B.TECH.

SIXTH SEMESTER EXAMINATION, 2007-2008

COMPUTER NETWORKS

Time – 3 hours **Total Marks – 100**

Note: (1) Attempt all questions.

(2) All questions carry equal marks.

(3) Be precise in your answer.

(4) No second answer book will be provided.

Q. 1. Attempt any four parts — 4×5

Q.1.(a). Differentiate between bit-rate and baud-rate. A modem constellation diagram has data point at coordinates: (1,1), (1,–1), (–1,1) and (–1,–1). How many bps can a modem with these parameters achieve at 1200 baud?

Ans. Bit rate Vs Baud Rate: Bit Rate is how many data bits are transmitted per second. A baud Rate is the measurement of the number of times per second a signal in a communications channel changes.

Bit rates measure of the number of data bits (that's 0's and 1's) transmitted in one second in a communication channel.

A baud rate, by definition, means the number of times a signal in a communications channel changes state or varies.

The main difference of the two is that one change of state can transmit one bit - or slightly more or less than one bit, that depends on the modulation technique used. So the bit bate (bps) and baud rate (baud per second) have this connection:

bps = baud per second x the number of bit per baud

The number of bit per baud is determined by the modulation technique. examples:

When FSK ("Frequency Shift Keying", a transmission technique) is used, each baud transmits one bit; only one change in state is required to send a bit. Thus, the modem's bps rate is equal to the baud rate: When we use a baud rate of 2400, you use a modulation technique called phase modulation that transmits four bits per baud. So:

2400 baud x 4 bits per baud = 9600 bps

Such modems are capable of 9600 bps operation.

Q.1.(b). A system is designed to sample analog signals, convert them to digital form with a 4-bit converter, and transmit them. What bit rate is required if the analog signal consists of frequencies between 400 Hz to 3400 Hz ?

Ans. According to question

Frequency bandwidth W = (3400 – 400) Hz

= 3000 Hz

We know that

Date rate = bit rate

$= 2 W \log_2 v = 2 W \log_2 4$

$= 2*3000 \log_2 2^2$

$= 2*2*3000$

= 12000 bps

= 12 Kbps **Ans.**

Q.1.(c). Compare and contrast circuit, message and packet switching techniques.

Ans. Circuit , message & packet switching:

Circuit switching and packet switching both are used in high-capacity networks. In circuit-switched networks, network resources are static, set in "copper" if you will, from the sender to receiver before the start of the transfer, thus creating a "circuit". The resources remain dedicated to the circuit during the entire transfer and the entire message follows the same path. In packet-switched networks, the message is broken into packets, each of which can take a different route to the destination where the packets are recompiled into the original message.

Message switching was the precursor of packet switching, where messages were routed in their entirety, one hop at a time. Message switching systems are nowadays mostly implemented over packet-switched or circuit-switched data networks. A message switch is "transactional". It can store data or change its format and bit rate, then convert the data back to their original form or an entirely different form at the receive end. Message switching multiplexes data from different sources onto a common facility.

Q.1.(d). What are the relative merits and demerits of a single-mode fiber in comparison to a multi-mode fiber? Describe the structure and composition difference between the two.

Ans. Single mode fiber Vs Multi mode fiber:

Multimode fiber optic cable has a large-diameter core that is much larger than the wavelength of light transmitted, and therefore has multiple pathways of light-several wavelengths of light are used in the fiber core.

Multimode fiber optic cable can be used for most general fiber applications. Use multimode fiber for bringing fiber to the desktop, for adding segments to your existing network, or in smaller applications such as alarm systems. Multimode cable comes with two different core sizes: 50 micron or 62.5 micron.

Singlemode fiber optic cable has a small core and only one pathway of light. With only a single wavelength of light passing through its core, singlemode realigns the light toward the center of the core instead of simply bouncing it off the edge of the core as with multimode.

Singlemode is typically used in long-haul network connections spread out over extended areas—longer than a few miles. For example, telcos use it for connections between switching offices. Singlemode cable features a 9-micron glass core.

Multimode fiber has a relatively large light carrying core, usually 62.5 microns or larger in diameter. It is usually used for short distance transmissions with LED based fiber optic equipment. Single-mode fiber has a small light carrying core of 8 to 10 microns in diameter. It is normally used for long distance transmissions with laser diode based fiber optic transmission equipment.

This depends on the application. Multimode fiber will allow transmission distances of up to about 10 miles and will allow the use of relatively inexpensive fiber optic transmitters and receivers. There will be bandwidth

limitations of a few hundred MHz per Km of length. Consequently, a 10 mile link will be limited to about 10 to 30 MHz. For CCTV this will be fine but for high speed data transmission it may not be.

Single-mode fiber on the other hand will be useful for distances well in excess of 10 miles but will require the use of single-mode transmitters (which normally use solid-state laser diodes). The higher cost of these optical emitters mean that single-mode equipment can be anywhere from 2 to 4 times as expensive as multimode equipment.

Q.1.(e). In a certain communication channel, the signal power is 100 W and the noise power is 10 W. In order to Send information at the rate of 10 kbps, what is required bandwidth?

Ans. According to question

Single power, S = 100 W

Noise power N = 10 W $\therefore$ S/N = 100/10 = 10

Data rate r = 10 Kbps = 10×10^3 bps

Band width W = ?

So, by Sharman theorem

$$r = W \log_2\left(1+\frac{S}{N}\right)$$

$$10 \times 10^3 = W \log_2(1+10)$$

$$\Rightarrow \quad W = \frac{10\times10^3}{\log_2(11)}\text{ Hz} = \frac{10}{\log_2 11}\text{ KHz} \quad \textbf{Ans.}$$

Q.1.(f). What is ISDN? Differentiate between-

(i) BRI & PRI

(ii) B-ISDN & N-ISDN

Ans. ISDN: ISDN is a circuit-switched telephone network system, that also provides access to packet switched networks, designed to allow digital transmission of voice and data over ordinary telephone copper wires, resulting in better voice quality than an analog phone. It offers circuit-switched connections (for either voice or data), and packet-switched connections (for data), in increments of 64 kbit/s. Another major market application is Internet access, where ISDN typically provides a maximum of 128 kbit/s in both upstream and downstream directions (which can be considered to be broadband speed, since it exceeds the narrowband speeds of standard analog 56k telephone lines). ISDN B-channels can be bonded to achieve a greater data rate, typically 3 or 4 BRIs (6 to 8 64 kbit/s channels) are bonded.

(i) BRI Vs PRI: The entry level interface to ISDN is the Basic Rate Interface (BRI), a 144 kbit/s service delivered over a pair of standard telephone copper wires. The 144 kbit/s rate is broken down into two 64 kbit/s bearer channels ('B' channels) and one 16 kbit/s signaling channel ('D' channel or Delta channel). BRI is sometimes referred to as 2B+D

PRI: The other ISDN service available is the Primary Rate Interface (PRI) which is carried over an E1 (2048 kbit/s) in most parts of the world. An E1 is 30 'B' channels of 64 kbit/s, one 'D' channel of 64 kbit/s and a timing and alarm channel of 64 kbit/s. In North America PRI service is delivered on one or more T1s (sometimes referred to as 23B+D) of 1544 kbit/s (24 channels). A T1 has 23 'B' channels and 1 'D' channel for signalling (Japan uses a circuit called a J1, which is similar to a T1).

In North America, NFAS allows two or more PRIs to be controlled by a single D channel, and is sometimes called "23B+D + n*24B". D-channel backup allows you to have a second D channel in case the primary fails. One popular use of NFAS is on a T3. PRI-ISDN is popular throughout the world, especially for connection of PSTN circuits to PBXs.

(ii) B-ISDN & N-ISDN: Broadband Integrated Services Digital Network (B-ISDN). This was designed in the 1990s as a logical extension of the end-to-end circuit switched data service, ISDN. The technology for B-ISDN was going to be Asynchronous Transfer Mode (ATM), which was intended to carry both synchronous voice and asynchronous data services on the same transport. The B-ISDN vision has been overtaken by the disruptive technology of the Internet. The ATM technology survives as a low-level layer in most DSL technologies, and as a payload type in some wireless technologies such as WiMAX.

N-ISDN (Narrowband ISDN): A single network capable of carrying several different types of service, based on voice, data, still or moving image - by means of digital transmission techniques. The ISDN (Integrated Services Digital Network) currently being deployed in Europe carries a communication of up to 2 Megabits/second (Narrowband ISDN). Future networks will carry higher speed communications (Broadband ISDN).

Q. 2. Attempt any four parts — 5×4

Q.2.(a). A series of 8-bit message blocks (frames) is to be transmitted across a data link using a CRC for error detection. A generator polynomial of 11001 is to be used. Use an example to illustrate the following:

(i) The CRC generation process

(ii) The CRC checking process.

Ans. According to question

Generator polynomial is, p = 11001

(i). CRC generator process:

Let 8 bit message block is 11110101

```
          101010101
      ____________
11001 | 111101010000  <-- M * 2^4
        11001
        -----
         ××11110
          11001
          -----
           ××11110
            11001
            -----
             ××11100
              11001
              -----
               ××10100
                11001
                -----
                 ×1101
```

So, CRC= Reminder= 1101

(ii). CRC checking process:

Transmitted message

$T = m*2^4 + CRC$

$= 111101010000 + 1101$

$= 111101011101$

So,

```
          101010101
      ____________
P --> 11001 | 111101011101  <-- T
        11001
        -----
         ××11110
          11001
          -----
           ××11111
            11001
            -----
             ××11010
              11001
              -----
               ××11001
                11001
                -----
                 0000  <-- R
```

Clearly, reminder is zero, so, received message is correct one.

Q.2.(b)(i). A bit string, 01111 0111110111111 0, needs to be transmitted at the data link layer. What is the string actually transmitted after bit stuffing?

Ans. Original bit stream

01111 011111 01111110

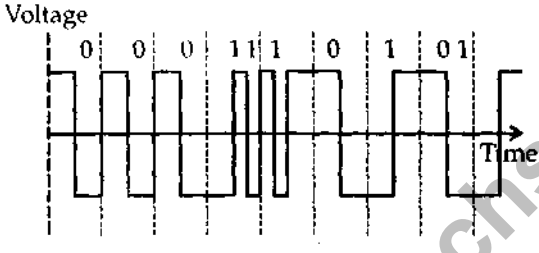
After bit stuffing

01111 0 111110 011111 0 10

(ii). Sketch the Manchester encoding for the bit stream: 0001110101.

Ans. Manchester encoding

Given bit stream



Q.2.(c). Discriminate between the send window and receive window for link and how are they related with

(i) a selective repeat retransmission scheme

ii) a go-back-N control scheme.

Ans. (i) Selective repeat Transmission Scheme: In selective repeat, the receiver's window size is more than one, and the receiver can receive a fixed number of out of order frames equal to the window size. If a frame is received with an error it simply sends a 'negative acknowledgement' signal corresponding to the faulty frame prompting the sender to resend the frame. In the meantime, the recipient can store the subsequent frames, which have already been sent, in a buffer and wait for the remaining frame to be received. Once received, the frames are transferred in sequence to the network layer.

(ii). a go-back-N Control Scheme: Go back n corresponds to the example above where the receiver's window size is one (the sender's window size is greater than one otherwise it's a trivial case). In this case, if an error occurs in receiving the nth frame the receiver simply discards subsequent frames and does not send the acknowledgement corresponding to the nth frame. This compels the sender to resend all frames starting with the nth and thereafter. This deteriorates the data rate.

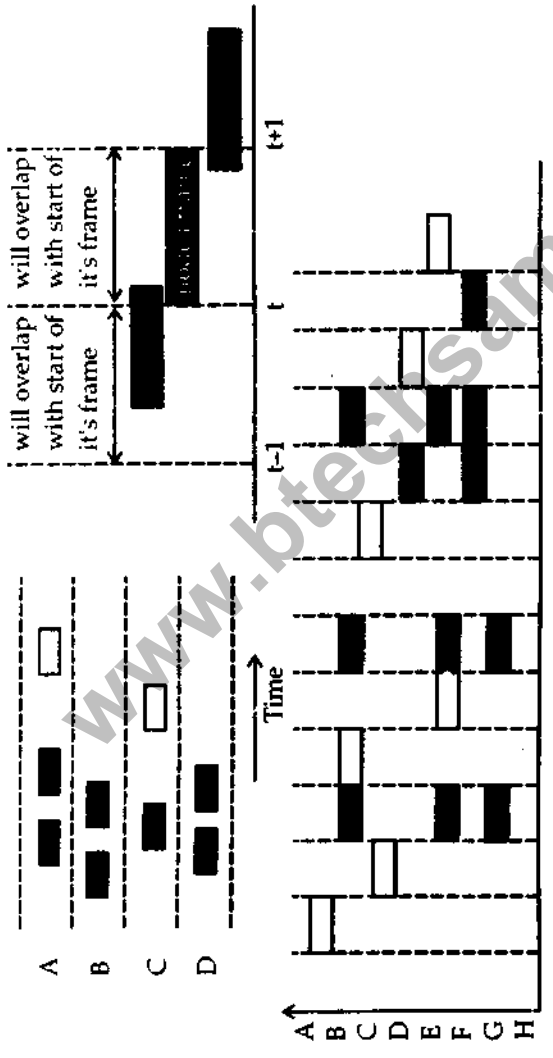
Q.2.(d).(i) Why is the channel throughput doubled in slotted ALOHA compared to pure ALOHA?

(ii) A stop-and-wait protocol uses a 100 kbps satellite link which employs a round-trip propagation delay of 250 ms approximately. Find out the percentage of time the sender is blocked to wait for acknowledgement, if the frame size is 1000 bits.

Ans. Pure Aloha had a maximum throughput of about 18.4%. This means that about 81.6% of the total available bandwidth was essentially wasted due to losses from packet collisions. The basic throughput calculation involves the assumption that the aggregate arrival process follows a Poisson distribution with an average number of arrivals of $2G$ arrivals per $2X$ seconds. Therefore, the lambda parameter in the Poisson distribution becomes $2G$. The mentioned peak is reached for $G = 0.5$ resulting

in a maximum throughput of 0.184, i.e. 18.4%.

An improvement to the original Aloha protocol was Slotted Aloha, which introduced discrete timeslots and increased the maximum throughput to 36.8%. A station can send only at the beginning of a timeslot, and thus collisions are reduced. In this case, the average number of aggregate arrivals is G arrivals per $2X$ seconds. This leverages the lambda parameter to be G. The maximum throughput is reached for $G = 1$.



(ii). According to question

Data rate, $r = 100$ Kbps

$= 100 \times 10^3$ bps

Rand trip propagation delay,

$RTT = 250$ ms $= 250 \times 10^{-3}$ sec.

Frame size = 1000 bits

Percentage of time and sender blocked= 0

$$= \frac{RTT}{Total\ Time} \times 100 = \frac{RTT}{RTT + Transmission\ Delay}$$

$$= \frac{250\times10^{-3}}{250\times10^{-3} + \frac{1000}{100\times10^{3}}} = \frac{250\times10^{-3}}{(250+10)10^{-3}}$$

$$= \frac{250\times10^{-3}}{260\times10^{-3}} = \frac{25}{26} = 0.961 = 96.1\%$$ **Ans.**

Q. 2(e). How do contention-free protocols differ from contention-oriented protocols? Discuss any one protocol of contention-free category.

Ans. Contention free protocols Vs Contention oriented protocols: A contention free protocol is one that guarantees that only one host may attempt transmission at a given time. By slicing time and specifying that a unique host may transmit, collisions become impossible. The token bus (802.4) protocol is an example of a contention free protocol, as only the host holding the token may transmit. A lightly loaded contention-free network is going to have higher wait times than necessary, and thus slightly reduced utilization. For heavy loads, however, contention-free protocols are ideal, as utilization becomes nearly optimal, and the wait time is lower since collision management does not pollute the traffic.

A contention oriented protocol may allow

only a subset of the hosts to transmit at a given time. Given high network load or collisions, such a protocol may divide time such that certain hosts may only attempt transmissions during certain time slices. The adaptive tree walk protocol is a limited contention protocol, where the response to a collision requires that first one half of the hosts be prevented from attempting transmission, and then (perhaps) the other half. A limited contention protocol exhibits low wait times and high utilization when the load is low, and adapts its behavior to reduce collisions when the load is high, thus keeping wait times as low as possible and maintaining high utilization.

The most common and major error was to believe a contention-based network is one that did not resolve or manage collisions at all. It is important to notice that a collision-based network is one that never differentiates between (and thus restricts) hosts in collision resolution. Often, this mistake was coupled with the erroneous claim that Ethernet (802.3) was an example of a limited contention protocol.

Q. 2(f). How is bridge different from a repeater? What are the advantages and disadvantages of each? State with reason whether an LAN can be extended to any size by increasing the number of repeaters or not?

Ans. Bridge different from Repeater: Bridge connects multiple network segments at the data link layer (layer 2) of the OSI model, and the term layer 2 switch is very often used interchangeably with bridge. Bridges are similar to repeaters or network hubs, devices that connect network segments at the physical layer; however with bridging, traffic from one network is managed rather than simply rebroadcast to adjacent network segments. In Ethernet networks, the term "bridge" formally means a device that behaves according to the IEEE 802.1D standard—this is most often referred to as a network switch in marketing literature. Bridges tend to be more complex than hubs or repeaters due to the fact that bridges are capable of analyzing incoming data packets on a network to determine if the bridge is able to send the given packet to another segment of that same network.

Part-ii Advantages of bridges

- Self configuring
- Primitive bridges are often inexpensive
- Reduce size of collision domain by microsegmentation in non switched networks
- Transparent to protocols above the MAC layer
- Allows the introduction of management - performance information and access control
- LANs interconnected are separate and physical constraints such as number of stations, repeaters and segment length don't apply
- it also helps minimize high bandwidth

Disadvantages of bridges

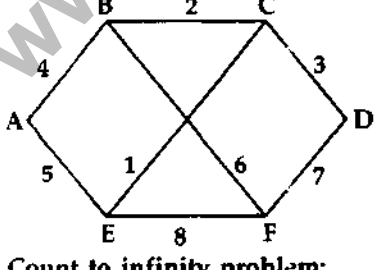
- Does not limit the scope of broadcasts
- Does not scale to extremely large networks
- Buffering introduces store and forward delays - on average traffic destined for bridge will be related to the number of stations on the rest of the LAN
- Bridging of different MAC protocols introduces errors

- Because bridges do more than repeaters by viewing MAC addresses, the extra processing makes them slower than repeaters
- Bridges are more expensive than repeaters

Part-III: repeater is the simplest multi-port active device in use. It has multiple input/output (I/O) ports, in which a signal introduced at the input of any port appears at the output of every port except (usually) the original incoming. With multiple computers, the network slows, due to packet collisions. A multiport repeater usually performs regenerative functions, i.e., it reshapes the digital signals. Therefore, we cannot extend beyond the certain limit.

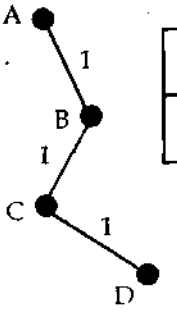
Q. 3. Attempt any two parts: 10×2

Q.3.(a). (i) What is count-to-infinity problem? How is it addressed in link sate routing protocol? For fol lowing subnet, distance vector routing is used and the vectors that have just come in to router C: from B: (5, 0, 8, 12, 6, 2); from D: (16, 12, 6, 0, 9, 10); and from E: (7, 6, 3, 9, 0, 4). The measured delays to B, D, and E are 6, 3, and 5 respectively. What is C's new routing table? Give both the outgoing line to use and the expected delay.



Ans. Count to infinity problem:

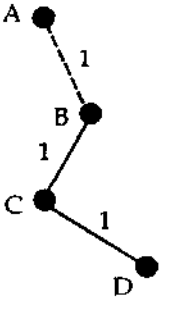
Followed illustration shows an imagined network and denotes the distances from router A to every other router. Until now every thing works fine.



B	C	D
1	2	3

The illustration shows that link (A, B) is broken. Router B observed it, but in his routing table he sees, that router C has a route to A with 2 hops. The problem is, that router B doesn't know that C has router B as successor in his routing table on the route to A. That occurs followed count-to-infinity problem. B actualizes his routing table and takes the route to A over router C.

In the next picture, we can see the new distances to A. In C's routing the route to A contains router B as next hop router, so if B has increase his costs to A, C is forced to do so. Router C increases his cost to A about B + 1 = 4. Now we see the consequence of the distributed Bellman-Ford protocol: Because router B takes the path over C to A, he reactualizes his routing table and so on! At the end this problem is going to immobilize the whole network.



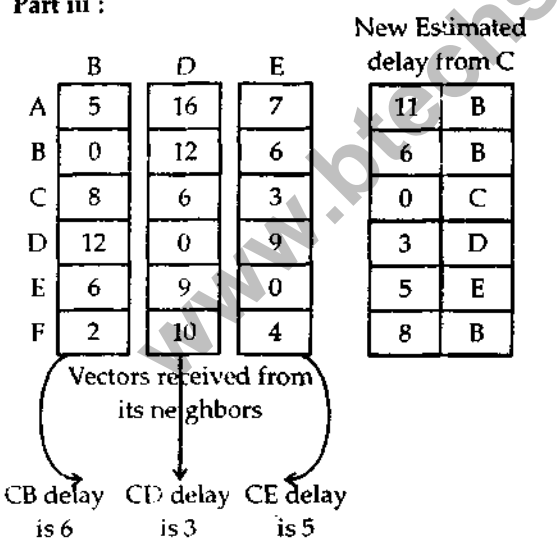
B	C	D
3	2	3

B	C	D
3	4	3

B	C	D
5	4	5

Part ii : The link-state protocol is performed by every *switching node* in the network (i.e. nodes which are prepared to forward packets; in the Internet, these are called routers). The basic concept of link-state routing is that every node constructs a *map* of the connectivity of the network, in the form of a graph showing which nodes are connected to which other nodes. The link-state routing requires each switching node in the network to send its information about its neighbours to the entire internetwork. Each node then independently calculates the best *next hop* from it for every possible destination in the network. (It does this using only its local copy of the map, and without communicating in any other way with any other node.) The collection of best next hops forms the routing table for the node.

Part iii :



Now,

For A: CA via B = CB + BA = 6 + 5 = 11
CA via D = CD + 16 = 3 + 16 = 19
CA via E = CE + EA = 5 + 7 = 12
Thus CA via B has minimum value = 11

For B: CB via B = CB + BB = 6 + 0 = 6
CB via D = CD + DB = 3 + 12 = 15
CB via E = CE + EA = 5 + 6 = 11
Thus CB via B has minimum value = 6

For D: CD via B = CB + BD = 6 + 12 = 18
CD via D = CD + DD = 3 + 0 = 3
CD via E = CE + ED = 5 + 9 = 14
Thus CD via D has minimum value = 3

For E: CE via B = CB + BE = 6 + 6 = 12
CE via D = CD + DE = 3 + 9 = 12
CE via E = CE + EE = 5 + 0 = 5
Thus CE via E has minimum value = 5

For F: CF via B = CB + BF = 6 + 2 = 8
CF via D = CD + DF = 3 + 10 = 13
CF via E = CE + EF = 5 + 4 = 9
Thus CF via B has minimum value = 8

(ii) What is fragmentation? Why do we need it? Discuss prps and cons of transparent and non-transparent fragmentation.

Ans. Fragmentation & Pro's & Con's:

The Internet Protocol allows IP fragmentation so that datagrams can be fragmented into pieces small enough to pass over a link with a smaller MTU than the original datagram size.

The Identification field, and Fragment offset field along with Don't Fragment and More Fragment Flags are used for Fragmentation and Reassembly of IP datagrams.

In a case where a router in the network receives a PDU larger than the next hop's MTU, it has two options. Drop the PDU and send an ICMP message which says "Packet too Big", or to Fragment the IP packet and send over the link with a smaller MTU.

If a receiving host receives an IP packet which is fragmented, it has to reassemble the IP packet

and hand it over to the higher layer. Reassembly always happens only in the receiving host.

The details of the fragmentation mechanism, as well as the overall architectural approach to fragmentation, are different in IPv4, the current version of the Internet Protocol, and IPv6, the newer version. In IPv4 routers do the fragmentation, whereas in IPv6, routers do not fragment, but drop the packets that are larger than the MTU size. Though the header formats are different for IPv4 and IPv6, similar fields are used for fragmentation, so the algorithm can be reused for fragmentation and reassembly.

IP fragmentation can cause excessive retransmissions when fragments encounter packet loss and reliable protocols such as TCP must retransmit all of the fragments in order to recover from the loss of a single fragment. Thus senders typically use two approaches to decide the size of IP datagrams to send over the network. The first is for the sending host to send an IP datagram of size equal to the MTU of the first hop of the source destination pair. The second is to run the "Path MTU discovery" algorithm, to determine the path MTU between two IP hosts, so that IP fragmentation can be avoided.

Q.3.(b)(i). Explain how do ARP and RARP map IP addresses onto data link layer such as Ethernet?

Ans. ARP Vs RARP: Address Resolution Protocol (ARP) is the method for finding a host's link layer (hardware) address when only its Internet Layer (IP) or some other Network Layer address is known. ARP has been implemented in many types of networks; it is not an IP-only or Ethernet-only protocol. It can be used to resolve many different network layer protocol addresses to interface hardware addresses, although, due to the overwhelming prevalence of IPv4 and Ethernet, ARP is primarily used to translate IP addresses to Ethernet MAC addresses. It is also used for IP over other LAN technologies, such as Token Ring, FDDI, or IEEE 802.11, and for IP over ATM.

Reverse Address Resolution Protocol (RARP) is a Link layer networking protocol used by a host computer to obtain its IPv4 address given only its link-layer address (such as an Ethernet address). RARP is described in IETF publication RFC 903. It has been rendered obsolete by Bootstrap Protocol and the modern Dynamic Host Configuration Protocol, which both support a much greater feature set than RARP.

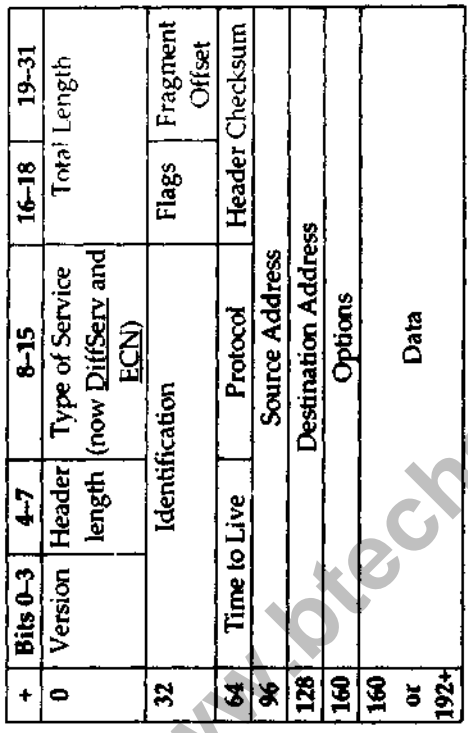
RARP requires one or more server hosts to maintain a database of mappings from Link Layer address to protocol address. MAC addresses needed to be individually configured on the servers by an administrator. RARP was limited, with respect to newer configuration protocols, to serving IP addresses only.

Reverse ARP differs from the Inverse Address Resolution Protocol (InARP, RFC 2390), which is designed to locate the IP address associated with another station's MAC address. InARP is the complement of the Address Resolution Protocol used for the reverse lookup. RARP was only used for lookup of a host's own IP address.

- Dynamic Host Configuration Protocol (DHCP)
- Bootstrap protocol (BOOTP)
- Address Resolution Protocol (ARP)
- Maintenance Operations Protocol (MOP)

(ii) Sketch the IP header neatly and explain the function of each field. List major differences between IPv4 and IPv6.

Ans. IP Header: **The header consists of 13 fields, of which only 12 are required. The 13th field is optional (red background in table) and aptly named: options.**



+	Bits 0–3	4–7	8–15	16–18	19–31
0	Version	Header length	Type of Service (now DiffServ and ECN)	Total Length	
32	Identification			Flags	Fragment Offset
64	Time to Live		Protocol	Header Checksum	
96	Source Address				
128	Destination Address				
160	Options				
160 or 192+	Data				

Version

The first header field in an IP packet is the four-bit version field. For IPv4, this has a value of 4 (hence the name IPv4).

Internet Header Length (IHL)

The second field (4 bits) is the Internet Header Length (IHL) telling the number of 32-bit words in the header. Since an IPv4 header may contain a variable number of options, this field specifies the size of the header (this also coincides with the offset to the data). The minimum value for this field is 5 , which is a length of 5×32 = 160 bits. Being a 4-bit value, the maximum length is 15 words or 480 bits.

Type of Service (TOS)

The following eight bits were allocated to a Type of Service (TOS) field:

- bits 0–2: Precedence (111 - Network Control, 110 - Internetwork Control, 101 - CRITIC/ECP, 100 - Flash Override, 011 - Flash, 010 - Immediate, 001 - Priority, 000 - Routine)
- bit 3: 0 = Normal Delay, 1 = Low Delay
- bit 4: 0 = Normal Throughput, 1 = High Throughput
- bit 5: 0 = Normal Reliability, 1 = High Reliability
- bits 6–7: Reserved for future use

This field is now used for DiffServ and ECN. The original intention was for a sending host to specify a preference for how the datagram would be handled as it made its way through an internet.

Total Length

This 16-bit field defines the entire datagram size, including header and data, in bytes. The minimum-length datagram is 20 bytes (20-byte header + 0 bytes data) and the maximum is 65,535 — the maximum value of a 16-bit word. The minimum size datagram that any host is required to be able to handle is 576 bytes, but most modern hosts handle much larger packets.

Identification

This field is an identification field and is primarily used for uniquely identifying fragments of an original IP datagram. Some experimental work has suggested using the ID

field for other purposes, such as for adding packet-tracing information to datagrams in order to help trace back datagrams with spoofed source addresses.

Flags

A three-bit field follows and is used to control or identify fragments. They are (in order, from high order to low order):

- Reserved; must be zero. As an April Fools joke, proposed for use in RFC 3514 as the "Evil bit".
- Don't Fragment (DF)
- More Fragments (MF)

If the DF flag is set and fragmentation is required to route the packet then the packet will be dropped. This can be used when sending packets to a host that does not have sufficient resources to handle fragmentation.

When a packet is fragmented all fragments have the MF flag set except the last fragment, which does not have the MF flag set. The MF flag is also not set on packets that are not fragmented -- an unfragmented packet is its own last fragment.

Fragment Offset

The fragment offset field, measured in units of eight-byte blocks, is 13 bits long and specifies the offset of a particular fragment relative to the beginning of the original unfragmented IP datagram. The first fragment has an offset of zero. This allows a maximum offset of 65,528 [$(2^{13} - 1) \times 8$]which would exceed the maximum IP packet length of 65,535 with the header length included.

Time To Live (TTL)

An eight-bit time to live (TTL) field helps prevent datagrams from persisting (e.g. going in circles) on an internet. Historically the TTL field limited a datagram's lifetime in seconds, but has come to be a hop count field. Each packet switch (or router) that a datagram crosses decrements the TTL field by one. When the TTL field hits zero, the packet is no longer forwarded by a packet switch and is discarded. Typically, an ICMP message (specifically the time exceeded) is sent back to the sender that it has been discarded. The reception of these ICMP messages is at the heart of how traceroute works.

Protocol

This field defines the protocol used in the data portion of the IP datagram. The Internet Assigned Numbers Authority maintains a list of Protocol numbers and were originally defined in RFC 790. Common protocols and their decimal values are shown below (*see Data*).

Header Checksum

The 16-bit checksum field is used for error-checking of the header. At each hop, the checksum of the header must be compared to the value of this field. If a header checksum is found to be mismatched, then the packet is discarded. Note that errors in the data field are up to the encapsulated protocol to handle — indeed, both UDP and TCP have checksum fields.

Since the TTL field is decremented on each hop and fragmentation is possible at each hop then at each hop the checksum will have to be recomputed. *The checksum field is the 16-bit one's complement of the one's complement sum of all 16-bit words in the header. For purposes of computing the checksum, the value of the checksum field is zero.*

Source address

An IPv4 address is a group of four eight-bit octets for a total of 32 bits. The value for this field is determined by taking the binary value of each octet and concatenating them together to make a single 32-bit value.

For example, the address 10.9.8.7 (00001010.00001001.00001000.00000111 in binary) would be 00001010000010010000 100000000111.

This address is the address of the sender of the packet. Note that this address may not be the "true" sender of the packet due to network address translation. Instead, the source address will be translated by the NATing machine to its own address. Thus, reply packets sent by the receiver are routed to the NATing machine, which translates the destination address to the original sender's address.

Destination address

Identical to the source address field but indicates the receiver of the packet.

Options

Additional header fields may follow the destination address field, but these are not often used. Note that the value in the IH L field must include enough extra 32-bit words to hold all the options (plus any padding needed to ensure that the header contains an integral number of 32-bit words). The list of options may be terminated with an EOL (End of Options List, 0x00) option; this is only necessary if the end of the options would not otherwise coincide with the end of the header. The possible options that can be put in the header are as follows:

Field	Size (bits)	Description
Copied	1	Set to 1 if the options need to be copied into all fragments of a fragmented packet.
Option Class	2	A general options category. 0 is for "*control*" options, and 2 is for "*debugging and measurement*". 1, and 3 are reserved.
Option Number	5	Specifies an option.
Option Length	8	Indicates the size of the entire option (including this field). This field may not exist for simple options.
Option Data	Variable	Option-specific data. This field may not exist for simple options.

Part-ii IPv6 has a number of new features designed to address the shortcomings of IPv4, including a new IP header format, a larger address space, a more efficient routing infrastructure, stateless and stateful address configurations, enhanced security, and standardised QoS support.

The new header format

The first notable feature of the IPv6 protocol is a newly designed IP header. It's designed to make the protocol more efficient by keeping overhead to a minimum. An IP packet header is made up of required components and optional components; in IPv6, the required components are moved to the front of the header. Optional components are moved to an extension header. This means that if optional components aren't used, the extension headers aren't necessary, reducing the packet size.

The downside to the new header is that it isn't compatible with IPv4. If a router is to handle both IPv4 and IPv6, it must be configured to recognise both protocols. You can't just set up a router to recognise IPv6 and expect it to be backward-compatible with IPv4.

Larger address space

Perhaps the most compelling reason for moving to IPv6 is the supposed shortage of IP addresses. IPv6 uses 128-bit source and destination addresses. There are theoretically over 3.4x10^38 possible addresses that can be addressed by the IPv6 protocol. Furthermore, this new structure allows for more levels of subnetting than are available with IPv4. Some people speculate that because of the large number of addresses that IPv6 allows, NAT technology may soon become a thing of the past.

More efficient routing

The Internet is hierarchical in nature, and the IPv6 protocol is designed with this in mind. Think about it. The computer you're using right now doesn't have a direct connection to an Internet backbone. Instead, you're probably behind a NAT firewall, which is connected to an ISP. That ISP may be connected to another ISP or to a backbone router. Either way, a packet must make quite a few hops to go from an Internet backbone router to you.

New configuration options

One of the coolest things about IPv6 is the way it's configured. While you can still manually configure IPv6, or lease an address from a DHCP server, there is a new automatic configuration option available. If an unconfigured PC tries to connect to a network that doesn't offer a DHCP server, the PC can look at either the network's router or the other PCs on the network and determine an address that would be appropriate for it to use. This technique is referred to as link local addressing.

Integrated security

IPSec is available in some implementations of IPv4, but it's completely integrated into IPv6. Any computer that's running IPv6 will support IPSec encryption, regardless of the computer's operating system.

Standardised QoS support

IPv6 also includes standardised support for QoS. The QoS implementation is set up so that routers can identify packets belonging to an individual QoS flow. This allows those routers to allocate the necessary amount of bandwidth to those packets. Furthermore, QoS instructions are included in the IPv6 packet header. This means that the packet body can be encrypted, but QoS will still function because the header portion containing the QoS instructions is not encrypted. This will make it possible to send streaming audio and video over the Internet with IPSec encryption, but in a manner that guarantees adequate bandwidth for real-time playback.

Get ready to move

IPv6 is a huge improvement over IPv4. You should expect many ISPs to start supporting both IPv4 and IPv6 as more demand is made for IPv6. The IPv4 protocol will be gradually phased out, but this may take some time.

Q.3.(c)(i). Explain token bucket algorithm. What problems of leaky bucket algorithm are addressed by it?

Ans. Token bucket algorithm: The algorithm can be conceptually understood as follows:

- A token is added to the bucket every $1 / r$ seconds.
- The bucket can hold at the most b tokens. If a token arrives when the bucket is full, it is

discarded.

- When a packet (network layer PDU) of n bytes arrives, n tokens are removed from the bucket, and the packet is sent to the network.
- If fewer than n tokens are available, no tokens are removed from the bucket, and the packet is considered to be *non-conformant*.

The algorithm allows bursts of up to b bytes, but over the long run the output of conformant packets is limited to the constant rate, r. Non-conformant packets can be treated in various ways:

- They may be dropped.
- They may be enqueued for subsequent transmission when sufficient tokens have accumulated in the bucket.
- They may be transmitted, but marked as being non-conformant, possibly to be dropped subsequently if the network is overloaded.

To calculate the time for which the Token Bucket Algorithm allows burst of maximum possible size, assume that the capacity of the Token Bucket is C bytes, the token arrival rate if R bytes/second and the maximum possible transmission rate is M bytes/second and S is the number of seconds for which it is possible to transmit at maximum rate. Then, the following equality holds $C + R * S = M * S$ which gives $S = C / (M - R)$ seconds

Implementers of this algorithm on platforms lacking the clock resolution necessary to add a single token to the bucket every $1 / r$ seconds may want to consider an alternative formulation. Given the ability to update the token bucket every S milliseconds, the number of tokens to add every S milliseconds = $(r * S) / 1000$.

Part-ii a leaky bucket implementation and a token bucket implementation. Sometimes they are mistakenly lumped together under the same name. Both these schemes have distinct properties and are used for distinct purposes. They differ principally in that the leaky bucket imposes a hard limit on the data transmission rate, whereas the token bucket allows a certain amount of burstiness while imposing a limit on the average data transmission rate.

(ii) With the aid of an example, explain why subnetting was introduced. Hence state the meaning of a subnet router and an address mask. What is the maximum number of hosts which a network on internet having a subnet mask of 255.255.240.0 can handle?

Ans. Subnetting: In a large organization's network, without subnetting, traffic levels can grow enough that excessive rates of Ethernet packet collisions become a bottle neck due to the nature of Ethernet (cf. carrier sense multiple access with collision detection). For this reason, subnetting can be used to break the network into smaller more efficient *subnets*. Such subnets can be arranged hierarchically, with the organization's network address space (see also Autonomous System) partitioned into a tree-like structure. Routers are used to manage traffic and constitute borders between subnets. Communication is of specific link-local character (Ethernet broadcast) only within the smallest subnet.

A typical subnet is a physical network served by one router, for instance an Ethernet network (consisting of one or several Ethernet segments or local area networks, interconnected by network switches and network bridges) or a Virtual Local Area Network (VLAN). However, subnetting allows the network to be logically divided regardless of the physical layout of a network, since it is possible to divide a physical

network into several subnets by configuring different host computers to use different routers.

While improving network performance, subnetting increases routing complexity, since each locally connected subnet is typically represented by one row in the routing tables in each connected router. However, with intelligent design of the network, routes to collections of more distant subnets within the branches of a tree-hierarchy can be aggregated by single routes.

Subnetting was originally conceived well before the introduction of classful network addresses in IPv4 to allow a single larger network to have a number of smaller networks within it, controlled by several routers. Existing subnetting functionality in routers made the introduction of Classless Inter-Domain Routing seamless.

Part ii: Given subnet mask = 255·255·240·0

Mask in binary = 11111111 · 11111111 ·
10001100 · 00000000

New mask complement
= 00000000 · 00000000 ·
01110011 · 11111111

Thus, number of mask = 115 + 255 + 1
= 377 **Ans.**

Q. 4. **Attempt any two parts:** **10×2**

Q.4(a). Draw the diagram of TCP header and explain the use of following:

(i) Source and destination port addresses

(ii) Sequence and acknowledgement numbers

(iii) Code bits

(iv) Window size

(v) Urgent pointer

Describe the role of checksum field and optional pad bytes.

Ans. The TCP header consists of 11 fields, of which only 10 are required. The eleventh field is optional (pink background in table) and aptly named "options".

TCP Header												
Bit offset	Bits 0–3	4–7	8–15								16–31	
0	Source port										Destination port	
32	Sequence number											
64	Acknowledgment number											
96	Data offset	Reserved	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN	Window Size	
128	Checksum										Urgent pointer	
160	Options (optional)											
160/192+	Data											

- Source port (16 bits) – identifies the sending port
- Destination port (16 bits) – identifies the receiving port
- Sequence number (32 bits) – has a dual role

- If the SYN flag is present, then this is the initial sequence number and the sequence number of the first data byte is this sequence number plus 1
- If the SYN flag is not present, then this is the sequence number of the first data byte
- Acknowledgement number (32 bits) – if the ACK flag is set then the value of this field is the next expected byte that the receiver is expecting.
- Data offset (4 bits) – specifies the size of the TCP header in 32-bit words. The minimum size header is 5 words and the maximum is 15 words thus giving the minimum size of 20 bytes and maximum of 60 bytes. This field gets its name from the fact that it is also the offset from the start of the TCP packet to the data.
- Reserved (4 bits) – for future use and should be set to zero
- Flags (8 bits) (aka Control bits) – contains 8 1-bit flags
- CWR (1 bit) – Congestion Window Reduced (CWR) flag is set by the sending host to indicate that it received a TCP segment with the ECE flag set (added to header by RFC 3168).
- ECE (ECN-Echo) (1 bit) – indicate that the TCP peer is ECN capable during 3-way handshake (added to header by RFC 3168).
- URG (1 bit) – indicates that the URGent pointer field is significant
- ACK (1 bit) – indicates that the ACKnowledgment field is significant
- PSH (1 bit) – Push function
- RST (1 bit) – Reset the connection
- SYN (1 bit) – Synchronize sequence numbers
- FIN (1 bit) – No more data from sender
- Window (16 bits) – the size of the receive window, which specifies the number of bytes (beyond the sequence number in the acknowledgment field) that the receiver is currently willing to receive (*see Flow control*)
- Checksum (16 bits) – The 16-bit checksum field is used for error-checking of the header *and data*
- Urgent pointer (16 bits) – if the URG flag is set, then this 16-bit field is an offset from the sequence number indicating the last urgent data byte
- Options (Variable bits) – the total length of the option field must be a multiple of a 32-bit word and the data offset field adjusted appropriately
- 0 - End of options list
- 1 - No operation (NOP, Padding)
- 2 - Maximum segment size (*see maximum segment size*)
- 3 - Window scale (*see window scaling for details*)
- 4 - Selective Acknowledgement ok (*see selective acknowledgments for details*)
- 5 -
- 6 -
- 7 -
- 8 - Timestamp

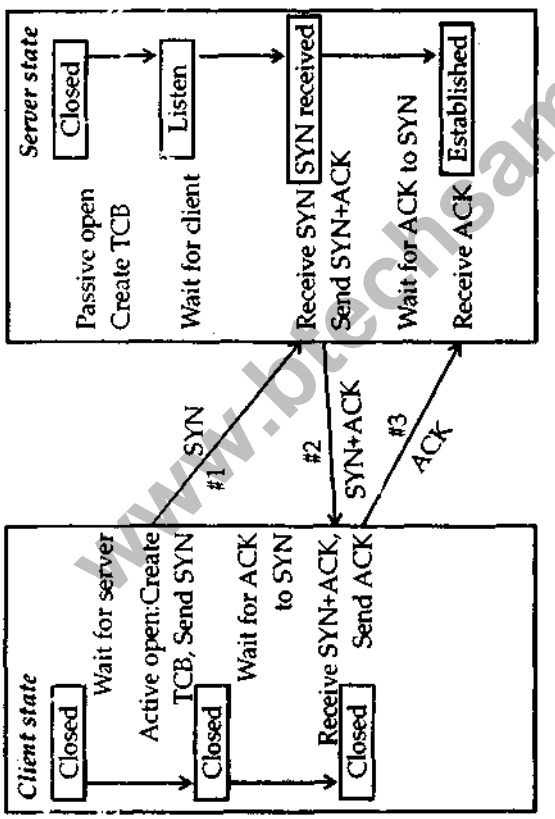
The last field is not a part of the header. The contents of this field are whatever the upper layer protocol wants but this protocol is not set in the header and is presumed based on the port selection.

Data (Variable bits): As you might expect, this

is the payload, or data portion of a TCP packet. The payload may be any number of application layer protocols. The most common are HTTP, Telnet, SSH, FTP, but other popular protocols also use TCP.

Q.4.(b). With the aid of a time sequence diagram, explain how a logical connection between two TCP entities is established using three-way handshake procedure. Include in your diagram the socket primitives at both the client and server side that trigger the sending of each segment. Also explain how the initial sequence number in each direction is seleted.

Ans. Three Way Handshake:

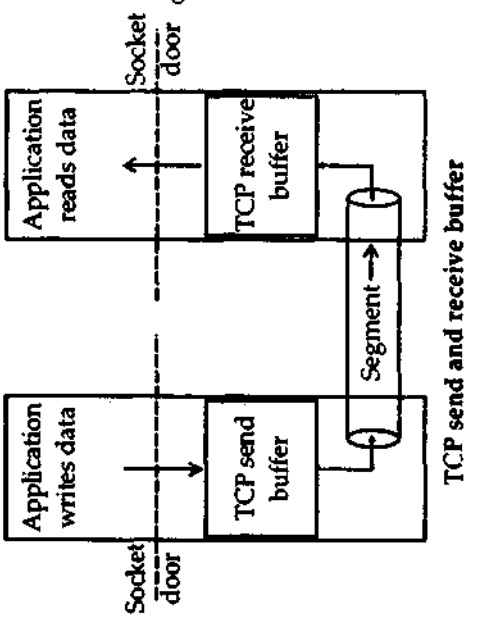


TCP "Three-Way Handshake" Connection Establishment Procedure

This diagram illustrates how a conventional connection is established between a client and server, showing the three messages sent during the process and how each device transitions from the *CLOSED* state through intermediate states until the session is *ESTABLISHED*.

Elements of Transport Protocol:

TCP provides a connection oriented, reliable, byte stream service. The term connection-oriented means the two applications using TCP must establish a TCP connection with each other before they can exchange data. It is a full duplex protocol, meaning that each TCP connection supports a pair of byte streams, one flowing in each direction. TCP includes a flow-control mechanism for each of these byte streams that allows the receiver to limit how much data the sender can transmit. TCP also implements a congestion-control mechanism.



TCP encapsulates each chunk of client data with TCP header, thereby forming TCP

segments. The segments are passed down to the network layer, where they are separately encapsulated within network-layer IP datagrams. The IP datagrams are then sent into the network. When TCP receives a segment at the other end, the segment's data is placed in the TCP connection's receive buffer. The application reads the stream of data from this buffer. Each side of the connection has its own send buffer and its own receive buffer. The send and receive buffers for data flowing in one direction are shown in Figure.

We see from this discussion that a TCP connection consists of buffers, variables and a socket connection to a process in one host, and another set of buffers, variables and a socket connection to a process in another host. As mentioned earlier, no buffers or variables are allocated to the connection in the network elements (routers, bridges and repeaters) between the hosts.

Applications send streams of octets (8 Bit or "Byte") to TCP for delivery through the network, and TCP divides the byte stream into appropriately sized segments (usually delineated by the maximum transmission unit (MTU) size of the data link layer of the network to which the computer is attached). TCP then passes the resulting packets to the Internet Protocol, for delivery through a network to the TCP module of the entity at the other end. TCP checks to make sure that no packets are lost by giving each packet a sequence number, which is also used to make sure that the data are delivered to the entity at the other end in the correct order. The TCP module at the far end sends back an acknowledgement for packets which have been successfully received; a timer at the sending TCP will cause a timeout if an acknowledgement is not received within a reasonable round-trip time (or RTT), and the (presumably lost) data will then be re-transmitted. The TCP checks that no bytes are damaged by using a checksum; one is computed at the sender for each block of data before it is sent, and checked at the receiver.

Q.4.(c)(i). What is cryptography? Distinguish between symmetric and asymmetric key cryptography.

Ans. Cryptography: Cryptography (or cryptology; derived from Greek κρύπτω *krypto* "hidden" and the verb γράφω *gráfo* "to write" or λέγειν *legein* "to speak") is the practice and study of hiding information. In modern times, cryptography is considered to be a branch of both mathematics and computer science, and is affiliated closely with information theory, computer security, and engineering. Cryptography is used in applications present in technologically advanced societies; examples include the security of ATM cards, computer passwords, and electronic commerce, which all depend on cryptography. So, we can say cryptography provides technique for providing network security.

Part-ii Symmetric-key cryptography refers to encryption methods in which both the sender and receiver share the same key (or, less commonly, in which their keys are different, but related in an easily computable way).

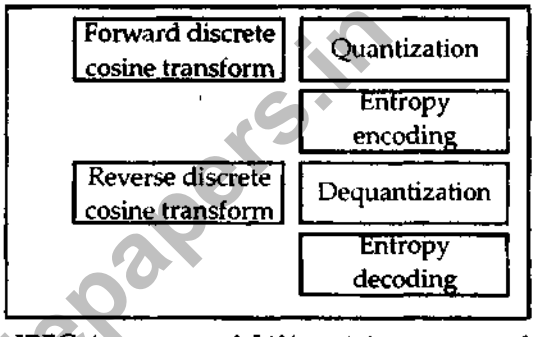
Symmetric-key cryptosystems use the same

key for encryption and decryption of a message, though a message or group of messages may have a different key than others. A significant disadvantage of symmetric ciphers is the key management necessary to use them securely. Each distinct pair of communicating parties must, ideally, share a different key, and perhaps each ciphertext exchanged as well. The number of keys required increases as the square of the number of network members, which very quickly requires complex key management schemes to keep them all straight and secret. The difficulty of securely establishing a secret key between two communicating parties, when a secure channel doesn't already exist between them.

In addition to encryption, public-key cryptography can be used to implement digital signature schemes. A digital signature is reminiscent of an ordinary signature; they both have the characteristic that they are easy for a user to produce, but difficult for anyone else to forge. Digital signatures can also be permanently tied to the content of the message being signed; they cannot then be 'moved' from one document to another, for any attempt will be detectable. In digital signature schemes, there are two algorithms: one for *signing*, in which a secret key is used to process the message (or a hash of the message, or both), and one for *verification*, in which the matching public key is used with the message to check the validity of the signature. RSA and DSA are two of the most popular digital signature schemes. Digital signatures are central to the operation of public key infrastructures and many network security schemes.

(ii). Discuss different steps of JPEG compression standard.

Ans. STEPS of JPEG Compression Standards:



JPEG (pronounced JAY-peg) is a commonly used standard method of compression for photographic images. The name JPEG stands for Joint Photographic Experts Group, the name of the joint ISO/CCITT committee which created the standard.

JPEG provides for lossy compression of images (although there are variations on the standard baseline JPEG which are lossless). The file format which employs this compression is commonly also called JPEG, the most common file extension for this format is .jpg, though .jpeg, .jfif, .JPG, and .JPE are also used.

JPEG itself specifies only how an image is transformed into a stream of bytes, not how those bytes are encapsulated in any particular storage medium. A further standard created by the Independent JPEG Group, called JFIF (JPEG File Interchange Format), specifies how to produce a file suitable for computer storage and transmission (such as over the Internet) from a JPEG stream. In common usage, when one speaks of a "JPEG file" the actual file is generally found to be JFIF, or sometimes an Exif JPEG file. There are, however, other JPEG-based file formats, such as JNG. Additionally, the TIFF

format can carry JPEG data.

JPEG/JFIF is the format most used for storing and transmitting photographs on the World Wide Web. For this application, it is preferred to formats such as GIF, which has a limit of 256 distinct colors that is insufficient for colour photographs, and PNG, which produces much larger image files for this type of image. The compression algorithm is *not* as well suited for line drawings and other textual or iconic graphics, and thus the PNG and GIF formats are preferred for these types of images.

Q. 5. Attempt any *two* parts— (10×2=20)

Q.5.(a). Explain why a DNS is required with the Internet and describe its main functional parts. In relation to DNS, explain why a hierarchical naming structure is used instead of a flat structure?

Ans. Domain Name System (DNS) is a hierarchical naming system for computers, services, or any resource participating in the Internet. It associates various information with domain names assigned to such participants. Most importantly, it translates human meaningful domain names to the numerical (binary) identifiers associated with networking equipment for the purpose of locating and addressing these devices world-wide. An often used analogy to explain the Domain Name System is that it serves as the "phone book" for the Internet by translating human-friendly computer hostnames into IP addresses. For example, *www.example.com* translates to *208.77.188.166*.

GenericDomainNames sells domain name registrations with a premium service set included in the discount price of the registration. You will find our domain names to be one of the best values around.

We offer great versatility to each domain name through easy to use and powerful tools for beginner users and advanced users alike. Our domain names have all tools included! You do not pay extra like the other discount registrars for such services as email forwarding.

Below the top-level domains in the domain name hierarchy are the second-level domain (SLD) names. These are the names directly to the left of .com, .net, and the other top-level domains. As an example, in the domain *en.wikipedia.org*, *wikipedia* is the second-level domain.

Next are third-level domains, which are written immediately to the left of a second-level domain. There can be fourth- and fifth-level domains, and so on, with virtually no limitation. An example of a working domain with four domain levels is *www.sos.state.oh.us*. The www preceding the domains is a host name of the World-Wide Web server. Each level is separated by a dot, or period symbol. 'sos' is said to be a sub-domain of 'state.oh.us', and 'state' a sub-domain of 'oh.us', etc. In general, Sub-domains are domains subordinate to their parent domain. An example of very deep levels of subdomain ordering are the IPv6 reverse resolution DNS zones, e.g., 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ip6.arpa, which is the reverse DNS resolution domain for the IP address of a loopback interface, or the localhost name.

Q.5.(b). What is the function of SMTP? Differentiate between a user agent (UA) and a mail transfer agent? How does MIME enhance SMTP?

Ans. SMTP: Simple Mail Transfer Protocol (SMTP) is an Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks. SMTP is a relatively simple, text-based protocol, in which one or more recipients of a message are specified (and in most cases verified to exist) along with the message text and possibly other encoded objects. The message is then transferred to a remote server using a series of queries and responses between the client and server. Either

an end-user's e-mail client, a.k.a. MUA (*Mail User Agent*), or a relaying server's MTA (Mail Transport Agents) can act as an *SMTP client*.

Part-ii

1. Simple Mail Transfer Protocol (SMTP) - which is used with the TCP/IP protocol suite? It has traditionally been limited to the text based electronic messages.
2. Multipurpose Internet Mail Extension (MIME) - Which allows the transmission and reception of mail that contains various types of data, such as speech, images, and motion video? It is a newer standard than STMP and uses much of its basic protocol.
3. S/MIME (Secure MIME). RSA Data security created S/MIME which supports encrypted e-mail transfer and digitally signed electronic mail.

A typical email-architecture contains four elements:

1. Post offices- where outgoing messages are temporally buffered (stored) before transmission and where incoming messages are stored. The post office runs the server software capable of routing messages (a message transfer agent) and maintaining the post office database.

2. Message transfer agents- for forwarding messages between post offices and to the destination clients. The software can either reside on the local post office or on a physically separate server.

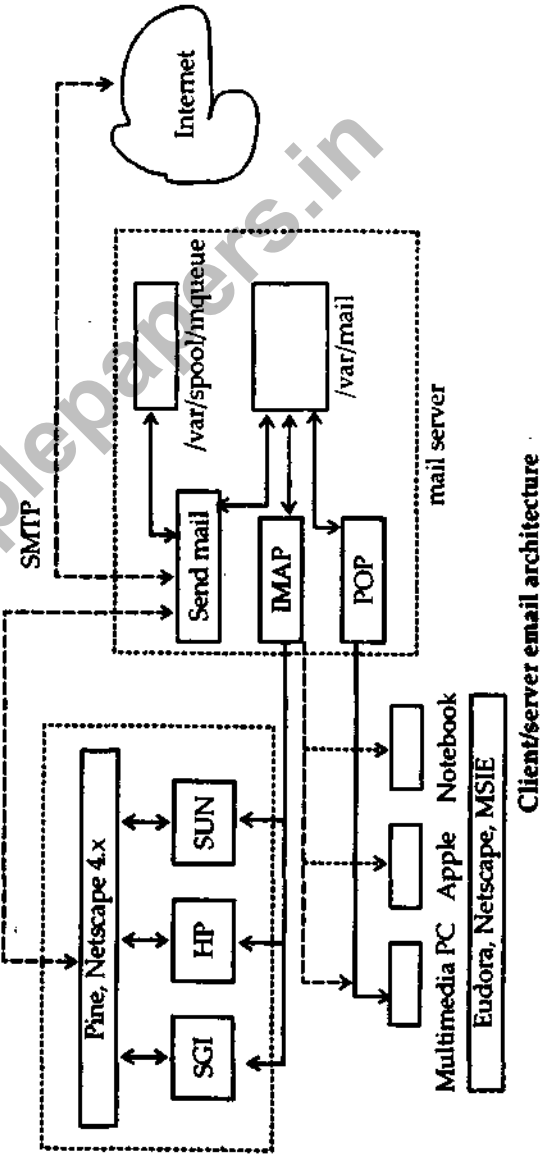
3. Gateways-which provide parts of the message transfer agent functionally. They translate between different e-mail systems, different e-mail addressing schemes and messaging protocols.

4. E-mail clients- normally the computer which connects to the post office. It contains three parts:

E-mail Application Program Interface (API), such as MAPI, VIM, MHS and CMC.

Messaging protocol. The main messaging protocols are SMTP or X.400. Where as x.400 is an OSI-defined e-mail message delivery standard.

Network transport protocol, such as Ethrnet, FDDI, and so on.



Part-iii : Multipurpose Internet Mail Extensions (MIME) is an Internet standard that extends the format of e-mail to support:

- text in character sets other than ASCII;
- non-text attachments;

• message bodies with multiple parts

• header information in non-ASCII character sets.

MIME's use, however, has grown beyond describing the content of e-mail to describing content type in general.

Virtually all human-written Internet e-mail and a fairly large proportion of automated e-mail is transmitted via SMTP in MIME format. Internet e-mail is so closely associated with the SMTP and MIME standards that it is sometimes called SMTP/MIME e-mail.

The content types defined by MIME standards are also of importance outside of e-mail, such as in communication protocols like HTTP for the World Wide Web. HTTP requires that data be transmitted in the context of e-mail-like messages, even though the data may not actually be e-mail.

Q.5.(c). Show the working of RSA algorithm with suitable example.

Ans. The RSA algorithm involves three steps, key generation, encryption and decryption.

Key Generation

RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated the following way:

1. Choose two distinct large random p and q
2. Computer $n = pq$

   n is used as the modulus for both the public and private keys
3. Compute the totient: $\varphi(n) = (p - 1)(q - 1)$
4. Choose an integer e such that $1 < e < \varphi(n)$ and e and $\varphi(n)$ share no factors other than 1 (i.e. e and $\varphi(n)$ are coprime)

   o e is released as the public key exponent
5. Compute d to satisfy the congruence relation $de \equiv 1 \pmod{\varphi(n)}$ i.e. $de = 1 + k\varphi(n)$ for some integer k.

   d is kept as the private key exponent.

Notes on the above steps:

• **Step 1:** Numbers can be probabilistically tested for primality.

Step 2: changed in PKCS#1 v2.0 to $\lambda(n) = \text{lcm}(p - 1, q - 1)$.where lcm is the least common multiple, instead of $\varphi(n) = (p - 1)(q - 1)$.

Step 3: A popular choice for the public exponents is $e = 2^{16} + 1 = 65537$. Some applications choose smaller values such as $e = 3$, 5, 17 or 257 instead. This is done to make encryption and signature verification faster on small devices like smart cards but small public exponents can lead to greater security risks.

• Steps 4 and 5 can be performed with the extended Euclidean algorithm; see modular arithmetic.

The public key consists of the modulus n andthe public (or encryption) exponent e. The private key consists of the modulus n and the private (or decryption) exponent which must be kept secret.

• For efficiency a different form of the private key can be stored:

p **and** q: the primes from the key generation,

d mod (p – 1) and d mod (q – 1),

q^{-1} mod (p).

All parts of the private key must be kept secret in this form. p and q are are sensitive since they are the factors of n, and allow computation of d given e. If p and q are not stored in this form of the private key then they are securely deleted along with other intermediate values from key generation.

• Although this form allows faster decryption and signing by using the Chinese Remainder Theorem, it is considerably less secure since it enables side channel attacks. This is a particular problem if implemented on smart cards, which benefit most from the improved efficiency. (Start with $y = x^e mod n$ and let the card

decrypt that. So it computes $y^d \pmod p$ or $y^d \pmod q$ whose results give some value z. Now, induce an error in one of the computations. Then gcd(z - x,n) will reveal p or q.)

Encryption

Alice transmits her public key (n, e) to Bob and keeps the private key secret. Bob then wishes to send message M to Alice.

He first turns M into a number $m < n$ by using an agreed-upon reversible protocol known as a padding scheme. He then computes the ciphertext c corresponding to:

$$c \equiv m^e \bmod n$$

This can be done quickly using the method of exponentiation by squaring. Bob then transmits c to Alice.

Decryption

Alice can recover m from c by using her private key exponent d by the following computation:

$$m = c^d \bmod n$$

Given m she can recover the message M.

The above decryption procedure works because first $c^d \equiv (m^e)^d \equiv m^{ed} \pmod n$

Now, $ed \equiv 1 \pmod{(p-1)(q-1)}$, and hence

$ed \equiv 1 \pmod{p-1}$ and

$ed \equiv 1 \pmod{q-1}$

which can also be written as

$ed = k(p-1) + 1$ and $ed = h(q-1) + 1$

for proper values of k and h. If m is not a multiple of p then m and p are coprime because p is prime, so by Fermat's little theorem

$$m^{(p-1)} \equiv 1 \pmod p$$

and therefore, using the first expression for ed.

$m^{ed} = m^{k(p-1)+1} = (m^{p-1})^k m \equiv 1^k m = m \pmod p$

If instead m is a multiple of p, then

$m^{ed} = 0^{ed} = 0 \equiv m \pmod p$

Using the second expression for ed, we similarly conclude that

$m^{ed} \equiv m \pmod q$

Since p and q are distinct prime numbers, they are relatively prime to each other, so the fact that both primes divide $m^{ed} - m$ implies their product pq divides $m^{ed} - m$, which means

$$m^{ed} \equiv m \pmod{pq}$$

Thus,

$$c^d \equiv m \pmod n$$

A worked example

Here is an example of RSA encryption and decryption.

1. Choose two prime numbers.
   $p = 61$ and $q = 53$.
2. Compute $n = pq$.
   $n = 61*53 = 3233$
3. Compute the totient $\varphi(n) = (p-1)(q-1)$
   $\varphi(n) = (61-1)(53-1) = 3120$
4. Choose e > 1 coprime to 3120
   $e = 17$
5. Compute d such that $de \equiv 1 \pmod{\varphi(n)}$ e.g. by computing the modular multiplicative inverse of e modulo $\varphi(n)$.
   $d = 2753$
   $17 * 2753 = 46801 = 1 + 15*3120$

The public key is ($n = 3233$, $e = 17$). For a padded message m the encryption function is:

$c = m^e \bmod n = m^{17}$, mod 3233

The private key is (n = 3233, d = 2753). The decryption function is:

$m = c^d \bmod n = c^{2753} \bmod 3233$.

For example, to encrypt m = 123, we calculate

$c = 123^{17} \bmod 3233 = 855$

To decrypt $c = 855$, we calculate

$m = 855^{2753} \bmod 3233 = 123$