4. Network Layer

The network layer is the third level of the Open Systems Interconnection Model (OSI Model) and the layer that provides data routing paths for network communication. Data is transferred in the form of packets via logical network paths in an ordered format controlled by the network layer. Logical connection setup, data forwarding, routing and delivery error reporting are the network layer's primary responsibilities.

A) Network Layer Design Issues

1. Store-and-Forward Packet Switching: A host with a packet to send transmits it to the nearest router, either on its own LAN or over a point-to-point link to the carrier. The packet is stored there until it has fully arrived so the checksum can be verified. Then it is forwarded to the next router along the path until it reaches the destination host, where it is delivered.

2. Services Provided to the Transport Layer: The network layer provides services to the transport layer at the network layer/transport layer interface. The network layer services have been designed with following goals:

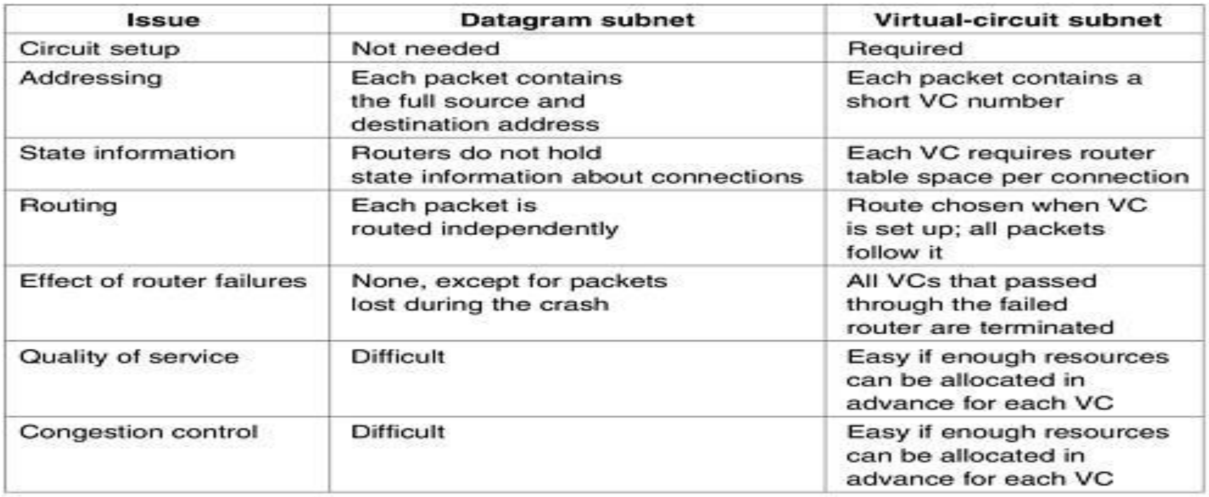
- The services should be independent of the router technology.
- The transport layer should be shielded from the number, type, and topology of the routers present.
- The network addresses made available to the transport layer should use a uniform numbering plan, even across LANs and WANs.

Given these goals, the designers of the network layer have a lot of freedom in writing detailed specifications of the services to be offered to the transport layer. This freedom often degenerates into a raging battle between two warring factions.

3. Implementation of Connectionless Service: If connectionless service is offered, packets are injected into the subnet individually and routed independently of each other. No advance setup is needed. In this context, the packets are frequently called datagrams and the subnet is called a datagram subnet.

4. Implementation of Connection-Oriented Service: For connection-oriented service, we need a virtual-circuit subnet. The idea behind virtual circuits is to avoid having to choose a new route for every packet sent. Instead, when a connection is established, a route from the source machine to the destination machine is chosen as part of the connection setup and stored in tables inside the routers. That route is used for all traffic flowing over the connection, exactly the same way that the telephone system works. When the connection is released, the virtual circuit is also terminated. With connection-oriented service, each packet carries an identifier telling which virtual circuit it belongs to.

5. Comparison of Virtual-Circuit and Datagram Subnets



Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

B) IP Based Networking:

1) Mobile IP:-

Mobile IP (or MIP) is an Internet Engineering Task Force (IETF) standard communications protocol that is designed to allow **mobile** device users to move from one network to another while maintaining a permanent **IP** address.
It enables the transfer of information to and from mobile computers, such as laptops and wireless communications. The mobile computer can change its location to a foreign network and still access and communicate with and through the mobile computer's home network.

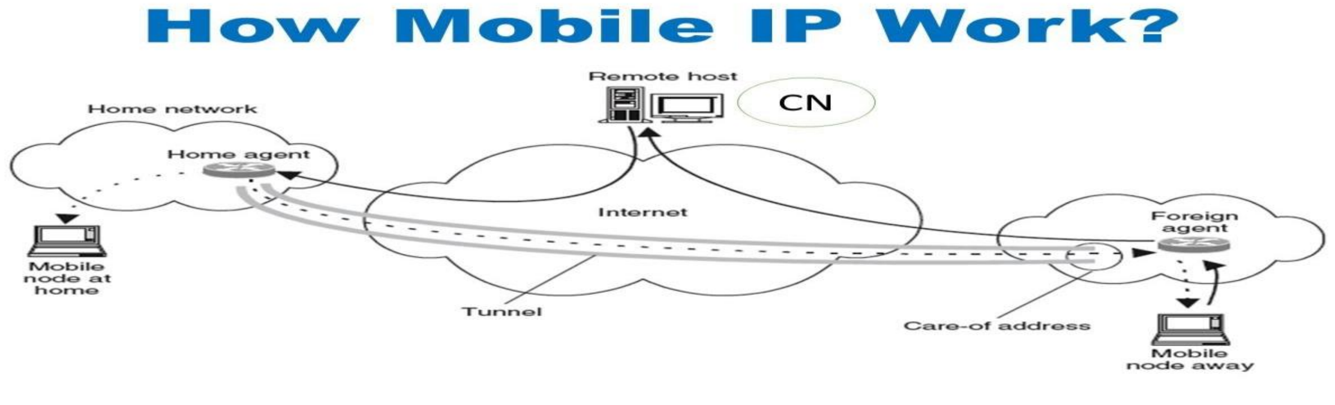
Mobile IP or IP-Mobility Management (IP-MM) is an open standard communication protocol defined by Internet Engineering Task Force (IETF) that allows mobile device users to move from one network to another without changing their IP address as a change in the IP address will interrupt ongoing TCP/IP communications. Mobile IP is an enhancement of the Internet Protocol (IP) which allows a node to change its point of attachment to the Internet without needing to change its IP address.

Mobile IP is independent of the physical layer technology as the mobility functions are performed at the network layer – any media that can support IP can support Mobile IP.

Components of a Mobile IP Network
Mobile IP has three major components as mentioned below –

- **Mobile Node:** A device such as a cell phone, personal digital assistant, or laptop whose software enables network roaming capabilities.
- **The Home Agent:** A router on the home network serving as the anchor point for communication with the mobile node; its tunnel packets from a device on the Internet, called a correspondent node, to the roaming mobile node.
- **The Foreign Agent:** A router that may function as the point of attachment for the mobile node when it roams to a foreign network delivers packets from the home agent to the mobile node.

The Mobile IP process has three main phases –

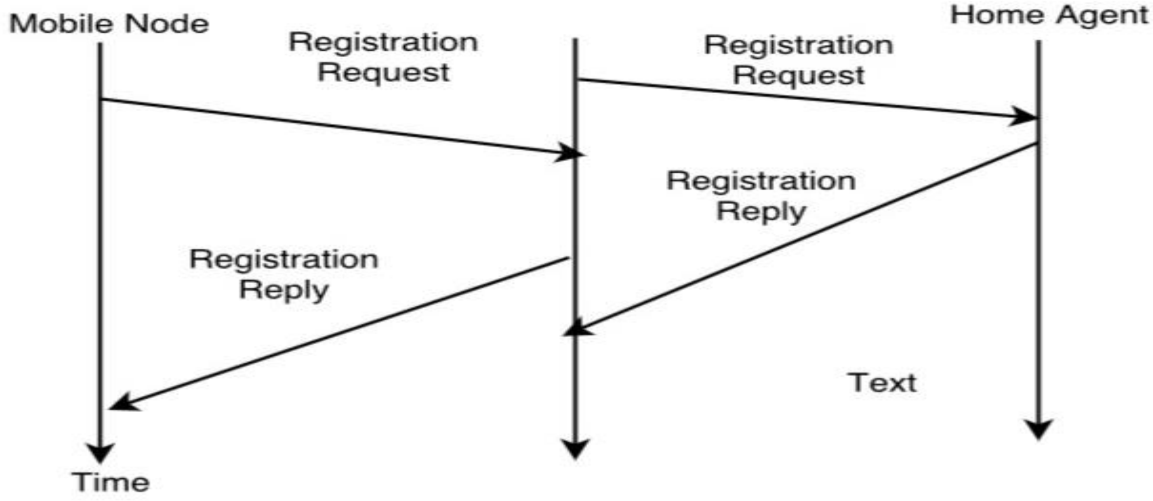


Phase I: Agent Discovery
This is the phase where mobile node discovers its foreign and home agents. A mobile node first determines its connected location by using ICMP router discovery messages. If it's connected location is with the local network, then the normal IP routing is used for the communication. When a mobile node determines that it has moved to a foreign network it obtains a care-of address from the foreign agent reflecting its current location.
Two types of "care-of" addresses exist –

- The care-of addresses acquired from a Foreign Agent: An IP address of a Foreign Agent that has an interface on the foreign network being visited by a Mobile node.
- The collocated care-of address: This represents the current position of the Mobile Node on the foreign network and can be used by only one Mobile Node at a time.

Phase II: Registration

This is the phase, where a mobile node registers its current location with the foreign agent and the home agent. If the connected location is identified as foreign location, then the mobile node looks for a foreign agent and registers itself with the foreign location and the foreign agent, in turn, notifies the home agent and creates a tunnel between itself and the home agent. During this phase, the Mobile node sends a registration request message to the foreign agent which forwards the message to the home agent. The home agent sends back a reply after updating its registration table with the home address and "care-of" address mapping. Thus, a successful Mobile IP registration sets up the routing mechanism for transporting packets to and from the mobile node as it roams.



Phase III: Tunneling

This is the phase where a reciprocal tunnel is set up by the home agent to the care-of address to route packets to the mobile node as it roams. The method by which mobile IP receives information from a network is called tunneling. It has two primary functions:

- Encapsulation of the data packet to reach the tunnel endpoint.
- Decapsulation, when the packet is delivered at that endpoint.

After the registration phase, the home agent now encapsulates all the packets intended for the mobile node and forwards those packets through the tunnel to the foreign agent. The foreign agent de-encapsulates the packet and forwards them to the mobile node. The return path from the mobile node is as per the standard IP routing principle where the foreign agent acts as a gateway for the mobile node.

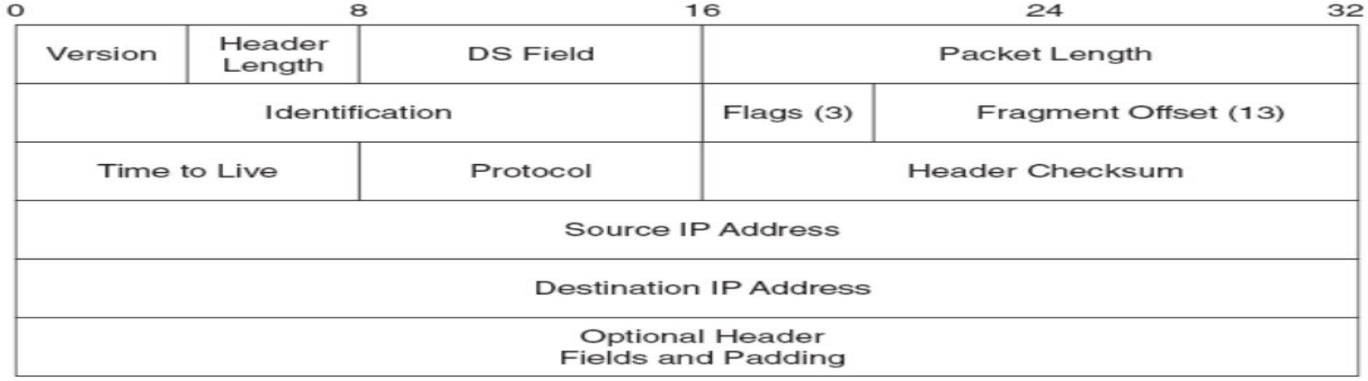
2) IP Addressing (IP Version Type)

A) IPV4

IPv4 is the most widely used Internet protocol across the internet today. Internet protocols are mostly responsible for addressing and forwarding of data in the Internet.

The network layer packet, also referred as datagram plays a central role in communication across the internet.

The basic format of the IPv4 datagram is shown below :



0		8	16	24 32
Version	Header Length	DS Field	Packet Length	
Identification			Flags (3)	Fragment Offset (13)
Time to Live		Protocol	Header Checksum	
Source IP Address				
Destination IP Address				
Optional Header Fields and Padding				

IPv4 Fields

- **Version**:- The first header field in an IP packet is the four-bit version field. The Version field indicates the format of the internet header. Version identifies the IP version to which the packet belongs. This four-bit field is set to binary 0100 to indicate version 4 (IPv4) or binary 0110 to indicate version 6 (IPv6).
- **Header length or Internet Header Length (IHL) :-** The second field (4 bits) is the Internet Header Length (IHL), this field specifies the size of the header.
- **Type of Service(ToS):-** now known as **Differentiated Services Code Point (DSCP).** The TOS field is used to carry information to provide quality of service features. An example is Voice over IP (VoIP) that is used for interactive data voice exchange.
- **Total Length:-** This 16-bit field defines the entire datagram size, including header and data, in bytes.
- **Identification, Flags, Fragment Offset:-** This three fields are used to fragment and reassemble large IP datagram.
- **Time To Live (TTL):-**It is of 8 bit field. This field specifies the maximum no of hops (Routers) the packet can travel before it is dropped.
- **Protocol:-**This field defines the protocol used in the data portion of the IP datagram. Eg. TCP, UDP etc.
- **Header Checksum:-** The 16-bit checksum field is used for error-checking of the header. At each hop, the checksum of the header must be compared to the value of this field. If a header checksum is found to be mismatched, then the packet is discarded.
- **Source address:-** Sets the source IP address.
- **Destination address:-** An IPv4 address indicating the receiver of the packet.
- **Options and Padding :-** Used to specify additional header fields if needed. They must be multiple of 32 bits.

B) IPV6

The exhaustion of the IPv4 address space has forced the technology world to look for newer solutions to the addressing scheme. Since IPv4 IP addresses use 32-bit addresses, there are a possible ~4.2 billion (2^32) possible IP addresses. Have we used them all up? No, but there are some address ranges that can't be used for technical/legacy reasons. Even if those technical/legacy reasons could be overcome, the IPv4 address space is still very constrained for a quickly growing Internet. In April 2010, the Regional Internet Registries (the "authorities") said that only 8% of the IPv4 addresses are unallocated and the remaining are expected to run within years.

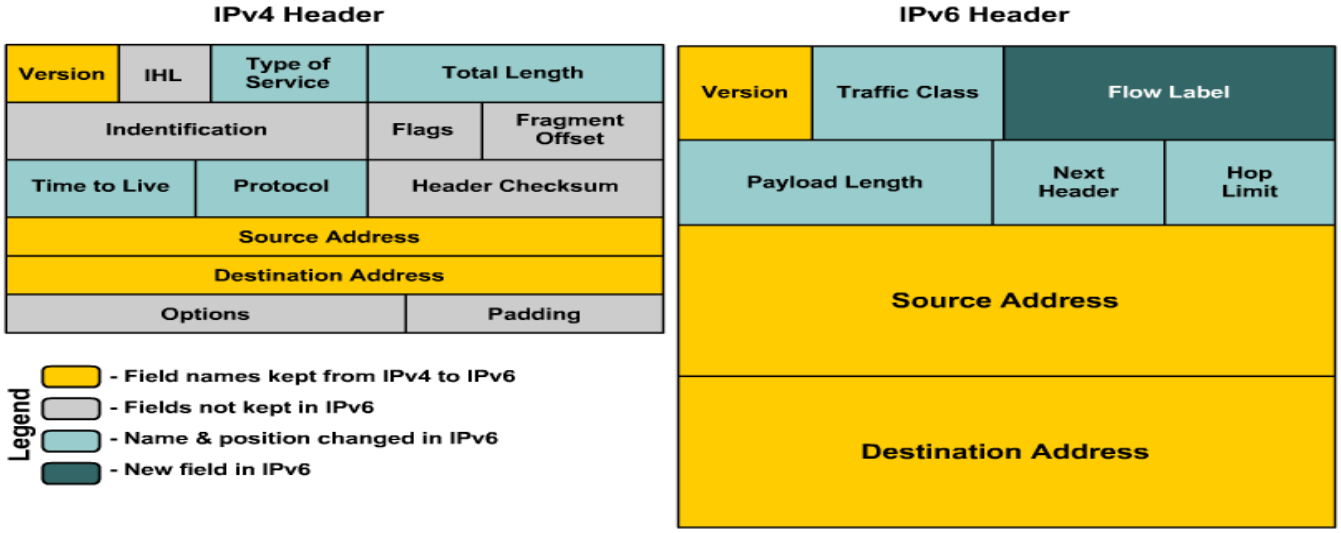
The main incentives to move to IPv6 are:

-IPv6 gives us 128 bits for address space that is (2^128) unique addresses.

-IPv6 has built-in feature for mobility.

-IPv6 has built-in support for IPSec.

-IPv6 supports for smooth transition from IPv4



The following list describes the function of each header field.

- **Version** – 4-bit Version number of Internet Protocol = 6.
- **Traffic Class** – 8-bit traffic class field. The nodes that originate a packet must identify different classes or different priorities of IPv6 packets. The nodes use the Traffic Class field in the IPv6 header to make this identification. The routers that forward the packets also use the Traffic Class field for the same purpose.
- **Flow Label** – 20-bit field. The IPv6 routers must handle the packets belonging to the same flow in a similar fashion. All packets that belong to the same flow must be sent with the same source address, same destination address.
- **Payload Length** – The 16-bit payload length field contains the length of the data field in octets/bits following the IPv6 packet header. The 16-bit Payload length field puts an upper limit on the maximum packet payload to 64 kilobytes.
- **Next Header** – 8-bit selector. Identifies the type of header that immediately follows the IPv6 header. wing the IPv6 header and located at the beginning of the data field (payload) of the IPv6 packet. This field usually specifies the transport layer protocol used by a packet's payload. The two most common kinds of Next Headers are TCP (6) and UDP (17), but many other headers are also possible. Similar to Protocol field in IPv4.
- **Hop Limit** – 8-bit integer. This field specifies the maximum no of hops (Routers) the packet can travel before it is dropped.
- **Source Address** – 128 bits. The address of the initial sender of the packet.
- **Destination Address** – 128 bits. The address of the intended recipient of the packet.

BASIS	IPV4	IPV6
Address Configuration	Supports Manual and DHCP configuration.	Supports Auto-configuration and renumbering
End-to-end connection integrity	Unachievable	Achievable
Address Space	It can generate 4.29 x 10^9 addresses.	It can produce quite a large number of addresses, i.e., 3.4 x 10^{38}.
Security features	Security is dependent on application	IPSEC is inbuilt in the IPv6 protocol
Address length	32 bits (4 bytes) 12:34:56:78	128 bits (16 bytes) 1234:5678:9abc: def0:1234:5678:9abc:def0
Address Representation	In decimal	In hexadecimal
Fragmentation performed by	Sender and forwarding routers	Only by the sender
Packet flow identification	Not available	Available and uses flow label field in the header
Checksum Field	Available	Not available
Message Transmission Scheme	Broadcasting	Multicasting and Anycasting
Encryption and Authentication	Not Provided	Provided
Packet size	576 bytes required, fragmentation optional	1280 bytes required without fragmentation
DNS records	Address (A) records, maps host names	Address (AAAA) records, maps host names
Local subnet group management	Internet Group Management Protocol (IGMP)	Multicast Listener Discovery (MLD)
IPSec	optional, external	required

3. Subnet Mask:

IP Subnetting is a process of dividing a large IP network in smaller IP networks. In Subnetting we create multiple small manageable networks from a single large IP network.

To best utilize available addresses if we put more than 16000000 hosts in a single network, due to broadcast and collision, that network will never work. If we put less hosts then remaining addresses will be wasted.

Subnetting provides a better way to deal with this situation. Subnetting allows us to create smaller networks from a single large network which not only fulfill our hosts' requirement but also offer several other networking benefits.

Network portion vs Host portion

Identifying network portion and host portion in an IP address is the first step of Subnetting. Subnetting can only be done in host portion. Subnet mask is used to distinguish the network portion from host portion in an IP address.

An IP address and a subnet mask both collectively provide a numeric identity to an interface. Both addresses are always used together. Without subnet mask, an IP address is an ambiguous address and without IP address a subnet mask is just a number.

Both addresses are 32 bits in length. These bits are divided in four parts. Each part is known as octet and contains 8 bits. Octets are separated by periods and written in a sequence.

Subnet mask assigns an individual bit for each bit of IP address. If IP bit belongs to network portion, assigned subnet mask bit will be turned on. If IP bit belongs to host portion, assigned subnet mask bit will be turned off. There are two popular notations to write the IP address and Subnet mask; Decimal notation and Binary notation. In decimal notation, a value range 1 to 255 represents a turned on bit while a value 0 (zero) represents a turned off bit. In binary notation, 1 (one) represents a turned on bit while 0 (zero) represents a turned off bit.

Slash Notation

It's a compact representation of Subnet mask. In this notation a slash (/) sign and total number of the on bits in subnet mask are written with IP address instead of full Subnet mask.

Following table lists some examples of IP addresses with Subnet mask in all three notations.

In Slash notation	In binary notation	In decimal notation
10.10.10.10/8	00001010.00001010.00001010.00001010 11111111.00000000.00000000.00000000	10.10.10.10 255.0.0.0
172.168.1.1/16	10101100.10101000.00000001.00000001 11111111.11111111.00000000.00000000	172.168.1.1 255.255.0.
192.168.1.1/24	11000000.10101000.00000001.00000001 11111111.11111111.11111111.00000000	192.168.1.1 255.255.255.0
192.168.1.1/28	11000000.10101000.00000001.00000001 11111111.11111111.11111111.11110000	192.168.1.1 255.255.255.240

IP classes

There are 4,294,967,296 IP addresses. Based on following rules, IP addresses are categorized in five classes; A, B, C, D and E.

Class of an IP address is determined by the value of first byte or octet.

Class	Starting Address	Ending Address	Subnet mask
A	0.0.0.0	127.255.255.255	255.0.0.0
B	128.0.0.0	191.255.255.255	255.255.0.0
C	192.0.0.0	223.255.255.255	255.255.255.0
D	224.0.0.0	239.255.255.255	Not applicable
E	240.0.0.0	255.255.255.255	Not applicable

Although we have nearly 4.3 billion IP addresses but not all are available for end devices. From these addresses, following addresses are reserved and cannot be assigned to end devices.

- **0.0.0.0**:- This address represents all networks.
- **127.0.0.0 to 127.255.255.255**: - This IP range is reserved for loopback testing.
- **224.0.0.0 to 239.255.255.255 (*Class D*)**: - This IP class is reserved for multicast.
- **240.0.0.0 to 255.255.255.254 (*Class E*)**: - This IP class is reserved for future use.
- **255.255.255.255**: - This address represents all hosts.

Besides these reserved address, we also cannot use the first and the last IP address of each network. First IP address is reserved for the network address and last IP address is reserved for the broadcast address. We can use only the addresses available between the network address and the broadcast address for end devices.

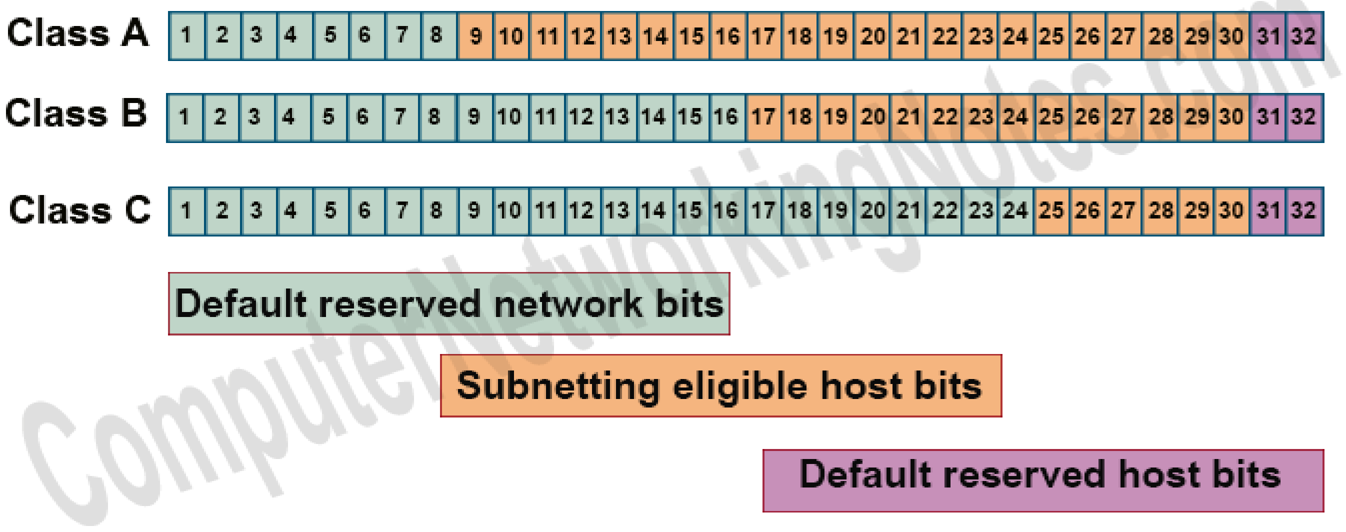
Reserve IP classes, network bits and host bits

In class A, B and C: -

- First 8, 16 and 24 bits are reserved for network portion respectively.
- Last 2 bits (31 & 32) are reserved for host portion.

Reserved network bits and host bits cannot be used in Subnetting.

Subnetting can be done only in Subnetting eligible bits.



Subnet

A subnet is a single small network created from a large network. In Subnetting we break a single large network in multiple small networks. These networks are known as subnets.

Network address and Broadcast address

In each network there are two special addresses; network address and broadcast address. Network address represents the network itself while broadcast address represents all the hosts which belong to it. These two addresses can't be assigned to any individual host in network. Since each subnet represents an individual network, it also uses these two addresses.

In simple language, in a single network only two IP addresses will be used for these addresses. But if we breaks this network in two small networks then four IP addressed will be used for these addresses. Network address and broadcast address are also known as Network ID and broadcast ID respectively.

Block Size

Block size is the sum of network address, valid host addresses and broadcast address. For example, if in a network there are 6 valid hosts than block size of that network is 8 (1 network address + 6 valid hosts + 1 broadcast address).

- A combination of all 32 represents a unique IP address.
- A combination of network bits in IP address represents the number of networks or subnets.
- A combination of host bits in IP address represents the number of total hosts.

To know how many combinations the number of bits provides or to get the number of combinations how many bits are required, we use the power of 2.

In 2^X the X is the number of bits.

Type of Subnetting

There are two types of Subnetting FLSM and VLSM. In FLSM, all subnets have equal number of host addresses and use same Subnet mask. In VLSM, subnets have flexible number of host addresses and use different subnet mask. FLSM is easy in implementation and simple in operation but wastes a lot of IP addresses. VLSM is hard in implementation and complex in operation but utilizes maximum IP addresses.

FLSM (Fixed Length Subnet Masks) Subnetting	**VLSM (Variable Length Subnet Masks) Subnetting**
All subnets are equal in size.	Subnets are variable in size.
All subnets have equal number of hosts.	Subnets have variable number of hosts.
All subnets use same subnet mask.	Subnets use different subnet masks.
It is easy in configuration and administration.	It is complex in configuration and administration.
It wastes a lot of IP addresses.	It wastes minimum IP addresses.
It is also known as classfull Subnetting.	It is also known as classless Subnetting.
It supports both classfull and classless routing protocols.	It supports only classless routing protocols.

A) FLSM (Fixed Length Subnet Masks) Subnetting

Let’s start with a simple class C network and play with some binary numbers:

192.168.1.0 (with the default subnet mask 255.255.255.0)

The network device knows which part is the network part and host part because of the subnet mask. The default subnet mask for network 192.168.1.0 is 255.255.255.0.

Here’s what that looks like in binary:

	Network	Network	Network	Hosts
IP address (decimal)	192	168	1	0
IP address (binary)	11000000	10101000	00000001	00000000
Subnet mask (decimal)	255	255	255	0
Subnet mask (binary)	11111111	11111111	11111111	00000000

The 1’s in the subnet mask indicate the network address part, the 0’s indicate the host part. In other words, the subnet mask tells us that the first 24 bits (192.168.1) are the network part and the remaining 8 bits (.0) are for hosts. Let’s write down these 8 host bits: Let’s set all of them to 1:

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

128 + 64 + 32 + 16 + 8 + 4 + 2 +1 = 255 with 8 bits the highest value we can create is 255, does this mean we can have 255 hosts in this network? The answer is no because for every network there are 2 addresses we can’t use:

Network address: this is the address where all the host bits are set to 0.

192	168	1	0
11000000	10101000	00000001	00000000

Broadcast address: this is the address where all host bits are set to 1.

192	168	1	255
11000000	10101000	00000001	11111111

We can use 192.168.1.1 – 192.168.1.254 as IP addresses for our hosts 254 Addresses.

Now let’s say I don’t want to have a single network where I can fit In 254 hosts, but I want to have 2 networks? Is this possible? It sure is! Basically what we are doing is taking a Class C network and chop it in 2 pieces, and this is what we call subnetting. The subnet mask defines the size of the network so if we want to create more subnets, we’ll have to “borrow” bits from the host part.

For every bit you borrow you can double the number of subnets, by borrowing 1 bit we create 2 subnets out of this single network. There are 8 host-bits so if we steal one to create more subnets this means we have only 7 bits left for hosts. Let’s do this, here’s what the new subnet mask will look like:

255	255	255	128
11111111	11111111	11111111	10000000

The first 24 bits are the same and we borrow the first bit from the 4th octet. This one has a value of 128 so our subnet mask becomes 255.255.255.128.

So what do our new subnets look like? Let’s zoom in on the 7 bits that we have left for our hosts:

128	64	32	16	8	4	2	1
N/A	0	0	0	0	0	0	0

We can’t use the first bit since it’s used for the network address now thanks to our subnet mask. What’s the largest decimal number we can create with these 7 bits?

64 + 32 + 16 + 8 + 4 + 2 + 1 = 127. Don’t forget that we start counting at 0 so in total we have 128 addresses. Our original class C network has now been subnetted into two subnets that each have 128 addresses.

Subnet #1

We start with 192.168.1.0 and the subnet mask is 255.255.255.128:

IP address	192	168	1	0
	11000000	10101000	00000001	00000000
Subnet mask	255	255	255	128
	11111111	11111111	11111111	10000000

Network address:

The network address has all host bits set to 0 so that’s why it is 192.168.1.0:

192	168	1	0
11000000	10101000	00000001	00000000

First usable host IP address:

The first usable host IP address is the one that comes after the network address, this will be 192.168.1.1:

192	168	1	1
11000000	10101000	00000001	00000001

Last usable host IP address:

The last IP address we can use for a host is the one before the broadcast address so this will be 192.168.1.126:

192	168	1	126
11000000	10101000	00000001	01111110

Broadcast address:

The broadcast address has all host bits set to 1 so the broadcast address we have is 192.168.1.127:

192	168	1	127
11000000	10101000	00000001	01111111

Subnet #2

The first subnet ended at 192.168.1.127 so we just continue with the next subnet at 192.168.1.128:

IP address	192	168	1	128
	11000000	10101000	00000001	10000000
Subnet mask	255	255	255	128
	11111111	11111111	11111111	10000000

Network address:

The network address has all host bits set to 0 so that's why it is 192.168.1.128:

192	168	1	0
11000000	10101000	00000001	10000000

First usable host IP address:

The first usable host IP address is the one that comes after the network address, this will be 192.168.1.129:

192	168	1	129
11000000	10101000	00000001	10000001

Last usable host IP address:

The last IP address we can use for a host is the one before the broadcast address so this will be 192.168.1.254:

192	168	1	254
11000000	10101000	00000001	11111110

Broadcast address:

The broadcast address has all host bits set to 1 so the broadcast address we have is 192.168.1.255:

192	168	1	255
11000000	10101000	00000001	11111111

That's it! That's the first network we just subnetted in 2 subnets and we found out what the network and broadcast addresses are, and what IP addresses we can use for hosts.

Let me show you another one, we take the same class C 192.168.1.0 network but now we want to have 4 subnets. For every host-bit we borrow we can double the number of subnets we can create, so by borrowing 2 host bits we can create 4 subnets.

Every host bit you "borrow" doubles the amount of subnets you can create.

Let's take a look at it in binary:

255.255.255.192 (11111111.11111111.11111111.11000000)

Calculate it from binary to decimal: 128+64 = 192.

With this subnet mask we only have 6 host bits to play with.

	Subnet #1	Subnet #2	Subnet #3	Subnet #4
Network Address	192.168.1.0	192.168.1.64	192.168.1.128	192.168.1.192
Block Size	2^6=64	2^6=64	2^6=64	2^6=64
Valid Block Size	2^6-2=62	2^6-2=62	2^6-2=62	2^6-2=62
Valid IP Range	192.168.1.1 - 192.168.1.62	192.168.1.65 - 192.168.1.126	192.168.1.129 - 192.168.1.190	192.168.1.193 - 192.168.1.254
Broadcast Address	192.168.1.63	192.168.1.127	192.168.1.191	192.168.1.255

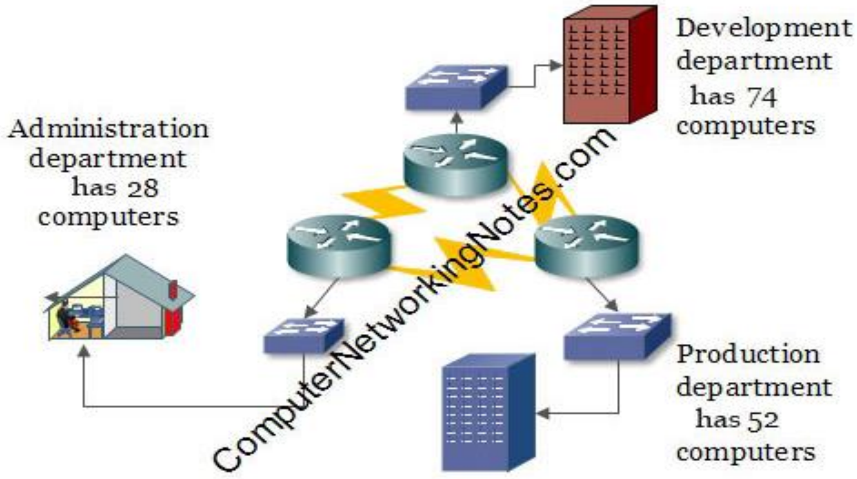
The smallest subnet you can create has a subnet mask of 255.255.255.252. This subnet has only 4 addresses, one network address, one broadcast address and two valid host IP addresses. This subnet is ideal for point-to-point links where we only require two usable IP addresses.

B) VLSM (Variable Length Subnet Masks) Subnetting

The biggest advantage of VLSM Subnetting is that, instead of forcing us to use a fixed size for all segments, it allows us to choose the individual size for each segment. This flexibility reduces the IP wastage. We can choose the size of subnet which closely matches with our requirement. Let's understand it with an example.

VLSM Example

Do the VLSM Subnetting of following network.


Development department has 74 computers
Administration department has 28 computers
ComputerNetworkingNotes.com
Production department has 52 computers

In this network: -

- Development department has 74 computers.
- Production department has 52 computers.
- Administration department has 28 computers.
- All departments are connected with each other via wan links.
- Each wan link requires two IP addresses.
- The given address space is 192.168.1.0/24.

Before we perform VLSM Subnetting for this network, let's understand how VLSM Subnetting actually works.

Basic concepts of VLSM Subnetting

VLSM Subnetting is the extended version of FLSM Subnetting. If you know how FLSM Subnetting works and how it is done, you already know the 90% of VLSM Subnetting. In FLSM, all subnets use same block size, thus Subnetting is required only one time. In VLSM, subnets use block size based on requirement, thus Subnetting is required multiple times.

The concept of VLSM Subnetting is relatively simple.

- Select block size for each segment. Block size must be greater than or equal to the actual requirement. Actual requirement is the sum of host addresses, network address and broadcast address.
- Based on block size arrange all segments in descending order.
- Do FLSM Subnetting for the block size of the first segment.
- Assign first subnet from subnetted subnets to the first segment.
- If next segment has similar block size, assign next subnet to it.
- If next segment has lower block size, do FLSM Subnetting again for the block size of this segment.
- From subnetted subnets exclude the occupied subnets. Occupied subnets are the subnets which provide the addresses which are already assigned.
- From available subnets, assign the first available subnet to this segment.
- Repeat above steps till the last segment of the network.

Let's implement above steps in our example network.

Step by step VLSM Subnetting

Step 1: The first step of VLSM Subnetting is selecting the appropriate block size for each segment. Following table lists all available block sizes.

2	4	8	16	32	64	128	256
512	1024	2048	4096	8192	16384	65536	32768
131072	262144	524288	1048576	2097152	4194304	8388608	16777216

To learn how block size is calculated, please see the third part of this tutorial.

While selecting appropriate block size for a given segment, always select a size which is adequate for host addresses plus two additional addresses; network address and broadcast address.

Actual requirement = Host requirement + Network address + broadcast address
Block Size >= Actual requirement

Following table shows the selection of block size in our example.

Segment	Host requirement	Actual requirement	Block size
Production	52	54	64
Wan link 1	2	4	4
Development	74	76	128
Wan link 2	2	4	4
Administration	28	30	32
Wan link 3	2	4	4

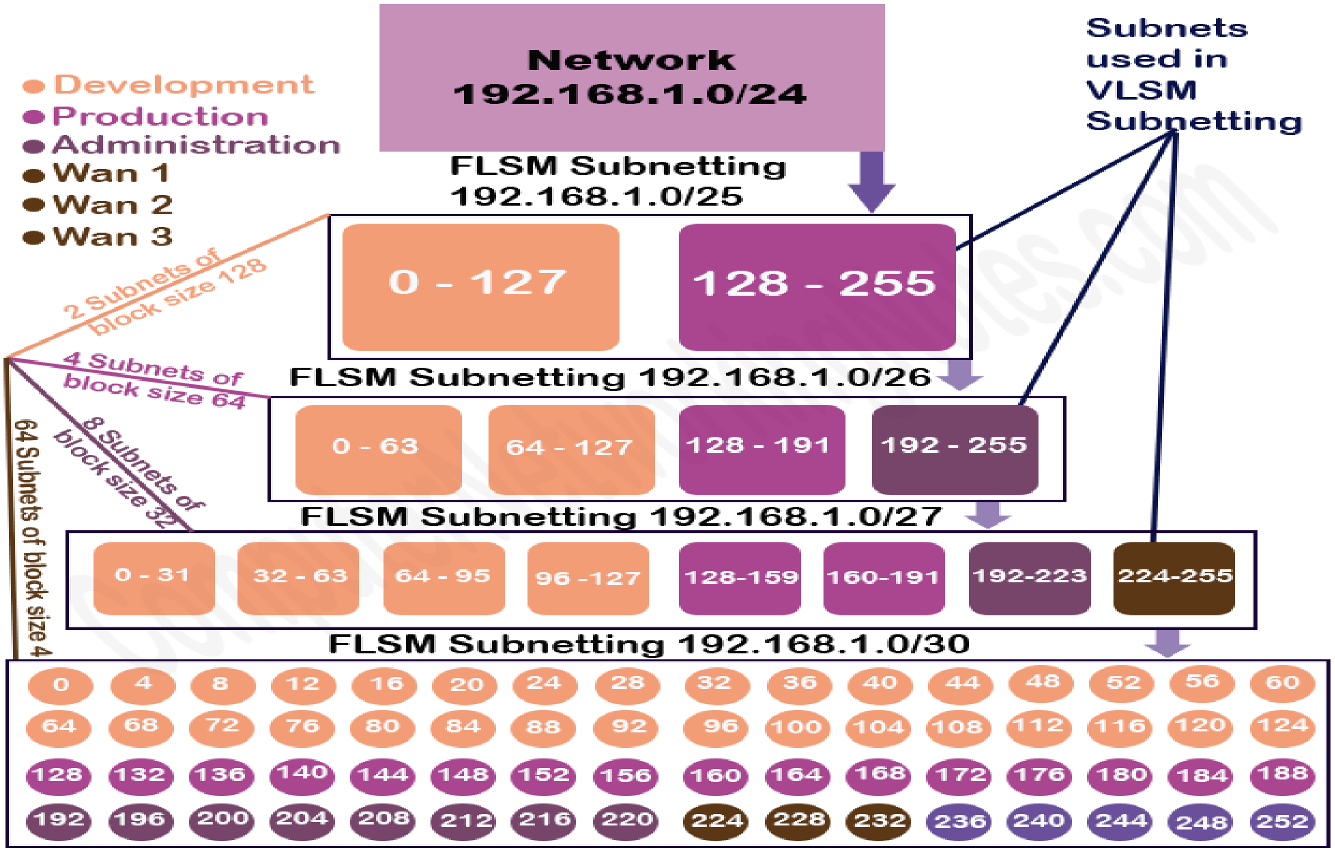
Step 2: The next step of VLSM Subnetting is arranging segments in descending order. Based on block size, following table arranges all segments in descending order.

Segment	Block size	Descending order
Development	128	1
Production	64	2
Administration	32	3
Wan link 1	4	4
Wan link 2	4	5
Wan link 3	4	6

Step 3: The next step of VLSM Subnetting is doing FLSM Subnetting and selecting appropriate subnets for segments from the subnetted subnets.

A single FLSM Subnetting provides a single block size for all of its subnets. If different block size is required, we have to perform the FLSM Subnetting again for that block size. How many times we have to perform the FLSM Subnetting is depend on how many unique block sizes we need. For instance, our example network requires four unique block sizes 128, 64, 32 and 4. For four block sizes, we have to perform FLSM Subnetting four times.

FLSM Subnetting is always performed in descending order. For ordering, block size is used. In our example, first we have to perform FLSM Subnetting for block size 128 then for block size 64 then for block size 32 and finally for block size 4.



Let's understand above process in more detail.

a) First largest segment (Block size 128)

Our first segment needs a block size of 128. The FLSM Subnetting of /25 provides us two subnets with the block size 128.

FLSM Subnetting of 192.168.1.0/25

Subnet	Subnet1	Subnet2
Network ID	192.168.1.0	192.168.1.128
First host address	192.168.1.1	192.168.1.129
Last host address	192.168.1.126	192.168.1.254
Broadcast ID	192.168.1.127	192.168.1.255

From Subnetted subnets assign first subnet to this segment.

Segment	Development
Requirement	74
CIDR	/25
Subnet mask	255.255.255.128
Network ID	192.168.1.0
First hosts	192.168.1.1
Last hosts	192.168.1.126
Broadcast ID	192.168.1.127

Since our second segment (Production) needs different block size (64), instead of using second subnet (Subnet2) for it, let's do Subnetting again.

b) Second largest segment (Block size 64)

The Subnetting of /26 provide us 4 subnets with block size 64.

Subnetting of 192.168.1.0/26

Subnet	Subnet 1	Subnet 2	Subnet 3	Subnet 4
Network ID	0	64	128	192
First address	1	65	129	193
Last address	62	126	190	254
Broadcast ID	63	127	191	255

From this Subnetting, we cannot use subnet 1 and subnet 2 as they are already occupied.

Subnet 1 and Subnet 2 provide addresses from 0 to 127 which are already assigned in the development department.

We can use subnet 3 for this segment (production).

Segment	Production
Requirement	52
CIDR	/26
Subnet mask	255.255.255.192
Network ID	192.168.1.128
First hosts	192.168.1.129
Last hosts	192.168.1.190
Broadcast ID	192.168.1.191

c) Third largest segment (block size 32)

The Subnetting of /27 provides us 8 network and 32 hosts.

Subnetting of 192.168.1.0/27

Subnet	Sub 1	Sub 2	Sub 3	Sub 4	Sub 5	Sub 6	Sub 7	Sub 8
Net ID	0	32	64	96	128	160	192	224
First Host	1	33	65	95	129	161	193	225
Last Host	30	62	94	126	158	190	222	254
Broadcast ID	31	63	95	127	159	191	223	255

Exclude the already occupied subnets (Sub1 to Sub6) and assign the first available subnet (Sub7) to this segment.

Segment	Administration
Requirement	28
CIDR	/27
Subnet mask	255.255.255.224
Network ID	192.168.1.192
First hosts	192.168.1.193
Last hosts	192.168.1.222
Broadcast ID	192.168.1.223

d) WAN Links (Block Size 4)

Last three segments require the block size of 4. The Subnetting of /30 gives us 64 subnets of block size 4.

Subnets of /30 Subnetting:-0, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 172, 176, 180, 184, 188, 192, 196, 200, 204, 208, 212, 216, 220, 224, 228, 232, 236, 240, 244, 248, 252, 256

Exclude already occupied subnets (0-56) and use first three available subnets 57, 58 and 59 for WAN links.

Subnet	Subnet 57	Subnet 58	Subnet 59
Network ID	224	228	232
First host	225	229	233
Last host	226	230	234
Broadcast ID	227	231	235

Assign subnet 57 to the WAN link 1.

Subnet	Subnet 57
Segments	Wan Link 1
Requirement	2
CIDR	/30
Subnet mask	255.255.255.252
Network ID	192.168.1.224
First hosts	192.168.1.225
Last hosts	192.168.1.226
Broadcast ID	192.168.1.227

Assign subnet 58 to the WAN link 2.

Subnet	Subnet 58
Segments	Wan Link 2
Requirement	2
CIDR	/30
Subnet mask	255.255.255.252
Network ID	192.168.1.228
First hosts	192.168.1.229
Last hosts	192.168.1.230
Broadcast ID	192.168.1.231

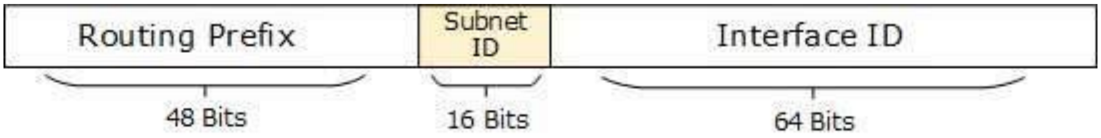
Assign subnet 59 to the WAN link 3.

Subnet	Subnet 59
Segments	Wan Link 3
Requirement	2
CIDR	/30
Subnet mask	255.255.255.252
Network ID	192.168.1.232
First hosts	192.168.1.233
Last hosts	192.168.1.234
Broadcast ID	192.168.1.235

We have assigned IP addresses to all segments. The subnets 60, 61, 62, 63 and 64 are still available for further use.

IPV6 Subnetting:

IPv6 addresses use 128 bits to represent an address which includes bits to be used for subnetting. The second half of the address (least significant 64 bits) is always used for hosts only. Therefore, there is no compromise if we subnet the network.



16 bits of subnet is equivalent to IPv4's Class B Network. Using these subnet bits, an organization can have another 65 thousands of subnets which is by far, more than enough.

Thus routing prefix is /64 and host portion is 64 bits. We can further subnet the network beyond 16 bits of Subnet ID, by borrowing host bits; but it is recommended that 64 bits should always be used for hosts addresses because auto-configuration requires 64 bits.

IPv6 subnetting works on the same concept as Variable Length Subnet Masking in IPv4.

/48 prefix can be allocated to an organization providing it the benefit of having up to /64 subnet prefixes, which is 65535 sub-networks, each having 2^{64}hosts. A /64 prefix can be assigned to a point-to-point connection where there are only two hosts (or IPv6 enabled devices) on a link.

3. Private and Public IP addresses (IP Types):

Private Ip Address:

A private IP address is an IP address that's reserved for internal use behind a router or other Network Address Translation (NAT) device, apart from the public. Private IP addresses are in contrast to public IP addresses, which are public and can *not* be used within a home or business network.

Sometimes a private IP address is also referred to as a *local IP address*.

Why Private IP Addresses Are Used

Instead of having devices inside a home or business network *each* use a public IP address, of which there's a limited supply, private IP addresses provide an entirely separate set of addresses that still allow access on a network but without taking up a public IP address space.

The devices in a network use the router to translate their requests through the public IP address, which can communicate with other public IP addresses and eventually to other local networks.

The hardware within a specific network that are using a private IP address can communicate with all the other hardware *within the confines of that network,* but will require a router to communicate with devices outside the network, after which the public IP address will be used for the communication.

For example, before landing on webpage, your device ,which uses a private IP address, requested this webpage through a router, which has a public IP address. Once the request was made and server responded to deliver the page, it was downloaded to your device through a public IP address before reaching your router, after which it got handed off to your private/local address to reach your device.

Public IP Address:

A public IP address is an IP address that your home or business router receives from your ISP. Public IP addresses are required for any publicly accessible network hardware, like for your home router as well as for the servers that host websites. Public IP addresses are what differentiate all devices that are plugged into the public internet. Each and every device that's accessing the internet is using a unique IP address. In fact, a public IP address is sometimes called an *Internet IP*.

Most networks that host websites will have static IP addresses because they want to make sure that users can have constant access to their server. Having an IP address that changes would defeat the purpose,
as DNS records would need to be updated once the IP changes, which might cause unwanted downtime.

Home networks, on the other hand, almost always are assigned dynamic IP addresses for the opposite reason. If an ISP gave your network an unchanging address, it may be more likely to be abused by customers who are hosting websites from home, or by hackers who can try the same IP address over and over until they breach your network.

Hiding Your Public IP Address

You can't hide your public IP address from your ISP because all of your traffic has to move through them before reaching anything else on the internet. However, you *can* hide your IP address from the websites you visit, as well as encrypt all of the data transfers (thus hiding *traffic* from your ISP), by first filtering all your data through a virtual private network (VPN).

More Information on IP Addresses

When a device like a router is plugged in, it receives a public IP address from an ISP. It's the devices that are then connected to the router that are given private IP addresses.

As we mentioned above, private IP addresses can't communicate directly with a public IP address. This means if a device that has a private IP address is connected directly into the internet, and therefore becomes non-routable, the device will have no network connection until the address is translated into a working address through a NAT, or until the requests it's sending are sent through a device that does have a valid public IP address.

All traffic from the internet can interact with a router. This is true for everything from regular HTTP traffic to things like FTP and RDP. However, because private IP addresses are hidden behind a router, the router must know which IP address it should forward information to if you're wanting something like an FTP server to be set up on a home network.

For this to work properly for private IP addresses, you must set up port forwarding. Forwarding one or more ports to a specific private IP address involves logging into the router to access its settings, and then choosing which port(s) to forward, and to where it should go.

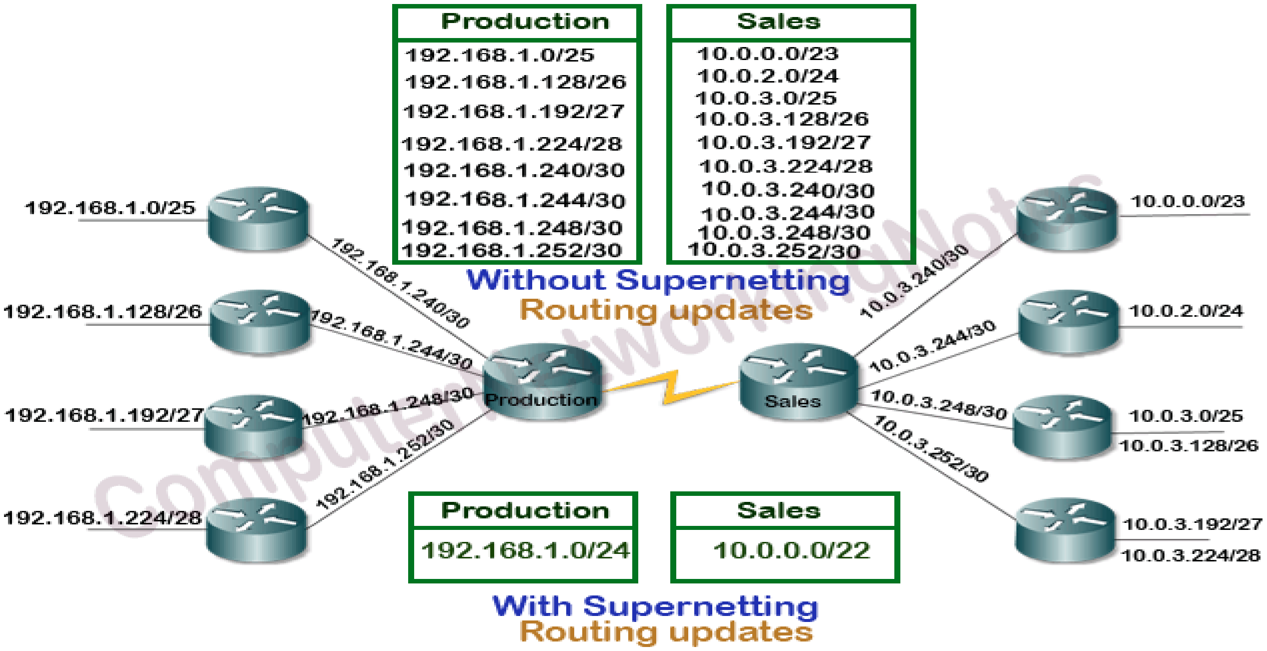
4. Supernetting/ CIDR (Classless Inter-Domain Routing)/ Route Summarization/ Route Aggregation

A supernet is created by combining several Internet Protocol (IP) networks or subnets into one network with a single classless interdomain routing (CIDR) prefix. The new combined network has the same routing prefix as the collection of the prefixes of the subnets. Supernetting enables organizations to modify their network size and minimize the extensive requirement of network routing devices by combining several independent routes. It also helps to conserve address space and helps the router to effectively store routing information and minimize processing overheads while matching the routes.

Why Supernetting is done?

Supernetting is mainly done for optimizing the routing tables. A routing table is the summary of all known networks. Routers share routing tables to find the new path and locate the best path for destination.

Without Supernetting, router will share all routes from routing tables as they are. With Supernetting, it will summarize them before sharing. Route summarization reduces the size of routing updates dramatically.



Production	Sales
192.168.1.0/25	10.0.0.0/23
192.168.1.128/26	10.0.2.0/24
192.168.1.192/27	10.0.3.0/25
192.168.1.224/28	10.0.3.128/26
192.168.1.240/30	10.0.3.192/27
192.168.1.244/30	10.0.3.224/28
192.168.1.248/30	10.0.3.240/30
192.168.1.252/30	10.0.3.244/30
	10.0.3.248/30
	10.0.3.252/30

Without Supernetting
Routing updates

Production	Sales
192.168.1.0/24	10.0.0.0/22

With Supernetting
Routing updates

Advantage of Supernetting

Supernetting provides following advantages.

- It reduces the size of routing updates.
- It provides a better overview of network.
- It decreases the use of resources such as Memory and CPU.
- It decreases the required time in rebuilding the routing tables.

Supernetting components

Each route advertises a certain number of addresses including network ID, broadcast ID and subnet mask. We can use a term Block size to refer all these addresses collectively.

In order to perform the Supernetting, we need Network ID, CIDR Value, Broadcast ID, Subnet Mask and Block Size of each route.

- Network ID and broadcast ID are used to check the alignment of routes. Supernetting can be performed only if routes are sequential.
- Block size is used to calculate the summarized route from given routes.

- Subnet mask and CIDR value is the same thing in different notations. Both are used to find the ON network bits in IP address. You may use any notation. Since, an advertise route is the combination of network ID and CIDR value, we only need to figure out the broadcast ID, subnet mask and block size.
  For block size use following formulas:-
  32 – CIDR Value = Number of host bits
  Block size = $2^{\text{Number of host bits}}$
  For example if CIDR value is 25 then block is 128.
  32 -25 = 7
  2^7 = 128
  Broadcast ID is the last address of network. Once you know the block size, to calculate the broadcast ID, simply count the addresses starting from network ID till the last address of the block.
  For example if network ID is 192.168.1.0/25 and block size is 128 and then broadcast ID will be 192.168.1.127/25.

Supernetting chart

CIDR	Subnet mask	Block Size
/8	255.0.0.0	16777216
/9	255.128.0.0	8388608
/10	255.192.0.0	4194304
…		
….		
/28	255.255.255.240	16
/29	255.255.255.248	8
/30	255.255.255.252	4

Key points of Supernetting

Supernetting can be done only in same address space. If address space is completely different between two or more routes, they cannot be summarized in a single route. For example, we can't summarize the route 192.168.1.0/25 with the route 193.168.1.128/25.

A route can be summarized only in a route which is bigger than it in block size. For example we can't summarize a route of block size 64 in a route of block size 32 but we can summarize two routes of block size 32 in a single route of block size 64.

The easiest way of calculating the summarized route is adding the block size of all sequential routes and using the Subnetting which provides the block size that is equal to the result of addition. For example if we have two sequential routes of block size 16, we can summarize them in a single route of block size 32.

Summarization can be done only in available bock sizes. For example if we have 5 routes of block size 8, we cannot summarize them in single route of block size 40 (8x5). 40 is not a valid block size. For valid block sizes see the Supernetting chart give above. In this case, the best choice is summarizing first four routes is single summarized route of block size 32 and keeping the fifth route as it is.

Supernetting Examples Explained Step by Step

Above we took the two examples of Supernetting. Let's understand how Supernetting was performed in them step by step.

Arrange all the routes in ascending order based on their after slash value (also known CIDR value). If CIDR value is same in two or more routes, use their IP addresses for ordering.

Supernetting Example 1	After slash value or CIDR Value	Supernetting Example 2	After slash value or CIDR Value
192.168.1.0/25	25	10.0.0.0/23	23
192.168.1.128/26	26	10.0.2.0/24	24
192.168.1.192/27	27	10.0.3.0/25	25
192.168.1.224/28	28	10.0.3.128/26	26
192.168.1.240/30	30	10.0.3.192/27	27
192.168.1.244/30	30	10.0.3.224/28	28
192.168.1.248/30	30	10.0.3.240/30	30
192.168.1.252/30	30	10.0.3.244/30	30
		10.0.3.248/30	30
		10.0.3.252/30	30

Write the CIDR value, Subnet Mask, Network ID, Broadcast ID and block size of each route.

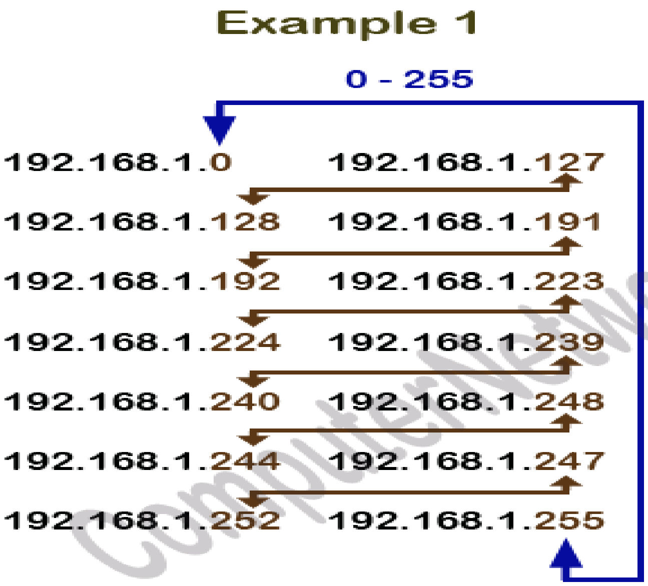
Supernetting Example 1

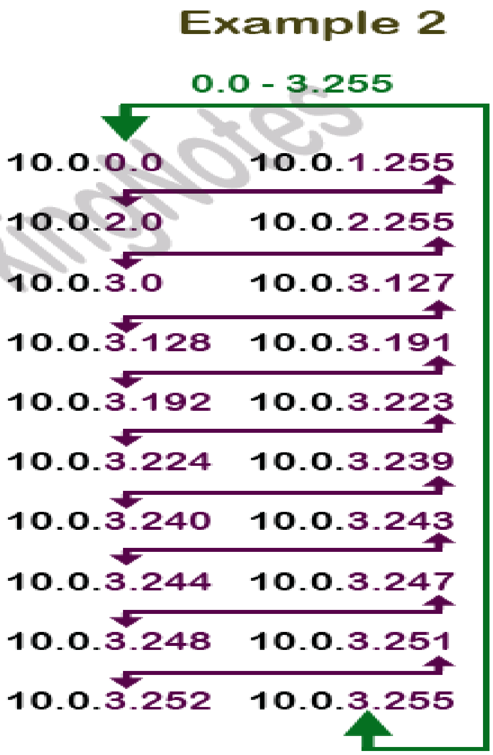
Route	CIDR value	Subnet Mask	Network ID	Broadcast ID	Block Size
192.168.1.0/25	25	255.255.255.128	192.168.1.**0**	192.168.1.**127**	128
192.168.1.128/26	26	255.255.255.192	192.168.1.**128**	192.168.1.**191**	64
192.168.1.192/27	27	255.255.255.224	192.168.1.**192**	192.168.1.**223**	32
192.168.1.224/28	28	255.255.255.240	192.168.1.**224**	192.168.1.**239**	16
192.168.1.240/30	30	255.255.255.252	192.168.1.**240**	192.168.1.**248**	4
192.168.1.244/30	30	255.255.255.252	192.168.1.**244**	192.168.1.**247**	4
192.168.1.248/30	30	255.255.255.252	192.168.1.**248**	192.168.1.**251**	4
192.168.1.252/30	30	255.255.255.252	192.168.1.**252**	192.168.1.**255**	4

Supernetting Example 2

Route	CIDR value	Subnet Mask	Network ID	Broadcast ID	Block Size
10.0.0.0/23	23	255.255.254.0	10.0.0.**0**	10.0.1.**255**	512
10.0.2.0/24	24	255.255.255.0	10.0.2.**0**	10.0.2.**255**	256
10.0.3.0/25	25	255.255.255.128	10.0.3.**0**	10.0.3.**127**	128
10.0.3.128/26	26	255.255.255.192	10.0.3.**128**	10.0.3.**191**	64
10.0.3.192/27	27	255.255.255.224	10.0.3.**192**	10.0.3.**223**	32
10.0.3.224/28	28	255.255.255.240	10.0.3.**224**	10.0.3.**239**	16
10.0.3.240/30	30	255.255.255.252	10.0.3.**240**	10.0.3.**243**	4
10.0.3.244/30	30	255.255.255.252	10.0.3.**244**	10.0.3.**247**	4
10.0.3.248/30	30	255.255.255.252	10.0.3.**248**	10.0.3.**251**	4
10.0.3.252/30	30	255.255.255.252	10.0.3.**252**	10.0.3.**255**	4

Group the routes based on sequence. If a route's network ID starts from where previous route's broadcast ID ends, it is a sequential route. But if it does not start from where previous route ends, it is not a sequential route.





Add the block size of all sequential routes.

In first example, sum of block sizes is 256 and in second example it is 1024.

Check the nearest valid block size which provides equal or less number of addresses. The block size 256 and 1024 exactly match with our requirement. The Subnetting of /24 and /22 give us the block size of 256 and 1024 respectively.

To write the summarize route, use the network ID of first route with the CIDR value or the subnet mask of the summarized route.

In first example, network ID of the first route is 192.168.1.0 and the CIDR value of summarized route is /24. Thus, the summarized route for first example will be 192.168.1.0/24.

Same way in second example, network ID of first route is 10.0.0.0 and the CIDR value of summarized route is /22. So, the summarize route for second example will be 10.0.0.0/22.

5. Multicast Address

A multicast address is a single IP data packet set that represents a network host group. Multicast addresses are available to process datagrams or frames intended to be multicast to a designated network service. Multicast addressing is applied in the link layer (Layer 2 of the OSI Model) and the Internet layer (Layer 3 of the OSI Model) for IP versions 4 (IPv4) and 6 (IPv6).
Datagrams with multicast address are simultaneously transmitted to one or more multicast host groups or networked computers.

Multicast addresses range from 224.0.0.0 to 239.255.255.255. Examples for IPV4-reserved addresses for multicasting are as follows:
224.0.0.0: Base address reserved
224.0.0.1: Used for all multicasting host groups
224.0.0.2: Used for all subnet routers
224.0.0.5 and 224.0.0.6: Used by Open Shortest Path First, an interior gateway protocol for all network segment routing information
Multicast addresses in IPV4 are defined using leading address bits of 1110, which originate from the classful network design of the early Internet when this group of addresses was designated as Class D. Multicast addresses in IPV6 have the prefix ff00::/8. IPv6 multicast addresses are generally formed from four-bit groups.

6. Broadcast Address

A broadcast address is a special Internet Protocol (IP) address used to transmit messages and data packets to network systems. Network administrators (NA) verify successful data packet transmission via broadcast addresses. Dynamic Host Configuration Protocol (DHCP) and Bootstrap Protocol (BOOTP) clients use broadcast IP addresses to locate and transmit respective server requests.

If you want to reach the device on network, you send a packet destined to its ip address. A broadcast IP is a special, reserved address used to send to all IP devices on a Local Area Network, and there is a special IP address reserved for it: 255.255.255.255 for IPv4 (though it was historically 0.0.0.0).

Broadcasting is convenient, but very resource wasteful. Broadcasts have to be replicated to the entire network and every single network interface has to ingest and process the broadcast packet. Because of this wastefulness, IPv6 eliminated the broadcast address in favor of a family of multicast and anycast addresses that could address smaller subsets of the network rather than blasting data to every device.

BASIS	BROADCAST	MULTICAST
Basic	The packet is transmitted to all the hosts connected to the network.	The packet is transmitted only to intended recipients in the network.
Transmission	One-to-all.	One-to-many.
Management	Broadcasting does not require any group management.	Multicasting requires group management to define the group of hosts/stations which will receive packets.
Bandwidth	Bandwidth is wasted.	Bandwidth is utilized efficiently.
Traffic	Unnecessarily huge amount traffic is generated in the network.	Traffic is under control.
Process	Slow.	Fast.

C) Routing

There are two basic methods of building a routing table:

1. Static Routing/ Non-Adaptive Routing

A static routing table is created, maintained, and updated by a network administrator, manually. A static route to every network must be configured on every router for full connectivity. This provides a granular level of control over routing, but quickly becomes impractical on large networks. Routers will not share static routes with each other, thus reducing CPU/RAM overhead and saving bandwidth. However, static routing is not fault-tolerant, as any change to the routing infrastructure (such as a link going down, or a new network added) requires manual intervention. Routers operating in a purely static environment cannot seamlessly choose a better route if a link becomes unavailable.

Advantages of Static Routing

- Minimal CPU/Memory overhead
- No bandwidth overhead (updates are not shared between routers)
- Granular control on how traffic is routed

Disadvantages of Static Routing

- Infrastructure changes must be manually adjusted
- No "dynamic" fault tolerance if a link goes down
- Impractical on large network

2. Dynamic Routing/ Adaptive Routing

A **dynamic routing** table is created, maintained, and updated by a routing protocol running on the router. Examples of routing protocols include RIP (Routing Information Protocol) and OSPF (Open Shortest Path First).Routers do share dynamic routing information with each other, which increases CPU, RAM, and bandwidth usage. However, routing protocols are capable of dynamically choosing a different (or better) path when there is a change to the routing infrastructure.

Advantages of Dynamic Routing

- Simpler to configure on larger networks
- Will dynamically choose a different (or better)

Disadvantages of Dynamic Routing

- Updates are shared between routers, thus consuming bandwidth
- Routing protocols put additional load on router CPU/RAM
- The choice of the "best route" is in the hands of the routing protocol, and not the network administrator.

Key Differences Between Static and Dynamic Routing

i. The routers are configured manually, and the table is also created manually in static routing whereas in dynamic routing the configuration and table creation is automatic and router driven.

ii. In static routing, the routes are user-defined while in dynamic routing the routes are updated as topology changes.

iii. Static routing does not employ complex algorithms. As against, dynamic routing uses the complex algorithm for calculating shortest path or route.

iv. Dynamic routing is suitable for large networks where the number of hosts is high. Conversely, static routing can be implemented in a small network.

v. When a link fails in static routing, the rerouting is discontinued and requires manual intervention to route traffic. In contrast, link failure in dynamic routing does not disrupt rerouting.

vi. The message broadcast and multicast in dynamic routing makes it less secure. On the other hand, static routing does not involve advertisement which makes it more secure.

vii. Dynamic routing involves protocols such as RIP, EIGRP, BGP, etc. Inversely, static routing does not require such protocols.

viii. Static routing does not need any additional resources while dynamic routing requires additional resources such as memory, bandwidth, etc.

BASIS	STATIC ROUTING	DYNAMIC ROUTING
Configuration	Manual	Automatic
Routing table building	Routing locations are hand-typed	Locations are dynamically filled in the table.
Routes	User defined	Routes are updated according to change in topology.
Routing algorithms	Doesn't employ complex routing algorithms.	Uses complex routing algorithms to perform routing operations.
Implemented in	Small networks	Large networks
Link failure	Link failure obstructs the rerouting.	Link failure doesn't affect the rerouting.
Security	Provides high security.	Less secure due to sending broadcasts and multicasts.
Routing protocols	No routing protocols are indulged in the process.	Routing protocols such as RIP, EIGRP, etc are involved in the routing process.
Additional resources	Not required	Needs additional resources to store the information.

D)**Routing Algorithm:**

1. Shortest path routing

Shortest path routing refers to the process of finding paths through a network that have a minimum of distance or other cost metric. Routing of data packets on the Internet is an example involving millions of routers in a complex, worldwide, multilevel network. Optimum routing on the Internet has a major impact on performance and cost.

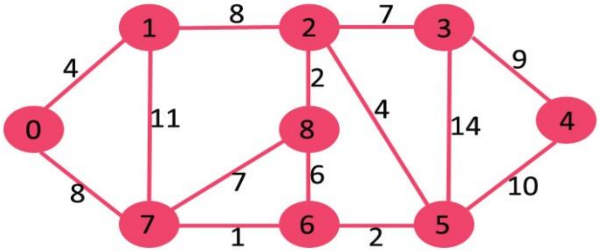
Limitations

The main limitations of simple shortest-path routing have to do with real-world problems that occur in large networks. We can't just keep adding nodes to a huge routing table at each and every node. Also, as nodes are added, the number of failing links, changes in topology, and other events that trigger re-builds throughout the network - these events will occur more frequently.
A routing algorithm is not enough to design a network. We need a complete routing protocol to deal with real-world issues.

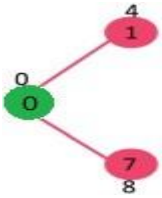
Below are the detailed steps used in Dijkstra's algorithm to find the shortest path from a single source vertex to all other vertices in the given graph:

a) Create a set *sptSet* (shortest path tree set) that keeps track of vertices included in shortest path tree, i.e., whose minimum distance from source is calculated and finalized. Initially, this set is empty.
b) Assign a distance value to all vertices in the input graph. Initialize all distance values as INFINITE. Assign distance value as 0 for the source vertex so that it is picked first.
c) While *sptSet* doesn't include all vertices
   i. Pick a vertex u which is not there in *sptSet* and has minimum distance value.
   ii. Include u to *sptSet*.
   iii. Update distance value of all adjacent vertices of u. To update the distance values, iterate through all adjacent vertices. For every adjacent vertex v, if sum of distance value of u (from source) and weight of edge u-v, is less than the distance value of v, then update the distance value of v.

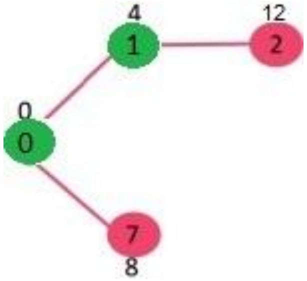
Let us understand with the following example:



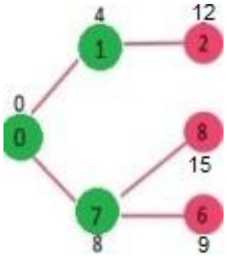
The set *sptSet* is initially empty and distances assigned to vertices are {0, INF, INF, INF, INF, INF, INF, INF} where INF indicates infinite. Now pick the vertex with minimum distance value. The vertex 0 is picked, include it in *sptSet*. So *sptSet* becomes {0}. After including 0 to *sptSet*, update distance values of its adjacent vertices. Adjacent vertices of 0 are 1 and 7. The distance values of 1 and 7 are updated as 4 and 8. Following subgraph shows vertices and their distance values, only the vertices with finite distance values are shown. The vertices included in SPT are shown in green colour.



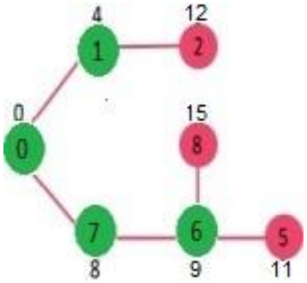
Pick the vertex with minimum distance value and not already included in SPT (not in sptSET). The vertex 1 is picked and added to sptSet. So sptSet now becomes {0, 1}. Update the distance values of adjacent vertices of 1. The distance value of vertex 2 becomes 12.



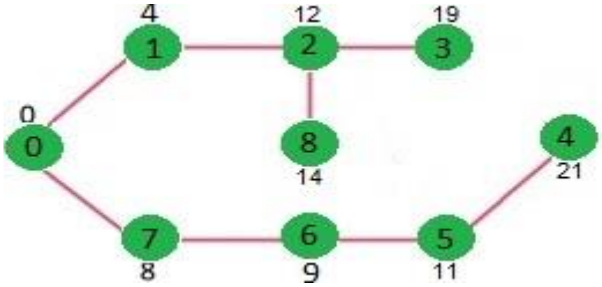
Pick the vertex with minimum distance value and not already included in SPT (not in sptSET). Vertex 7 is picked. So sptSet now becomes {0, 1, 7}. Update the distance values of adjacent vertices of 7. The distance value of vertex 6 and 8 becomes finite (15 and 9 respectively).



Pick the vertex with minimum distance value and not already included in SPT (not in sptSET). Vertex 6 is picked. So sptSet now becomes {0, 1, 7, 6}. Update the distance values of adjacent vertices of 6. The distance value of vertex 5 and 8 are updated.



We repeat the above steps until *sptSet* doesn't include all vertices of given graph. Finally, we get the following Shortest Path Tree (SPT).



2. Flooding

Flooding is the static routing algorithm. In this algorithm, every incoming packet is sent on all outgoing lines except the line on which it has arrived.

One major problem of this algorithm is that it generates a large number of duplicate packets on the network.

Several measures are takes to stop the duplication of packets. These are:

1. One solution is to include a hop counter in the header of each packet. This counter is decremented at each hop along the path. When this counter reaches zero the packet is discarded. Ideally, the hop counter should become zero at the destination hop, indicating that there are no more intermediate hops and destination is reached. This requires the knowledge of exact number of hops from a source to destination.

2. Another technique is to keep the track of the packed that have been flooded, to avoid sending them a second time. For this, the source router put a sequence number in each packet it receives from its hosts. Each router then needs a list per source router telling which sequence numbers originating at that source have already been seen. If an incoming packet is on the list, it is not flooded.

3. Another solution is to use **selective flooding.** In selective flooding the routers do not send every incoming packet out on every output line. Instead packet is sent only on those lines which are approximately going in the right direction.

Advantages

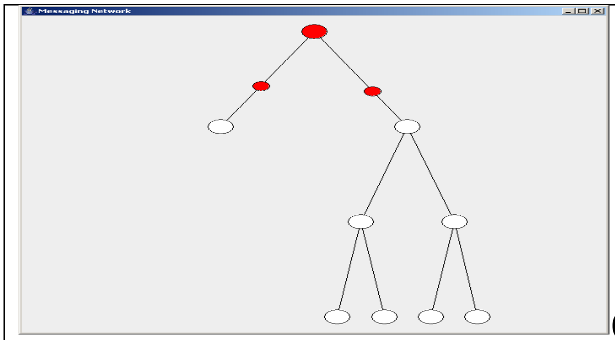
- If a packet can be delivered, it will (probably multiple times).
- Since flooding naturally utilizes every path through the network, it will also use the shortest path.
- This algorithm is very simple to implement.

Disadvantages

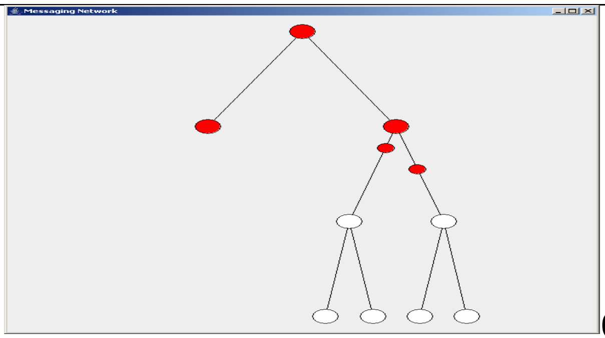
- Flooding can be costly in terms of wasted bandwidth. While a message may only have one destination it has to be sent to every host. In the case of a ping flood or a denial of service attack, it can be harmful to the reliability of a computer network.
- Messages can become duplicated in the network further increasing the load on the networks bandwidth as well as requiring an increase in processing complexity to disregard duplicate messages.

Examples

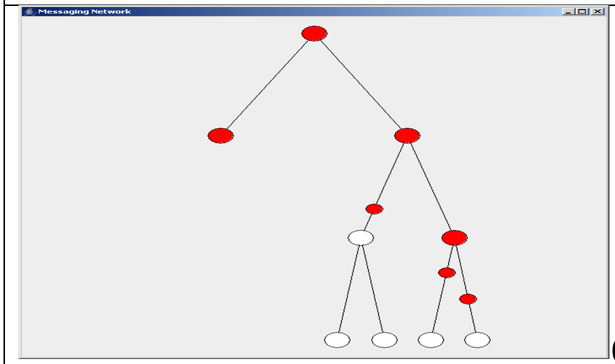
Open Shortest Path First (OSPF) - used for transferring updates to the topology (LSAs)



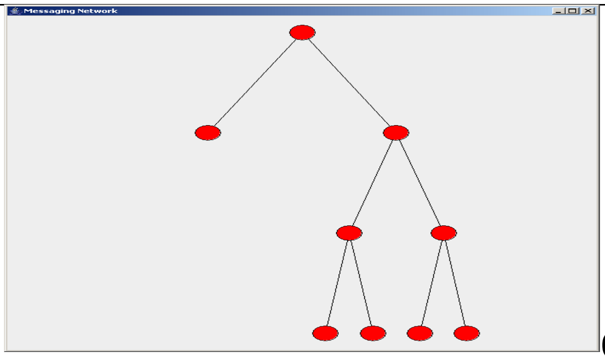
(1)



(2)



(3)



(4)

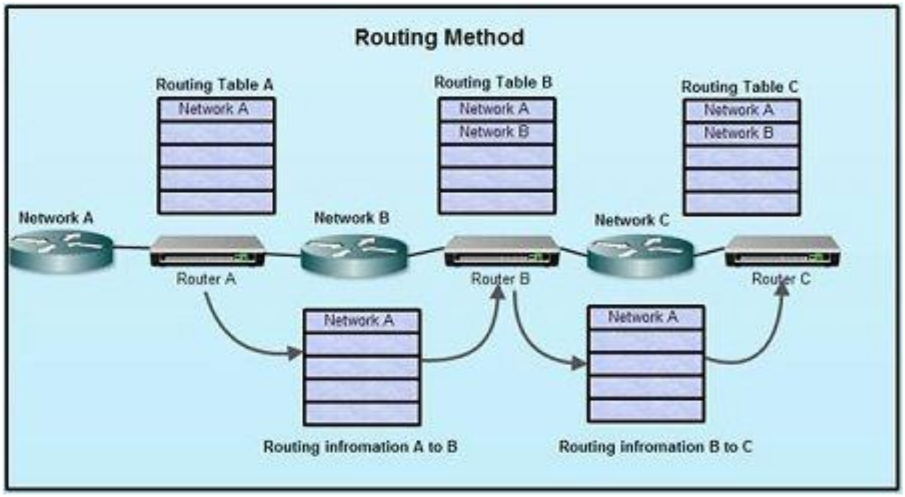
3. Distance Vector routing Protocol

The distance-vector routing Protocol is a type of algorithm used by routing protocols to discover routes on an interconnected network. The primary distance-vector routing protocol algorithm is the Bellman-Ford algorithm. Dynamic routing protocols assist in the automatic creation of routing tables. Network topologies are subject to change at any time. A link may fail unexpectedly, or a new link may be added. A dynamic routing protocol must discover these changes, automatically adjust its routing tables, and inform other routers of the changes. The process of rebuilding the routing tables based on new information is called convergence. Distance-vector routing refers to a method for exchanging route information. A router will advertise a route as a vector of direction and distance.

Direction refers to a port that leads to the next router along the path to the destination, and distance is a metric that indicates the number of hops to the destination, although it may also be an arbitrary value that gives one route precedence over another. Inter network routers exchange this vector information and build route lookup tables from it.

- It is a dynamic routing algorithm in which each router computes distance between itself and each possible destination i.e. its immediate neighbors.
- The router share its knowledge about the whole network to its neighbors and accordingly updates table based on its neighbors.
- The sharing of information with the neighbors takes place at regular intervals.
- It makes use of Bellman Ford Algorithm for making routing tables.
- Problems – Count to infinity problem which can be solved by splitting horizon.
  - Good news spread fast and bad news spread slowly.
  - Persistent looping problem i.e. loop will be there forever.

It is an Algorithm where each router exchanges its routing table with each of its neighbors. Each router will then merge the received routing tables with its own table, and then transmit the merged table to its neighbors. This occurs dynamically after a fixed time interval by default, thus requiring significant link overhead.



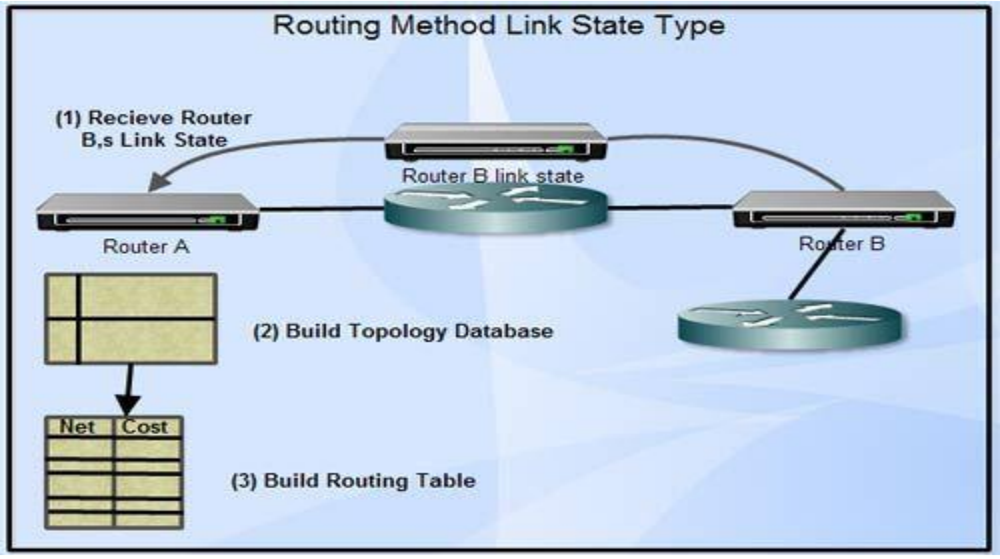
There are problems, however, such as:

a. If exchanging data among routers every 90 seconds for example, it takes 90 x 10 seconds that a router detects a problem in router 10, routers ahead and the route cannot be changed during this period.
b. Traffic increases since routing information is continually exchanged.
c. There is a limit to the maximum amount of routing information (15 for RIP), and routing is not possible on networks where the number of hops exceeds this maximum.
d. Cost data is only the number of hops, and so selecting the best path is difficult.

4. Link-State Protocol

Algorithm where each router in the network learns the network topology then creates a routing table based on this topology. Each router will send information of its links (Link-State) to its neighbor who will In turn propagate the information to its neighbors, etc. This occurs until all routers have built a topology of the network. Each router will then prune the topology, with itself as the root, choosing the least-cost-path to each router, then build a routing table based on the pruned topology.

In link-state protocols, there are no restrictions. in number of hope as in distance-vector protocols, and these are aimed at relatively large networks such as Internet backbones. The load on routers will be large however, since processing is complex.


Routing Method Link State Type
(1) Recieve Router B,s Link State
Router B link state
Router A
Router B
(2) Build Topology Database
Net
Cost
(3) Build Routing Table

- It is a dynamic routing algorithm in which each router shares knowledge of its neighbors with every other router in the network.
- A router sends its information about its neighbors only to all the routers through flooding.
- Information sharing takes place only whenever there is a change.
- It makes use of Dijkastra's Algorithm for making routing tables.
- Problems –
  - Heavy traffic due to flooding of packets.
  - Flooding can result in infinite looping which can be solved by using Time to leave (TTL) field.

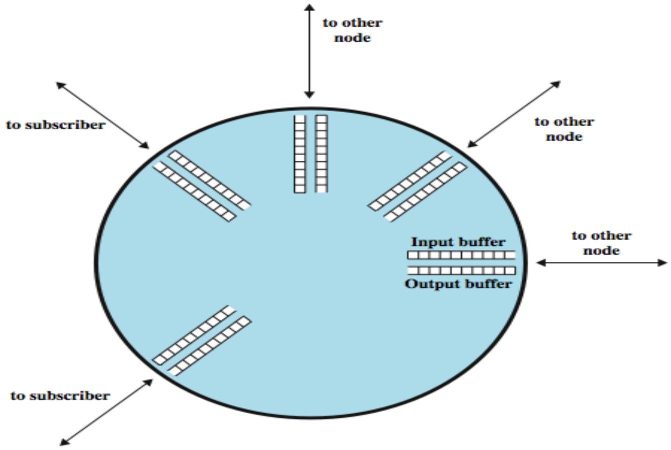
Distance Vector Routing	Link State Routing
Bandwidth required is less due to local sharing, small packets and no flooding.	Bandwidth required is more due to flooding and sending of large link state packets.
Based on local knowledge since it updates table based on information from neighbors.	Based on global knowledge i.e. it have knowledge about entire network.
Make use of Bellman Ford algorithm	Make use of Dijkastra's algorithm
Traffic is less	Traffic is more
Converges slowly i.e. good news spread fast and bad news spread slowly.	Converges faster.
Count to infinity problem.	No count to infinity problem.
Persistent looping problem i.e. loop will there forever.	No persistent loops, only transient loops
Practical implementation is RIP and IGRP.	Practical implementation is OSPF and ISIS.

E) Congestion Control:

Congestion, refers to a network state where a node or link carries so much data that it may deteriorate network service quality, resulting in queuing delay, frame or data packet loss and the blocking of new connections. In a congested network, response time slows with reduced network throughput. Congestion occurs when bandwidth is insufficient and network data traffic exceeds capacity.

(i) If packet arrival rate of packet in node exceeds the packets transmission rate the queue size grows without bound and (ii) Delay in delivery of packets leads to retransmissions. Whenever these two things happen the thumb rule says: When the line for which the packets are queuing becomes more than 80% that means its original capacity may be 100% but whenever it becomes more that 80% utilized the queue length grows alarmingly. That means if the utilization of a link increases more than 80% that means the network has become overloaded and we can say that the network is in congestion. There will be a port and when too many packets arrive at the port the performance degrades in the packet switched network, this is known as congestion.

Queues at a Node



Let's Have Bigger View of Particular node.
It has a storage which includes input queue and output queue which are used as packet buffers where packets are stored before being transmitted.

a. In a normal situation what happens is it goes to node from input buffer and goes from that node to other output buffer.
b. That is normal Situation, but what happens when the traffic increases suddenly.
   We know that the data communication network is bursty in nature. Suddenly increase in load on a short period of time:
   i. One situation is that the buffer may get filled up and when there is no more storage, the packet gets discarded. That means when buffer becomes full there is no empty space to the storage then the packet is discarded.
   ii. Another possibility is that whenever there is big queue at the output nodes suppose going out, what can happen is the particular packet takes very long time to reach the front of the queue, before transmitting it takes long time in the buffer. As it can sequence delay increases significantly and whenever delay increases, what happens is source node after waiting for some time doesn't get an acknowledgement as it can sequence and retransmit the same packet, which also increases the load of the network.

So these two things happen and as a result when packets don't get delivered and delay increases to large extend, then we can say congestion occurs.

Common Causes of Congestion

1. As packets arrive at a node they are stored in an input buffer if packets arrive too fast because of bursty nature and incoming packet may find that there is no available buffer space that is one possibility of a packet getting discarded.
2. Even very large buffer cannot prevent congestion because of delay, timeout and retransmissions:

One can argue that since the buffer space is insufficient why not you increase the buffer space. But increasing the buffer size will increase the size of the queue and the packet which is at the end of the queue will take the long time to reach the front of the queue before it gets transmitted and this will lead to timeout, lead to retransmissions which will increase the traffic in the network and which in other words contribute towards congestion.

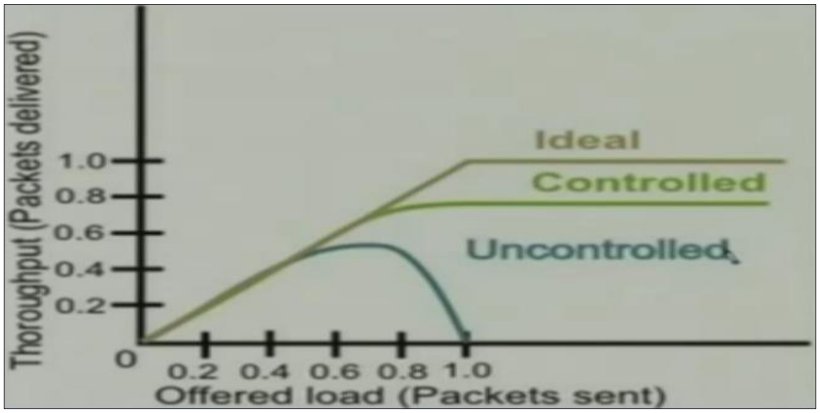
3. **Then the slow processors may also be responsible for congestion.**

Although the link may be of high speed but the slow processors may take very long time to process a packet so it has to do the buffer management, it has to do some housekeeping and for all these things the processors will take sometime. If the processor is slow that may take quite sometime to do this processing. As a result that may delay a packet leading to congestion.

4. **The low bandwidth line may also lead to congestion:**

As we know the network may have links of various data rates or line capacity and as a consequence if the bandwidth of a particular link is small even that can lead to the increase in congestion because the buffer size will increase and the packet may not be delivered.

Effect of Congestion:



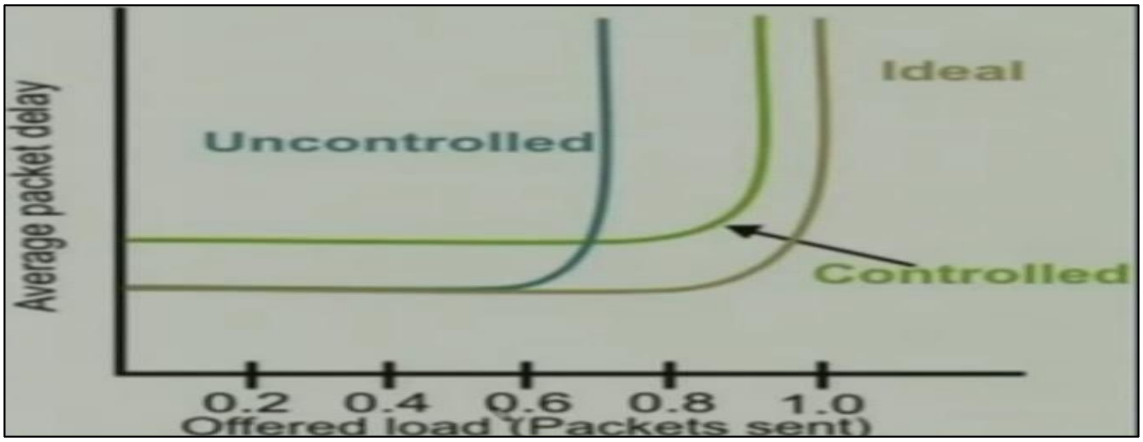
As you can see this greenish curve (bottom) corresponds to uncontrolled that means no controlled measure has been taken for overcoming congestion. In such a situation initially as the offered load or the number of packets introduced in the network increases all the packets gets delivered that means the throughput follows linearly. That means if it is 0.2% of the total capacity then here also it is 0.2% so it rises linearly. But as when it crosses 0.6 to 0.8 mark there is delay and that delay leads to retransmission and as a result the throughput decreases and the rate of increase decreases initially then the throughput suddenly drops although the offered load increases and at certain point it becomes 0 which is known as thrashing situation.

In **thrashing situation** although the stations introduce the packets in the network but not a single one is delivered because of long delay and retransmission and various other problems and that situation is known as thrashing. Thrashing occurs when the throughput becomes 0.

Ideally it should follow the linear curve (upper curve in fig) if there is no congestion but because of congestion it behaves in the uncontrolled manner.

And by taking suitable congestion control approaches the congestion can be controlled. As we see the throughput is less than the ideal curve in controlled curve. That means there is some overhead for implementing congestion control and because of that overhead the throughput is less because there are some overhead packets which are also transmitted resulting in a decrease in throughput but it will not drop like the congested network. As a result although the throughput is less it will never reach the thrashing situation.

Delay due to effect in Congestion



Delay is a very important parameter and as you can see in the ideal situation as long as the offered load is within the capacity of the network the delay is very small which is decided only by the propagation time. Delay is decided by Propagation time plus Transmission Time.

Propagation time and the transmission time of the packet is the only delay and there is no other delay in the network. Now, in the uncontrolled case as you can see it can handle low offered load and delay can be very high compared to the ideal situation.

On the other hand, by using control although the overall delay increases (average packet delay) ultimately it can sustain very high offered load compared to the uncontrolled situation. So we see that how the delay parameter is affected in all the three conditions ideal, uncontrolled case and whenever congestion control measures are taken. This curve shows how delay is affected.

Therefore we have seen two important effects of congestion, one on throughput and another on delay.

Congestion Control and Prevention Algorithms or Techniques:

A) Leaky Bucket Algorithm

It is used to shape bursty rate traffic into fixed rate traffic.

Its working can be compared with a tap with bursty flow of water. If the water is collected in a bucket with a hole at the bottom for a continuous drainage of water.

Its working as follows as illustrated in the figure below:

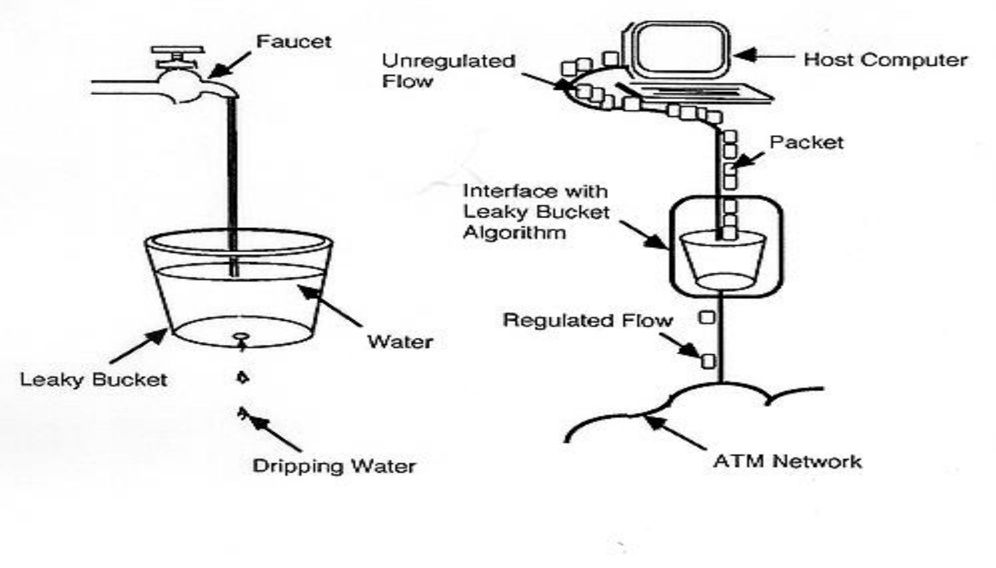


Figure 16 - The Leaky-Bucket Idea

1. A host sent data of bursty nature.
2. This data is kept in a storage buffer by the OS or the NIC.
3. This stored data is then sent out to n/w at a uniform rate.

But it has two major disadvantages

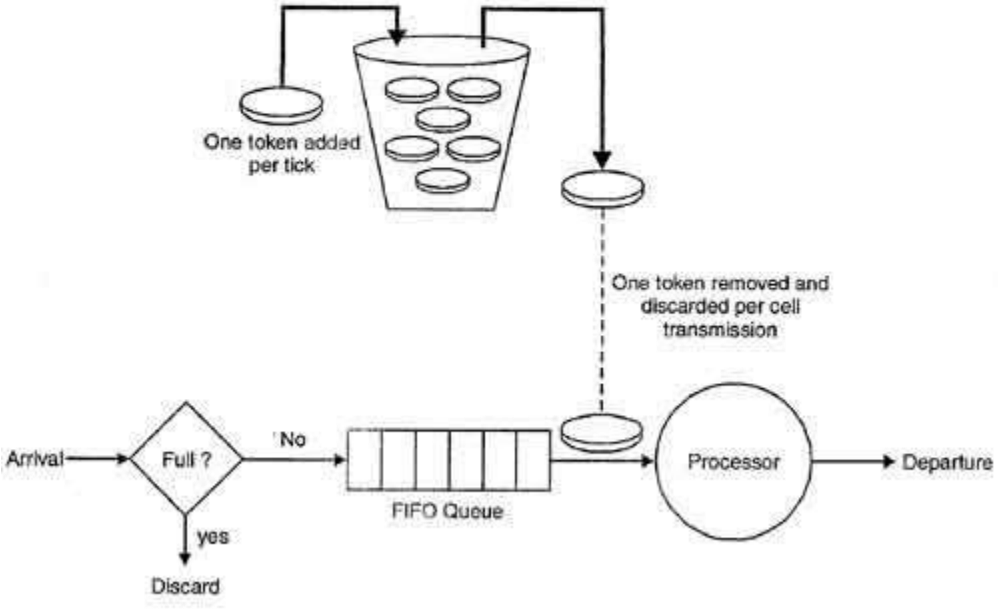
1. Traffic is lost when the bucket is full.
2. The o/p rate is fixed even if there is no congestion.

B) Token Bucket Algorithm

Token bucket was introduced to address the two major drawback of leaky bucket i.e. loss of traffic when the bucket is full and fixed o/p rate even in case of no congestion.

This algorithm works as follow

1. At every tick (or time interval) a token is added in the bucket.
2. Let us say there are 3 token at the bucket at a given time when the host sends a bursty data containing 5 packets as shown in figure 1. below.
3. Since the bucket has 3 token it can send 3 packets immediately as shown in figure 2 below.
4. The rest of the data are sent at subsequent ticks.



Token bucket algorithm

F) Internetworking, Tunneling and Routing, ATM Internetworking, Mobile Routing Schemes

1. Internetworking

Internetworking is combined of 2 words, inter and networking which implies an association between totally different nodes or segments. This connection area unit is established through intercessor devices akin to routers or gateway. The first term for associate degree internetwork was catenet. This interconnection is often among or between public, private, commercial, industrial, or governmental networks. Thus, associate degree internetwork could be an assortment of individual networks, connected by intermediate networking devices, that functions as one giant network. Internetworking refers to the trade, products, and procedures that meet the challenge of making and administering internetworks.

To enable communication, every individual network node or phase is designed with similar protocol or communication logic, that is Transfer Control Protocol (TCP) or Internet Protocol (IP). Once a network communicates with another network having constant communication procedures, it's called Internetworking. Internetworking was designed to resolve the matter of delivering a packet of information through many links.

There a minute difference between extending the network and Internetworking. Merely exploitation of either a switch or a hub to attach 2 local area networks is an extension of LAN whereas connecting them via the router is associate degree example of Internetworking. Internetworking is enforced in Layer three (Network Layer) of OSI-ISO model. The foremost notable example of internetworking is that the Internet.

There are chiefly 3 unit of Internetworking:

a) **Extranet –** It's a network of the internetwork that's restricted in scope to one organization or entity however that additionally has restricted connections to the networks of one or a lot of different sometimes, however not essential. It's very lowest level of Internetworking, usually enforced in an exceedingly personal area. Associate degree extranet may additionally be classified as a Man, WAN, or different form of network however it cannot encompass one local area network i.e. it should have a minimum of one reference to associate degree external network.

b) **Intranet –** This associate degree computer network could be a set of interconnected networks, which exploits the Internet Protocol and uses IP-based tools akin to web browsers and FTP tools, that's underneath the management of one body entity. That body entity closes the computer network to the remainder of the planet and permits solely specific users. Most typically, this network is the internal network of a corporation or different enterprise. An outsized computer network can usually have its own internet server to supply users with browseable data.

c) **Internet –** A selected Internetworking, consisting of a worldwide interconnection of governmental, academic, public, and personal networks based mostly upon the Advanced analysis comes Agency Network (ARPANET) developed by ARPA of the U.S. Department of Defense additionally home to the World Wide Web (WWW) and cited as the 'Internet' to differentiate from all different generic Internetworks. Participants within the web, or their service suppliers, use IP Addresses obtained from address registries that management assignments.

Internetworking has evolved as an answer to a few key problems: isolated LANs, duplication of resources, and an absence of network management. Isolated LANs created transmission problem between totally different offices or departments. Duplication of resources meant that constant hardware and code had to be provided to every workplace or department, as did a separate support employee. This lack of network management meant that no centralized methodology of managing and troubleshooting networks existed.

Internetwork Addressing –

Internetwork addresses establish devices severally or as members of a bunch. Addressing schemes differ based on the protocol family and therefore the OSI layer. Three kinds of internetwork addresses area unit ordinarily used:

a) **Data Link Layer addresses:** A data-link layer address unambiguously identifies every physical network association of a network device. Data-link addresses typically area unit cited as physical or hardware addresses. Data-link addresses sometimes exist among a flat address area and have a pre-established and usually fastened relationship to a selected device. End systems usually have just one physical network association, and therefore have just one data-link address. Routers and different internetworking devices usually have multiple physical network connections and so eventually have multiple data-link addresses.

b) **MAC Addresses:** Media Access management (MAC) addresses encompass a set of data-link layer addresses. MAC addresses establish network entities in LANs that implement the IEEE MAC addresses of the data-link layer. MAC addresses different area unit distinctively for every local area network interface. MAC addresses are forty-eight bits long and are expressed in form of twelve hexadecimal digits. The primary half dozen hexadecimal digits, that are usually administered by the IEEE, establish the manufacturer or merchant and therefore comprise the Organizational Unique Identifier (OUI). The last half dozen positional notation digits comprise the interface serial variety or another price administered by the particular merchant. MAC addresses typically area unit referred to as burned-in addresses (BIAs) as a result of burned into read-only memory(ROM) and are traced into random-access memory (RAM) once the interface card initializes.

c) **Network-Layer Addresses:** Network addresses sometimes exist among a gradable address area and typically area unit referred to as virtual or logical addresses. the connection between a network address and a tool is logical and unfixed, it usually relies either on physical network characteristics or on groupings that don't have any physical basis. finish systems need one network-layer address for every network-layer protocol they support. Routers and different Internetworking devices need one network-layer address per physical network association for every network-layer protocol supported.

Challenges to Internetworking –

Implementing a useful internetwork isn't at any certainty. There are several challenging fields, particularly in the areas of dependableness, connectivity, network management, and adaptability and each and every space is essential in establishing associate degree economical and effective internetwork. Few of them are:-

- The initial challenge lies when we are trying to connect numerous systems to support communication between disparate technologies. For example, Totally different sites might use different kinds of media, or they could operate at variable speeds.
- Another essential thought is reliable service that should be maintained in an internetwork. Individual users and whole organizations depend upon consistent, reliable access to network resources.
- Network management should give centralized support associate degreed troubleshooting capabilities in an internetwork. Configuration, security, performance, and different problems should be adequately addressed for the internetwork to perform swimmingly.
- Flexibility, the ultimate concern, is important for network enlargement and new applications and services, among different factors.

2. Tunneling and Routing

a) Tunneling or Port Forwarding

Tunneling, also known as "port forwarding," is the transmission of data intended for use only within a private, usually corporate network through a public network in such a way that the routing nodes in the public network are unaware that the transmission is part of a private network. Tunneling is generally done by encapsulating the private network data and protocol information within the public network transmission units so that the private network protocol information appears to the public network as data. Tunneling allows the use of the Internet, which is a public network, to convey data on behalf of a private network.

One approach to tunneling is the Point-to-Point Tunneling Protocol (PPTP) developed by Microsoft and several other companies. The PPTP keeps proprietary data reasonably secure, even though part of the path(s) between or among end users exists in public communication channels. The PPTP makes it possible for authorized users to gain access to a private network - called a virtual private network (VPN) -through an Internet service provider (ISP) or online service. Another commonly used tunneling protocol is generic routing encapsulation (GRE), developed by Cisco Systems. There are numerous, less common tunneling protocols.

Tunneling, and the use of a VPN, is not intended as a substitute for encryption/decryption. In cases where a high level of security is necessary, the strongest possible encryption should be used within the VPN itself, and tunneling should serve only as a convenience.

b) Routing

Routing is the process of moving packets across a network from one host to a another. It is usually performed by dedicated devices called routers.

Packets are the fundamental unit of information transport in all modern computer networks, and increasingly in other communications networks as well. They are transmitted over packet switched networks, which are networks on which each message (i.e., data that is transmitted) is cut up into a set of small segments prior to transmission. Each packet is then transmitted individually and can follow the same path or a different path to the common destination. Once all of the packets have arrived at the destination, they are automatically reassembled to recreate the original message.

Routing is a key feature of the Internet and it, together with a great deal of deliberate redundancy of high capacity transmission lines (e.g., optical fiber cable and microwave), is a key factor in the robustness (i.e., resistance to equipment failure) of the Internet. Each intermediary router performs routing by passing along the message to the next router. Part of this process involves analyzing self-configuring routing tables to determine the best (i.e., optimal) path.

Routing is sometimes confused with bridging, which performs a somewhat similar function. The main difference is that the latter occurs at a lower level of the OSI (open systems interconnect) model and is thus more of a hardware function; the former occurs at a higher level where the software component is more important, and thus it can perform more complex analysis to determine the optimal path for each packet.

Routing is also used by circuit switched networks, in which a dedicated circuit is established for the duration of the transmission of each message. The dominant circuit switched network is the public switched telephone network (PSTN), which is the worldwide collection of interconnected public telephone networks that was designed primarily for voice traffic.

c) Asynchronous Transfer Mode (ATM)

A high-speed, broadband transmission data communication technology based on packet switching, which is used by telcos, long distance carriers, and campus-wide backbone networks to carry integrated data, voice, and video information.

ATM is a connection-oriented protocol that can work with either permanent virtual circuits (PVCs) or switched virtual circuits (SVCs), depending on your wide area network (WAN) traffic needs. ATM networks use bandwidth at maximum efficiency, while maintaining guaranteed quality of service (QoS) for users and applications that require it. The two main benefits of ATM are its high transmission speeds and its flexible bandwidth-on-demand capability.

How it works

The «asynchronous» in ATM means ATM devices do not send and receive information at fixed speeds or using a timer, but instead negotiate transmission speeds based on hardware and information flow reliability. The "transfer mode" in ATM refers to the fixed-size cell structure used for packaging information. This cell-based transmission is in contrast to typical local area network (LAN) variable-length packet mechanisms, which means that ATM connections are predictable and can be managed so that no single data type or connection can monopolize the transmission path.

ATM technology originated in broadband ISDN (B-ISDN) technology and works primarily at layer 2 of the Open Systems Interconnection (OSI) reference model. ATM connects devices over a WAN using virtual channels (VCs) and virtual paths (VPs). Virtual channels consist of one or more physical ATM links connected in a series for transmitting data between remote stations. A VC exists only while data is being transmitted on it, and all cells in a given ATM transmission follow the same VC to ensure reliable data transmission. A virtual path is a collection of VCs having the same source and destination points that can be used to pool traffic being transmitted to a given destination.

ATM is a connection-oriented technology that requires the establishment of a specific network path between two points before data can be transported between them.

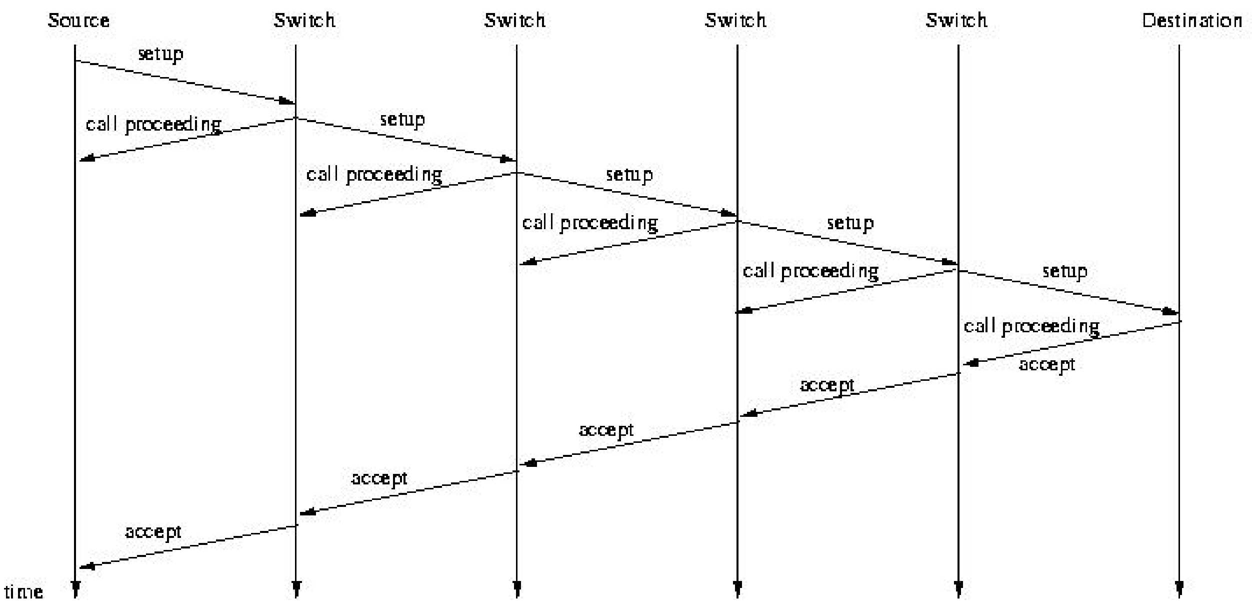
ATM uses fixed-size packets called "cells." Each 53-byte ATM cell contains 48 bytes of data payload and 5 bytes of control and routing information in the header. The header provides addressing information for switching the packet to its destination. The payload section carries the actual information, which can be data, voice, or video. The payload is properly called the user information field. The reason for choosing 48 bytes as the payload size is to compromise between the optimal cell sizes for carrying voice information (32 bytes) and data information (64 bytes). The fixed size of an ATM cell makes ATM traffic simple and predictable, and makes it possible for ATM to operate at high speeds.

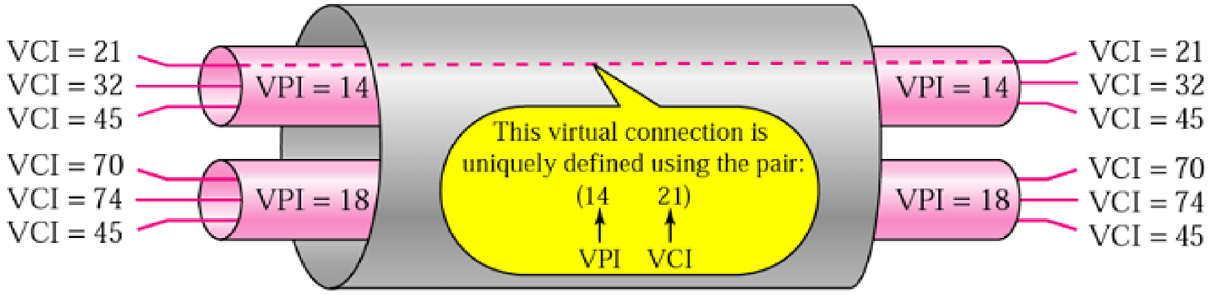
ATM also includes a mechanism for allocating bandwidth dynamically; that is, bandwidth is allocated only in required amounts and the required direction. As a result, when an ATM link is idle, it utilizes no bandwidth, which can result in considerable cost savings depending on the needs of your network.

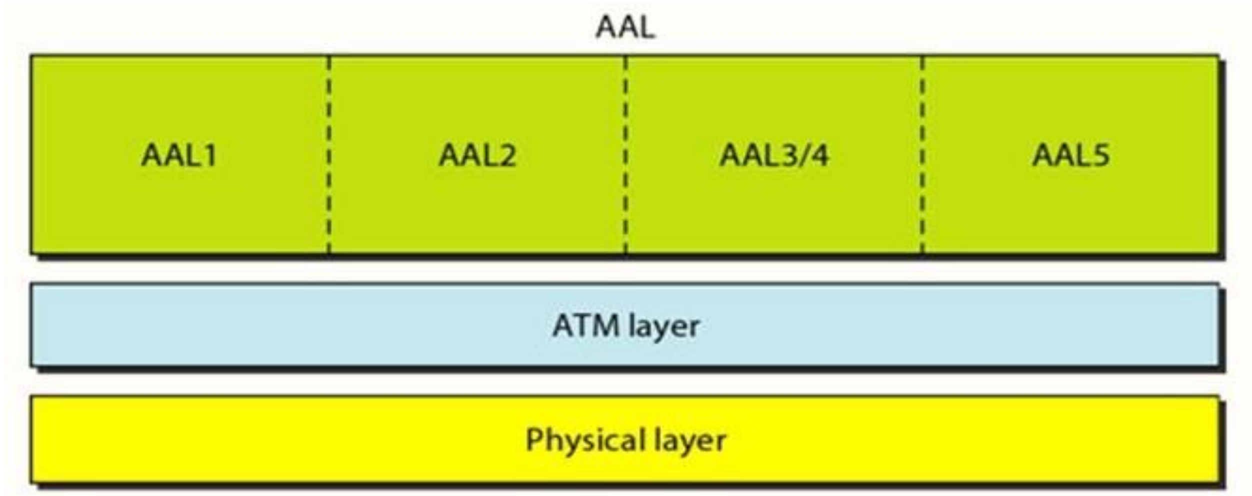
- ATM is a connection-oriented protocol and uses the virtual circuit mechanism for data transfer.
- The main purpose of the ATM cell header is to identify the virtual connection of the cell.
- An ATM virtual connection is specified by the combination of a 12-bit virtual path identifier and a 16-bit virtual channel identifier. Virtual paths are bundles of virtual channels.
- The ATM switch switches these cells using this VPI and VCI from one channel to another.

ATM provides the following advantages:

- High-speed, fast-switched integrated data, voice, and video communication that is not bound by the physical or architectural design constraints of traditional LAN networking technologies.
- A standards-based solution formalized by the International Telecommunication Union (ITU) that allows ATM to easily replace existing telephony network infrastructures. ATM provides a global telephony standard, and over 70 percent of U.S. telcos have migrated their internal networks to ATM.
- Interoperability with standard LAN/WAN technologies. ATM networks can be interconnected with Ethernet and token ring LANs using LAN emulation (LANE) services to provide TCP/IP over ATM.
- QoS technologies that enable a single network connection to reliably carry voice, data, and video simultaneously and manage bandwidth on a per-connection basis depending on the priority of the service required.







G) Network Layer Protocols

1. Internet Protocol

The Internet Protocol (IP) is the principal communications protocol in the Internet protocol suite for relaying datagrams across network boundaries. Its routing function enables internetworking, and essentially establishes the Internet. IP has the task of delivering packets from the source host to the destination host solely based on the IP addresses in the packet headers. For this purpose, IP defines packet structures that encapsulate the data to be delivered. It also defines addressing methods that are used to label the datagram with source and destination information.

The first major version of IP, Internet Protocol Version 4 (IPv4), is the dominant protocol of the Internet. Its successor, Internet Protocol Version 6 IPv6, has been growing in adoption for the last years, reaching almost 20% of the Internet traffic as of April, 2018.

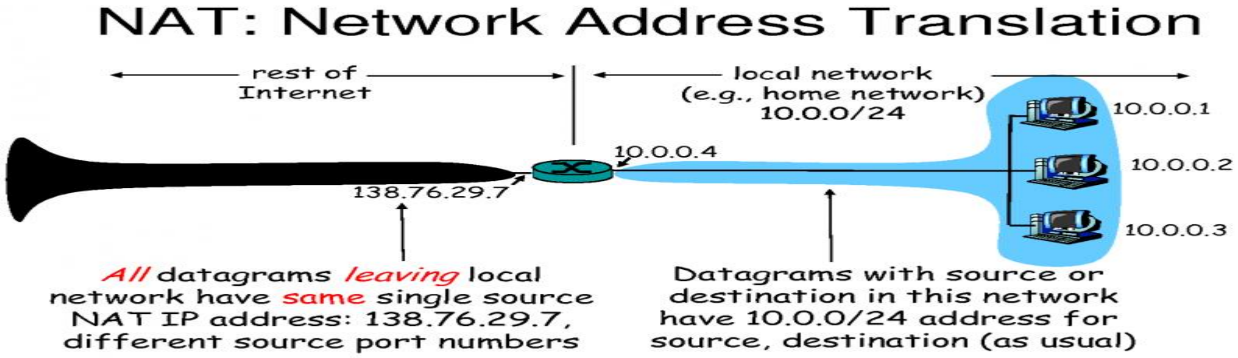
The Internet Protocol is responsible for addressing hosts, encapsulating data into datagrams (including fragmentation and reassembly) and routing datagrams from a source host to a destination host across one or more IP networks. Each datagram has two components: a header and a payload. The IP header includes source IP address, destination IP address, and other metadata needed to route and deliver the datagram. The payload is the data that is transported. This method of nesting the data payload in a packet with a header is called encapsulation.

2. Network Address Translation (NAT)

A Network address translation is a process by which an address can be changed into another address. For example an IPv4 address can be translated to IPv6 address. One of the most widely used implementation of NAT is the conversion of private IP address to public IP address. Since a LAN in an organization uses private address it cannot access the internet; as a system needs to have an public address to access the internet. But a NAT router converts the private address of the LAN computer to a public address thus allowing for the communication with the Internet.

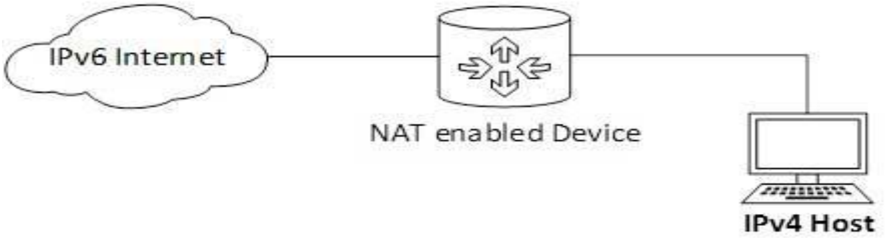
Private IPs are valid just on your network. The computers on your network can see each other, but no one outside your network can see those computers, because they don't have public IPs. Your router on the other hand get a public IP address, and anyone in the world can see your router. So, all of the equipment in your house communicates to outside web sites using that public IP of the router.

Network address translation (NAT) is a method of remapping one IP address space into another by modifying network address information in the IP header of packets while they are in transit across a traffic routing device. The technique was originally used as a shortcut to avoid the need to readdress every host when a network was moved.


NAT: Network Address Translation
rest of Internet
local network (e.g., home network) 10.0.0/24
10.0.0.1
10.0.0.4
10.0.0.2
138.76.29.7
10.0.0.3
All datagrams leaving local network have same single source NAT IP address: 138.76.29.7, different source port numbers
Datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

IP masquerading is a technique that hides an entire IP address space, usually consisting of private IP addresses, behind a single IP address in another, usually public address space. The address that has to be hidden is changed into a single (public) IP address as "new" source address of the outgoing IP packet so it appears as originating not from the hidden host but from the routing device itself. Because of the popularity of this technique to conserve IPv4 address space, the term NAT has become virtually synonymous with IP masquerading.

This is another important method of transition to IPv6 by means of a NAT-PT (Network Address Translation – Protocol Translation) enabled device. With the help of a NAT-PT device, actual can take place happens between IPv4 and IPv6 packets and vice versa.
See the diagram below:



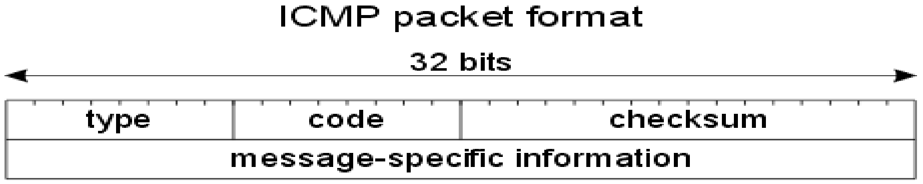
A host with IPv4 address sends a request to an IPv6 enabled server on Internet that does not understand IPv4 address. In this scenario, the NAT-PT device can help them communicate. When the IPv4 host sends a request packet to the IPv6 server, the NAT-PT device/router strips down the IPv4 packet, removes IPv4 header, and adds IPv6 header and passes it through the Internet. When a response from the IPv6 server comes for the IPv4 host, the router does vice versa.

3. Internet Control Message Protocol (ICMP)

Since IP does not have a inbuilt mechanism for sending error and control messages. It depends on Internet Control Message Protocol(ICMP) to provide an error control. It is used for reporting errors and management queries. It is a supporting protocol and used by networks devices like routers for sending the error messages and operations information. To allow routers in an internet to report errors or to provide information about unexpected circumstances, the designers added a special-purpose message mechanism to the TCP/IP- protocols. The mechanism, known as the *Internet Control Message Protocol (ICMP)*, is considered a required part of IP and must be included in every IP implementation. ICMP packet may be used to

a) inform that a datagram has been discarded due to not finding the destination
b) Time-to-live parameter in a datagram expired. **TraceRoute** is a tool which maps network routes by sending packets with small TTL values and watching the ICMP timeout announcements.
c) Requests a host to reduce the rate at which datagrams are sent.
d) Checks the reachability of a specified host or gateway. **Ping**, a common network management tool, is based on this feature. Ping will transmit a series of packets, measuring average round--trip times and computing loss percentages.

Icmp uses codes and type field to specify the types of message it is trying to transmit. For example a **ping request** uses type 8 and code 0 (**echo request**) and **ping reply** uses type 0 code 0 (**echo reply**) . The general format of ICMP packet is shown below


ICMP packet format

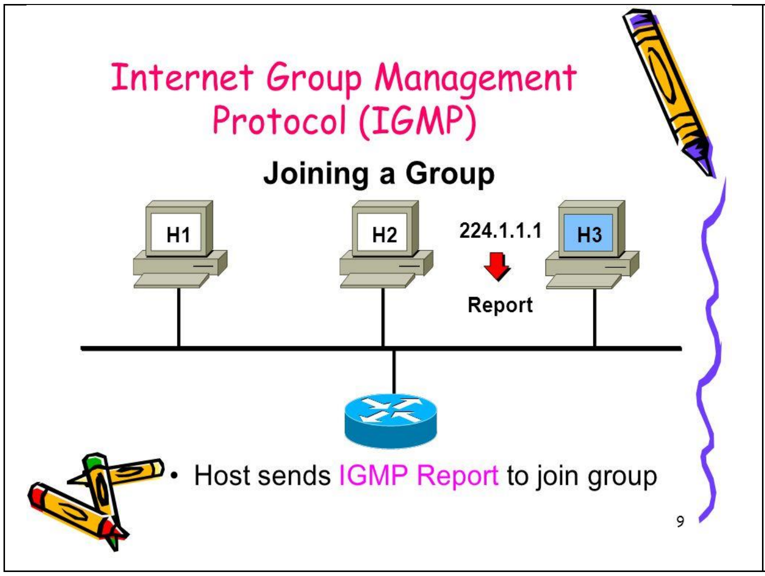
type	code	checksum
message-specific information		

4. Internet Group Management Protocol (IGMP)

- Internet Group Management Protocol is a group management protocol that mainly manages the group membership in a multicast network.
- In a multicast network, multicast routers are used to route packets to all the computers that are having membership of a particular group.
- The multicast routers use the information from IGMP to determine which hosts are having membership of which group.
- A multicast router generally receives thousands of multicast packets that have to be transmitted to various groups. If a router has no knowledge about the group membership, it will broadcast packet to every host and this will increase the load on the network.
- In order to save the network from such a problem, a list of groups IS maintained when members of the group are present in the network.
- Thus, IGMP helps the multicast router to create and update this list.
- This protocol uses three different messages: query message, membership report and leave report.

Working of IGMP

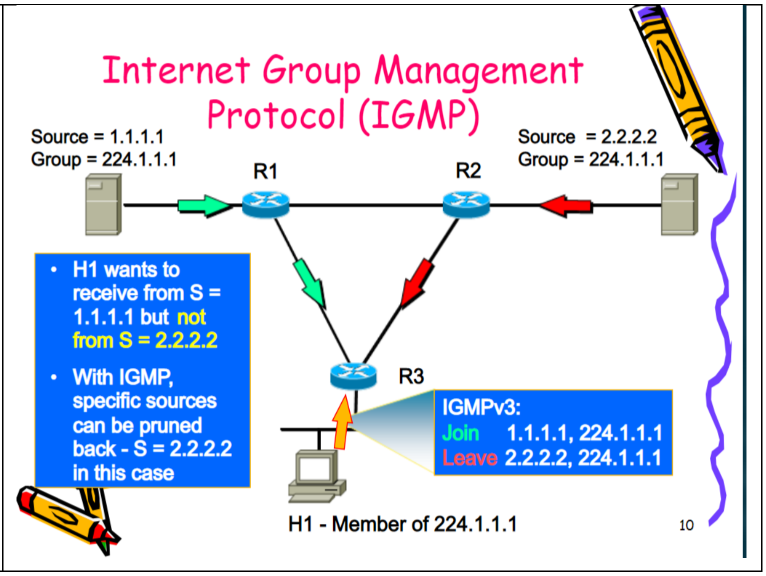
- The multicast router of the network has a list of multicast address for which the network is having any member.
- There is one multicast router for each group that distributes multicast packet to members of that group. It means the network will have two multicast routers, if there are two multicast groups.
- A host or a multicast router can be a member of the group.
- When a host is having membership, it means that any process running on that host is a member of the group and when a router is having membership of group, it means one of the networks connected to the router is having membership of the group.



Internet Group Management Protocol (IGMP)

Joining a Group

- Host sends IGMP Report to join group



Internet Group Management Protocol (IGMP)

- H1 wants to receive from S = 1.1.1.1 but not from S = 2.2.2.2
- With IGMP, specific sources can be pruned back - S = 2.2.2.2 in this case

IGMPv3:
Join 1.1.1.1, 224.1.1.1
Leave 2.2.2.2, 224.1.1.1

Joining a Group

- Both the host and a router can join a group.
- When a process on the host wants to join a group it sends the request to the host.
- The host adds the name of the process and group name to its list.
- If this is the first entry for that particular group, the host sends *membership report* message to the multicast router of the group.
- If it is not the first entry for the requested group there is no need of sending such a message.

Leaving a Group

- Whenever a host sees no process interested in a group, it sends a *leave report* message.
- The membership is not purged by the multicast router of the group, rather it immediately transmits query packets repeatedly to see if anyone still interested.
- If the response comes in the form of membership report message, the membership of the host or network is preserved.

5. Routing Information Protocol (RIP)

Routing Information Protocol (RIP) is a dynamic routing protocol which uses hop count as a routing metric to find the best path between the source and the destination network. It is a distance vector routing protocol which has AD value 120 and works on the application layer of OSI model. RIP uses port number 520.

- RIP request is used, by a router upon startup to inquire of its neighbor router about route information to obtain routing information.
- RIP response includes a destination host address and cost information in the address part. Response is sent to the neighbor router in case of the following:
  a) Receipt of RIP request
  b) Regularly
     Response is sent every 30 seconds even if no RIP request is issued. All routers delete route information from their routing table if no route information is received within a specified period of time. This is intended to allow detection of fault of neighbor router.
  c) In case of changes made to routing table contents
     If changes are made to the routing table because changes to the network configuration have been detected, information relating to these changes is sent to the neighbor router.

Hop Count :

Hop count is the number of routers occurring in between the source and destination network. The path with the lowest hop count is considered as the best route to reach a network and therefore placed in the routing table. RIP prevents routing loops by limiting the number of hopes allowed in a path from source and destination. The maximum hop count allowed for RIP is 15 and hop count of 16 is considered as network unreachable.

Features of RIP :

- Updates of the network are exchanged periodically.
- Updates (routing information) are always broadcast.
- Full routing tables are sent in updates.
- Routers always trust on routing information received from neighbor routers. This is also known as *Routing on rumours*.

RIP versions :

RIP V1	**RIP V2**	**RIPNG**
Sends update as broadcast	Sends update as multicast	Sends update as multicast
Broadcast at 255.255.255.255	Multicast at 224.0.0.9	Multicast at FF02::9 (RIPng can only run on IPv6 networks)
Doesn’t support authentication of update messages	Supports authentication of RIPv2 update messages	–
Classful routing protocol	Classless protocol, supports classful	Classless updates are sent

RIP v1 is known as *Classful* Routing Protocol because it doesn’t send information of mask in its routing update.
RIP v2 is known as *Classless* Routing Protocol because it doesn’t send information of mask in its routing update.

RIP timers :

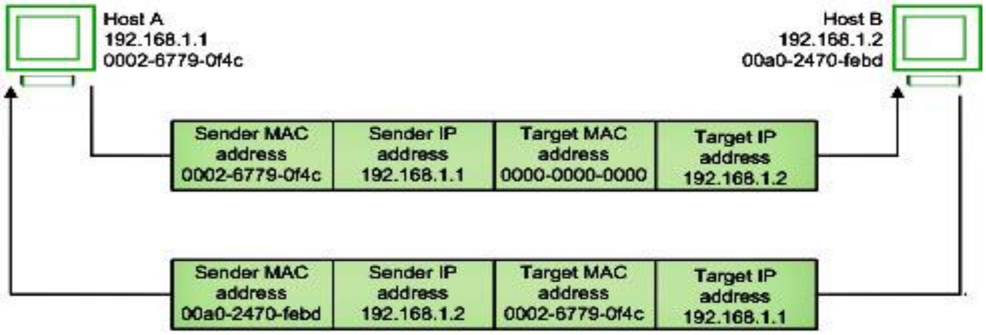
- **Update timer :** The default timing for routing information being exchanged by the routers operating RIP is 30 seconds. Using Update timer, the routers exchange their routing table periodically.
- **Invalid timer:** If no update comes until 180 seconds, then the destination router consider it as invalid. In this scenario, the destination router mark hop count as 16 for that router.
- **Hold down timer :** This is the time for which the router waits for neighbour router to respond. If the router isn’t able to respond within a given time then it is declared dead. It is 180 seconds by default.
- **Flush time :** It is the time after which the entry of the route will be flushed if it doesn’t respond within the flush time. It is 60 seconds by default. This timer starts after the route has been declared invalid and after 60 seconds i.e time will be 180 + 60 = 240 seconds.

6. Address Resolution Protocol (ARP)

IP addresses are assigned independently of the hardware addresses of the machines. To send a datagram on the Internet, the network software must convert the IP address into a physical address, used to transmit the frame.

It's ARP (Address Resolution Protocol) performing this translation between the IP world and Ethernet based on the physical network. ARP enables machines to resolve addresses without using static table that lists all addresses of both worlds. A machine uses ARP to determine the recipient's physical address by broadcasting an ARP request to the subnet containing the IP address to be translated. The machine with the relevant IP address responds with its physical address. To make ARP more efficient, each machine maintains in memory a table of addresses resolved and thus reduces the number of Broadcast emissions.

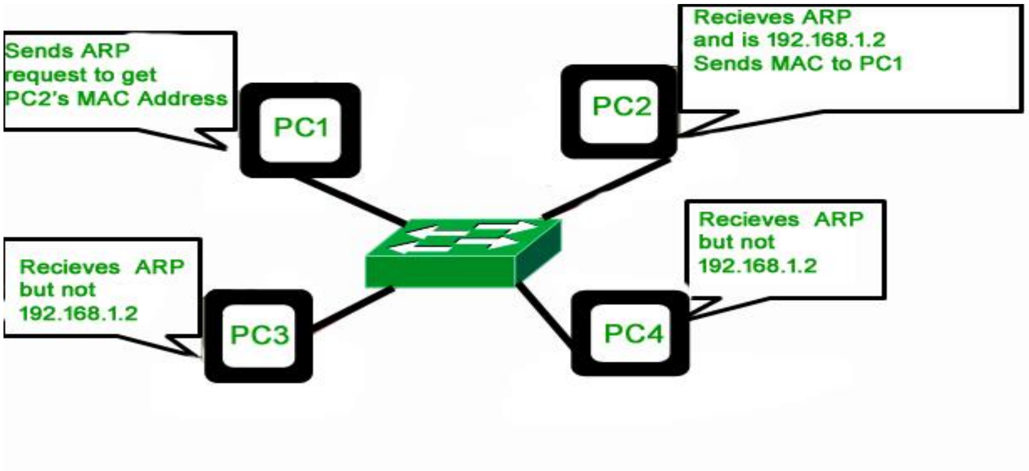
Typically, ARP is a network layer to data link layer mapping process, which is used to discover MAC address for given Internet Protocol Address. In order to send the data to destination, having IP address is necessary but not sufficient; we also need the physical address of the destination machine. ARP is used to get the physical address (MAC address) of destination machine.



Sender MAC address	Sender IP address	Target MAC address	Target IP address
0002-6779-0f4c	192.168.1.1	0000-0000-0000	192.168.1.2

Sender MAC address	Sender IP address	Target MAC address	Target IP address
00a0-2470-febd	192.168.1.2	0002-6779-0f4c	192.168.1.1

Before sending the IP packet, the MAC address of destination must be known. If not so, then sender broadcasts the ARP-discovery packet requesting the MAC address of intended destination. Since ARP-discovery is broadcast, every host inside that network will get this message but the packet will be discarded by everyone except that intended receiver host whose IP is associated. Now, this receiver will send a unicast packet with its MAC address (ARP-reply) to the sender of ARP-discovery packet. After the original sender receives the ARP-reply, it updates ARP-cache and start sending unicast message to the destination.

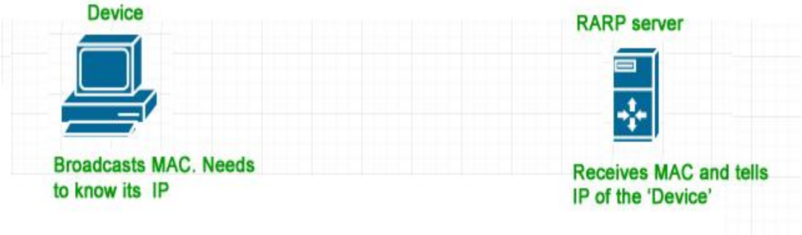

Sends ARP request to get PC2's MAC Address
PC1
PC2
Recieves ARP and is 192.168.1.2 Sends MAC to PC1
Recieves ARP but not 192.168.1.2
PC3
PC4
Recieves ARP but not 192.168.1.2

The important terms associated with ARP are :

a) **ARP Cache:** After resolving MAC address, the ARP sends it to the source where it stores in a table for future reference. The subsequent communications can use the MAC address from the table
b) **ARP Cache Timeout:** It indicates the time for which the MAC address in the ARP cache can reside
c) **ARP request:** This is nothing but broadcasting a packet over the network to validate whether we came across destination MAC address or not
d) **ARP response:** It is the MAC address response that the source receives from the destination which aids in further communication of the data

7. Reverse Address Resolution Protocol (RARP)

Reverse ARP is a networking protocol used by a client machine in a local area network to request its Internet Protocol address (IPv4) from the gateway-router's ARP table. The network administrator creates a table in gateway-router, which is used to map the MAC address to corresponding IP address.
When a new machine is setup or any machine which don't have memory to store IP address, needs an IP address for its own use. So the machine sends a RARP broadcast packet which contains its own MAC address in both sender and receiver hardware address field.


Device
Broadcasts MAC. Needs to know its IP
RARP server
Receives MAC and tells IP of the 'Device'

A special host configured inside the local area network, called as RARP-server is responsible to reply for these kind of broadcast packets. Now the RARP server attempt to find out the entry in IP to MAC address mapping table. If any entry matches in table, RARP server send the response packet to the requesting device along with IP address.

- LAN technologies like Ethernet, Ethernet II, Token Ring and Fiber Distributed Data Interface (FDDI) support the Address Resolution Protocol.
- RARP is not being used in today's networks. Because we have much great featured protocols like BOOTP (Bootstrap Protocol) and DHCP(Dynamic Host Configuration Protocol).

8. Open Shortest Path First (OSPF)

Open Shortest Path First (OSPF) is a link-state routing protocol that was developed for IP networks and is based on the Shortest Path First (SPF) algorithm. OSPF is an Interior Gateway Protocol (IGP).

In an OSPF network, routers or systems within the same area maintain an identical link-state database that describes the topology of the area. Each router or system in the area generates its link-state database from the link-state advertisements (LSAs) that it receives from all the other routers or systems in the same area and the LSAs that itself generates. An LSA is a packet that contains information about neighbors and path costs. Based on the link-state database, each router or system calculates a shortest-path spanning tree, with itself as the root, using the SPF algorithm.

RIP successfully implemented dynamic routing, where routing tables change if the network topology changes. But RIP did not adapt its routing according to changing network conditions, such as data-transfer rate. OSPF was developed so that the shortest path through a network was calculated based on the cost of the route, taking into account bandwidth, delay and load. Therefore OSPF undertakes route cost calculation on the basis of link-cost parameters.

As a link state routing protocol, OSPF maintains link state databases, which are really network topology maps, on every router on which it is implemented. The state of a given route in the network is the cost, and OSPF algorithm allows every router to calculate the cost of the routes to any given reachable destination. A router interface with OSPF will then advertise its link cost to neighbouring routers through multicast, known as the *hello procedure*. All routers with OSPF implementation keep sending *hello packets*, and thus changes in the cost of their links become known to neighbouring routers. The information about the cost of a link, that is the speed of a point to point connection between two routers, is then cascaded through the network because OSPF routers advertise the information they receive from one neighbouring router to all other neighbouring routers. This process of flooding link state information through the network is known as synchronisation. Based on this information, all routers with OSPF implementation continuously update their link state databases with information about the network topology and adjust their routing tables.

OSPF detects changes in the topology, such as link failures, and converges on a new loop-free routing structure within seconds.

OSPF has the following key advantages:

- Compared with distance-vector routing protocols such as the Routing Information Protocol (RIP), OSPF is more suitable for serving large, heterogeneous internetworks. OSPF can recalculate the routes in a short amount of time when the network topology changes.
- With OSPF, you can divide an Autonomous System (AS) into areas and keep area topologies separate to decrease the OSPF routing traffic and the size of the link-state database of each area.
- OSPF provides equal-cost multipath routing. You can add duplicate routes to the TCP stack using different next hops.

OSPF’s big idea

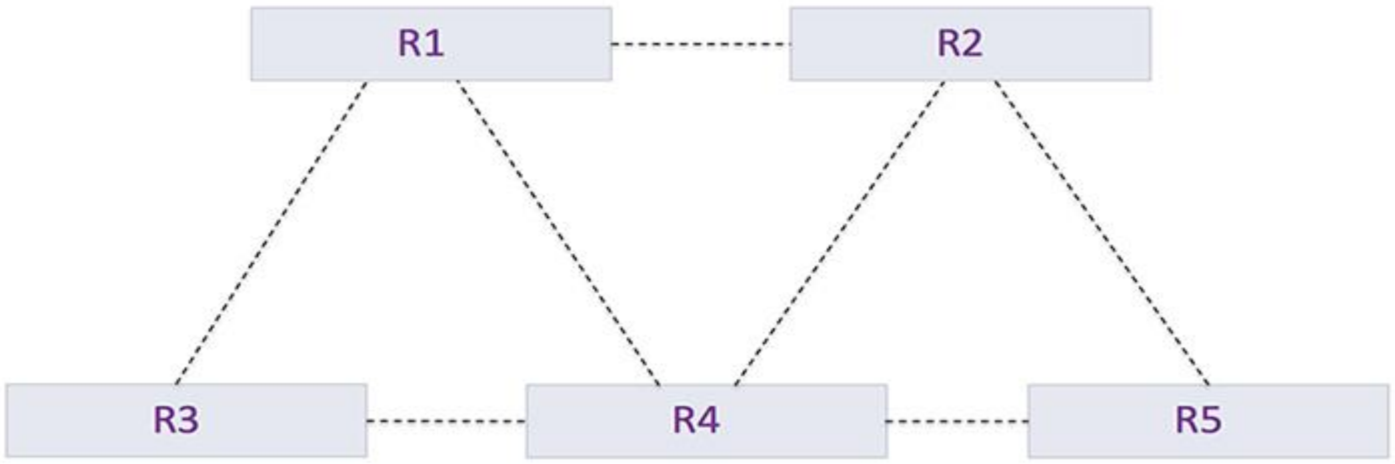
OSPF is a routing protocol. Two routers speaking OSPF to each other exchange information about the routes they know about and the cost for them to get there.

When many OSPF routers are part of the same network, information about all of the routes in a network are learned by all of the OSPF routers within that network—technically called an **area**.

Each OSPF router passes along information about the routes and costs they've heard about to all of their adjacent OSPF routers, called **neighbors**.

OSPF routers rely on **cost** to compute the shortest path through the network between themselves and a remote router or network destination. The shortest path computation is done using Djikstra's algorithm. This algorithm isn't unique to OSPF. Rather, it's a mathematical algorithm that happens to have an obvious application to networking.

Consider a simple example of five routers connected as shown in the diagram below. Assuming all links have the same cost, what's the fastest way for R3 to connect to R5? Through R4 — R4 is the lowest cost path. (R3's path to R5 via R1, for example, adds another link and therefore additional cost.)



OSPF interfaces

Another important idea in OSPF is that interfaces used to exchange information with OSPF neighbors have different types. There are too many types to discuss here but you should be aware of two important ones.

a) An OSPF **broadcast** interface is connected to a shared network, like Ethernet.

b) An OSPF **point-to-point** interface is connected to a link where there can only be a single OSPF router on either end, such as a WAN link or a purpose-built Ethernet link.

The reason for the various interface types is to make sure that all routers know about all routes from all other routers.

On point-to-point links, there's no mystery — the two routers know they're the only OSPF routers on the link and so they exchange routes with each other.

On broadcast links, there's a potential for many different OSPF routers to be on the network segment. To minimize the number of neighbor relationships that form on broadcast links, OSPF elects a **designated router**(as well as a backup) whose job it is to neighbor with all other OSPF routers on the segment and share everyone's routes with everyone else.

9. IGRP (Interior Gateway Routing Protocol)

Improved version of RIP, IGRP was designed by Cisco Systems for its own routers. It integrates multipath routing, management of default routes, dissemination of information every 90 seconds instead of every 30 seconds, detection of closures, that is to say, returns to a point whereby the packet has already passed, etc. The protocol itself has been extended for better protection against the loops by EIGRP (Extended IGRP).

10. Enhanced Interior Gateway Routing Protocol (EIGRP)

Enhanced Interior Gateway Routing Protocol (EIGRP) is an advanced distance-vector routing protocol that is used on a computer network for automating routing decisions and configuration. The protocol was designed by Cisco Systems as a proprietary protocol, available only on Cisco routers.

EIGRP is used on a router to share routes with other routers within the same autonomous system. Unlike other well-known routing protocols, such as RIP, EIGRP only sends incremental updates, reducing the workload on the router and the amount of data that needs to be transmitted.

EIGRP replaced the Interior Gateway Routing Protocol (IGRP) in 1993. One of the major reasons for this was the change to classless IPv4 addresses in the Internet Protocol, which IGRP could not support.

In addition to the routing table, EIGRP uses the following tables to store information:

a) **Neighbor Table:** The neighbor table keeps a record of the IP addresses of routers that have a direct physical connection with this router. Routers that are connected to this router indirectly, through another router, are not recorded in this table as they are not considered neighbors.

b) **Topology Table:** The topology table stores routes that it has learned from neighbor routing tables. Unlike a routing table, the topology table does not store all routes, but only routes that have been determined by EIGRP. The topology table also records the metrics for each of the listed EIGRP routes, the feasible successor and the successors. Routes in the topology table are marked as "passive" or "active". Passive indicates that EIGRP has determined the path for the specific route and has finished processing. Active indicates that EIGRP is still trying to calculate the best path for the specific route

Information in the topology table may be inserted into the router's routing table and can then be used to forward traffic. If the network changes (for example, a physical link fails or is disconnected), the path will become unavailable. EIGRP is designed to detect these changes and will attempt to find a new path to the destination. The old path that is no longer available is removed from the routing table.

When a router running EIGRP is connected to another router also running EIGRP, information is exchanged between the two routers. They form a relationship, known as an adjacency. The entire routing table is exchanged between both routers at this time. After the exchange has completed, only differential changes are sent. EIGRP does not send its routing table periodically, but will only send routing table data when an actual change has occurred. This behavior is more inline with link-state routing protocols, thus EIGRP is mostly considered a hybrid protocol.

Features

- Support for Classless Inter-Domain Routing (CIDR) and variable length subnet masking. Routes are not summarized at the classful network boundary unless auto summary is enabled.
- Support for load balancing on parallel links between sites.
- MD5 and SHA-2 authentication between two routers.
- Sends topology changes, rather than sending the entire routing table when a route is changed.
- Periodically checks if a route is available, and propagates routing changes to neighboring routers if any changes have occurred.
- Backwards compatibility with the IGRP routing protocols.

11. Border Gateway Protocol (BGP)

Border Gateway Protocol (BGP) is a standardized exterior gateway protocol designed to exchange routing and reachability information among autonomous systems (AS) on the Internet. The protocol is classified as a path vector protocol. The Border Gateway Protocol makes routing decisions based on paths, network policies, or rule-sets configured by a network administrator and is involved in making core routing decisions.

Operation

BGP neighbors, called peers, are established by manual configuration between routers to create a TCP session on port 179. A BGP speaker sends 19-byte keep-alive messages every 60 seconds to maintain the connection. Among routing protocols, BGP is unique in using TCP as its transport protocol.

When BGP runs between two peers in the same autonomous system (AS), it is referred to as Internal BGP (iBGP or Interior Border Gateway Protocol). When it runs between different autonomous systems, it is called External BGP (eBGP or Exterior Border Gateway Protocol).

How routes are propagated can be controlled in detail via the route-maps mechanism. This mechanism consists of a set of rules. Each rule describes, for routes matching some given criteria, what action should be taken. The action could be to drop the route, or it could be to modify some attributes of the route before inserting it in the routing table.

Autonomous systems

In the world of BGP, each routing domain is known as an autonomous system, or AS. What BGP does is help choose a path through the Internet, usually by selecting a route that traverses the least number of autonomous systems: the shortest AS path.

You might need BGP, for example, if your corporate network is connected to two large ISPs. To use BGP you would need an AS number, which you can get from the American Registry of Internet Numbers (ARIN).

Once BGP is enabled, your router will pull a list of Internet routes from your BGP neighbors, who in this case will be your two ISPS. It will then scrutinize them to find the routes with the shortest AS paths. These will be put into the router's routing table. (If you only connect to a single ISP then you don't need BGP. That's because there's only one path to the Internet, so there's no need for a routing protocol to select the best path.)

Generally, but not always, routers will choose the shortest path to an AS. BGP only knows about these paths based on updates it receives.

Route updates

Unlike Routing Information Protocol (RIP), a distance-vector routing protocol which employs the hop count as a routing metric, BGP does not broadcast its entire routing table. At boot, your peer will hand over its entire table. After that, everything relies on updates received.

Route updates are stored in a Routing Information Base (RIB). A routing table will only store one route per destination, but the RIB usually contains multiple paths to a destination. It is up to the router to decide which routes will make it into the routing table, and therefore which paths will actually be used. In the event that a route is withdrawn, another route to the same place can be taken from the RIB.

The RIB is only used to keep track of routes that could possibly be used. If a route withdrawal is received and it only existed in the RIB, it is silently deleted from the RIB. No update is sent to peers. RIB entries never time out. They continue to exist until it is assumed that the route is no longer valid.

BGP path attributes

In many cases, there will be multiple routes to the same destination. BGP therefore uses path attributes to decide how to route traffic to specific networks.

The easiest of these to understand is Shortest AS_Path. What this means is the path which traverses the least number of AS "wins."

Another important attribute is Multi_Exit_Disc (Multi-exit discriminator, or MED). This makes it possible to tell a remote AS that if there are multiple exit points on to your network, a specific exit point is preferred.

The Origin attribute specifies the origin of a routing update. If BGP has multiple routes, then origin is one of the factors in determining the preferred route.

BGP issues

To get a true sense of how BGP works, it's important to spend some time talking about the issues that plague the Internet.

First, we have a very big problem with routing table growth. If someone decides to deaggregate a network that used to be a single /16 network, they could potentially start advertising hundreds of new routes. Every router on the Internet will get every new route when this happens. People are constantly pressured to aggregate, or combine multiple routes into a single advertisement. Aggregation isn't always possible, especially if you want to break up a /19 into two geographically separate /20s. Routing tables are approaching 200,000 routes now, and for a time they were appearing to grow exponentially.

Second, there is always a concern that someone will "advertise the Internet." If some large ISP's customer suddenly decides to advertise everything, and the ISP accepts the routes, all of the Internet's traffic will be sent to the small customer's AS. There's a simple solution to this. It's called route filtering. It's quite simple to set up filters so that your routers won't accept routes from customers that you aren't expecting, but many large ISPs will still accept the equivalent of "default" from peers that have no likelihood of being able to provide transit.

Finally, we come to flapping. BGP has a mechanism to "hold down" routes that appear to be flaky. Routes that flap, or come and go, usually aren't reliable enough to send traffic to. If routes flap frequently, the load on all Internet routes will increase due to the processing of updates every time someone disappears and reappears. Dampening will prevent BGP peers from listening to all routing updates from flapping peers. The amount of time one is in hold-down increases exponentially with every flap. It's annoying when you have a faulty link, since it can be more than an hour before you can get to many Internet sites, but it is very necessary.

This quick discussion of BGP should be enough to get you thinking the right way about the protocol but is by no means comprehensive. Spend some time reading the RFCs if you're tasked with operating a BGP router. Your peers will appreciate it.