Solved Paper 2012-13
Computer Network
B.Tech (IT) VI semester

Section A

1. (a) Which of the communication modes support two way traffic but in only one direction?

Ans:- Half Duplex mode

(b) Explain the difference between 10 Base T and 10 Base 2 cabling

Ans:-IEEE shorthand identifiers, such as ***10Base5, 10Base2, 10BaseT, and 10BaseF*** include three pieces of information:

The number 10: At the front of each identifier, 10 denotes the standard data transfer speed over these media - ten megabits per second (10Mbps).
The word Base: Short for Baseband, this part of the identifier signifies a type of network that uses only one carrier frequency for signaling and requires all network stations to share its use. **The segment type or segment length:** This part of the identifier can be a digit or a letter
Digit - shorthand for how long (in meters) a cable segment may be before attenuation sets in. For example, a 10Base5 segment can be no more than 500 meters long
Letter - identifies a specific physical type of cable. For example, the
T at the end of 10BaseT stands for twisted-pair.

10BaseT

This particular implementation uses four-pair UTP wiring using RJ-45 connectors. Each cable is connected from each network device to a central hub in a physical star topology. Within the hub, the signals are repeated and forwarded to all other nodes on the network because it is a logical bus topology.

10Base2

10Base2, also called ThinNet, is one of the two Ethernet specifications that use coaxial cable.,

(c) **Why and how is bit stuffing used in framing?**

Ans:- Bit stuffing is the insertion of one or more bits into a transmission unit as a way to provide signaling information to a receiver. The receiver knows how to detect and remove or disregard the stuffed bits.

example of bit stuffing, a standard HDLC packet begins and ends with 01111110. To make sure this sequence doesn't appear again before the end of the packet, a 0 is inserted after every five consecutive 1s.

(d)In what situation contention based Mac protocols are suitable

Ans:- Contention based MAC protocols are suitable for bursty nature of traffic under light to moderate load. These techniques are always decentralized, simple and easy to implement.

(e)What are the goals needed in achieving a good routing algorithm

Ans:- Routing algorithms often have one or more of the following design goals:

1. Optimality
2. Simplicity and low overhead
3. Robustness and stability
4. Rapid convergence
5. Flexibility

Optimality refers to the capability of the routing algorithm to select the best route, which depends on the metrics and metric weightings used to make the calculation. For example, one routing algorithm may use a number of hops and delays, but it may weigh delay more heavily in the calculation. Naturally, routing protocols must define their metric calculation algorithms strictly.

Routing algorithms also are designed to be as simple as possible. In other words, the routing algorithm must offer its functionality efficiently, with a minimum of software and utilization overhead. Efficiency is particularly important when the software implementing the routing algorithm must run on a computer with limited physical resources.

Routing algorithms must be robust, which means that they should perform correctly in the face of unusual or unforeseen circumstances, such as hardware failures, high load conditions, and incorrect implementations. Because routers are located at network junction points, they can cause considerable problems when they fail. The best routing algorithms are often those that have withstood the test of time and that have proven stable under a variety of network conditions.

(f)What is a broadcast IP address

Ans:- In computer networking, a broadcast address is an IP address that allows information to be sent to all machines on a given subnet rather than a specific machine. The exact notation can vary by operating system, but the standard is laid out in RFC 919.

The broadcast address is generally obtained by performing a bitwise OR between the bit complement of the subnet mask and the IP address.

Example: to broadcast a packet to an entire class B subnet using a private IP address space, the broadcast address would be 172.16.255.255.
This can be found from the subnet mask (255.240.0.0) and the IP address (eg. 172.16.48.196) - the complement of the subnet mask is 0.15.255.255, and 172.16.48.196 | 0.15.255.255 = 172.31.255.255.

(g)What is Piggy backing?

Ans:- Piggybacking is a bi-directional data transmission technique in the network layer (OSI model). It makes the most of the sent data frames from receiver to emitter, adding the confirmation that the data frame sent by the sender was received successfully (ACK acknowledge). This practically means, that instead of sending an acknowledgement in an individual frame it is piggy-backed on the data frame.

(h)Briefly describe any two session related services?

Ans:- The session layer establishes a communications session between processes running on different computers and can support message-mode data transfer.
Functions of the session layer include the following tasks:

- Permits application processes to register unique process addresses, such as NetBIOS names. The session layer uses these stored addresses to help resolve the addresses of network adapters from process addresses.
- Establishing, monitoring, and terminating a virtual-circuit session between two processes identified by their unique process addresses. A virtual-circuit session is a direct link that exists between the sender and receiver.

(i)List some of the major security problems that exist on the internet

1. Validation of input and output data
2. Direct data access (and theft)
3. Data poisoning
4. Malicious file execution
5. Authentication and session management
6. Phishing
7. Denial of service
8. System information leakage

(j) What are the email gateways?

Ans:- Key Secure Email Gateway Features

1. Protection from email-based, blended threats
2. Prevents data loss
3. Improves productivity by reducing employee spam management
4. Ensures compliance with regulations and AUPs

Section B

2.(a) What are the salient features of ISDN? Discuss the functions of different layers of ISDN?

Ans:- ISDN PRI, also known as Integrated Service Digital Network Primary Rate Interface, is also offered by Tele1ten for all size of business, be it short-term, medium-term or long-term business. Such types of business require intense amount of communication. All the information can be accessed with the help of the internet but in order to take the advantage of such enhancements, you are in dire need of a high speed internet connection. Cleanest connection is provided by the ISDN digital technology so you won't be interrupted, and certainly will not be slowed down in case re-transmissions that happens sometimes.

PRI is basically used for official purposes. The foundation is located on a T1 line in US, and E1 line in Europe. It tends to be an ISDN which is equivalent to a T1 circuit. It is generally a part of ISDN. It offers 23 phone lines. The remaining channels are used in providing signaling and data. The ISDN is said to be an integrated line which possess the ability to carry both telephone calls and internet data. If you are out looking for constant phone and reliable internet service, then ISDN could be the best choice for you. The integration of voice and data circuit will signify the best values for phone services delivering both voice and data at a time. Multiple DSO voice as well as data between two particular physical locations is transported by ISDN.

Salient features of ISDN:

1. DID or Direct Inward Calling feature
2. Reversed Direct Inward Dialing
3. Caller Line Identification

4. Call by call routing feature
5. Dynamic allocation of port

Benefits of ISDN

1. It is economical as well as efficient
2. It increases speed and quality
3. It is more flexible and increases productivity
4. It also provides instant access on voice and net
5. It provides increased channels for multiple call at a time

Applications used on ISDN are as follows:

1. Electronic data processing
2. If private lines go down, Dial backup links can be used
3. Data transmission on both LAN and WAN
4. PBX-to-central office connections

various ISDN specifications for Layer 1, Layer 2, and Layer 3.

Layer 1

ISDN physical layer (Layer 1) frame formats differ depending on whether the frame is outbound (from terminal to network) or inbound (from network to terminal). Both physical layer interfaces are shown in

Layer 2

Layer 2 of the ISDN signaling protocol is Link Access Procedure, D channel (LAPD). LAPD is similar to High-Level Data Link Control (HDLC) and Link Access Procedure, Balanced (LAPB). As the expansion of the LAPD acronym indicates, this layer is used across the D channel to ensure that control and signaling information flows and is received properly. The LAPD frame is very similar to that of HDLC; like HDLC, LAPD uses supervisory, information, and unnumbered frames. The LAPD protocol is formally specified in ITU-T Q.920 and ITU-T Q.921.

Layer 3

Two Layer 3 specifications are used for ISDN signaling: ITU-T (formerly CCITT) I.450 (also known as ITU-T Q.930) and ITU-T I.451 (also known as ITU-T Q.931). Together, these protocols support user-to-user, circuit-switched, and packet-switched connections. A variety of call-establishment, call-termination, information, and miscellaneous messages are specified, including SETUP, CONNECT, RELEASE, USER INFORMATION, CANCEL, STATUS, and DISCONNECT. These messages are functionally similar to those provided by the X.25 protocol.

(b) Explain Performance issues for the following data link control protocols:

(i) Go-back-n

Ans:- Go-Back-N ARQ is a specific instance of the automatic repeat request (ARQ) protocol, in which the sending process continues to send a number of frames specified by a *window size* even without receiving an acknowledgement (ACK) packet from the receiver. It is a special case of the general sliding window protocol with the transmit window size of N and receive window size of 1.

The receiver process keeps track of the sequence number of the next frame it expects to receive, and sends that number with every ACK it sends. The receiver will ignore any frame that does not have the exact sequence number it expects – whether that frame is a "past" duplicate of a frame it has already ACK'ed or whether that frame is a

"future" frame past the last packet it is waiting for. Once the sender has sent all of the frames in its *window*, it will detect that all of the frames since the first lost frame are *outstanding*, and will go back to sequence number of the last ACK it received from the receiver process and fill its window starting with that frame and continue the process over again.

Go-Back-N ARQ is a more efficient use of a connection than Stop-and-wait ARQ, since unlike waiting for an acknowledgement for each packet, the connection is still being utilized as packets are being sent. In other words, during the time that would otherwise be spent waiting, more packets are being sent. However, this method also results in sending frames multiple times – if any frame was lost or damaged, or the ACK acknowledging them was lost or damaged, then that frame and all following frames in the window (even if they were received without error) will be re-sent. To avoid this, Selective Repeat ARQ can be used.

(ii) Sliding Window

Ans:- A **sliding window protocol** is a feature of packet-based data transmission protocols. Sliding window protocols are used where reliable in-order delivery of packets is required, such as in the Data Link Layer (OSI model) as well as in the Transmission Control Protocol (TCP).

Conceptually, each portion of the transmission (packets in most data link layers, but bytes in TCP) is assigned a unique consecutive sequence number, and the receiver uses the numbers to place received packets in the correct order, discarding duplicate packets and identifying missing ones. The problem with this is that there is no limit on the size of the sequence numbers that can be required.

By placing limits on the number of packets that can be transmitted or received at any given time, a sliding window protocol allows an unlimited number of packets to be communicated using fixed-size sequence numbers. The term "window" on transmitter side represents the logical boundary of the total number of packets yet to be acknowledged by the receiver. The receiver informs the transmitter in each acknowledgment packet the current maximum receiver buffer size (window boundary). The TCP header uses a 16 bit field to report the receive window size to the sender. Therefore, the largest window that can be used is 2^{16} = 64 kilobytes. In slow-start mode, the transmitter starts with low packet count and increases the number of packets in each transmission after receiving acknowledgment packets from receiver. For every ack packet received, the window slides by one packet (logically) to transmit one new packet. When the window threshold is reached, the transmitter sends one packet for one ack packet received. If the window limit is 10 packets then in slow start mode the transmitter may start transmitting one packet followed by two packets (before transmitting two packets, one packet ack has to be received), followed by three packets and so on until 10 packets. But after reaching 10 packets, further transmissions are restricted to one packet transmitted for one ack packet received. In a simulation this appears as if the window is moving by one packet distance for every ack packet received. On the receiver side also the window moves one packet for every packet received. The sliding window method ensures that traffic congestion on the network is avoided. The application layer will still be offering data for transmission to TCP without worrying about the network traffic congestion issues as the TCP on sender and receiver side implement sliding windows of packet buffer. The window size may vary dynamically depending on network traffic.

For the highest possible throughput, it is important that the transmitter is not forced to stop sending by the sliding window protocol earlier than one round-trip delay time (RTT). The limit on the amount of data that it can send

before stopping to wait for an acknowledgment should be larger than the bandwidth-delay product of the communications link. If it is not, the protocol will limit the effective bandwidth of the link.

© Explain briefly, the new features in IPV6 as compared to IPV4. What is the purpose of multiple headers? Explain briefly how IPV6 handles multiple headers.

Ans:-IPv6 provides a substantially larger IP address space than IPv4

IPv6 uses 128 bits for IPv6 addresses which allows for 340 billion billion billion billion (3.4x1038) unique addresses.

IPv6 provides better end-to-end connectivity than IPv4

IPv6 with its large address space no longer requires NAT and can ensure true end-to-end connectivity. This means peer-to-peer applications like VoIP or streaming media can work very effectively and efficiently with IPv6.

IPv6 has better ability for autoconfiguring devices than IPv4

IPv6 offers automatic configuration and more importantly, simple configuration mechanisms. Known as plug-and-play autoconfiguration, these capabilities are way beyond what IPv4 currently offers. IPv6 offers DHCPv6, which is an autoconfiguration similar to IPv4 DHCP and offers stateful address autoconfiguration. In addition, IPv6 also offers stateless or serverless address autoconfiguration.

IPv6 contains simplified Header Structures leading to faster routing as compared to IPv4

When compared to IPv4, IPv6 has a much simpler packet header structure, which is essentially designed to minimize the time and efforts that go in to header processing. This has been achieved by moving the optional fields as well as the nonessential fields to the extension headers that are placed only after the IPv6 header. Consequently, the IPv6 headers are processed more efficiently at the intermediate routers without having to parse through headers or recompute network-layer checksums or even fragment and reassemble packets. This efficiency allows for reduced processing overhead for routers, making hardware less complex and allowing for packets to be processed much faster.

IPv6 provides better security than IPv4 for applications and networks

In IPv6, **IPSec** is a major protocol requirement and is one of the factors in ensuring that IPv6 provides better security than IPv4.

IPSec contains a set of cryptographic protocols for ensuring secure data communication and key exchange. The main protocols used are:

1.**Authentication Header** (AH) protocol, which enables authentication and integrity of data.
2.**Encapsulating Security Payload** (ESP) protocol, which enables both authentication and integrity of data as well as privacy of data.
3.**Internet Key Exchange** (IKE) protocol. This protocol suite helps to initially set up and negotiate the security parameters between two end points. It then also keeps track of this information so that the communication stays secure till the end.

IPv6 gives better Quality of Service (QoS) than IPv4

QoS is given a special boost in the IPv6 protocol with the IPv6 header containing a new field, called Flow Label field that defines how particular packets are identified and handled by the routers. The Flow Label field allows packets that belong to a particular flow, in other words, that start from a particular host and head to a particular destination, to be identified and handled quickly and efficiently by the routers.

IPv6 provides better Multicast and Anycast abilities compared to IPv4

IPv6 extends the multicasting capabilities of IPv4 by offering a large multicast address range. Obviously, this limits the degree to which the information packets have now to be propagated and significantly improves the network efficiency.

Multiple Header and its Handling

Extension headers carry optional Internet Layer information, and are placed between the fixed header and the upper-layer protocol header. The headers form a chain, using the*Next Header* fields. The *Next Header* field in the fixed header indicates the type of the first extension header; the *Next Header* field of the last extension header indicates the type of the upper-layer protocol header in the payload of the packet.

All extension headers are a multiple of 8 octets in size; some extension headers require internal padding to meet this requirement.

There are several extension headers defined, and new extension headers may be defined in the future. Extension headers are to be examined and processed at the packet's destination only, except for *Hop-by-Hop Options*, which need to be processed at every intermediate node on the packet's path, including sending and receiving node. The defined extension headers below are listed in the preferred order, should there be more than one extension header following the fixed header. Note that all extension headers are optional and should only appear at most once, except for the *Destination Options* header, which may appear twice.

If a node does not recognize a specific extension header, it should discard the packet and send an *Parameter Problem* message (ICMPv6 type 4, code 1). When a Next Header value 0 appears in a header other than the fixed header a node should do the same.

Extension Header	Type	Description
Hop-by-Hop Options	0	Options that need to be examined by all devices on the path.
Destination Options (before routing header)	60	Options that need to be examined only by the destination of the packet.
Routing	43	Methods to specify the route for a datagram

		(used with Mobile IPv6).
Fragment	44	Contains parameters for fragmentation of datagrams.
Authentication Header (AH)	51	Contains information used to verify the authenticity of most parts of the packet.
Encapsulating Security Payload (ESP)	50	Carries encrypted data for secure communication.
Destination Options (before upper-layer header)	60	Options that need to be examined only by the destination of the packet.
Mobility (currently without upper-layer header)	135	Parameters used with Mobile IPv6.

Value 59 (No Next Header) in the Next Header field indicates that there is no next header *whatsoever* following this one, not even a header of an upper-layer protocol. It means that, from the header's point of view, the IPv6 packet ends right after it: the payload should be empty. There could, however, still be data in the payload if the payload length in the first header of the packet is greater than the length of all extension headers in the packet. This data should be ignored by hosts, but passed unaltered by routers.

(d) What is public key cryptography? List its advantages and disadvantages. Expalin How RSA works.

Ans:- Public-key cryptography refers to a cryptographic system requiring two separate keys, one of which is secret and one of which is public. Although different, the two parts of the key pair are mathematically linked. One key locks or encrypts the plaintext, and the other unlocks or decrypts the ciphertext. Neither key can perform both functions by itself. The public key may be published without compromising security, while the private key must not be revealed to anyone not authorized to read the messages.

Advantages

- **Convenience:** It solves the problem of distributing the key for encryption.Everyone publishes their public keys and private keys are kept secret.

- **Provides for message authentication:** Public key encryption allows the use of digital signatures which enables the recipient of a message to verify that the message is truly from a particular sender.

- **Detection of tampering:** The use of digital signatures in public key encryption allows the receiver to detect if the message was altered in transit. A digitally signed message cannot be modified without invalidating the signature.

- **Provide for non-repudiation:** Digitally signing a message is akin to physically signing a document. It is an acknowledgement of the message and thus, the sender cannot deny it.

Disadvantages

- **Public keys should/must be authenticated:** No one can be absolutely sure that a public key belongs to the person it specifies and so everyone must verify that their public keys belong to them.

- **Slow:** Public key encryption is slow compared to symmetric encryption. Not feasible for use in decrypting bulk messages.

- **Uses up more computer resources:** It requires a lot more computer supplies compared to single-key encryption.

- **Widespread security compromise is possible:** If an attacker determines a person's private key, his or her entire messages can be read.

- **Loss of private key may be irreparable:** The loss of a private key means that all received messages cannot be decrypted.

Working of RSA

An RSA public-key / private-key pair can be generated by the following steps:
1. Generate a pair of large, random primes p and q.
2. Compute the modulus n as n = pq.
3. Select an odd public exponent e between 3 and n-1 that is relatively prime to p-1 and q-1.
4. Compute the private exponent d from e, p and q.
5. Output (n, e) as the public key and (n, d) as the private key.

The encryption operation in the RSA cryptosystem is exponentiation to the eth power modulo n:

c = ENCRYPT (m) = me mod n .

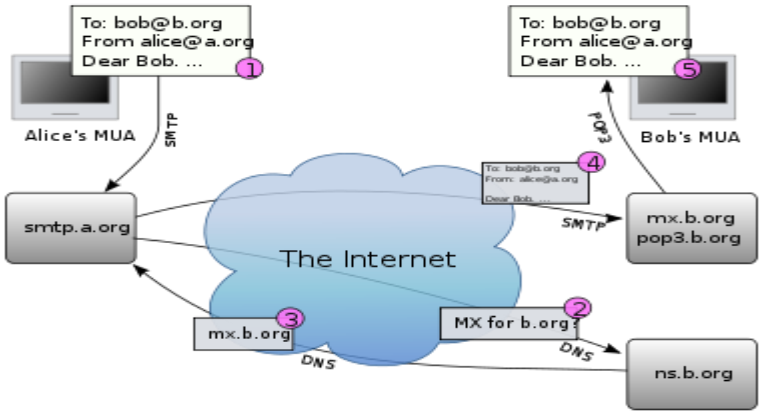
The input m is the message; the output c is the resulting ciphertext. In practice, the message m is typically some kind of appropriately formatted key to be shared. The actual message is encrypted with the shared key using a traditional encryption algorithm. This construction makes it possible to encrypt a message of any length with only one exponentiation. The decryption operation is exponentiation to the dth power modulo n:

m = DECRYPT (c) = cd mod n .

(e) Explain how does E-mail reach to destination. Explain in brief SMTP emphasizing the role and function of USER AGENT (UA) and mail transfer agent (MTA)

Ans:- Operation of email

The diagram to the right shows a typical sequence of event that takes place when Alice composes a message using her mail user agent (MUA). She enters the email addressof her correspondent,



and hits the "send" button.

1. Her MUA formats the message in email format and uses the Submission Protocol (a profile of theSimple Mail Transfer Protocol (SMTP), see RFC 6409) to send the message to the local mail submission agent (MSA), in this case smtp.a.org, run by Alice's internet service provider (ISP).
2. The MSA looks at the destination address provided in the SMTP protocol (not from the message header), in this case bob@b.org. An Internet email address is a string of the formlocalpart@exampledomain. The part before the @ sign is the *local part* of the address, often theusername of the recipient, and the part after the @ sign is a domain name or a fully qualified domain name. The MSA resolves a domain name to determine the fully qualified domain name of the mail exchange server in the Domain Name System (DNS).
3. The DNS server for the b.org domain, ns.b.org, responds with any MX records listing the mail exchange servers for that domain, in this case mx.b.org, a message transfer agent (MTA) server run by Bob's ISP.
4. smtp.a.org sends the message to mx.b.org using SMTP.

This server may need to forward the message to other MTAs before the message reaches the final message delivery agent (MDA).

1. The MDA delivers it to the mailbox of the user bob.
2. Bob presses the "get mail" button in his MUA, which picks up the message using either the Post Office Protocol (POP3) or the Internet Message Access Protocol (IMAP4).

That sequence of events applies to the majority of email users. However, there are many alternative possibilities and complications to the email system:

1. Alice or Bob may use a client connected to a corporate email system, such as IBM Lotus Notes or Microsoft Exchange. These systems often have their own internal email format and their clients typically communicate with the email server using a vendor-specific, proprietary protocol. The server sends

or receives email via the Internet through the product's Internet mail gateway which also does any necessary reformatting. If Alice and Bob work for the same company, the entire transaction may happen completely within a single corporate email system.

2. Alice may not have a MUA on her computer but instead may connect to a webmail service.
3. Alice's computer may run its own MTA, so avoiding the transfer at step 1.
4. Bob may pick up his email in many ways, for example logging into mx.b.org and reading it directly, or by using a webmail service.
5. Domains usually have several mail exchange servers so that they can continue to accept mail when the main mail exchange server is not available.
6. Email messages are not secure if email encryption is not used correctly.

Many MTAs used to accept messages for any recipient on the Internet and do their best to deliver them. Such MTAs are called *open mail relays*. This was very important in the early days of the Internet when network connections were unreliable. If an MTA couldn't reach the destination, it could at least deliver it to a relay closer to the destination. The relay stood a better chance of delivering the message at a later time. However, this mechanism proved to be exploitable by people sending unsolicited bulk email and as a consequence very few modern MTAs are open mail relays, and many MTAs don't accept messages from open mail relays because such messages are very likely to be spam.

Mail Transfer Agent

Within Internet message handling services (MHS), a **message transfer agent** or **mail transfer agent** (**MTA**) or **mail relay** is software that transfers electronic mail messages from one computer to another using a client–server application architecture. An MTA implements both the client (sending) and server (receiving) portions of the Simple Mail Transfer Protocol. The terms *mail server*, *mail exchanger*, and *MX host* may also refer to a computer performing the MTA function. The Domain Name System (DNS) associates a mail server to a domain with mail exchanger (MX) resource records containing the domain name of a host providing MTA services.

A mail server is a computer that serves as an electronic post office for email. Mail exchanged across networks is passed between mail servers that run specially designed software. This software is built around agreed-upon, standardized protocols for handling mail messages and the graphics they might contain.

USER AGENT

A mail user agent (MUA) is a program that allows you to receive and send e-mail messages; it's usually just called an e-mail program. To use an MUA such as Eudora or Microsoft Outlook, you install the MUA program on your computer and then use it to download and store e-mail messages to your computer; it will also allow you to read or write messages offline. Web-based MUAs, such as Hotmail and Yahoo, store messages on their own mailservers and allow access to them through a Web page. An MUA is sometimes called an e-mail agent or an e-mail client.

Section C

Q3(a)Explain guided transmission media. Mention the characteristics that distinguish optical from twisted pair or coxial cable.

Ans: **Guided Transmission Media**

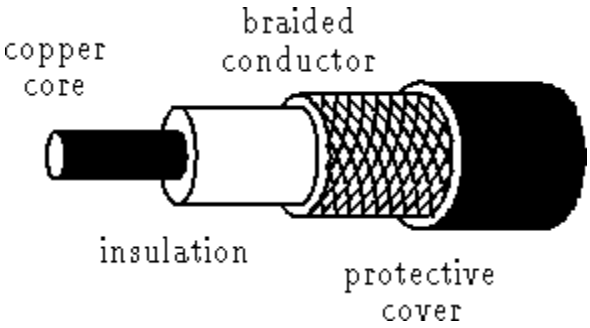
Waves are guided along solid medium

Twisted Pair

1. Pair of twisted conductors
   - 
2. Twisting reduces interference (two parallel wires constitute a simple antenna; a twisted pair does not.)
3. Cheap medium
4. Commonly used for communications within buildings and in telephone networks
5. Produced in unshielded (UTP) and shielded (STP) forms, and in different performance categories.
6. Cables may hold hundreds of pairs. Neighbor pairs typically have different twist lengths to reduce crosstalk.

Coaxial Cable

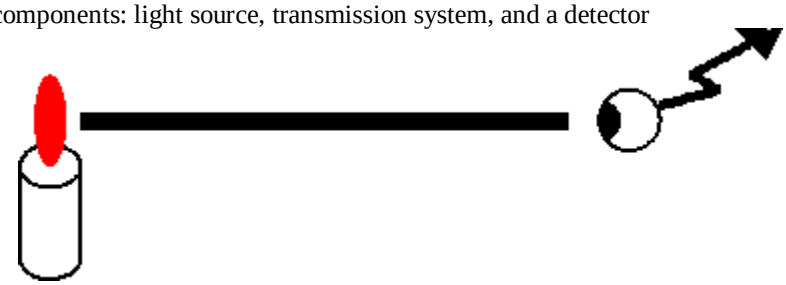
Pair of conductors separated by insulation



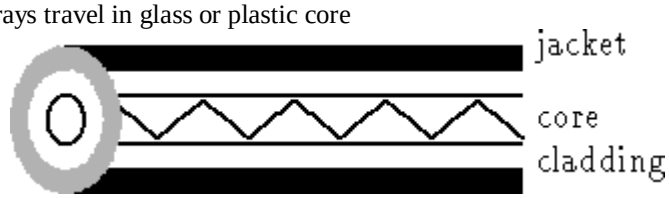
1. Offers longer distances and better speeds than twisted pair, due to better shielding.
2. Used for cable TV and local-area networks. Had been widely used in telephone systems, but optical fibre is now assuming this task.

Baseband Coaxial Cable 50-ohm cable, commonly used for digital transmission.
Broadband Coaxial Cable 75-ohm cable, commonly used for analog transmission.

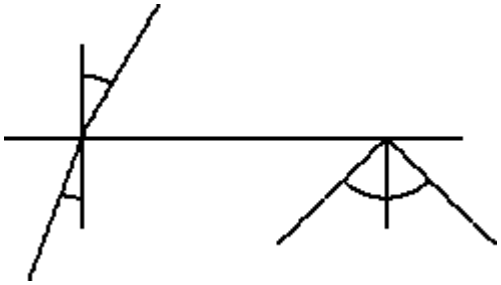
Optical Fibre

1. Three components: light source, transmission system, and a detector
   - 
2. The detector generates an electric pulse when hit by light

3. 1-a pulse of light; 0-missing pulse of light.
4. optical rays travel in glass or plastic core



5. When light move from one medium to another it bend at the boundary. The amount of bending depends on the properties of the media.



6. Light at shallow angles propagate along the fibre, and those that are less than critical angle are absorbed in the jacket
7. The cladding is a glass or plastic with properties that differ from those of the core
8. Used in long distance communication, in locations having small amount of space, and with reduction in price is starting to get also to LANs.
9. Not affected by external electromagnetic fields, and do not radiate energy. Hence, providing high degree of security from eavesdropping.
10. Provide for **multimode** of propagation at different angles of reflections. Cause signal elements to spread out in time, which limits the rate in which data can be accurately received.
11. Reduction of the radius of the core implies less reflected angles. **Single mode** is achieved with sufficient small radius.
12. A **multimode graded index** transmission is obtained by varying the index of reflection of the core to improve on the multi mode option without resolving to the cost of single mode. (index of reflection=speed in vacuum / speed in medium.)
13. 1 Gbps is the current limitation, with the bottle neck in the conversion from electrical to optical signals. Large improvements are expected.

• Coaxial cable	Optical Fiber
1. Data rate in Mbps relativelylow in comparison ofOptical Fiber .	1. Data rate in Gbps .
2. High attenuation	2. Low Attenuation .
3. Transmission through electro magnetic waves .	3. Transmission through Light Pulse .
4. Possibility of cross talk .	4. No Possibility of cross talk.
5. Best suited for Cable distribution and large LAN's .	5. best suited for LANs where high quality of facilities are required .

(b)What are the key benefits of layered network? What do you mean by service access point?

Ans: The key principle of modern networking is the separation of network functionality into independent layers; this is also true in the tele and datacoms industry. The 'layered' thinking is also a very fundamental and visible aspect in a number of standardization initiatives and industry forums, such as Multiservice Switching Forum, led by several of the largest operators and manufactures.
3G mobile communication systems are designed with these principles in mind, and hence offer a number of advantages and possibilities. Special attention needs to be put on meeting the requirements for time-to-market, low cost of ownership, openness and future evolution towards an "all-IP" solution.

Key characteristics :

1. Offers an open and versatile architecture capable of meeting the current and future demand in a fast changing telecommunications environment.
2. Provides and inherent flexibility for coping with growth and/or changing traffic patterns and traffic mixes.
3. Independence between layers allows each layer to evolve independently.
4. Provides great transport flexibility and allows different transport technologies, both existing and new. To be deployed without impacting the control or services/application layers.
5. Allows common transport arrangements for multi-services network.
6. Allows access independent and seamless services through a common service/applications layer.
7. Provides efficient use of network resources.
8. Relies on proven and stable protocols and design.
9. Allow a very flexible re-use of investments in the GSM infrastructure.

Why Layered architecture?

1. To make the design process easy by breaking unmanageable tasks into several smaller and manageable tasks (by divide-and-conquer approach).

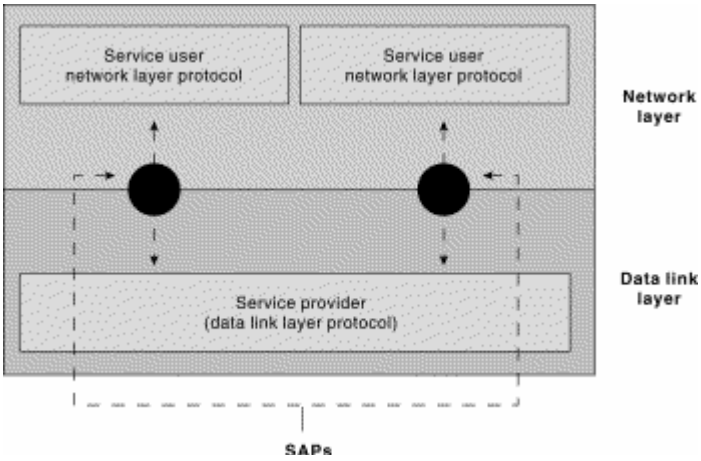
2. Modularity and clear interfaces, so as to provide comparability between the different providers' components.

3. Ensure independence of layers, so that implementation of each layer can be changed or modified without affecting other layers.

4. Each layer can be analyzed and tested independently of all other layers.

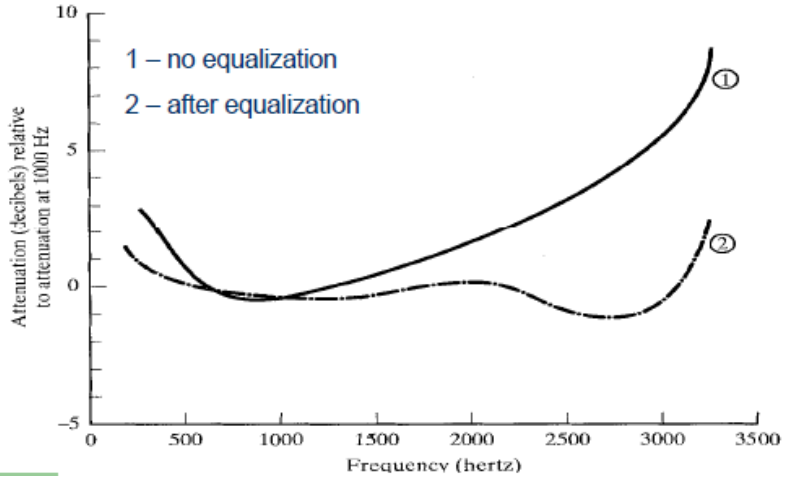
Service access Point: The Network Service Access Point (NSAP) is one of two types of hierarchical addresses (the other type is the *network entity title*) used to implement Open Systems Interconnection (OSI)network layer addressing. The NSAP is the logical point between the network and transport layers where network services are delivered to the transport layer; the location of this point is identified to the OSI network service provider by the NSAP address. There are two NSAP address fields, Initial Domain Part (IDP) and the Domain-Specific Part (DSP).

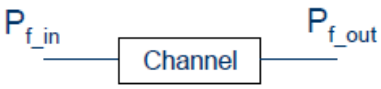
The IDP consists of the Authority Format Identifier (AFI) and the Initial Domain Identifier (IDI). AFI serves to provide information about the makeup of IDI and DSP, indicating, for example, whether the DSP uses decimal or binary notation, or whether or not the IDI may be of variable length.



©Explain degradation of signal quality due to attenuation and delay distortion.

Ans: Signal strength decreases with distance. It depends on the transmission medium. Received signal strength: must be high enough, allowing the signal to bedetected. It must be sufficiently higher than noise, such that the signal to be received accurately (Solution: repeaters, amplifiers) higher the transmission frequency, higher the attenuation is (mainly concerns the analog signals).

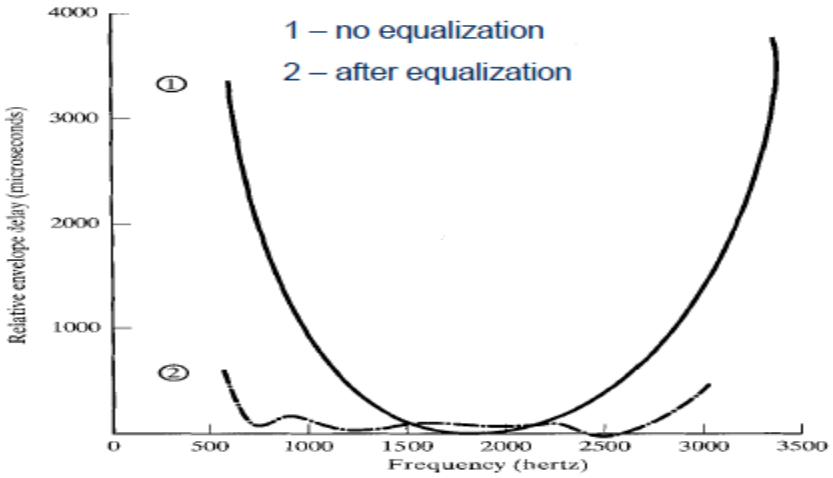




$$N_f = -10\log_{10}\frac{P_f}{P_{1000}}$$

The frequency dependency of the attenuation versus distance curve is particularly upsetting for analog data. Fig shows the specific attenuation curve for an analog telephone channel, whose frequency band spans from 300 to 3400 Hz. As for digital data, most of their energy is concentrated around a "fundamental frequency", which is, basically, equal to the data rate. Consequently, the higher frequency components have lower energy and the frequency dependency of the attenuation has no meaningful impact. Notice that in the figure the attenuation uses as a reference the signal power at 1000 Hz, and, consequently, both curves cross zero dB at that frequency. For the un-equalized curve, the attenuation is much higher towards the superior edge of the dedicated band, which causes degradation of the voice quality. The equalized curve is much smoother, meaning that all the frequency components are similarly attenuated during the transmission. Such an equalization can be performed using amplifiers, that amplify more the higher frequency components and less the lower frequency part of the signal.

Delay Distortion: Specific to guided media (wires). Signal propagation speed depends on the frequency. Frequency selectivity arises: various frequency components of the signal will arrive at receiver with different delays. A kind of Inter Symbol Interference (ISI) occurs. It particularly annoying for digital data.



The so called "delay distortion" actually means that the velocity of propagation of different frequency components of the transmitted signal is a function of frequency.
So, we deal again with a kind of "frequency selectivity" of the channel. For a band limited signal, the propagation speed tens to be higher in the center of the dedicated bandwidth and lower towards its edges. Thus, various frequency components arrive at reception with different delays and thus distort the signal. Equalization techniques must be used to alleviate the negative effect of the delay distortion.

4. (a) Consider the use of 10 K-bit size fames on a 10 Mbps satellite channel with 270 ms delay. What is the link utilization for stop and wait ARQ technique assuming P= 10^-3?

Ans:- Link utilization = (1-P) / (1+2a) Where
a = (Propagation Time) / (Transmission Time)
Propagation time = 270 msec
Transmission time = (frame length) / (data rate)
= (10 K-bit) / (10 Mbps)
= 1 msec
Hence, a = 270/1 = 270
Link utilization = 0.999/(1+2*270) ≈0.0018 =0.18%

(b) Prove that for a slotted ALOHA system , the maximum throughput happens at G=1 where G is the number of attempts per packet time.

Ans:- An improvement to the original ALOHA protocol was "Slotted ALOHA", which introduced discrete timeslots and increased the maximum throughput.[10] A station can send only at the beginning of a timeslot, and thus collisions are reduced. In this case, we only need to worry about the transmission-attempts within 1 frame-time and not 2 consecutive frame-times, since collisions can only occur during each timeslot. Thus, the probability of there being zero transmission-attempts in a single timeslot is:

$$Prob_{slotted} = e^{-G}$$

the probability of k packets is:

$$Prob_{slotted}k = e^{-G}(1 - e^{-G})^{k-1}$$

The throughput is:

$$S_{slotted} = Ge^{-G}$$

The maximum throughput is *1/e* frames per frame-time (reached when *G* = 1), which is approximately 0.368 frames per frame-time, or 36.8%.

© How is line coding implemented in FDDI.

Ans:- The ANSI standard (X3T9-5) for a dual, counter-rotating, fiber optic, token-passing ring LAN.The specification pegs the signaling rate at 125 Mbps and the transmission rate (i.e., data rate) at 100 Mbps due to the 4B/5B line coding technique. FDDI is intended for backbone applications, interconnecting major computing resources such as high speed switches, routers, and servers. As the FDDI maximum frame size is 9000 symbols (1 symbol = 4 bytes), Ethernet and Token Ring frames can easily be encapsulated within FDDI frames for backbone transport. FDDI specifies devices separations of as much as 1.2 miles (2 kilometers) over multimode fiber (MMF) and 37.2 miles (62 kilometers) over single-mode fiber (SMF), with excellent error performance.The dual counter-rotating ring provides considerable redundancy, but requires that all directly connected devices be dual-attached, which adds to the cost and complexity. In consideration of the high cost and fragility of optical fiber, standards were developed to extend connectivity to workstations via unshielded twisted pairs.Those standards are known variously as *CDDI* (Copper Distributed Data Interface) and *TPDDI*(Twisted Pair Distributed Data Interface). FDDI is considered obsolete, having been overwhelmed by simpler, higher speed switched Ethernet technologies such as 1000Base-LX, 1000Base-SX, and 10GBase-LR, LW.

FDDI line coding

- 4b5b line code
  - successive 4 bit blocks of data are translated into five bit blocks for transmission
  - code translation is chosen to achieve DC balance
  - e.g.

Symbol	Code Group
0	11110
1	01001
2	10100
3	10101

FDDI uses a 4b/5b coding scheme. This scheme take each 4 bit block to be transmitted and translates it into a unique 5 bit block. The choice of 5 bit blocks is determined by considerations like: placing a limit on the number of consecutive 1s to ensure a reasonably constant DC optical power level so that simpler receivers can be designed.□s and 1□s that appear on the transmission line (limited to 3 by this code), achieving a balance between the number of 0□FDDI uses a 4b/5b coding scheme. This scheme take each 4 bit block to be transmitted and translates it into a unique 5 bit block. The choice of 5 bit blocks is determined by considerations like: placing a limit on the number of consecutive 0

The table shows a sample of the symbols 0-F (needing a 4 bit representation) and their 5 bit representation. Clearly there are twice as many 5b symbols as there are 4b symbols to represent. Some of these are used to represent control

symbols that will not appear in the data and the remainder are known as violation symbols as these should never be found in the transmitted frame.

Q5(a).How is subnet mask useful in addressing? Explain with an example.

Ans: An IP address has two components, the network address and the host address. A subnet mask separates the IP address into the network and host addresses (<network><host>). Subnetting further divides the host part of an IP address into a subnet and host address (<network><subnet><host>). It is called a subnet mask because it is used to identify network address of an IP address by perfoming bitwise AND operation on the netmask.

A Subnet mask is a 32-bit number that masks an IP address, and divides the IP address into network address and host address. Subnet Mask is made by setting network bits to all "1"s and setting host bits to all "0"s. Within a given network, two host addresses are reserved for special purpose. The "0" address is assigned a network address and "255" is assigned to a broadcast address, and they cannot be assigned to a host.

Subnetting an IP network is to separate a big network into smaller multiple networks for reorganization and security purposes. All nodes (hosts) in a subnetwork see all packets transmitted by any node in a network. Performance of a network is adversely affected under heavy traffic load due to collisions and retransmissions.

Applying a subnet mask to an IP address separates network address from host address. The network bits are represented by the 1's in the mask, and the host bits are represented by 0's. Performing a bitwise logical AND operation on the IP address with the subnet mask produces the network address. For example, applying the Class C subnet mask to our IP address 216.3.128.12 produces the following network address:

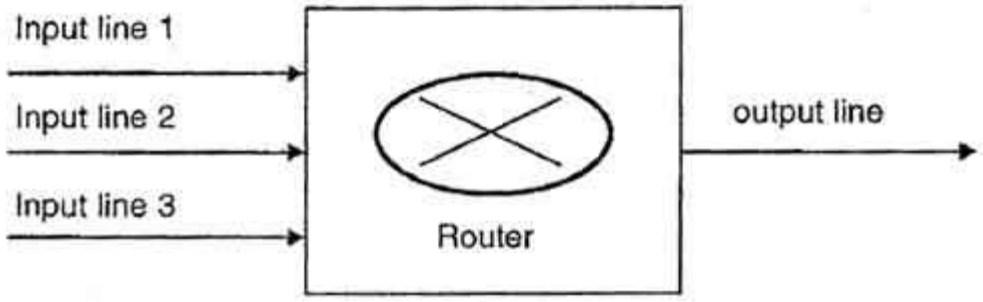
IP: 1101 1000 . 0000 0011 . 1000 0000 . 0000 1100 (216.003.128.012)
Mask: 1111 1111 . 1111 1111 . 1111 1111 . 0000 0000 (255.255.255.000)

1101 1000 . 0000 0011 . 1000 0000 . 0000 0000 (216.003.128.000)

(b)What are the reasons for congestion in a network? Describe any one method for congestion control.

Ans: Congestion occurs when the number of packets being transmitted through the network approaches the packet handling capacity of the network. Congestion control aims to keep number of packets below level at which performance falls off dramatically. Data network is a network of queues. Generally 80% utilization is critical. Finite queues mean data may be lost. The various causes of congestion in a subnet are:

1.The input traffic rate exceeds the capacity of the output lines.



Data from three input lines at same time

2.The routers are too slow to perform bookkeeping tasks (queuing buffers, updating tables, etc.).

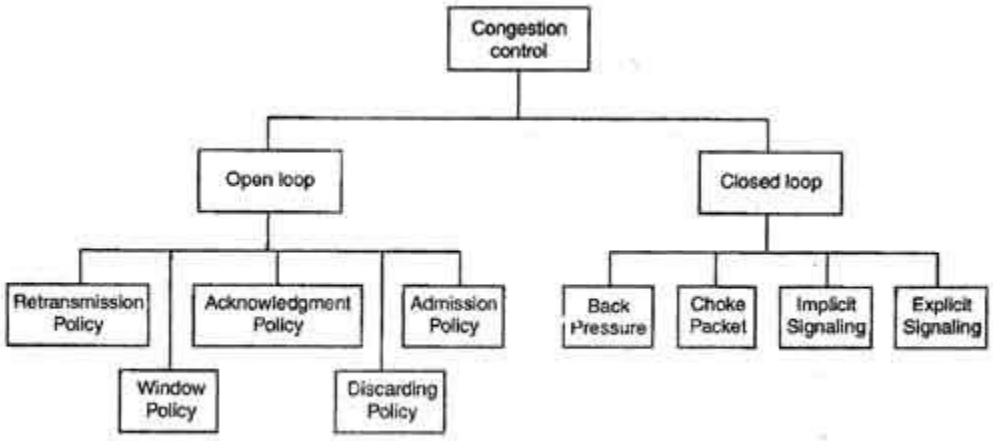
3.The routers' buffer is too limited.

4.Congestion in a subnet can occur if the processors are slow. Slow speed CPU at routers will perform the routine tasks such as queuing buffers, updating table etc slowly. As a result of this, queues are built up even though there is excess line capacity.

5.Congestion is also caused by slow links.

How to correct the Congestion Problem:

Congestion Control refers to techniques and mechanisms that can either prevent congestion, before it happens, or remove congestion, after it has happened. Congestion control mechanisms are divided into two categories, one category prevents the congestion from happening and the other category removes congestion after it has taken place.



Types of Congestion Control Methods

These two categories are:

1. Open loop

2. Closed loop

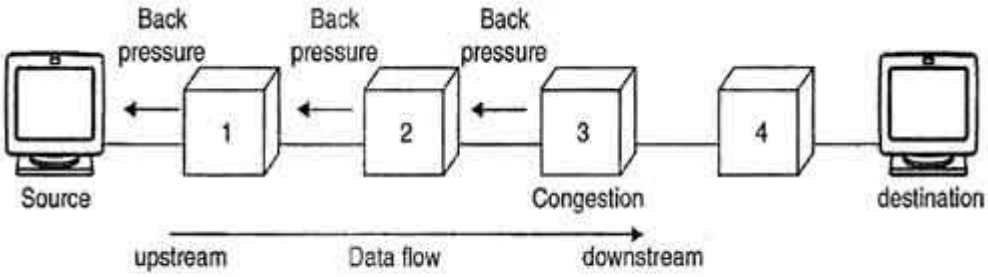
Closed Loop Congestion Control

• Closed loop congestion control mechanisms try to remove the congestion after it happens.

• The various methods used for closed loop congestion control are:

1. Backpressure

• Backpressure is a node-to-node congestion control that starts with a node and propagates, in the opposite direction of data flow.



Backpressure method

• The backpressure technique can be applied only to virtual circuit networks. In such virtual circuit each node knows the upstream node from which a data flow is coming.

• In this method of congestion control, the congested node stops receiving data from the immediate upstream node or nodes.

• This may cause the upstream node on nodes to become congested, and they, in turn, reject data from their upstream node or nodes.

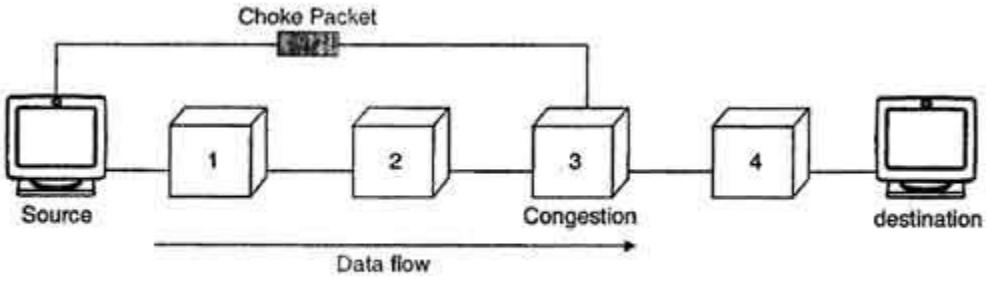
• As shown in fig node 3 is congested and it stops receiving packets and informs its upstream node 2 to slow down. Node 2 in turns may be congested and informs node 1 to slow down. Now node 1 may create congestion and informs the source node to slow down. In this way the congestion is alleviated. Thus, the pressure on node 3 is moved backward to the source to remove the congestion.

2. Choke Packet

• In this method of congestion control, congested router or node sends a special type of packet called choke packet to the source to inform it about the congestion.

• Here, congested node does not inform its upstream node about the congestion as in backpressure method.

• In choke packet method, congested node sends a warning directly to the source station *i.e.* the intermediate nodes through which the packet has traveled are not warned.



Choke Packet Method

3. Implicit Signaling

• In implicit signaling, there is no communication between the congested node or nodes and the source.

• The source guesses that there is congestion somewhere in the network when it does not receive any acknowledgment. Therefore the delay in receiving an acknowledgment is interpreted as congestion in the network.

• On sensing this congestion, the source slows down.

• This type of congestion control policy is used by TCP.

4. Explicit Signaling

• In this method, the congested nodes explicitly send a signal to the source or destination to inform about the congestion.

• Explicit signaling is different from the choke packet method. In choke packed method, a separate packet is used for this purpose whereas in explicit signaling method, the signal is included in the packets that carry data .

• Explicit signaling can occur in either the forward direction or the backward direction .

• In backward signaling, a bit is set in a packet moving in the direction opposite to the congestion. This bit warns the source about the congestion and informs the source to slow down.

• In forward signaling, a bit is set in a packet moving in the direction of congestion. This bit warns the destination about the congestion. The receiver in this case uses policies such as slowing down the acknowledgements to remove the congestion.

©Discuss the problem of count to infinity associated with distance vector routing technique.

Ans: Networks using distance-vector routing are susceptible to loops and issues with count to infinity. How does this problem develop? In the following illustration, everything is working fine on the network, and the network is converged.

Problems can happen with your routing protocol when a link or a router fails. In this figure, a failure happens on *Router3* with interface fa0/0. When this link goes down, the route to 10.4.0.0/16 is no longer available; however, if you look at what follows, you can see the issue.

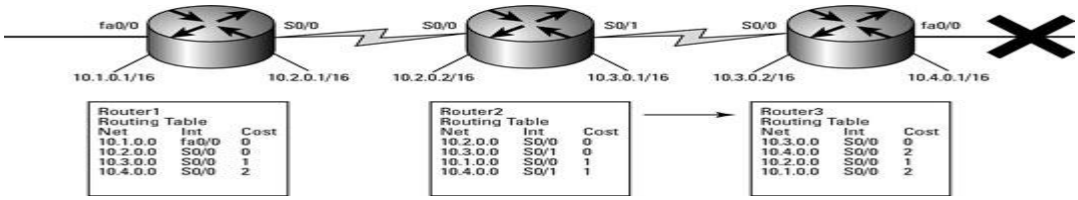
1. ***Router3* initially marks the route to 10.4.0.0 as a link down in its routing table.**

2. ***Router2* sends out its routing table to each of its neighbors.**

   This includes *Router3*, telling them that it has a path to 10.4.0.0 with a hop count of 1.

3. ***Router3* then updates its routing table with this new information.**

   The new information states that the route to 10.4.0.0/16 is now 2 hops away, as shown in Figure 6-3.



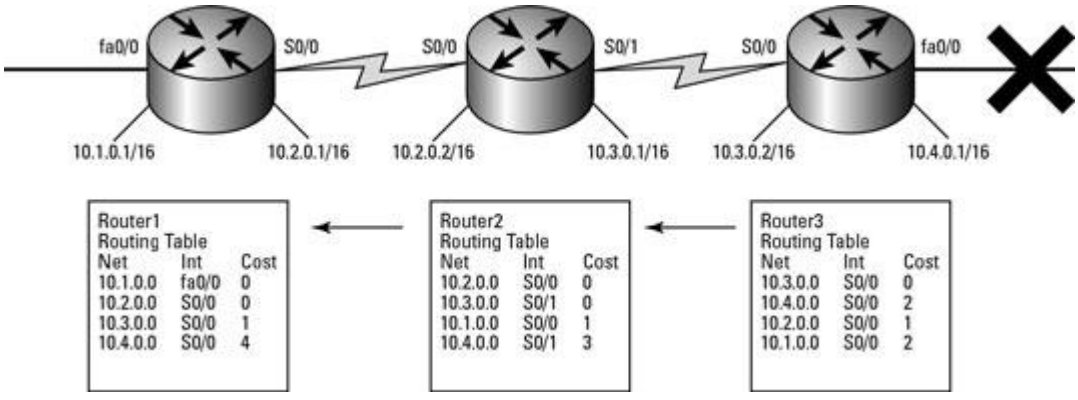
4. **Armed with the new information that 10.4.0.0/16 is available.**

   Albeit through another interface, *Router3* sends out its routing table to its neighbors.

5. ***Router2* gets the update.**

   It then identifies that the router that previously said it knew about 10.4.0.0/16 has updated the route from a hop count of 0 to 2, so Router2 updates its own routing table. The old route may have been identified as an updated route, or it may have timed out of the routing table, depending on the routing protocol that is in use.

6. ***Router2* then passes its own routing information out through its other interface (S0/0) to propagate the change to *Router1*.**



7. ***Router3* eventually receives the update from *Router2*.**

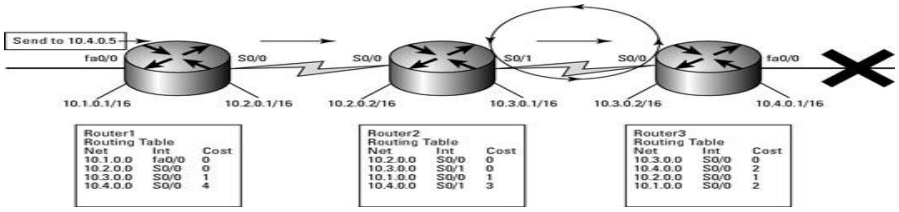
This update tells Router2 that the hop count to 10.4.0.0/16 has been updated to 3, and this process now continues.

This process continues to infinity because no mechanism is in place, in this case, to stop the process from continuing.

However, the RIP routing protocol has a built-in safety mechanism, to a degree. RIP has a maximum hop count of 16, and when the route to a network exceeds the 16-hop rule, the RIP protocol marks that network as unreachable so that it does not further propagate the route. This scenario does not change the information found in the router's routing table — it only limits how far the error is propagated.

When you send data to a host or device on the 10.4.0.0/16 network, it comes through the fa0/0 interface on *Router1* and *Router1* thinks that it can get to 10.4.0.0/16 within 4 hops by sending the data out through interface S0/0 based on *Router1's* routing table. The following figure shows what happens when the data is sent.

As it arrives at *Router3*, *Router3* determines that the route to 10.4.0.0/16 is back through *Router2*, which then causes the data to loop infinitely. There is a Time to Live (TTL) on IP packets, which defines the maximum amount of time which an IP packet can remain on a network. After spending some time looping, the data will be dropped from the network and a message sent back to the sender of the data.



Q 6. (a) Explain how a session layer establishes, maintains and synchronizes the interaction between two communicating hosts.

Ans: In the seven-layer OSI model of computer networking, the **session layer** is **layer 5**.

The session layer provides the mechanism for opening, closing and managing a session between end-user application processes, i.e., a semi-permanent dialogue. Communication sessions consist of requests and responses that occur between applications. Session-layer services are commonly used in application environments that make use of remote procedure calls (RPCs).

An example of a session-layer protocol is the OSI protocol suite session-layer protocol, also known as X.225 or ISO 8327. In case of a connection loss this protocol may try to recover the connection. If a connection is not used for a long period, the session-layer protocol may close it and re-open it. It provides for either full duplex or half-duplex operation and provides synchronization points in the stream of exchanged messages.

Other examples of session layer implementations include Zone Information Protocol (ZIP) – the AppleTalk protocol that coordinates the name binding process, and Session Control Protocol (SCP) – the DECnet Phase IV session-layer protocol.

Within the service layering semantics of the OSI network architecture, the session layer responds to service requests from the presentation layer and issues service requests to the transport layer.

An RPC is initiated by the *client*, which sends a request message to a known remote *server* to execute a specified procedure with supplied parameters. The remote server sends a response to the client, and the application continues its process. While the server is processing the call, the client is blocked (it waits until the server has finished processing before resuming execution), unless the client sends an asynchronous request to the server, such as an XHTTP call. There are many variations and subtleties in various implementations, resulting in a variety of different (incompatible) RPC protocols.

An important difference between remote procedure calls and local calls is that remote calls can fail because of unpredictable network problems. Also, callers generally must deal with such failures without knowing whether the remote procedure was actually invoked. Idempotent procedures (those that have no additional effects if called more than once) are easily handled, but enough difficulties remain that code to call remote procedures is often confined to carefully written low-level subsystems.

Sequence of events during a RPC

1. The client calls the client stub. The call is a local procedure call, with parameters pushed on to the stack in the normal way.
2. The client stub packs the parameters into a message and makes a system call to send the message. Packing the parameters is called marshalling.
3. The client's local operating system sends the message from the client machine to the server machine.
4. The local operating system on the server machine passes the incoming packets to the server stub.
5. The server stub unpacks the parameters from the message . Unpacking the parameters is called unmarshalling.
6. Finally, the server stub calls the server procedure. The reply traces the same steps in the reverse direction.

(b)How does the transport layer ensure that the complete message arrives at the destination and in the proper order?

Ans: Transport layer provides for the segmentation of data and the control necessary to reassemble these pieces into the various communication streams. Its primary responsibilities to accomplish this are:

- Tracking the individual communication between applications on the source and destination hosts
- Segmenting data and managing each piece
- Reassembling the segments into streams of application data
- Identifying the different applications

Tracking Individual Conversations

Any host may have multiple applications that are communicating across the network. Each of these applications will be communicating with one or more applications on remote hosts. It is the responsibility of the Transport layer to maintain the multiple communication streams between these applications.

Segmenting Data

As each application creates a stream data to be sent to a remote application, this data must be prepared to be sent across the media in manageable pieces. The Transport layer protocols describe services that segment this data from the Application layer. This includes the encapsulation required on each piece of data. Each piece of application data requires headers to be added at the Transport layer to indicate to which communication it is associated.

Reassembling Segments

At the receiving host, each piece of data may be directed to the appropriate application. Additionally, these individual pieces of data must also be reconstructed into a complete data stream that is useful to the Application layer. The protocols at the Transport layer describe the how the Transport layer header information is used to reassemble the data pieces into streams to be passed to the Application layer.

Controlling the conversation

The primary functions specified by all Transport layer protocols include:

Segmentation and Reassembly - Most networks have a limitation on the amount of data that can be included in a single PDU. The Transport layer divides application data into blocks of data that are an appropriate size. At the destination, the Transport layer reassembles the data before sending it to the destination application or service.

Conversation Multiplexing - There may be many applications or services running on each host in the network. Each of these applications or services is assigned an address known as a port so that the Transport layer can determine with which application or service the data is identified. In addition to using the information contained in the headers, for the basic functions of data segmentation and reassembly, some protocols at the Transport layer provide:

- Connection-oriented conversations
- Reliable delivery
- Ordered data reconstruction
- Flow control

Establishing a Session

The Transport layer can provide this connection orientation by creating a sessions between the applications. These connections prepare the applications to communicate with each other before any data is transmitted. Within these sessions, the data for a communication between the two applications can be closely managed.

Reliable Delivery

For many reasons, it is possible for a piece of data to become corrupted, or lost completely, as it is transmitted over the network. The Transport layer can ensure that all pieces reach their destination by having the source device to retransmit any data that is lost.

Same Order Delivery

Because networks may provide multiple routes that can have different transmission times, data can arrive in the wrong order. By numbering and sequencing the segments, the Transport layer can ensure that these segments are reassembled into the proper order.

Flow Control

Network hosts have limited resources, such as memory or bandwidth. When Transport layer is aware that these resources are overtaxed, some protocols can request that the sending application reduce the rate of data flow. This is done at the Transport layer by regulating the amount of data the source transmits as a group. Flow control can prevent the loss of segments on the network and avoid the need for retransmission.

Supporting Reliable Communication

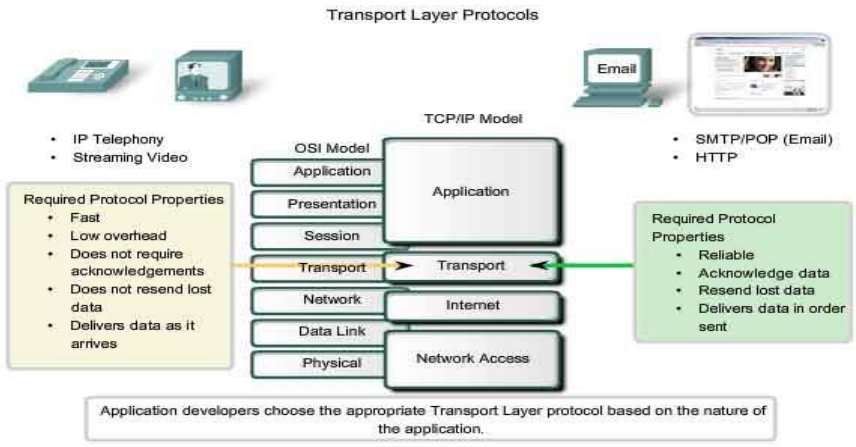
Recall that the primary function of the Transport layer is to manage the application data for the conversations between hosts. However, different applications have different requirements for their data, and therefore different Transport protocols have been developed to meet these requirements. A Transport layer protocol can implement a method to ensure reliable delivery of the data. In networking terms, reliability means ensuring that each piece of data that the source sends arrives at the destination. At the Transport layer the three basic operations of reliability are:

- tracking transmitted data
- acknowledging received data
- retransmitting any unacknowledged data

This requires the processes of Transport layer of the source to keep track of all the data pieces of each conversation and the retransmit any of data that did were not acknowledged by the destination. The Transport layer of the receiving host must also track the data as it is received and acknowledge the receipt of the data. These reliability processes place additional overhead on the network resources due to the acknowledgement, tracking, and retransmission. To support these reliability operations, more control data is exchanged between the sending and receiving hosts. This control information is contained in the Layer 4 header. This creates a trade-off between the value of reliability and the burden it places on the network. Application developers must choose which transport protocol type is appropriate based on the requirements of their applications. At the Transport layer, there are protocols that specify methods for either reliable, guaranteed delivery or best-effort delivery. In the context of networking, best-effort delivery is referred to as unreliable, because there is no acknowledgement that the data is received at the destination.

Determining the Need for Reliability

Applications, such as databases, web pages, and e-mail, require that all of the sent data arrive at the destination in its original condition, in order for the data to be useful. Any missing data could cause a corrupt communication that is either incomplete or unreadable. Therefore, these applications are designed to use a Transport layer protocol that implements reliability. The additional network overhead is considered to be required for these applications. Other applications are more tolerant of the loss of small amounts of data. For example, if one or two segments of a video stream fail to arrive, it would only create a momentary disruption in the stream. This may appear as distortion in the image but may not even be noticeable to the user. Imposing overhead to ensure reliability for this application could reduce the usefulness of the application. The image in a streaming video would be greatly degraded if the destination device had to account for lost data and delay the stream while waiting for its arrival. It is better to render the best image possible at the time with the segments that arrive and forego reliability. If reliability is required for some reason, these applications can provide error checking and retransmission requests.



(c)What is TCP? Connection termination in TCP is symmetric, whereas connection establishment is not. Why?

Ans: Transmission Control Protocol accepts data from a data stream, divides it into chunks, and adds a TCP header creating a TCP segment. The TCP segment is then encapsulated into an Internet Protocol (IP) datagram, and exchanged with peers.

Connection establishment is symmentric whereas connection termination is not

To establish a connection, TCP uses a three-way handshake. Before a client attempts to connect with a server, the server must first bind to and listen at a port to open it up for connections: this is called a passive open. Once the passive open is established, a client may initiate an active open. To establish a connection, the three-way (or 3-step) handshake occurs:

1. **SYN**: The active open is performed by the client sending a SYN to the server. The client sets the segment's sequence number to a random value A.
2. **SYN-ACK**: In response, the server replies with a SYN-ACK. The acknowledgment number is set to one more than the received sequence number i.e. A+1, and the sequence number that the server chooses for the packet is another random number, B.
3. **ACK**: Finally, the client sends an ACK back to the server. The sequence number is set to the received acknowledgement value i.e. A+1, and the acknowledgement number is set to one more than the received sequence number i.e. B+1.

At this point, both the client and server have received an acknowledgment of the connection. The steps 1, 2 establish the connection parameter (sequence number) for one direction and it is acknowledged. The steps 2, 3 establish the connection parameter (sequence number) for the other direction and it is acknowledged. With these, a full-duplex communication is established.

It uses to stop its half of the connection, it transmits a FIN packet, which the other end acknowledges with an ACK. Therefore, a typical tear-down requires a pair of FIN and ACK segments from each TCP endpoint. After both FIN/ACK exchanges are concluded, the side which sent the first FIN before receiving one waits for a timeout before finally closing the connection, during which time the local port is unavailable for new connections; this prevents confusion due to delayed packets being delivered during subsequent connections.

A connection can be "half-open", in which case one side has terminated its end, but the other has not. The side that has terminated can no longer send any data into the connection, but the other side can. The terminating side should continue reading the data until the other side terminates as well.

It is also possible to terminate the connection by a 3-way handshake, when host A sends a FIN and host B replies with a FIN & ACK (merely combines 2 steps into one) and host A replies with an ACK. This is perhaps the most common method.

It is possible for both hosts to send FINs simultaneously then both just have to ACK. This could possibly be considered a 2-way handshake since the FIN/ACK sequence is done in parallel for both directions.

Some host TCP stacks may implement a half-duplex close sequence, as Linux or HP-UX do. If such a host actively closes a connection but still has not read all the incoming data the stack already received from the link, this host sends a RST instead of a FIN This allows a TCP application to be sure the remote application has read all the data the former sent—waiting the FIN from the remote side, when it actively closes the connection. But the remote TCP stack cannot distinguish between a *Connection Aborting RST* and *Data Loss RST*. Both cause the remote stack to lose all the data received.

Q7.(a)What is multipurpose Internet Mail Extension(MIME) and for what it is used?

Ans: MIME (*Multipurpose Internet Mail Extensions*) is a standard which was proposed by Bell Communications in 1991 in order to expand upon the limited capabilities of email, and in particular to allow documents (such as images, sound, and text) to be inserted in a message. It was originally defined by RFCs 1341 and 1342 in June 1992. Using headers, MIME describes the type of message content and the encoding used.

MIME adds the following features to email service:

- Be able to send multiple attachments with a single message;
- Unlimited message length;
- Use of character sets other than ASCII code;
- Use of rich text (layouts, fonts, colors, etc)
- Binary attachments (executables, images, audio or video files, etc.), which may be divided if needed.

MIME uses special header directives to describe the format used in a message body, so that the email client can interpret it correctly:

- MIME-Version: This is the version of the MIME standard used in the message. Currently only version 1.0 exists.
- Content-type: Describes the data's type and subtype. It can include a "charset" parameter, separated by a semi-colon, defining which character set to use.
- Content-Transfer-Encoding: Defines the encoding used in the message body
- Content-ID: Represents a unique identification for each message segment
- Content-Description: Gives additional information about the message content.
- Content-Disposition: Defines the attachment's settings, in particular the name associated with the file, using the attribute *filename*.

(b)Differentiate between SMTP and HTTP.

Ans: There are three main differences between HTTP and SMTP:

1) HTTP is mainly a pull protocol--someone loads information on a web server and users use HTTP to pull the information from the server. On the other hand, SMTP is primarily a push protocol--the sending mail server pushes the file to receiving mail server.

2) SMTP requires each message, including the body of each message, to be in seven-bit ASCII format. If the message contains binary data, then the message has to be encoded into seven-bit ASCII format. HTTP does not have this restriction.

3) HTTP encapsulate each object of message in its own response message while SMTP places all of the message's objects into one message.

©How does FTP work? Differentiate between passive and active FTP.

Ans: FTP stands for File Transfer Protocol. It is the standard Internet protocol for transferring files from one computer to another. FTP is part of the TCP/IP protocol suite. TCP/IP is the basic protocol that runs the whole Internet. Whether you are checking your email, visiting a web site or downloading files, you are using TCP/IP. There are a number of smaller protocols that run on top of TCP/IP, such as email, HTTP, and Telnet. FTP is one of these. Its sole function is to move a file from a server to a client (download) or from a client to a server (upload).

FTP requires two computers, one running an FTP server, the other running an FTP client. The exchange is initiated by the client which logs in under an accepted user name and password. Once this occurs, a session is opened and stays open until closed by either the client or the server, or until it times out. While the session is open, the client may execute numerous FTP commands on the server. These include commands to change directories, list files, get files and put files.

FTP is an unusual protocol in that it uses two ports, one for commands and the other for data. (This is one of the reasons it is superior to HTTP for transferring large files.) Active FTP was invented first. The client initiates a connection on the servers command port. The server then initiates a connection with the client from its data port. In Passive FTP, the client initiates both connections with the server, which remains "passive".

Active FTP may cause problems if your client is behind a firewall. From the firewalls point of view, the FTP server that is trying to initiate a connection with your client looks like an intruder and is usually blocked. This is why many users have difficulty using FTP to download files from behind a firewall.

Passive FTP solves this problem, but creates other problems, notably where FTP server security is concerned. The server must listen on a large number of ports. This requires the firewall to let a lot of unqualified traffic through. Most firewall administrators do not like this.