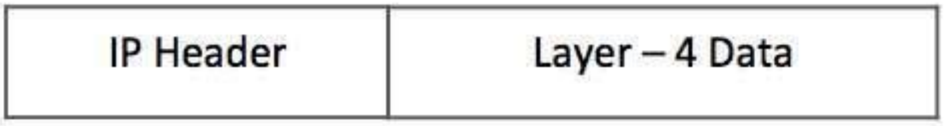
Internet Protocol Version 4 (IPv4)

Internet Protocol is one of the major protocols in the TCP/IP protocols suite. This protocol works at the network layer of the OSI model and at the Internet layer of the TCP/IP model. Thus this protocol has the responsibility of identifying hosts based upon their logical addresses and to route data among them over the underlying network.

IP provides a mechanism to uniquely identify hosts by an IP addressing scheme. IP uses best effort delivery, i.e. it does not guarantee that packets would be delivered to the destined host, but it will do its best to reach the destination. Internet Protocol version 4 uses 32-bit logical address.

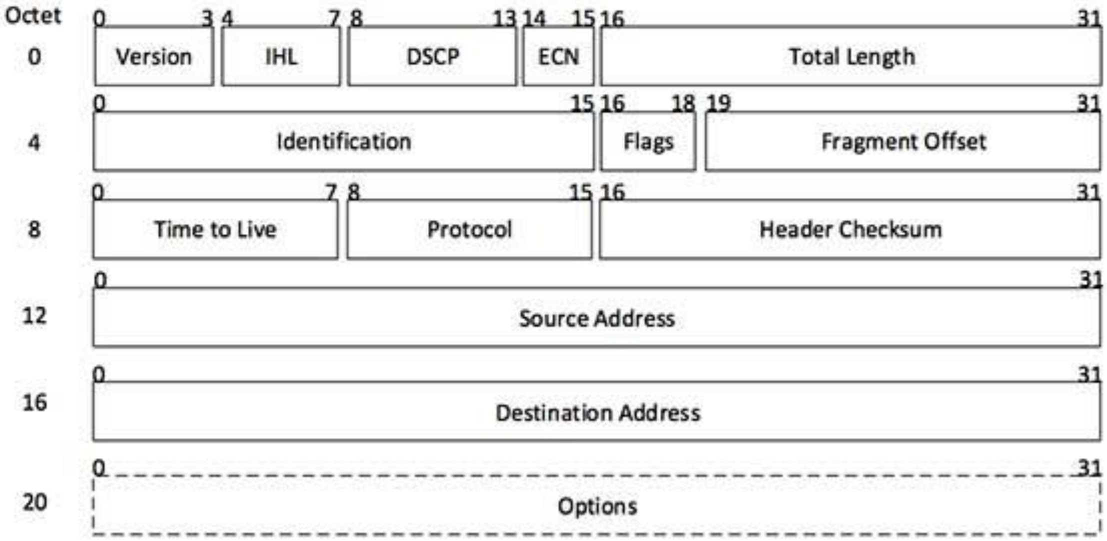
IPv4 - Packet Structure

Internet Protocol being a layer-3 protocol (OSI) takes data Segments from layer-4 (Transport) and divides it into packets. IP packet encapsulates data unit received from above layer and add to its own header information.



(IP Encapsulation)

The encapsulated data is referred to as IP Payload. IP header contains all the necessary information to deliver the packet at the other end.



[Image: IP Header]

IP header includes many relevant information including Version Number, which, in this context, is 4. Other details are as follows −

Version − Version no. of Internet Protocol used (e.g. IPv4).

IHL − Internet Header Length; Length of entire IP header.

DSCP − Differentiated Services Code Point; this is Type of Service.

ECN − Explicit Congestion Notification; It carries information about the congestion seen in the route.

Total Length − Length of entire IP Packet (including IP header and IP Payload).

Identification − If IP packet is fragmented during the transmission, all the fragments contain same identification number. to identify original IP packet they belong to.

Flags − As required by the network resources, if IP Packet is too large to handle, these flags tells if they can be fragmented or not. In this 3-bit flag, the MSB is always set to 0.

Fragment Offset − This offset tells the exact position of the fragment in the original IP Packet.

Time to Live − To avoid looping in the network, every packet is sent with some TTL value set, which tells the network how many routers (hops) this packet can cross. At each hop, its value is decremented by one and when the value reaches zero, the packet is discarded.

Protocol − Tells the Network layer at the destination host, to which Protocol this packet belongs to, i.e. the next level Protocol. For example protocol number of ICMP is 1, TCP is 6 and UDP is 17.

Header Checksum − This field is used to keep checksum value of entire header which is then used to check if the packet is received error-free.

Source Address − 32-bit address of the Sender (or source) of the packet.

Destination Address − 32-bit address of the Receiver (or destination) of the packet.

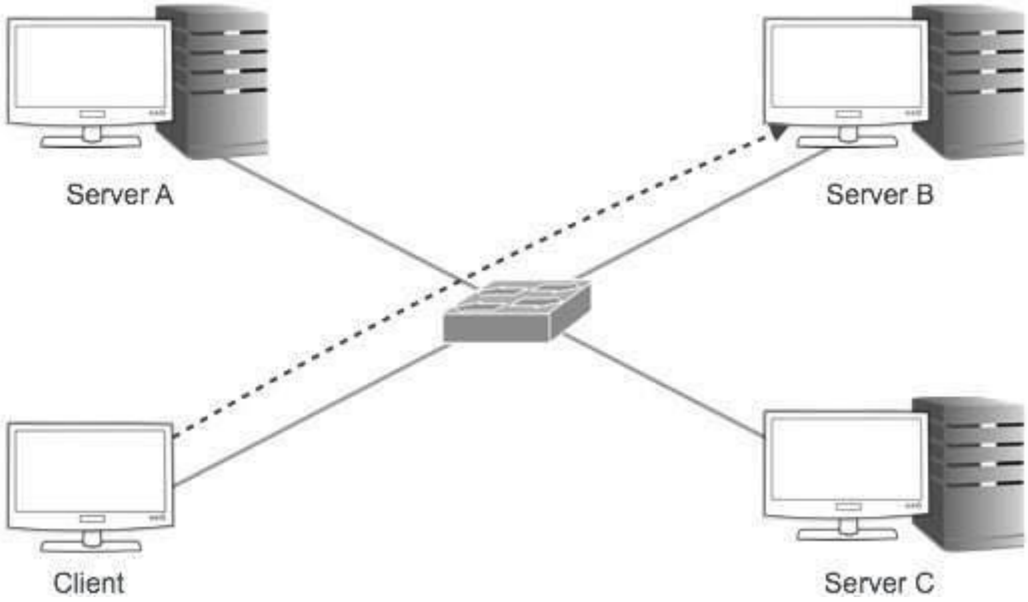
Options − This is optional field, which is used if the value of IHL is greater than 5. These options may contain values for options such as Security, Record Route, Time Stamp, etc.

IPv4 - Addressing

IPv4 supports three different types of addressing modes. −

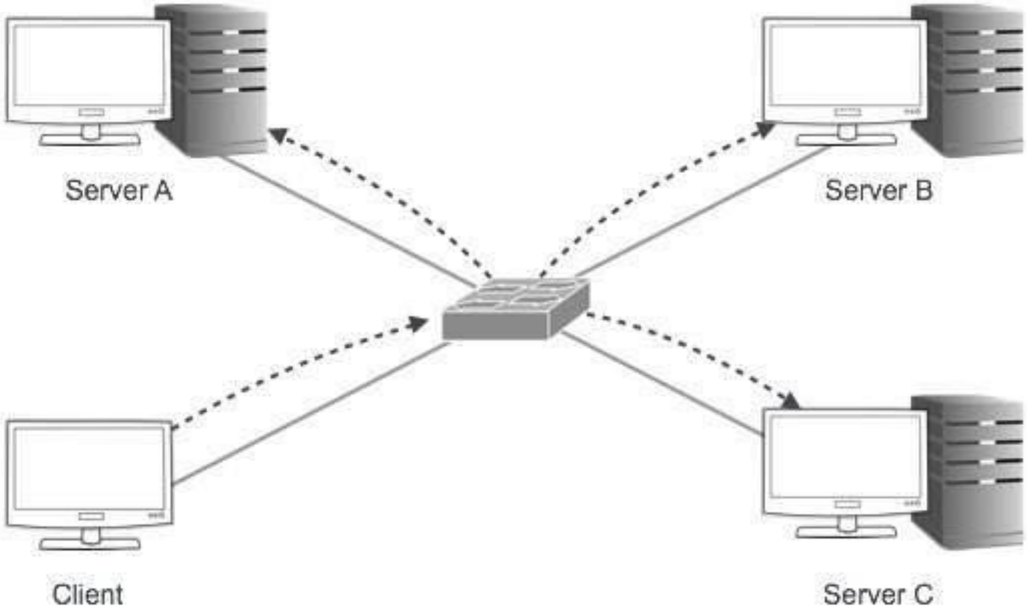
Unicast Addressing Mode

In this mode, data is sent only to one destined host. The Destination Address field contains 32- bit IP address of the destination host. Here the client sends data to the targeted server –



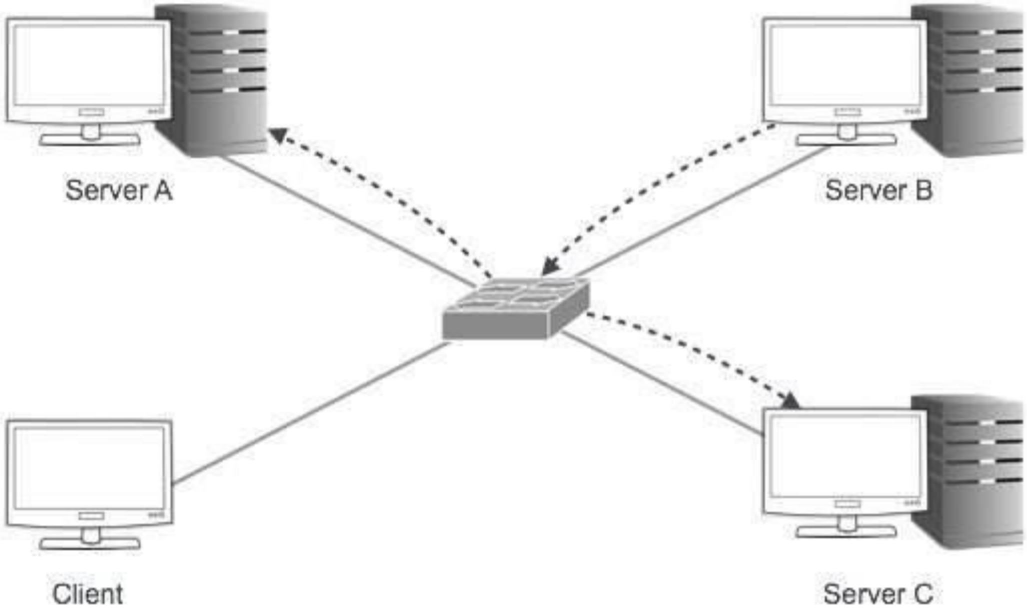
Broadcast Addressing Mode

In this mode, the packet is addressed to all the hosts in a network segment. The Destination Address field contains a special broadcast address, i.e. 255.255.255.255. When a host sees this packet on the network, it is bound to process it. Here the client sends a packet, which is entertained by all the Servers –



Multicast Addressing Mode

This mode is a mix of the previous two modes, i.e. the packet sent is neither destined to a single host nor all the hosts on the segment. In this packet, the Destination Address contains a special address which starts with 224.x.x.x and can be entertained by more than one host.



Here a server sends packets which are entertained by more than one servers. Every network has one IP address reserved for the Network Number which represents the network and one IP address reserved for the Broadcast Address, which represents all the hosts in that network.

Various organizations are currently working with IPv4 technology and in a very short time, we can not switch directly from IPv4 to IPv6. Instead of only using IPv6, we use a combination of both and transition means not replacing IPv4 but co-existing of both.

When we want to send a request from an IPv4 address to an IPv6 address, it is not possible because IPv4 and IPv6 transition is not compatible. For a solution to this problem, we use some technologies that help in an easy transition from IPv4 to IPv6. These technologies are mentioned below:

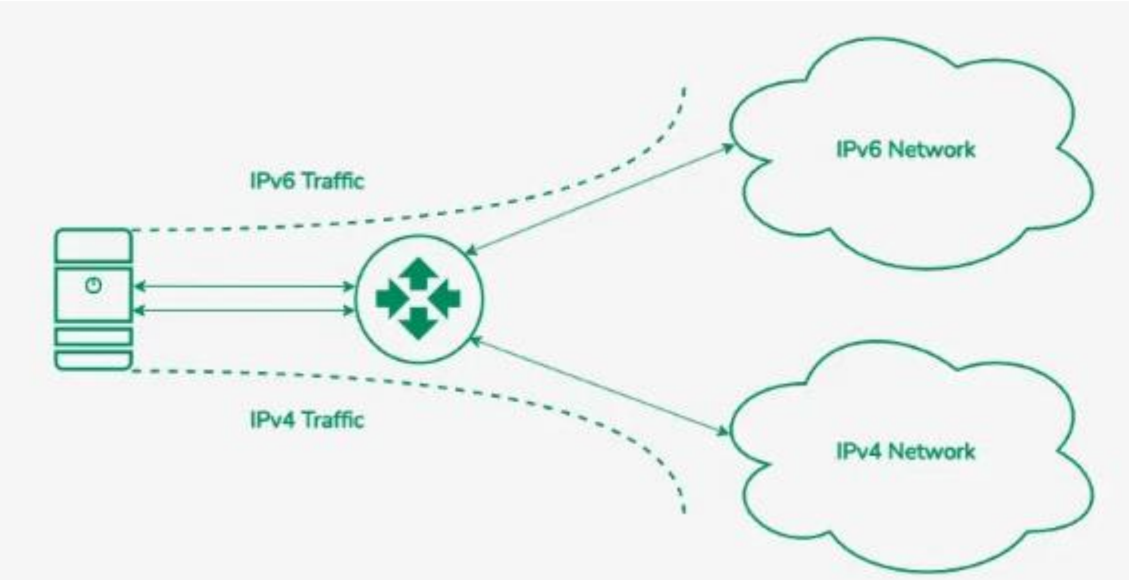
Dual Stack Routers

Tunneling

NAT Protocol Translation

Dual-Stack Routers

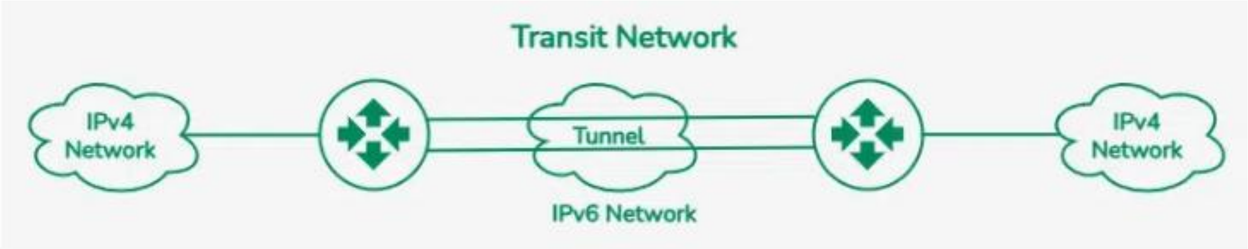
A dual-stack router is a network device that can support both IPv4 and IPv6 protocols simultaneously. It allows communication between devices using any of the protocol, making it a key component during the transition from IPv4 to IPv6. In dual-stack router, A router's interface is attached with IPv4 and IPv6 addresses configured are used in order to transition from IPv4 to IPv6.



In this above diagram, A given server with both IPv4 and IPv6 addresses configured can communicate with all hosts of IPv4 and IPv6 via dual-stack router (DSR). The dual stack router (DSR) gives the path for all the hosts to communicate with the server without changing their IP addresses.

Tunneling

Tunneling is a technique used to enable communication between IPv4 and IPv6 networks during the transition from IPv4 to IPv6. Tunneling encapsulates IPv6 packets within IPv4 packets (or vice versa). Tunneling is used as a medium to communicate the transit network with the different IP versions.



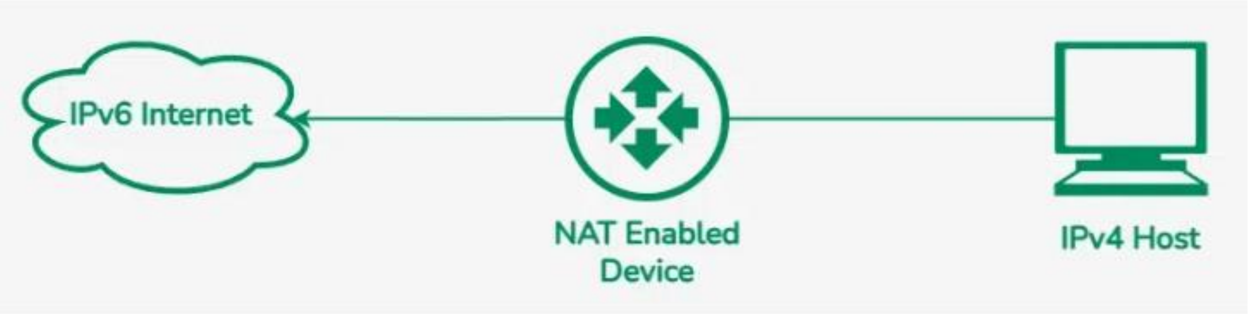
Tunneling

In this above diagram, the different IP versions such as IPv4 and IPv6 are present. The IPv4 networks can communicate with the transit or intermediate network on IPv6 with the help of the Tunnel. It's also possible that the IPv6 network can also communicate with IPv4 networks with the help of a Tunnel.

NAT Protocol Translation

NAT (Network Address Translation) Protocol Translation (NAT-PT), is a technique used to enable communication between IPv4 and IPv6 networks by translating one protocol to the other. With the help of the NAT Protocol Translation technique, the IPv4 and IPv6 networks can also communicate with each other without understanding the address of different IP version.

Generally, an IP version doesn't understand the address of different IP version, for the solution of this problem we use NAT-PT device which removes the header of first (sender) IP version address and add the second (receiver) IP version address so that the Receiver IP version address understand that the request is sent by the same IP version, and its vice-versa is also possible.



NAT Protocol Translation

In the above diagram, an IPv4 address communicates with the IPv6 address via a NAT-PT device to communicate easily. In this situation, the IPv6 address understands that the request is sent by the same IP version (IPv6) and it responds.

What is Congestion Control in Computer Networks?

Congestion occurs when the amount of data transmitted over a network exceeds its capacity, leading to a slowdown or interruption of data transmission. Congestion control is necessary to ensure that data can be transmitted efficiently and effectively over a network, even when the network is under heavy load.

Effective congestion control can prevent network outages, minimize delays, and reduce the risk of data loss. With congestion control, network performance can improve significantly, improving user experience.

Effects of Congestion Control

The effects of congestion control include

Improved network performance: Congestion control techniques help prevent congestion and reduce packet loss, resulting in improved network performance and better end-user experience.

Fairness: Congestion control mechanisms ensure that network resources are shared fairly among all users, preventing one user from dominating the network.

Efficient use of network resources: Congestion control techniques help ensure that network resources are used efficiently, preventing wastage and maximizing the network's capacity.

Reduced latency: Congestion control mechanisms help to reduce network latency by preventing network congestion and packet loss.

Improved scalability: Congestion control mechanisms help to ensure that the network can handle increasing amounts of traffic without becoming congested, thus improving the scalability of the network.

Congestion Control Algorithms

Congestion control algorithms manage network data traffic to avoid bottlenecks by adjusting how much data can be sent at a time. They monitor network performance and adapt data transmission rates accordingly. Examples include TCP Tahoe, Reno, and BBR, which balance efficiency and fairness to prevent network congestion while maximizing data transfer speed.

Leaky Bucket Algorithm

The Leaky Bucket Algorithm is a traffic-shaping algorithm used in computer networks to control the rate at which data is transmitted. It regulates the flow of packets to ensure that they are transmitted at a controlled and steady rate, preventing bursts of traffic that can lead to network congestion. The Leaky

Bucket Algorithm helps to provide a smoother transmission experience and prevents network congestion. It is generally used in quality of service (QoS) implementations, bandwidth allocation, and network traffic management.

Let us consider an example to understand:

Consider there is a bucket in which data gets loaded. The bucket has a hole in it. The bucket can only transfer the data that can be accumulated in that hole. That means when data packets arrive in a bucket, only those packets that can be transferred from that hole only get moved onto the network layer. When the bucket is full, no further data is accepted. Either the data is kept in the queue or discarded as the bucket size is full but once the bucket has space, it starts accepting the data, and in this way the congestion control of the data takes place.

Token bucket Algorithm

The Token Bucket Algorithm is a traffic-shaping algorithm used in computer networks to control the rate of data transmission. It allows for burst traffic while maintaining an overall average transmission rate. The Token Bucket Algorithm ensures the transmission rate stick to the defined average rate. This helps to manage network congestion and enforce quality of service (QoS) policies.

Need of token bucket Algorithm

In networking and computer systems, the Token Bucket Algorithm is typically used for traffic shaping and rate restriction. It aids in regulating how quickly a system handles incoming requests or data, ensuring fair usage and reducing congestion.

The requirement for the token bucket is as follows:

Congestion Avoidance: The Token Bucket Algorithm controls the data flow to help reduce congestion when network or system resources are constrained. It ensures that traffic stays within a certain rate, preventing overloading and preserving consistent performance.

Traffic Control: It's crucial to control the speed at which data is transferred or processed in network environments, particularly when several users or applications share resources. The Token Bucket Algorithm aids in preventing abrupt spikes in traffic that can cause congestion and deterioration of service.

Rate Limiting: The token bucket algorithm is used to enforce rate limitations on data transmissions or requests. The method makes sure that only a certain quantity of data may be communicated over a specific period by regulating the rate at which tokens (representing units of data) are added to the bucket. This stops resource abuse or indiscriminate use.

Steps of this algorithm can be described as follows:

The token bucket algorithm is an algorithm that constantly works to refill, consume, and process the packet that comes along. The algorithm goes as discussed below:

It is a control algorithm that sets when the token should go from the bucket and the capacity of the bucket to hold the data. The capacity of the bucket is the number of tokens it can hold.

It looks for the time when the bucket has to be refilled. At certain levels, or periodically, the token bucket is filled with the maximum capacity.

Whenever the data packet arrives, it checks the token bucket; if the token bucket is empty, it sends the packet, and if it cannot accommodate, the packet is either delayed or marked with lower priority.

Based on the size of the packet, tokens are removed from the bucket. When excess packets arrive or bursts happen, either the packet is at lower priority or is delayed.

This process constantly runs and helps decide the data packet to be transmitted and received.

Let’s understand with an example.

Consider a bucket that can hold 5 tokens at a time. The token bucket arrival rate is three tokens and the packet size is 2. At first, the token bucket is initially empty, and all the packets are transferred. After every transmission, the token size is checked and if the size is less or equal the packet is transferred. If excess packets arrive it is either set for lower priority or is kept in queue. The bucket can only accommodate packets equal to its token size. In this manner, token bucket algorithm helps in congestion control flow of data.

What is the Quality of Service?

Quality of Service (QoS) is a group of technologies that operate on a network to ensure that high-priority traffic and applications may be reliably carried out even when the network's capacity is constrained.

Additionally, the QoS specifies that supporting priority for one or more flows will not fail other flows. A flow can consist of a packet from a particular application or an incoming interface as well as source and destination addresses, source and destination socket numbers, session identifiers, and packets.

The companies can avoid interruptions in real-time communications applications such as VoIP (voice over IP), AoIP (audio over IP), and others by using quality of service (QoS).

How Does QoS in Computer Networks Work?

QoS facilitates the manipulation of packet loss, postponement, and jitter in your community infrastructure. Since we are operating with a finite quantity of bandwidth, our first order of enterprise is to become aware of what packages could benefit from handling those three things.

Once community and alertness directors become aware of the packages that want to have precedence over bandwidth in a community, the following step is to become aware of those visitors. There are numerous approaches to become aware of or mark the visitors. Class of Service (CoS) and Differentiated Services Code Point (DSCP) are examples.

We may utilize this information to set policies on those groups in order to give some data streams preferential treatment over others now that we can group data streams into different groups. Queuing is the term for this. The routing or switching device will advance these packets/frames to the front of the queue and transmit them right away, for instance, if voice traffic is tagged and a policy is developed to grant it access to the bulk of network bandwidth on a channel.

However, if a typical TCP data transfer stream is given a lower priority designation, it will wait (be queued) until enough bandwidth is available to send. These lower-priority packets/frames are the first to be dropped if the queues get overcrowded.

QoS networking generation works through marking packets to become aware of provider types, then configuring routers to create separate digital queues for every utility, primarily based totally on their precedence.

As a result, bandwidth is reserved for essential packages or websites that have been assigned precedence to get entry.

QoS technology offers the ability and manage allocation to particular flows of community visitors. This allows the community administrator to assign the order wherein packets are treated and offer an appropriate quantity of bandwidth to every utility or visitor going with the drift.

Difference between Integrated Services and Differentiated Services.

Feature	Integrated Services (IntServ)	Differentiated Services (DiffServ)
QoS Guarantees	Individualized guarantees for each session	Class-based guarantees for groups of traffic
Resource Reservation	Requires explicit resource reservation	Does not require explicit resource reservation
Scalability	Potentially lower scalability due to per-flow state	Better scalability due to class-based approach
Signaling	Uses RSVP for resource reservation	Does not use RSVP
Complexity	More complex, requiring individual session-level management	Simpler, with per-class traffic management

Quality of Service (QoS) is a type of Networking Technology that can guarantee a specific level of output for a specific connection, path, or type of traffic. QoS mechanisms provide control on both quality and availability of bandwidth whereas another network provides only a best-effort delivery. QoS feature is used when there is traffic congestion in-network, it gives priority to certain real-time media. A high level of QoS is used while transmitting real-time multimedia to eliminate latency and dropouts. Asynchronous Transfer Mode (ATM) is a networking technology that uses a certain level of QoS in data transmission. The Quality of Service in ATM is based on following: Classes, User-related attributes, and Network-related attributes. These are explained as following below. **1. Classes :** The ATM Forum defines four service classes that are explained below -

1. **Constant Bit Rate (CBR) -** CBR is mainly for users who want real-time audio or video services. The service provided by a dedicated line. For example, T line is similar to CBR class service.
2. **Variable Bit Rate (VBR) -** VBR class is divided into two sub classes -
    - **(i) Real-time (VBR-RT) :** The users who need real-time transmission services like audio and video and they also use compression techniques to create a variable bit rate, they use VBR-RT service class.
    - **(ii) Non-real Time (VBR-NRT) :** The users who do not need real-time transmission services but they use compression techniques to create a variable bit rate, then they use VBR-NRT service class.
3. **Available Bit Rate (ABR) -** ABR is used to deliver cells at a specific minimum rate and if more network capacity is available, then minimum rate can be exceeded. ABR is very much suitable for applications that have high traffic.
4. **Unspecified Bit Rate (UBR) -** UBR class and it is a best-effort delivery service that does not guarantee anything.

DNS

The DNS name space is the set of all domain names that are registered in the DNS. These domain names are organized into a tree-like structure, with the top of the tree being the root domain. Below the root domain, there are a number of top-level domains, such as .com, .net, and .org. Below the top-level domains, there are second-level domains, and so on. Each domain name in the DNS name space corresponds to a set of resource records, which contain information about that domain name, such as its IP address, mail servers, and other information.

The DNS name space is hierarchical, meaning that each domain name can have subdomains beneath it. For example, the domain name "example.com" could have subdomains such as "www.example.com" and "mail.example.com". This allows for a very flexible and scalable naming structure for the Internet.

Symmetric Key Cryptography	**Asymmetric Key Cryptography**
There is only one key (symmetric key) is used, and the similar key can be used to encrypt and decrypt the message.	There are two different cryptographic keys (asymmetric keys), known as the public and the private keys, are used for encryption and decryption.
It is effective as this technique is recommended for high amounts of text.	It is inefficient as this approach is used only for short messages.
Symmetric encryption is generally used to transmit bulk information.	It is generally used in smaller transactions. It is used for making a secure connection channel before transferring the actual information.
Symmetric key cryptography is also known as secret-key cryptography or private key cryptography.	Asymmetric key cryptography is also known as public-key cryptography or a conventional cryptographic system.
Symmetric key cryptography uses fewer resources as compared to asymmetric key cryptography.	Asymmetric key cryptography uses more resources as compared to symmetric key cryptography.
The length of the keys used is frequently 128 or 256 bits, based on the security need.	The length of the keys is much higher, such as the recommended RSA key size is 2048 bits or higher.

Difference Between AES and DES

AES	**DES**
AES stands for Advanced Encryption Standard	DES stands for Data Encryption Standard
The date of creation is 2001.	The date of creation is 1977.
Byte-Oriented.	Bit-Oriented.
Key length can be 128-bits, 192-bits, and 256-bits.	The key length is 56 bits in DES.
Number of rounds depends on key length: 10(128-bits), 12(192-bits), or 14(256-bits)	DES involves 16 rounds of identical operations
The structure is based on a substitution-permutation network.	The structure is based on a Feistel network.
The design rationale for AES is open.	The design rationale for DES is closed.
The selection process for this is secret but accepted for open public comment.	The selection process for this is secret.
AES is more secure than the DES cipher and is the de facto world standard.	DES can be broken easily as it has known vulnerabilities. 3DES(Triple DES) is a variation of DES which is secure than the usual DES.
The rounds in AES are: Byte Substitution, Shift Row, Mix Column and Key Addition	The rounds in DES are: Expansion, XOR operation with round key, Substitution and Permutation
AES can encrypt 128 bits of plaintext.	DES can encrypt 64 bits of plaintext.
It can generate Ciphertext of 128, 192, 256 bits.	It generates Ciphertext of 64 bits.
AES cipher is derived from an aside-channel square cipher.	DES cipher is derived from Lucifer cipher.
AES was designed by Vincent Rijmen and Joan Daemen.	DES was designed by IBM.
No known crypt-analytical attacks against AES but side channel attacks against AES implementations possible. Biclique attacks have better complexity than brute force but still ineffective.	Known attacks against DES include Brute-force, Linear crypt-analysis, and Differential crypt-analysis.
It is faster than DES.	It is slower than AES.
It is flexible.	It is not flexible.