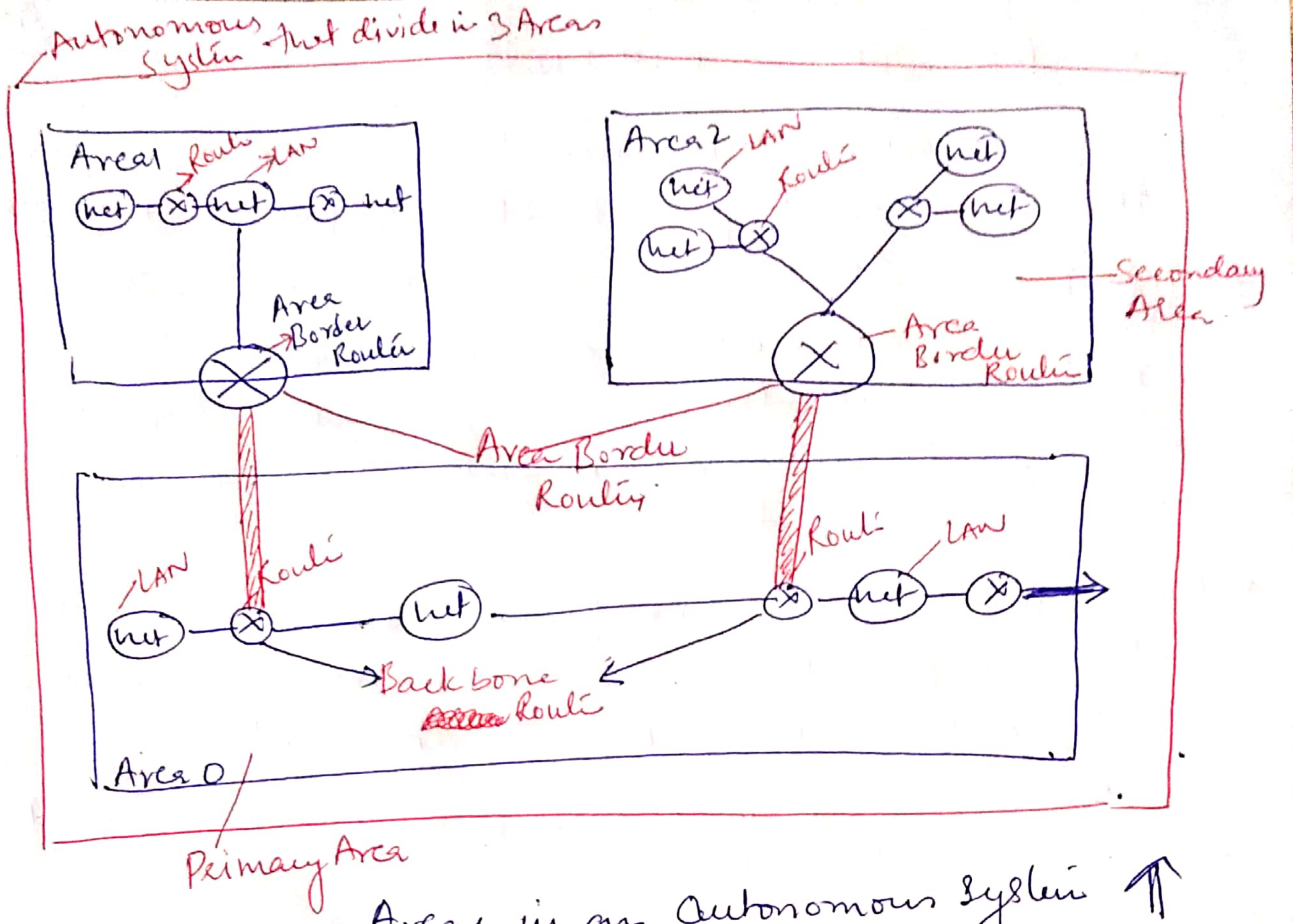
* Routers inside an area flood the area with Routing Information. At the border of an area Special Routers called Area border routers. Summarize the information about the area and send it to other areas.

* Among the areas inside an autonomous system is a special area called the Backbone all the areas inside an autonomous system must be connected to the Backbone.

* In other words, the backbone serves as primary area and the other areas as secondary area.

* This does not means that the routers within area cannot be connected to each other. The routers inside the backbone are called the Backbone routers. Backbone router can also be an area border router.

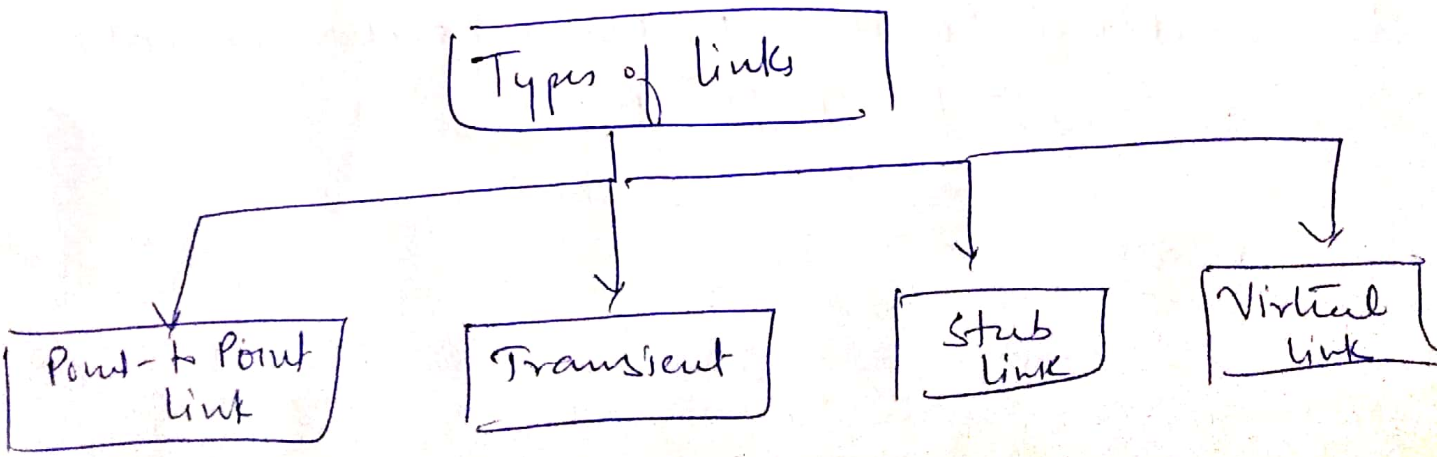




Areas in an Autonomous System
Architecture diagram of OSPF

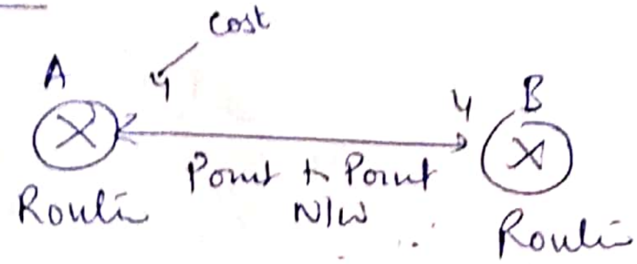
Types of links :

A connection is called links. Four types of links have been defined
- → point to point
- → Transient link.
- → Stub link
- → Virtual link

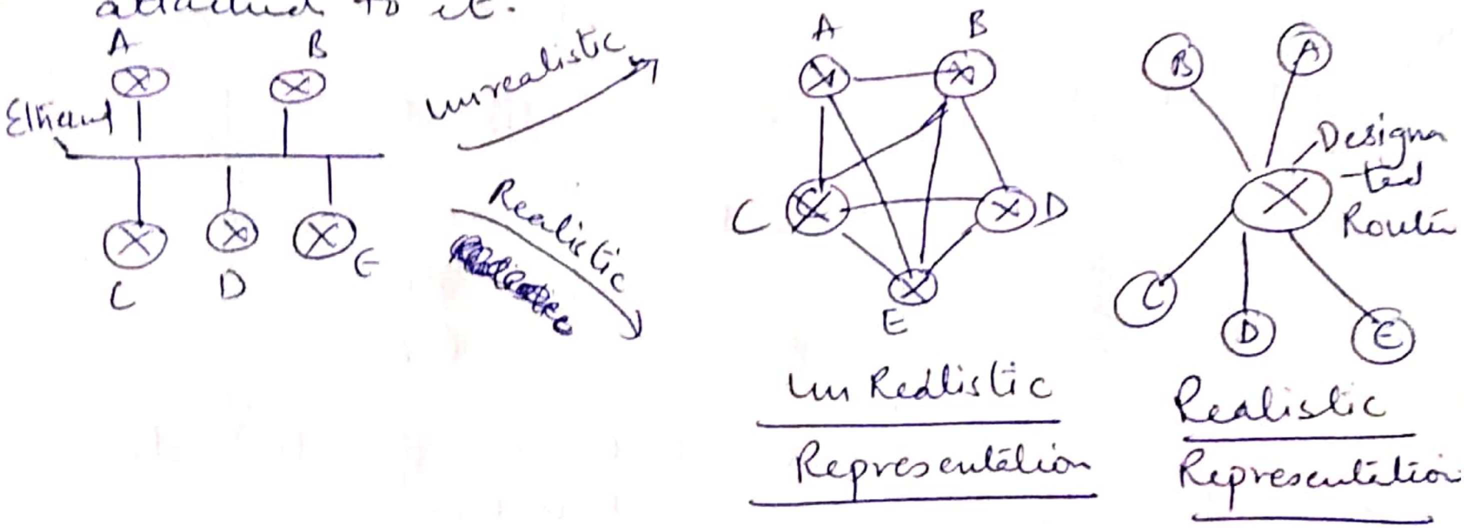


(1) Point-to Point Link :-

Point to Point link connect two router without any other host or router in between. In other words purpose of link (Network) is just to connect the two routers. An Example this type of link is two router connected by a Telephone line or a T line.

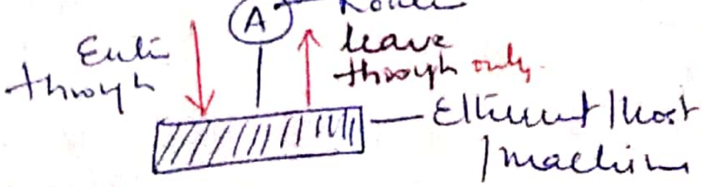


(2) Transient link:- It is between with several router attached to it.

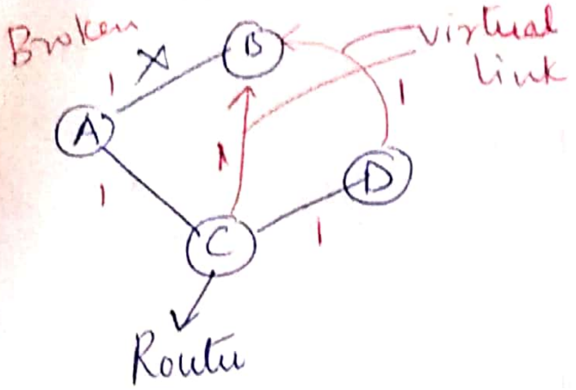


The Data can enter through any of the router and leave through any router.

(3) Stub link :- A Network that connected to only one Router. The Data packet enter the Network through this single router and leave the N/W through this same router.



(4) Virtual link :- When the link betn two Router is broken, the administration may create a Virtual link between them using a longest path that probably goes through several router.



Administration create an virtual from C and D to travel packet from A → B via C or D.

OSPF Packet Format :-

1 Hello
2 Database Description
3 Link state Request
4
Link state update
5 Link state Ack

32-bit

Version (8)	Type (8)	Message Length (16)
Source IP address		
Area identification		
Checksum	Authentication type	
Authentication (32)		

Packet type (1-5)

Length of total message + Header

OSPF protocol version 1,2

sending router IP address

define the area within Router happen

2 type — 0 (None), 1 (For Password)

Error detection/Correction

Actual value of Authentication data value.

Hello → Hello msg is used to create neighbourhood relationship and to test the reachability of Neighbour. Connectivity for using hello msg between.

Database Description :- Once the database connect to network hello packet are sent to Neighbour. then first time Neighbour Router conn" the packet send database description according modification are made.

Link state Req - It is sent by Router that need information about specific router. Suppose (A) → get LS information required from (B).

Link state update - ↳ State of link information Broadcast.

Link state Ack - How OSPF in this A send update request to B and C. A require to get ack from B and C as well as.

3) PATH VECTOR ROUTING

* Distance Vector and link state Routing are both Intradomain routing protocol.
* They can be used inside an Autonomous system but not betn Autonomous system. These two protocol are not suitable for Interdomain routing mostly because of scalability.

Path vector routing proved to be useful for Inter-domain routing. The principle of path vector routing is similar to that of distance vector Routing.
* It is an Exterior Routing protocol proved to be useful for interdomain or inter-AS routing
* In this routing a router has list of N/ws that can be reached with the path to reach each one.
* As the name suggests, it tells us the path

Speaker node: Speaker node in a AS create a routing table and advertises it to speaker node in Neighboring ASs. The idea is the same as for distance vector routing, except that only speaker nodes in each AS can communicate with each other.

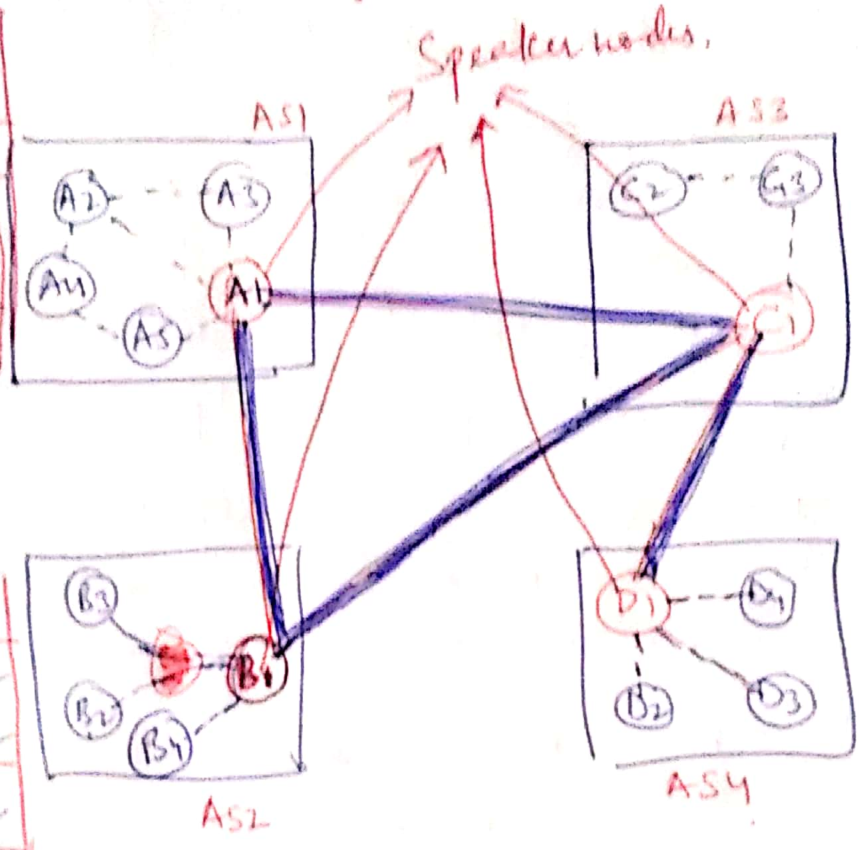
An speaker node advertises the path, not the metric of the nodes, in its autonomous system or other autonomous system.

Reachability:- Reachability of nodes inside an Autonomous System. Shows the initial table for each speaker node in a system made of four ASs.

Initial Routing tables in Path Vector Routing

A1 Table

Dest	Path
A1	AS1
A2	AS1
A3	AS1
A4	AS1
A5	AS1



C1 Table

Dest	Path
C1	AS3
C2	AS3
C3	AS3

B1 table

Dest	Path
B1	AS2
B2	AS2
B3	AS2
B4	AS2

D1 Table

Dest	Path
D1	AS4
D2	AS4
D3	AS4
D4	AS4

→ Node A1 is the speaker node for AS1
→ Node B1 is the speaker node for AS2
→ Node C1 is the speaker node for AS3
→ Node D1 is the speaker node for AS4

* Node A1 creates an initial table that shows A1 to A5 are located in AS1 and can be reached through it.

* Node B1 advertises that B1 to B4 are located in AS2 and can be reached through it.

* Node C1 advertises that C1 to C3 are located in AS3 and can be reached through it and so on.

Sharing – Just as distance vector Routing, in path vector routing, a speaker in an autonomous system share its table with immediate neighbors.

Node A1 shares its table with node B1 and C1.

Node C1 share its table with node D1, B1 & A1.
Node B1 share its table with C1 and A1.
Node D1 share its table with C1

Routing table:- A Path vector Routing table for each router can be created if 'AS' share their rechability list with each other.

Stablized tables for three autonomous system

A1 Table

Dest	Path
A1	AS1
A5	AS1
B1	AS1-AS2
B4	AS1-AS2
C1	AS1-AS3
C3	AS1-AS3
D1	AS1-AS3-AS4
D4	AS1-AS3-AS4

B1 Table

Dest	Path
A1	AS2-AS1
A5	AS2-AS1
B1	AS2
B4	AS2
C1	AS2-AS3
C3	AS2-AS3
D1	AS2-AS3-AS4
D4	AS2-AS3-AS4

C1 table

Dest	Path
A1	AS3-AS1
A5	AS3-AS1
B1	AS3-AS2
B4	AS3-AS2
C1	AS3
C3	AS3
D1	AS3-AS4
D4	AS3-AS4

D1 Table

Dest	Path
A1	AS4-AS3-AS1
A5	AS4-AS3-AS1
B1	AS4-AS3-AS2
B4	AS4-AS3-AS2
C1	AS4-AS3
C3	AS4-AS3
D1	AS4
D4	AS4

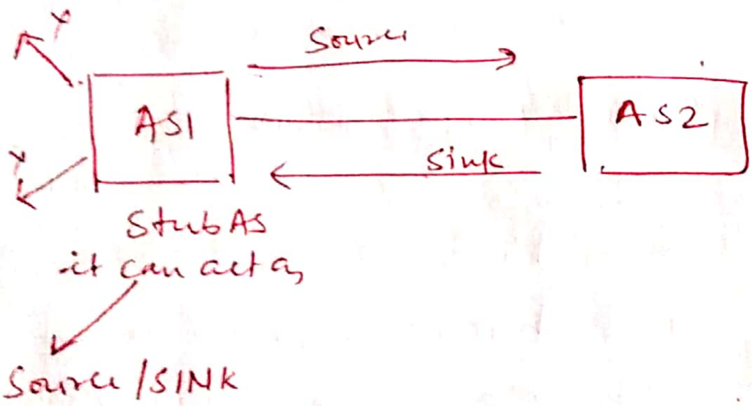
Loop Prevention When a router Recieve a Rechability message, it Checks to see if its Autonomous system is in the Path list to any destination. If it is looping is involved and the message is ignored/discarded.

Border Gateway Protocol (BGP):-

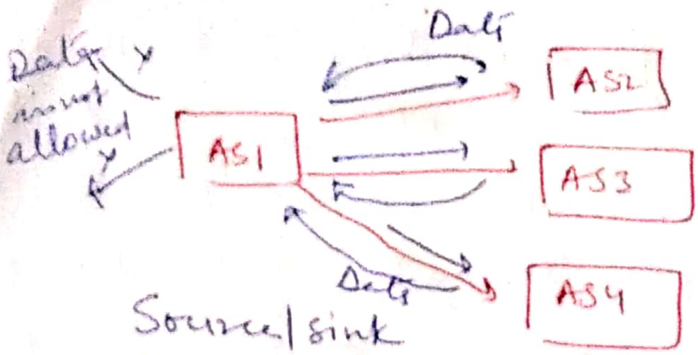
BGP is an Interdomain routing protocol using path Vector Routing.

Types of Autonomous System:- An Internet divided into hierarchical domain called Autonomous system For Example a large coorporation that manages its own N/w and has full control over it is an AS.

1) **Stub AS** - A stub AS has only one connection to another AS. The Interdomain data traffic in Stub AS can be either created or terminated in the AS. The host in the AS can send data traffic to other ASs. The host in AS can Receive data Comming from hosts in other ASs Data traffic, cannot pass through a stub AS. A Stub AS is either a Source or Sink. For Example stub AS is small coorporation or a small local ISP

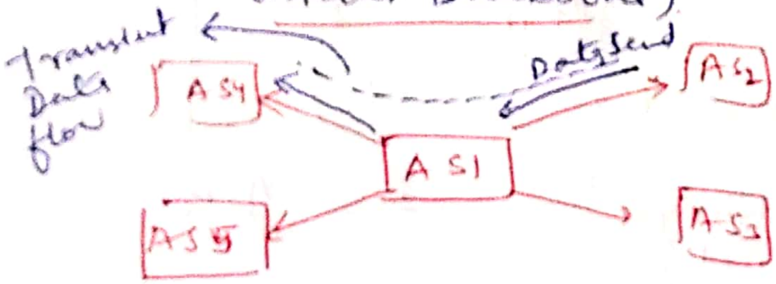


2) **Multihomed AS** :- A multihomed AS has more than one connection to other AS. but it is still only a source or sink for data traffic. It can receive Data traffic from more than one AS. It can send data traffic to more than one AS. But there is No Transient traffic. It does not allow data Comming from one AS and going to another AS to pass through. Ex. Large coorporation



Source | sink

③ Transit AS - A transit AS is a Multi homed AS that allows transient traffic. Good Example of Transit AS are national and international ISPs (Internet Backbone)



Path Attributes :-

The Path was presented as a list of AS. Each attribute gives some information about the path.
Attributes are divided into two broad categories
Well Known and optional

* A well Known attribute is one that every BGP router must Recognize.
* Optional attribute is one that needs not be recognized by every router.

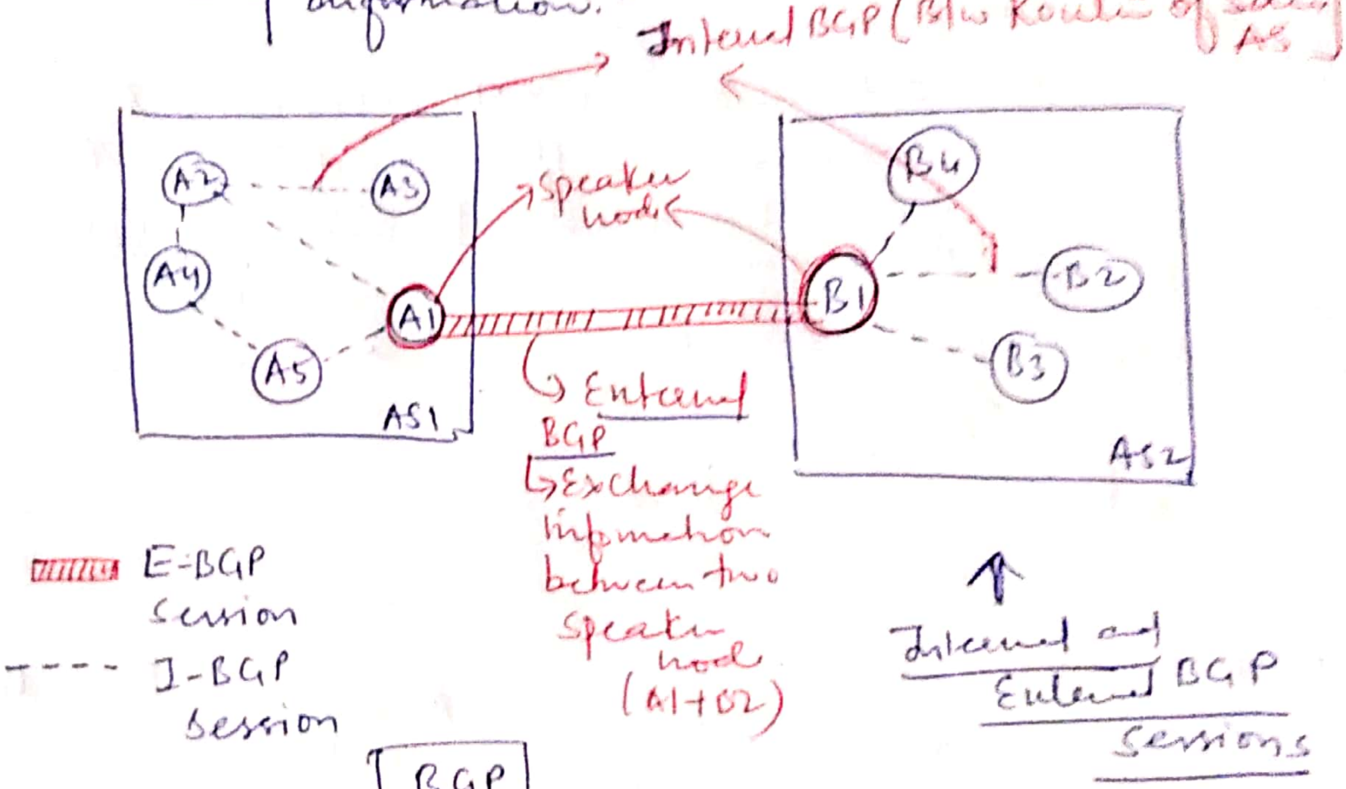
Path attributes → Well Known Attributes {Every BGP Router must Recognize} by every Router.

↓

optional Attribute [Not Mandatory]
Not Recognized by every Router

Scanned with CamScanner

BGP Session :- The Exchange of Routing Information between routers using BGP takes place in a session.
A Session is a Connection that is established betn two BGP routers only for the sake of Exchanging routing Information.



BGP
- external BGP Session (E-BGP)
- Internal BGP (I-BGP) session

1) **E-BGP session** - It is used to exchange Information betn two Speaker nodes belonging to two different Autonomous System.

2) **I-BGP session** - It is used to exchange Routing Information betn two Routers inside an Autonomous system.

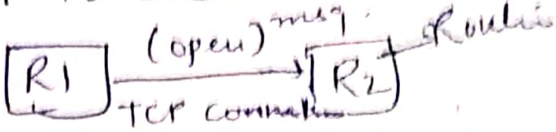
The session established betn AS1 and AS2 is E-BGP session. Two speaker node Exchange Information they know about N/W in the Internet.

(4)

There are two routes used to collect Info" from other routers in the autonomous system. This is done using I-BGP Session.

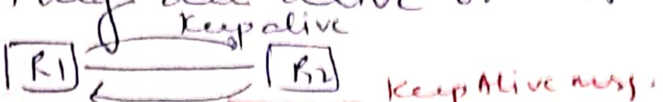
Types of BGP Session — External (E-BGP)
— Internal (I-BGP)

(*) Types of Packets :- (1) Open - It is used to create an Neighbourhood Relation.



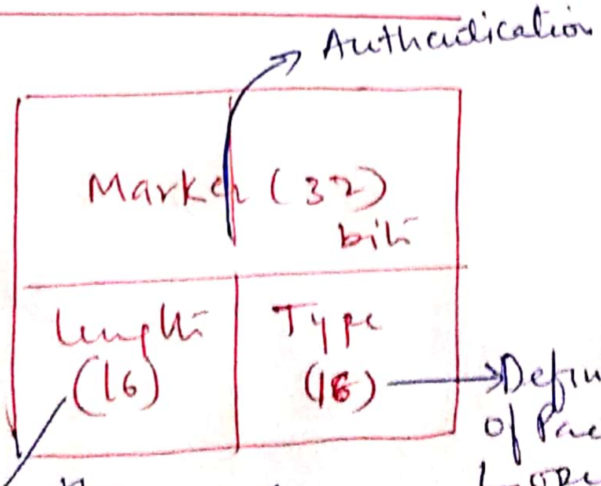
(2) update - It is used to withdraw destination that have been advertised previously, announce a route to new Destination or both.

(3) - KeepAlive - Exchange regularly to tell other router whether they are alive or not.



(4) - Notification - Sent by an Router whenever an error condition is detected or a Router wants to close the connection (when we want to terminate)

(*) BGP Packet Format :-



Type (16) → Defines the type of Packet
- open
- update
- Keepalive
- Notification.

Length (16) → To defines the length of the total message including Header