**UNIT – III**

**NETWORK LAYER SERVICES**

* Packetizing
* Routing and forwarding
* Other series,
* Error control
* Flow control
* Congestion control
* Security

**PACKETIZING:**

* The first duty of the network layer is packetizing
* Duty of the new layer is to carry a payload from the source to the destination without changing it.
* The network layer is doing the service of a carrier such as the postal office, which is responsible for delivery of packages from a sender to a receiver without changing the contents.
* The source host receive the payload from an upper layer protocol, adds a header that contains the source and destination addresses and some other information that is required by the network layer protocol and delivers the packet to the data link layer.
* The destination host receive the network- Layer packet from its data link layer, decapsulates the packet, and delivers the payload to the corresponding upper layer protocol.
* The routers are not allowed to change the source and destination addresses either. The just inspect the addresses for the purpose of forwarding the packet to the next network on the path.

**ROUTING:**

* The network layer is responsible for routing the packet from its source to the destination
* A physical network is a combination of networks and routers that connect them, this means that there is more than one route from the source to the destination.
* The network layer is responsible for finding the best one among these possible routes.
* Then network layer need to have some specific strategies for defining the best route.
* In the internet today, this is done by running some routing protocols to help the routers co-ordinate their knowledge about the neighborhood.

**FORWARDING:**

* Forwarding can be defined as the action applied by each router when a packet arrives at one of its interface.
* The decision making table is a also called forwarding table or routing table.
* When a router receives a packet from one of its attached networks, it needs to forward the packet to another attached network.
* To make this decision, the router uses a piece of information in the packet header, which can be the destination address or a label.

|  |  |
| --- | --- |
| B | DATA |

**ERROR CONTROL:**

* The designers of the network layer, have added a checksum field to the datagram to control any corruption in the header, but not in the whole datagram, this checksum may prevent any changes or corruption in the header of the datagram

**FLOW CONTORL:**

* Flow control regulates the amount of data a source can send without overwhelming the receiver.
* To control the flow of data, the receiver needs to send some feed back to the sender to inform the latter that it is overwhelmed with data.
* The new layer in the internet, does not directly provide any flow control.

**CONGESTION CONTROL:**

* Congestion in the network layer is a situation in which too many data grams are present in an area of the internet.
* Congestion may occur if the number of data grams sent by source computers is beyond the capacity of the network or routers.
* In this situation some routers may drop some of the data grams, however as more data grams are dropped, the situation may become worse.
* If the congestion continues, sometimes a situation may reach a point where the system collapses and no data grams are delivered.

**SECURITY:**

* Security was not a concern when the internet was originally designed because it was used by a small number of users at universities for research activities other people had no access to the internet.
* The network layer was designed with no security provision.

**PACKET SWITCHING**

**INTRODUCTION:**

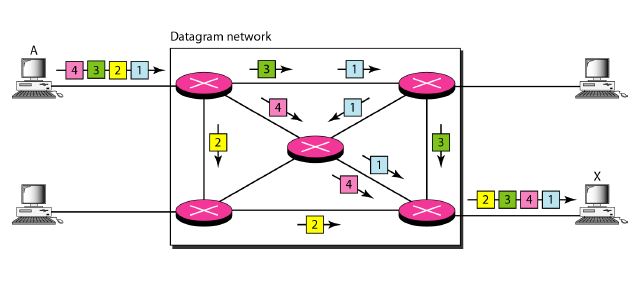
* A switch that creates a connection between an input port and an output port, just as an electrical switch connects the input to the output to let electricity flow.

* Switching techniques are divided into two broad categories
* Circuit switching
* Packet switching
* Only packet switching is used at the network layer because the unit of the data at this layer is packet.
* Circuit switching is mostly used at the physical layer.
* A packet switched network can use to different approaches to route the packets,

1. The data gram approaches (connection less)
2. Virtual circuit approaches (connection oriented)

**Datagram approach:**

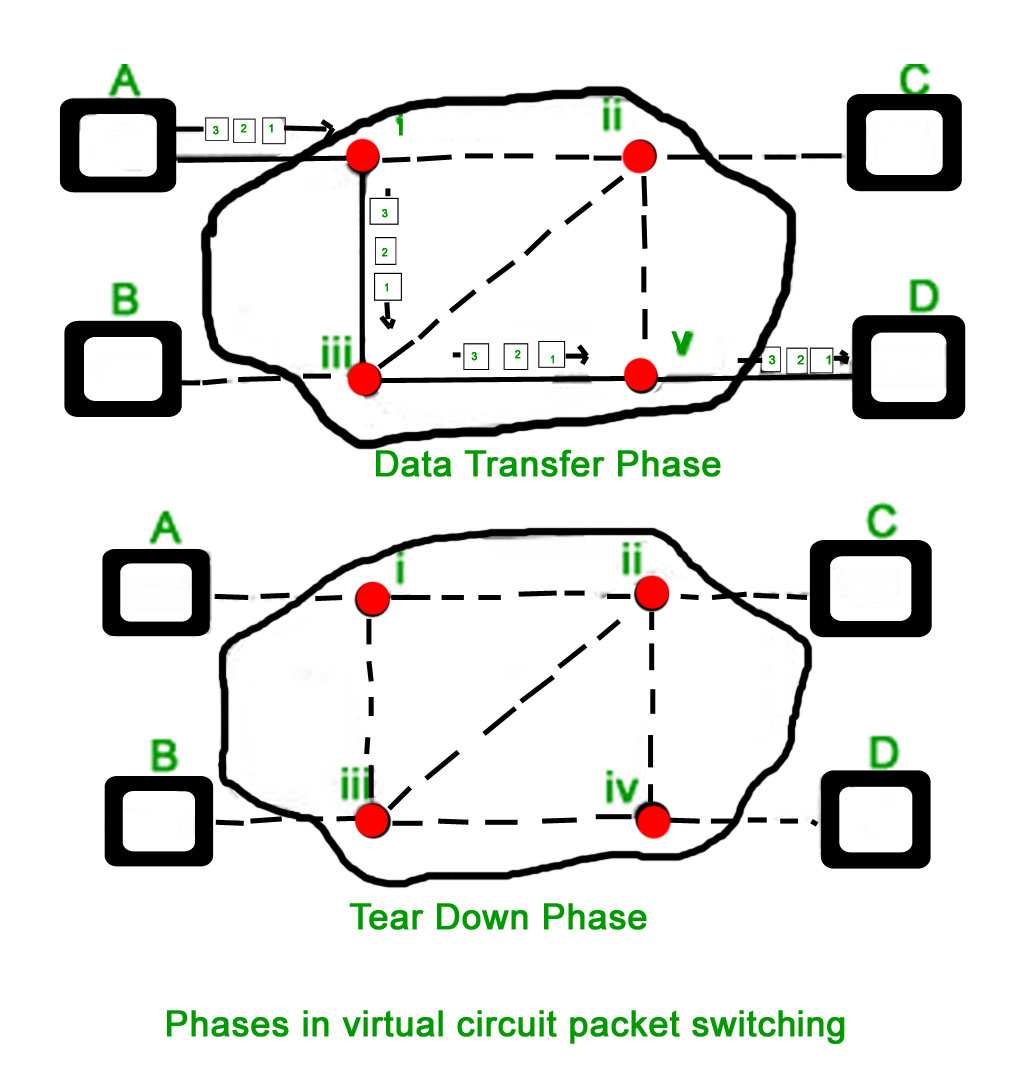
* Connection less service.



* In the data gram approach, the forwarding decision is based on the destination address of the packet.
* The network layer was designed to provide a connection less service in which the network layer protocol treats each packet independently, with each packet having no relationship to any other packet.
* The idea was that the new layer is only responsible for delivery of packets from the source to the destination.
* In this approach, the packets in a message may or may not travel in the same path to their destination.
* There is no relationship between packets belonging to the same message.
* The switches in this type of network are called routers
* Each packet is routed based on the information contained in its header, source and destination addresses.
* The destination address defines where it should go.
* The source address defines where it come from.
* The router in this case routes the packet based on the destination address.

**Virtual circuit approach:**

* It is also called connection oriented service.
* In this approach, there is a relationship between all packets belonging to a message.



* Before all data grams in a message can be sent, a virtual connection should be set up to define the path for the data grams.
* After connection setup, the data grams can all follow the same path.
* In this type of service, not only must the packet contains the source and destination addresses, it must also contain a flow label.
* In this method, three phase process is used.
  + Set up phase,
  + Data transfer phase
  + Tear down phase
* In the setup phase, the source and destination address of the sender and receiver are used to make table entries for the connection oriented service.
* In the teardown phase, the source and destination inform the router to delete the corresponding entries.
* In the data transfer phase, data transfer occurs between these two phases.

**NETWORK LAYER PERFORMANCE**

* The performance of a network can be measured in terms of delay, throughput, packet loss, congestion control.

**Delay:**

* All of us expect instant response from a network, but a packet from its source to its destination encounters delay.

**Types of delay:**

* Transmission delay
* Propagation delay
* Processing delay
* Queuing delay

**Transmission delay:**

* A source host or a router cannot send a packet instantly.
* A sender needs to put the bits in a packet on the line one by one.
* Transmission delay of the packet is(t2-t1).
* The transmission delay is longer for a longer packet and shorter if the sender can transmit faster.
* Delay tr=(packet length)/(transmission rate)

**Propagation delay:**

* Propagation delay is the time it takes for a bit to travel from point A to point B in the transmission media.
* The propagation delay depend on speed of the media and distance of the link.
* Delay pg=(distance)/(propagation speed)

**Processing delay:**

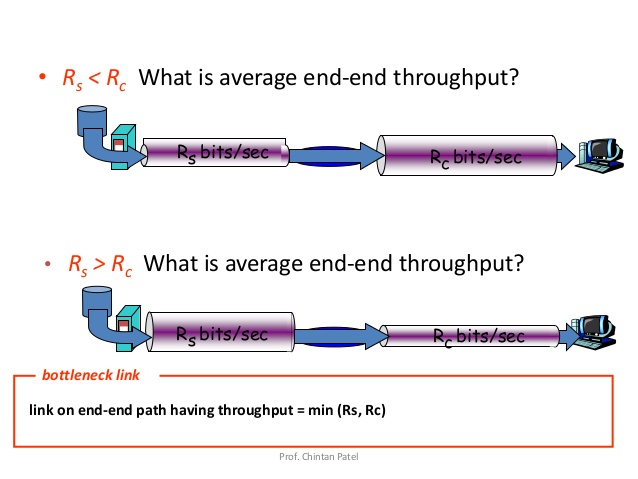
* The processing delay is the time required for a router or a destination host to receive a packet from its input port, remove the header, perform an error detection procedure, and deliver the packet to the output port.
* Delay pr=time required to process a packet in a destination host.

**Queuing delay:**

* Queuing delay can normally happen in a router.
* The queuing delay for a packet in a router is measured as the time a packet wait in the input queue and output queue of a router.
* Best known example is busy airport.
* delay qu= the time a packet waits in input and output queues in a router.

**Throughput:**

* throughput at any point in a network is defined as the number of bits passing through the point in a second, which is actually the transmission rate of data at that point.



* in the above diagram the data can flow at the rate of 200 kbps in link-1, 100 kbps in link 2, and 150 kbps in link3.
* The average rate of the path is 100 kbps, the minimum of the three different data rates,
* Throughtput=minimum {TR1,TR2,…TRn}

**Packet loss:**

* Another issue that severely affects the performance of communication is the number of packets lost during transmission.
* A router has an input butter with a limited size.
* A time may come when the buffer is full and the next packet need to be dropped.
* The effect of packet loss on the internet network layer is that the packet needs to be resend, which in turn may create overflow and cause more packet loss.

**Congestion control:**

* Congestion control is a mechanism for improving performance
* It can either prevent congestion before it happens or remove congestion after it has happened.
* We can divide congestion control mechanisms into two broad categories.
* Open – loop congestion control (prevention)
* Closed – loop congestion control (removal)

**Open – loop congestion control (prevention)**

* In this method, policies are applied to prevent congestion before it happens.
* Congestion control is handled by either the source or the destination.

**Policies:**

* Retransmission policy
* Window policy
* Acknowledgement policy
* Discarding policy

**Retransmission policy:**

* Retransmission is sometimes unavoidable, retransmission in general may increase congestion in the network the retransmission policy and the retransmission timers must be designed to optimize efficiency and at the same time prevent congestion.

**Window policy:**

* The type of window at the sender may also affect congestion. The selective repeat window is better then the Go – Back-N window for congestion control.
* In the Go – Back-N window, when the timer for a packet times out, several packets may be resent but the selective repeat window, tries to send the lost or corrupted packets only.

**Acknowledgement policy:**

* The acknowledgment policy imposed by receiver may also affect congestion.
* A receiver may decide to acknowledge only N packets at a time. Sending fewer acknowledgment means imposing less load on the network.

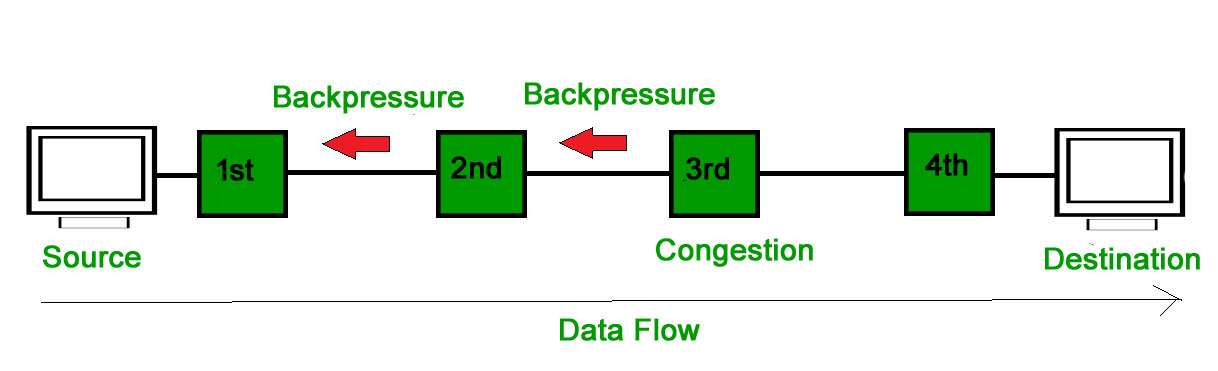
**Discarding policy:**

* A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission.
* Ex. Audio transmission.

**Closed loop congestion control:**

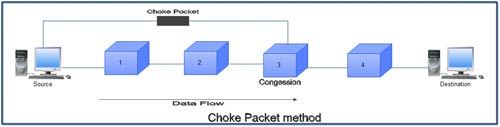
* Back pressure
* Choke packet
* Implicit signaling
* Explicit signaling
* Closed loop congestion control mechanisms try to alleviate congestion after it happens.

**Back pressure:**



* The technique of back pressure refers to a congestion control mechanism in which a congested node stops receiving data from the immediate up stream node or nodes.

**Choke packet:**



* A choke packet is a packet sent by a node to the source to inform it of congestion.
* In back pressure the warning is from one node to its up stream node, although the warning may eventually reach the source station.
* In the choke packet method the warning is from the router which has encountered congestion, directly to the source station.

**Implicit signaling:**

* In this method there is no communication between the congested node or nodes and the source.
* The source guess that there is congestion some where in the network from other symptoms.
* For example when a source sends several packets and there is no acknowledgement for a while, one assumption is that the network is congested.

**Explicit signaling:**

* The node that experiences congestion can explicitly send a signal to the source or destination.
* In this method, the signal is included in the packets that carry data.

**IPV4 ADDRESSES**

**Introduction:**

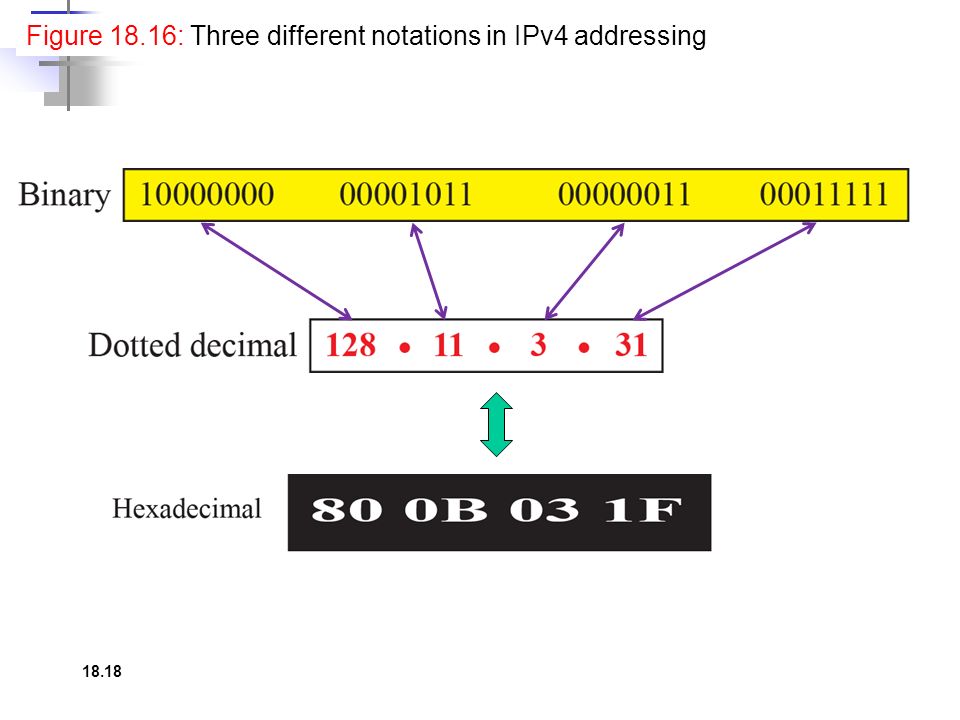
* The identifier used in the IP layer of the TCP/IP Protocol suite.
* To identity the communication of each devices to the internet is called the internet address or IP address.
* An ipv4 address is a 32 bit address that uniquely and universally defines the connection of a host or a router to the internet.
* Ipv4 addresses are unique.
* Ipv4 addresses are universal in the sence that the addressing system must be accepted by any host that wants to be connected to the internet.

**Address Space:**

* An address space is the total number of addresses used by the protocol.
* If a Protocol uses b bits to define an addresses, the addresses space is 2b, because each bit can have twodifferent values(0or1).
* Ipv4 uses 32 bit addresses, which means that the address space is 232 or 4,294,967,296, more than 4 billions devices could be connected to the internet.

**Notations:**

* There are three common notations to show an Ipv4 address,



**Binary notations (base 2):**

* In binary notation, an Ipv4 address is displayed, as 32 bits, to make the address more readable one or more space are usually inserted between each octet (8bits) each octet is often referred to as a byte.

**Dotted-decimal notation (base 256):**

* To make the Ipv4 address more compact and easier to read, it is usually written in decimal form with a decimal point (dot) separated by bytes.

**Hexa - decimal notation (base 16):**

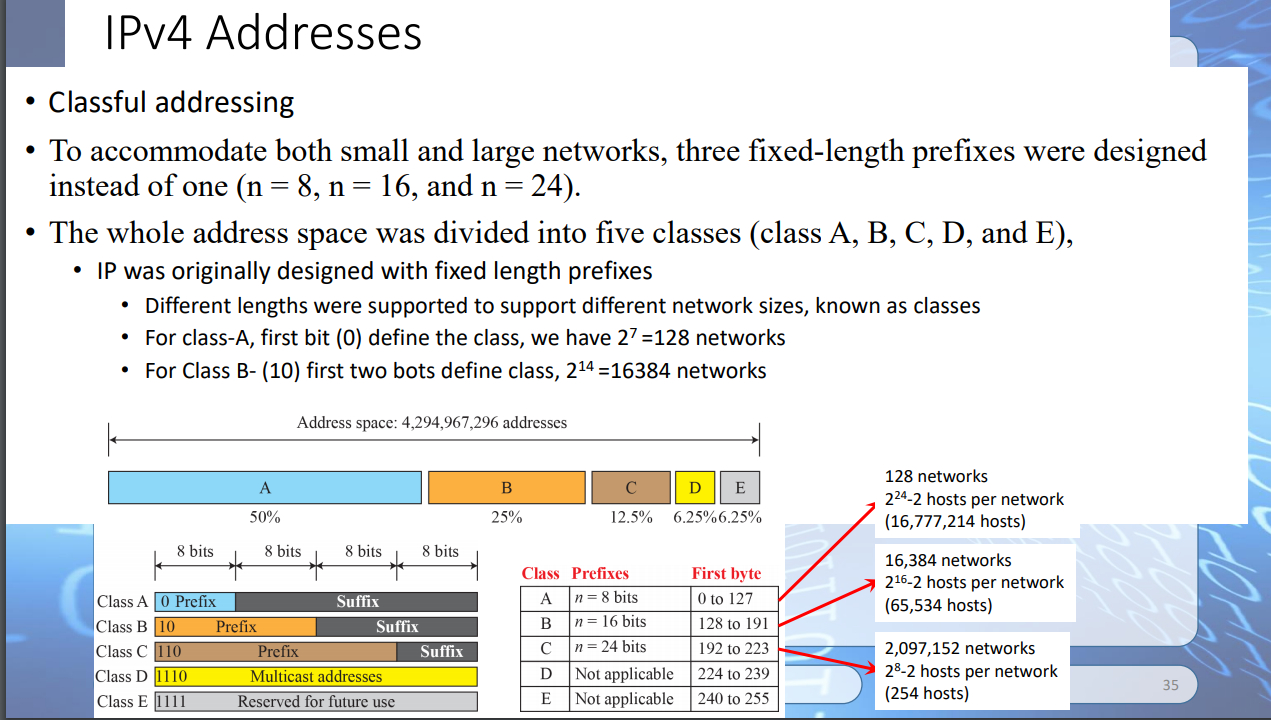
* Each hexadecimal digit is equivalent to four bits, this means that a 32 bit address has 8 hexa decimal digits.

**Hierarchy in addressing:**

* Any communication network the addressing system is hierarical,
* In a postal network, postal address include the country state, city, street, house number and the name of mail recipient.
* Similarly a telephone number is divided into the country code, area code, local exchange and the connection.
* A -32 bit Ipv4 address is also hieratical but divided only into two part.
* Prefix and suffix
* Prefix: the first part of the address called the prefix, defines the network.
* Suffix: the second part of address called the suffix defines the node.

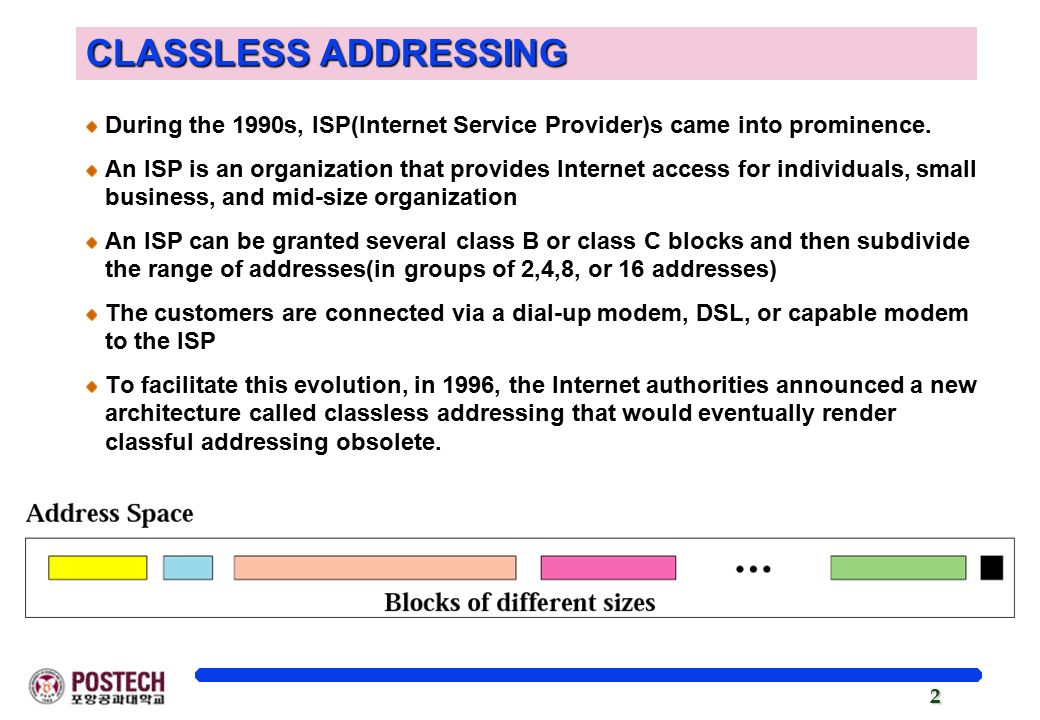
**Class ful addressing:**

* An Ipv4 address was designed with a fixed length prefix, but to accommodate both small and large networks, three fixed length prefixes were designed instead of one (n=8,n=16,n=24).
* The whole address space was divided into five classes (class A,B,C,D,E). this scheme is reffered as classful addressing.



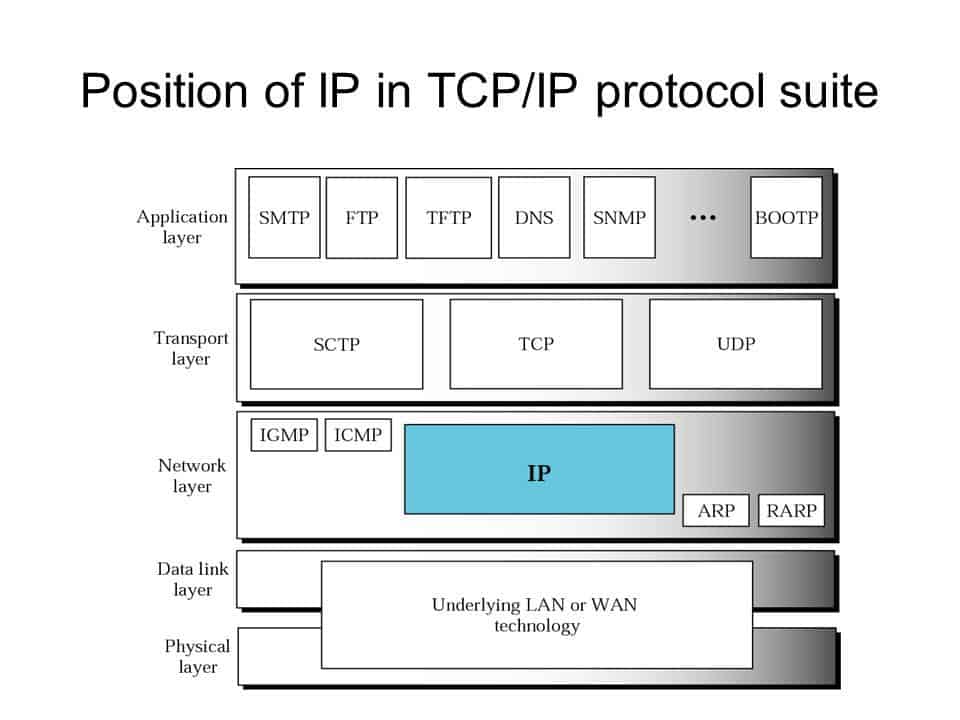
**Class less addressing:**

* In class ful addressing did not really solve the address depletion problem.
* In 1996, the internet authorities announced a new architecture called class less addressing.
* In class less addressing, variable length blocks are used that belong to no classes.
* We can have a block of 1 address, 2 addresses, 4 addresses, 128 addresses and so on.



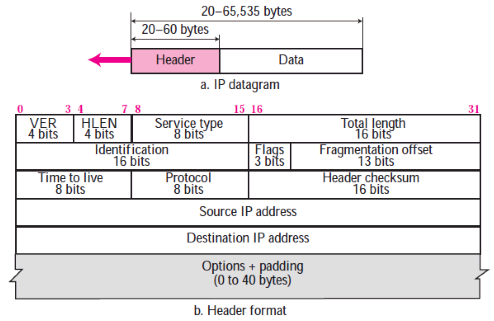
**INTERNET PROTOCOL(IP)**

* The network layer in version 4 can have one main protocol and three auxiliary ones.
* The main protocol, internet protocol version 4 (Ipv4) is responsible for packetizing, forwarding and delivery of a packet at the network layer.
* ICMP- Internet Control Message Protocol version 4 to handle some errors.
* IGMP- Internet Group Management Protocol is used to help multicasting.
* ARP- Address Resolution Protocol is used to glue the network and datalink layers,



* Ipv4 is an unreliable datagram protocol a best effort delivery service.
* Ipv4 is also a connection less protocol that uses the datagram approach this means that each datagram is handled independently, and each datagram can follow a different route to the destination.
* This implies that datagram sent by the same source to the same destination could arrive out of order, the Ipv4 relies on a higher level protocol to take care of all these problem.

**Datagram format:**



* Packets used by the IP are called datagrams.
* A data gram is variable length packet consisting of two parts : header and payload (data).
* The header is 20 to 60 bytes in length and contains information essential to routing and delivery.

**Version number:**

* The 4 bit version number (VER) field defines the version of the Ipv4 Protocol, its value is 4

**header length:**

* The 4 bit Header Length(HELN) field defines the total length of the datagram header.

**Service type:**

* In the original design of the IP header, this field was referred to as type of service (TOS) which defined how the datagram should be handled.

**Total length :**

* This 16 bit field defines the total length of the IP datagram in bypes.
* A 16 bit number can define a total length of upto 65,535.

**Identification flags and fragmentation off set:**

* These three field are related to the fragmentation of the IP datagram when the size of the datagram is larger than the underlying network can carry.

**Time to live:**

* The time to live (TTL) field is used to control the maximum number of hopes (routers) visited by the datagram.
* When a source host sends the datagram, it stores a number in this field, this value is approximately two times the maximum number of routers between any two hosts.
* Each router that processes the datagram decrements this numbers by one. If this value, after being decremented is zero, the router discard the datagram.

**Protocol:**

* set of rules and regulations.
* There are many protocol are available such as ICMP , IGMP, TCP, UDP.

**Header checksum:**

* IP is not a reliable protocol
* It does not check whether the payload carried by a datagram is corrupted during the transmission.
* IP puts the burden of error checking of the payload on the protocol that owns the payload such as UDP or TCP.
* The datagram header is added by IP, and its error checking is the responsibility of IP.
* Errors in the IP header can be disaster.

**Example:**

* If the destination address is corrupted the packet can be delivered to the wrong host.
* If the protocol field is corrupted the payload maybe delivered to the wrong protocol
* For this reason, IP adds a header checksum field to check the header but not the payload.

**Source and destination address:**

* 32 bit source and destination address fields define the IP address of the source and destination.
* The source host should know its IP address .
* Note that the value of these fields must unchanged during the time the IP datagram travels from the source host to the destination host.

**Options:**

* A data gram header can have up to 40 bytes of options.
* Options can be used for network testing and debugging
* Options are not required part of the IP header, option processing is required of the IP software.

**Payload:**

* Payload or data is the main reason for creating a datagram.
* Payload is the packet coming from other protocols that use the service of IP.
* comparing a datagram to a postal package, payload is the content of the package, the header is only the information written on the package.

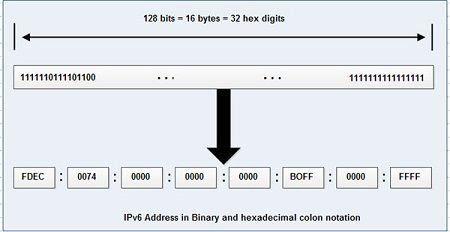
**IPV 6 ADDRESSING**

**Introduction:**

* An IPV6 address is 128 bits or 16 bytes (octets)long, four times the address length in IPV4.
* The main reason for migration from IPV4 to IPV6 is the small size of the address space in IPV4.
* Representation
* Address space
* Address space allocation
* Auto configuration
* Renumbering

**Representation:**

* A computer normally stores the address in binary but it is clear that 128 bits cannot easily be handled by humans.
* Several notation have been proposed to represent IPv6 addresses, when they are handled by humans.
* Two of these notations : binary, colon hexadecimal.



* Binary notations is used when the addresses are stored in a computer the colon hexadecimal divide the address into eight sections separated by colons.
* Abbreviation
* Mixed notation.

**Abbreviation:**

* An IPv6 address, in hexadecimal format is very long many of the digits are zeros.
* The leading zeros of a section can be omitted.
* 0074 can be written as 74, 000F as F , 0000 as 0, note that 3210 cannot be abbreviated.
* In hexadecimal notation. If there are consecutive sections consisting of zeros only, we can remove all the zeros and replace them with a double semi colons is called zero compression.

**FDEC: 0:0:0:0:BBFF:0: FFF ->FDEC::BBRR:0:FFFF**

* Note that this type of abbreviation is allowed only one per address, If there is more than one run of zero sections, only one of them can be compressed.

**Mixed notation:**

* Sometimes we see a mixed representation an IPV6 address. Colon hex and dotted decimal notation.
* This is appropriate during the transistion period in which an IPv4 address is embedded in an IPV6 address.

**Address space:**

* The address space of IPV6 contains 2 128 addresses, this address space is 2 96 times greater than the IPV4 address the size of the space is:

**340,282,366,920,938,463,374,607,431,768,211,456.**

* We assume that only 1/64 (almost 2 percent ) of the address in the space can be assigned to the people on planet earth and the rest are reserved for special purpose. Address depletion in this version is impossible.

**Three address types:**

* In IPv6 a destination address can belong to one of three categories.
* Uni cast
* Any cast
* Multi cast

**Unicast Address:**

* A unicast address defines a single interface. The packet sent to a unicast address will be routted to the intended receipient.

**Any cast Address:**

* It defines a group of computers that all share a single address.
* A packet with an any cast address is delivered to only one member of the group, the most reachable one.

**Multicast address:**

* A multicast address also defines a group of computers there is a difference between anycasting and multicasting.
* In any casting only one copy of the packet is sent to one of the members of the group, in multicasting each member of the group receives a copy.

**Address space allocation**

* The address space of Ipv6 is divided in to several blocks of varying size and each block is allocated for a special purpose.
* Most of the blocks are still unassigned and have been set aside for future use.

|  |  |  |  |
| --- | --- | --- | --- |
| BLOCK PREFIX | CIDR | BLOCK ASSIGNMENT | FRACTION |
| 0000 0000 | 0000::/8 | Special Address | 1/256 |
| 001 | 2000::/3 | Global unicast | 1/8 |
| 1111 110 | FC00::/7 | Uniqe local unicast | 1/128 |
| 1111 1110 10 | FE80::/10 | Link Local addresses | 1/1024 |
| 1111 1111 | FF00::/8 | Multicast addresses | 1/256 |

**Global unicast address**

* The block in the address space that is used for unicast (one-to-one) communication between two hosts in the internet is called the global unicast address block.

**Special addresses**

* Addresses that use the prefix(**0000::/8**) are reserved , but part of this block is used to define some special addresses.

**Unique local unicast block**

* It can be privately created and used by a site.
* The packet carrying this type of address as the destination address is not expected to be routed.

**Link local block**

* It is designed for private addresses.
* A sub block in this block can be used as a private address in a network.

**Auto configuration**

* One of the interest features of Ipv6 addressing is the auto configuration of hosts.
* In Ipv4 the host and routers are originally configured manually by the network manager.
* In Ipv6 the host can also configure itself using DHCP (Dynamic Host Configuration Protocol) protocol.

**Renumbering**

* To allow sites to change the service provider.
* Renumbering of the address prefix(n)was built in to Ipv6 addressing.
* Each site is given a prefix by the service provider to which it is connected, if the site changes the provider ,the address prefix needs to be changed.

**ROUTING ALGORITHM**

* **Distance Vector Routing,**
* **Link – State Routing.**

**Distance Vector Routing:**

* Its Goal is to Find the best Route.
* The first thing each node creates is its own Least – cost tree with the information it has about its immediate neighbors.
* In Distance Vector Routing, A router continuously tells all of its neighbors what it knows about the whole internet.
* In this method we discuss two important topics.
* Bellman – Ford Equation,
* Distance Vector.

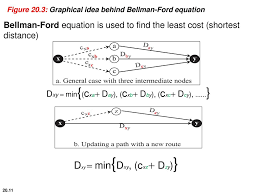
**Bellman – Ford Equation:**

* The heart of the Distance Vector Routing is the Famous Bellman – Ford Equation.
* This Equation is used to find the least cost (shortest Distance) between a Source node X and a Destination node Y Through some Intermediatery nodes and the least cost between the intermediatery nodes and the destination are given.

**DXY = Min{(Cxa + Day), (Cxb + Dby), (Cxc +Dzy )}**

* The Equations Become Simpler,

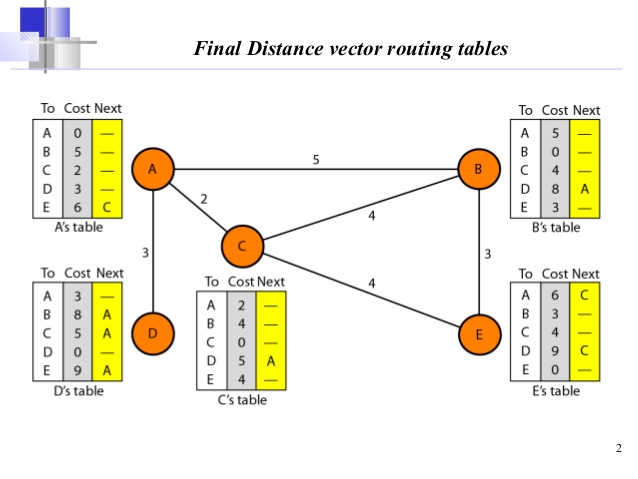
**DXY = Min{Dxy, (Cxz + Dzy)}**

****

* We can say that the Bellman – Ford Equation enables us to build a new Least – Cost path from previously established Least – Cost Paths.
* We can think of (a->y), (b->y) and (C-> y) as previously established Least – Cost paths and (x->y) as the Least – Cost path.

**Distance Vectors:**

* A Least – Cost tree is a Combination of least – cost paths from the root of the tree to all destinations. These paths are graphically glued together to form the tree.
* Distance vector Routing unglues these paths and creates a distance vector, a one – Dimensional array to represent a tree.

****

* Note that the name of the distance vector defines the root, the indexes define the destinations, and the value of each cell defines the least cost from the root to the destination.
* Later we can change a distance vector to a forwarding table, but we first need to find all distance vectors for an internet.
* These vectors cannot help the internet to effectively forward a packet, for example, node A thinks that it is not connected to node G because the corresponding cell shows the lowest cost of infinity.
* To improve these vectors, the node in the internet need to help each other by exchanging information.
* After each node has created its Vector, it sends a copy of the vector to all its immediate neighbors.
* After a node receives a distance vector from a neighbor, it updates its distance vector using the Bellman – Ford Equation.

**Distance Vector Routing Algorithm:**

Distance\_Vector\_Routing()

{

//Initialize (create initial vectors for the node)

D[myself] = 0

For(Y=1 to N)

{

If(Y is a Neighbor)

D[y]= c[myself][y]

else

D[y]==

}

Send vector {D[1],D[2],….D[N] to all neighbors}

//update(improve the vector with the vector received from a neighbor)

Repeat (Forever)

{

Wait (for a vector Dw from a neighbor w or any change in the link)

For (y=1 to N)

{

D[y]= min[D[y], (c[myself][w] + Dw[y])]

}

If (any change in the vector)

Send vector {D[1],D[2],….D[N] to all neighbors}

}

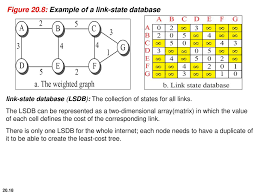
}//end of Distance Vector

**Link – State Routing:**

* A Routing Algorithm that Directly follows our discussion for Creating Least – Cost trees and forwarding tables is Link – State (LS) Routing.
* The term Link – State to Define the characteristics of a link that represent a network in the internet.

**Link – State Database (LSDB):**

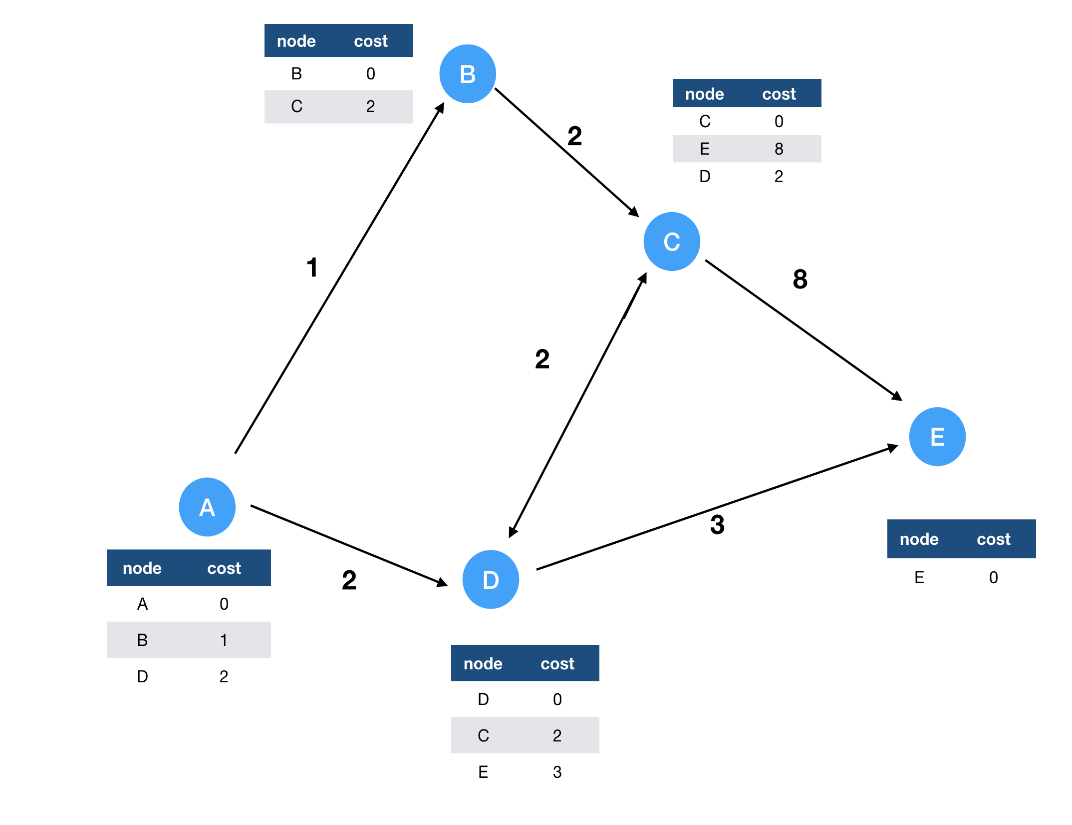
* To create a Least – Cost tree with this method, each node needs to have a complete map of the network, which means it needs to know the state of each link.
* This collection of states for all links is called the Link – State Database(LSDB).



* There is only one LSDB for the whole internet.

**Flooding:**

* Now the question is how each node can create this LSDB that contains information about the whole internet. This can be done by a process called Flooding.



**Dijkstra’s Algorithm:**

Dijkstra’s Algorithm()

{

//Initialization

Tree = {root}

For(y=1 to N)

{

If(Y is the root)

D[y] = 0

Elseif (Y is a neighbor)

D[y] = c [root][y]

Else

D[y] = ∞

}

//calculation

Repeat

{

Find a node w, with D[w] minimum among all nodes not in the Tree

Tree = Tree U {w} //add w to tree

// update distances for all neighbors of w

For (every node x , which is a neighbor of w and not in the Tree)

{

D[x] = min{D[x],(D[w] + c[w][x])}

}

} (until all nodes included in the Tree)

}//End of Dijkstra