CHAPTER 10

COMPUTER NETWORKS

Syllabus: Computer networks: ISO/OSI stack, LAN technologies (Ethernet, token ring), Flow-and error-control techniques, Routing algorithms, Congestion control, TCP/UDP and sockets, IP(v4), Application-layer protocols (ICMP, DNS, SMTP, POP, FTP, HTTP); Basic concepts of hubs, switches, gateways and routers; Network security: basic concepts of public key and private key cryptography, digital signature, firewalls.

10.1 INTRODUCTION

Computer network is a collection of autonomous devices interconnected via a medium. The medium may be a guided medium, a wireless medium or a satellite communication. To understand data communication, different layered architectures have been presented. All layers execute different protocols to communicate the data successfully from one end to the other. The intermediary devices such as switch, hub, router or gateway help in advancing this communication. Although the security requirements may be different depending upon the application, network security cannot be left aside when studying networks.

10.2 NETWORK

Two or more devices connecting to each other through any medium forms a network. To connect the devices there are two possible ways:

1. **Point-to-point connection:** This provides a dedicated link between the two devices.
2. **Multipoint connection:** This is also known as multi-drop connection. In this case, channel capacity is shared by more than two devices. In general, this is used in practice. All the five examples in Fig. 10.1 are multipoint.

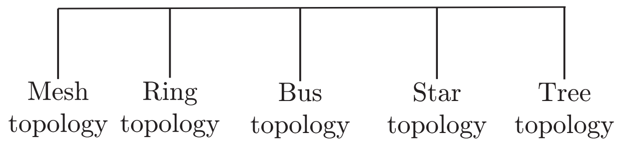


Figure 10.1 Network topology.

10.2.1 Network Topology

The physical or logical view of interconnection among devices is known as topology.

Ring and mesh topology are examples of peer-to-peer relationship because here all the devices share the link equally. Bus, star and tree topologies are examples of primary—secondary relationship, where one device controls and the other devices have to transmit through it.

10.2.1.1 Bus Topology

In a bus network, all the devices are connected with one cable. The major benefit here is, we require less cable length than star topology and expansion is quite easy with the help of repeaters.

The issues associated with bus topology are as follows:

1. Only one device can send the data at one time. All the devices will listen at that time.
2. The data communication is only in one direction.
3. In a bus topology, if the network shuts down then there is problem in identifying the culprit device.

10.2.1.2 Ring Topology

In a ring topology, each deviceis connected exactly to two devices to form the ring. Repeaters are used to regenerate and retransmit each bit. Data travels around the network in one direction. Data travels in the form of token. Additional components do not affect the performance of the network. Even if the load on the network increases, the performance of a ring topology is better than a bus topology. The problems may be listed as follows:

1. Failure of one computer in the ring may lead to entire communication loss.
2. Network scaling is difficult.

For example, token ring is defined by IEEE 802.5 standard.

10.2.1.3 Star Topology

In a star topology, a hub is placed at the central location and all the devices are connected to the hub. All communication is possible through the hub only. If the hub is active, it may amplify or regenerate the signals. The following are the drawbacks of a star topology:

1. If the central hub fails, no communication is possible.
2. Cabling cost is more.

For example, Ethernet 10 base T is a popular example.

10.2.1.4 Mesh Topology

All the devices are connected through peer-to-peer links. A fully connected mesh will have $n(n{-}1)/2$ physical channels to connect n devices. The advantage of mesh topology is security and privacy. A dedicated link will eliminate traffic problems, and fault diagnose is easy here. But cabling cost and other hardware required make it difficult to implement in real practice. It is better to use this topology in backbone network and other topologies for further network configuration.

10.2.1.5 Tree Topology

A tree topology maintains the devices connected to a central hub as well as to some secondary hubs, which are again connected to the central hub. It allows an isolated network which prioritizes communication from different computers. It also faces the same problem as in a star topology; failure of central hub will crash the entire network. Also, it has high cabling cost.

10.3 LAN TECHNOLOGIES

LAN (local area network) is an integral part to create a network. There are many LAN technologies such as Ethernet, token ring, token bus, FDDI (fiber distributed data interface) and ATM LAN. Table 10.1 shows the comparison of Ethernet, token bus and token ring.

10.3.1 Ethernet

Xerox Corporation, Digital Equipment Corporation and Intel Corporation developed Ethernet LAN technology in 1976. Ethernet is based on the IEEE 802.3 specification. It is a linear-bus logical topology. Ethernet is the most widely used LAN technology in the world. It has passed four generations: Standard Ethernet (10 t Mbps), Fast Ethernet (100 Mbps), Gigabit Ethernet (1 Gbps) and Ten-Gigabit Ethernet (10 Gbps). Earlier Ethernet designed were of two types: Thicknet (thick coaxial main trunk cable of 10 mm and Thinnet (thin coaxial cable: RG-58 of 5 mm). In 1990, unshielded twisted-pair (10Base-T) Ethernet came into existence. Ethernet uses

Table 10.1 Comparison of Ethernet, token bus and token ring

Attribute	Ethernet	Token Bus	Token Ring
Physical Topology	Linear	Linear	Star
Logical Topology	None	Ring	Ring
Connection	Random	By token	By token
Node Addition	Node added anywhere and anytime	Distributed algorithms are responsible for node addition	Between two specified nodes
Cable Used	Twisted pair, co-axial and fibre optic	Co-axial	Twisted pair and fibre optic
Cable Length	50–2000 m	200–500 m	50–1000 m
Frequency	10–100 Mbps	10 Mbps	4–100 Mbps
Frame Structure	1500 byte	8191 bytes	5000 bytes
IEEE Standard	802.3	802.4	802.5
Maintenance	No central maintenance	By distributed algorithms	By a designated monitor node
Performance	Immediately transmitted by the nodes, heavy traffic can reduce the effectiveness of transmission	Nodes must wait for the token if no other node is transmitting. During heavy traffic, token passing provides fair access to all nodes	Nodes must wait for token even if no other node is transmitting. During heavy traffic, token passing provides fair access to all nodes
Maximum Delay before Transmitting	None	Bounded, depending on distance spanned and number of nodes	Bounded, depending on distance spanned and number of nodes

Carrier-Sense Multiple Access with Collision Detection (CSMA/CD) access control scheme. The drawback of this topology is that if one of the links between the two adjacent nodes fails, the whole network fails.

10.3.2 Token Bus

Token bus is a bus (physical view) ring (logical view) topology. In token bus, nodes are connected linearly. However, they make a logical ring, as each node knows the address of its successor.

10.3.3 Token Ring

Token ring is a star (physical view) ring (logical view) topology. The hub acts as a connector. The wiring inside the hub makes the ring; the stations are connected to this ring through the two wire connections. It is more efficient; if a link goes down, it will bypass the hub and operate the other nodes. It also improves the scalability of the network.

In early token release,

$$\text{Throughput for single station or } N \text{ stations} = \frac{\text{Data}}{\text{Transmission time} + (\text{Ring latency}/\text{Number of stations})}$$

In delayed token release,

$$\text{Throughput for single station} = \frac{\text{Data}}{\text{Transmission time} + \text{Ring latency} + (\text{Ring latency}/\text{Number of stations})}$$

$$\text{Throughput for } N \text{ stations} = \frac{\text{Data}}{\text{Ring latency} + (\text{Ring latency}/\text{Number of stations})}$$

10.4 ISO/OSI STACK

There are two reference models based on the network architectures. One is the International Standards Organization/Open Systems Interconnection (Reference Model 1984) (ISO/OSI) model and other is the Transmission Control Protocol/Internet Protocol (TCP/IP) model. The layered architecture of both the models has been compared in Fig. 10.2.

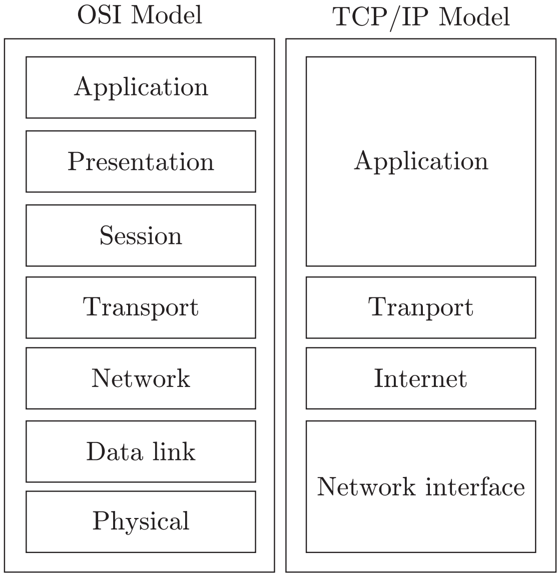


Figure 10.2 | OSI and TCP/IP model.

The ISO/OSI model has seven layers while TCP/IP has only four layers. Session and presentation layers are completely missing. Characteristics of the session layer are provided by the transport layer, and the application layer bears the accountability of the presentation layer. Host to network is the lowest layer in the TCP/IP model and its duty is to send IP packets to the network. In the ISO/OSI model, the network layer provides connection-oriented as well as connectionless services, but the TCP/IP model provides only connectionless services.

If M is a message and H is the header that is added at every layer and N layers are present in hierarchy, then the fraction of header that is passed in the total content is calculated as follows:

$$\text{Fraction of data} = \frac{M}{NH + M}$$

10.4.1 ISO/OSI Model

10.4.1.1 Layer 1: Physical Layer

The major responsibilities of physical layer are transmission of raw bit stream and to form the physical interface between two communicating devices. The conversion of analog to digital and digital to analog is performed at this layer.

The following are some issues listed which may create problems before/during transmission:

1. Compatibility of mechanical and electrical interfaces
2. Working of physical transmission media
3. Deciding on the number of bits per second to be sent
4. Finding out whether transmission is simplex or duplex
5. Establishing and terminating the initial communication when both sender and receiver are finished

$$\text{Transmission time} = \frac{\text{Message size (bits)}}{\text{Bandwidth (bits/s)}}$$

$$\text{Propagation time} = \frac{\text{Distance}}{\text{Velocity}}$$

Problem 10.1: Message size = 1 Kb

$$\text{Bandwidth} = 1\ \text{Mbps}$$

$$\text{Transmission time} = \frac{1\ \text{Kb}}{1\ \text{Mbps}} = \frac{10^3 \times 2^3 \text{bits}}{10^6 \text{bits/s}} = 8 \times 10^{-3}\ \text{s}$$

If link utilisation is 50%, what is the relation between transmission time and propagation time?

Solution:

Link utilization of sender

$$= \frac{\text{Transmission time}}{\text{Transmission time} + 2 \times \text{Propagation time}}$$

$$\frac{1}{2} = \frac{\text{Transmission time}}{\text{Transmission time} + 2 \times \text{Propagation time}}$$

Transmission time + 2 × Propagation time
= 2 × Transmission time

Therefore, Transmission time = 2 × Propagation time

$$\text{Transmission time} = 2 \times \text{Propagation time}$$

or $$\text{Transmission time} = \text{Round-trip time}$$

Let L be the message, B the bandwidth and R the round-trip time (RTT),

$$\text{Link utilization of sender} = \frac{L/B}{(L/B) + R}$$

$$\eta = \frac{L}{L + BR}$$

In the above expression,

1. if L = BR, $\eta = 50\%$
2. if $L > BR$, $\eta > 50\%$
3. if $L < BR$, $\eta < 50\%$
4. if $L >>> BR$, $\eta \simeq 100\%$
5. if $L <<< BR$, $\eta \simeq 0\%$

10.4.1.2 Layer 2: Data Link Layer

Data link layer provides reliable transfer of information between two adjacent nodes. Also, it provides frame-level error control and flow control. It provides communication between machines on the same network. Communication between two devices can be simplex, half-duplex, or full-duplex. In simplex, the communication is unidirectional. In half-duplex, each device can both transmit and receive, but not at the same time. In full-duplex, both devices can transmit and receive simultaneously.

This layer is responsible for encoding and decoding i.e. converting bits to signals at sender site and recovering bits from received signals at receiver side; frame creation i.e. deciding a minimum unit for sending bits; error detection and/or correction of frames through parity or CRC and flow control using ARQ, sliding WINDOW etc. The functionality of data link layer is as follows:

1. **Encoding:** Signals propagate over a physical medium – (1) modulate electromagnetic waves (varying voltage); (2) encode binary data onto signals (e.g., 0 as low signal or non-return to zero, NRZ, and 1 as high signal or non-return to zero inverted, NRZI); make a transition from current signal to encode a 1 or stay at the current signal to encode a 0;(3) Manchester (transmit XOR of the NRZ-encoded data and the clock only 50% efficient).
   In Manchester encoding, a clock signal and data signal are mixed together by XORing operation. The clock makes a clock transition in every bit time, so it runs at twice the bit rate. When it is XORed with 0 then it makes low-to-high transition, it acts as a clock and when it is XORed with 1 then it makes high-to-low transition, it acts as a data signal.
2. **Framing:** The basic data unit at the date link layer is called a 'frame' which is a collection of bits in sequence boundary. In order to mark boundaries of frame starting and ending characters are used.
3. **Flow control:** It is a mechanism which informs the sender about the amount of data transmission before receiving an acknowledgement from the receiver. As the receiving device has limited speed for processing the incoming data and limited memory to store data, so it informs the sending device by sending few frames and stop. The receiving device has a buffer, a block of memory, for storing extra incoming data before processing.
4. **Error control:** It is a mechanism which informs the sender about the retransmission of the damaged and lost frames during transmission. It is a method of error detection and error correction. Automatic repeat request (ARQ) is a process in which whenever an error is detected, the receiving device sends the request for retransmission to sender.
5. **Techniques of flow and error control:**
   - *Stop-and-wait automatic repeat request:* In this protocol, the sender starts the timer and keeps the copy of the sent frame. If the timer expires and there is no acknowledgement (ACK) for the sent frame, the frame is resent, the copy is held and the timer is restarted. For the corrupted and lost ACK frame, sequence numbers can be used. In the data frame, a field is added for the sequence number. The sequence numbers are based on modulo-2 arithmetic. This protocol is also having acknowledgement numbers, which specifies the sequence number of the next frame expected by the receiver. A data frame uses a sequence number and an ACK frame uses an acknowledgement number. The control variable of sender keeps the sequence number for the next frame to be sent (0 or 1). The control variable of receiver keeps the number of the next frame expected. When a frame is sent, the value of the control variable of sender is incremented. When a frame is received, the value of the control variable of receiveris incremented. The stop-and-wait ARQ protocol is very inefficient if the channel is thick (large bandwidth) and long (long round-trip delay). Other drawback of this protocol is that it does not support pipelining, as it does not support multiple frames.Pipelining helps in improving the efficiency of the transmission, if the number of bits in transition is large with respect to the bandwidth-delay product.
     For stop-and-wait ARQ,

$$\text{Transmission time} = \frac{\text{Message size}}{\text{Bandwidth}}$$

$$\text{Propagation delay} = \frac{\text{Distance of the link}}{\text{Velocity}}$$

     Link utilisation of sender or throughput is given by

$$\eta = \frac{\text{Transmission time}}{\text{Transmission time} + 2 \times \text{Propagation delay}}$$

   - *Go-back-N automatic repeat request:* This protocol helps in improving the efficiency of transmission by filling the pipe. It supports multiple frames during wait for acknowledgement. The sequence numbers are modulo 2^m, where m is the size of the sequence number field in bits. Sliding window defines the range of sequence numbers related to the sender and receiver. When the timer expires, the sender resends all outstanding frames. Stop-and-wait ARQ is a special case of Go-Back-N ARQ in which the size of the send window is 1. It supports one receiver window size. This protocol is very inefficient for a noisy link. In noisy link, frames are resending again and again, which uses bandwidth and slow down the transmission. For Go-Back-N ARQ,

     Sender window size $< 2m$
     Maximum sequence number $= 2m - 1$
     where m is the number of segment bits.

If maximum sequence number is s, then the number of sequence bits = log $(s+1)$

$$\text{Transmission time} = \frac{\text{Message size}}{\text{Bandwidth}}$$

$$\text{Propagation delay} = \frac{\text{Distance}}{\text{Velocity}}$$

Link utilisation of sender or throughput is given by

$$\eta = \frac{\text{Transmission time}}{\text{Transmission time} + 2 \times \text{Propagation delay}}$$

$$\text{Number of frames (Window size)} = \frac{\text{Total bits}}{\text{Frame size}}$$

- *Selective repeat automatic repeat request:* In this protocol, the damaged frame is resent in the network. It is efficient for noisy links, but it requires complex processing at the receiver end.

  Sender window size
  = Receiver window size $\leq 2^m - 1$

  If Q is the size of the window, then the number of sequence bits = $\log_2 Q + 1$

$$\text{Number of frames (Window size)} = \frac{\text{Total bits}}{\text{Frame size}}$$

Problem 10.2: Calculate the link utilisation for stop-and-wait flow control mechanism if the frame size is 4800 bits, bit rate is 9600 bps and distance between devices is 2000 km. Given propagation speed is 200000 km/s.

Solution:

$$\eta = \frac{4800/9600}{(4800/9600) + 2 \times (2000/200000)} = 0.96$$

6. **Parity bits:** Append a single parity bit to a sequence of bits
   - If using 'odd' parity, the parity bit is calculated as making the total number of 1's in the bit sequence odd;
   - If using 'even' parity, the parity bit makes the total number of 1's in the bit sequence even

   For example, if $-Q$ is for even parity, what's the parity bit for 00010101? The problem with parity bit is that it only detects when there are an odd number of bit errors.
7. **Polynomial codes:** It can detect errors on large chunks of data; has low overhead; is more robust than parity bit; and requires the use of a code polynomial.
8. **Cyclic Redundancy Check (CRC):** Example of a polynomial code

   Procedure:
   - Let r be the degree of the code polynomial. Append r zero bits to the end of the transmitted bit string. Call the entire bit string $S(x)$.
   - Divide $S(x)$ by the code polynomial using modulo-2 division.
   - Subtract the remainder from $S(x)$ using modulo-2 subtraction.

   The result is the checksummed message.
9. **Decoding a CRC procedure:**
   - Let n be the length of the checksummed message in bits.
   - Divide the checksummed message by the code polynomial using modulo-2 division. If the remainder is zero, there is no error detected.
10. **Choosing a CRC polynomial:** The longer the polynomial, the smaller the probability of undetected error.

Problem 10.3: If the frame is 1101011011 and generator is $x^4 + x + 1$, what would be the transmitted frame?

Solution: The polynomial $x^4 + x + 1$ corresponds to divisor $10011 (k = 5 \text{ bits})$

Data word (1101011011) of N = 10 bits is augmented with $(k - 1)$ zero's.

Dividend = 11010110110000

$$
\begin{array}{rl}
 & 110000101 \\ \cline{2-2}
10011\,\big) & 11010110110000 \\
 & 10011 \\ \cline{2-2}
 & 010011 \\
 & 10011 \\ \cline{2-2}
 & 0000010110 \\
 & 10011 \\ \cline{2-2}
 & 0010100 \\
 & 10011 \\ \cline{2-2}
 & 001110 \\ \cline{2-2}
\end{array}
$$

After dividing the message 1101011011 by 10011 the remainder is 1110, which is CRC. The transmitted data is data + CRC, which is 1101011011 + 1110 = 11010110111110.

Data link layer is divided into two sublayers:

1. **Multiple access control sublayer:** It provides controlled access to shared transmission media.
2. **Logical link control sublayer:** It is responsible for error and flow control.

When multiple users share a common communication link, multiple access protocols are used to coordinate access to common link (Fig. 10.3).

1. **Random Access Protocols:** The following are the different random access protocols used for accessing the shared transmission channel:
   - *Pure ALOHA:* It says that whenever a station is having the data, they can send immediately. The time at which the collision occurs is called vulnerable time.

$$T_p = \text{Maximum propagation time} = \frac{\text{Distance between two stations}}{\text{Velocity}}$$

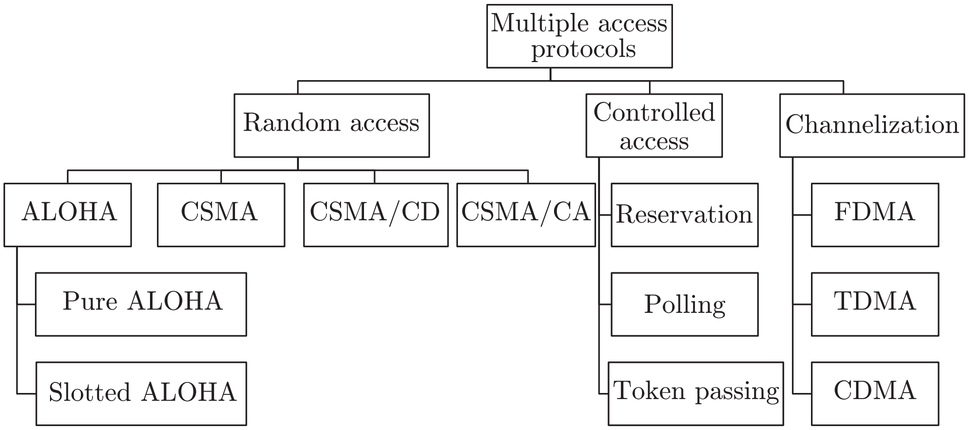


Figure 10.3 Multiple access protocols.

Suppose T_{fr} is the average transmission time for a frame, then

$$T_{fr} = \frac{\text{Frame size}}{\text{Bandwidth}} = G$$

$$\text{Throughput } (S) = G \times e^{-2G}$$

$$\text{Vulnerable time} = 2 \times T_{fr}$$

$$S_{\max} = 18.4\%$$

- *Slotted ALOHA:* It says that if stations are ready with the data, they have to wait for the required time slot and can transmit data exactly at that timeslot.

$$\text{Throughput } (S) = G \times e^{-G}$$

$$\text{Vulnerable time} = T_{fr}$$

$$S_{\max} = 36.8\%$$

- *CSMA (Carrier Sense Multiple Access):* If a station is ready with the data, it senses the channel, and if channel found idle, data is transmitted, otherwise the station has to wait for random amount of time.

10.4.1.3 Layer 3: Network Layer

The network layer is responsible for host-to-host delivery and path selection between endsystems (routing). The fragmentation, reassembly and translation between different network types are also performed at this layer. In other words, communication between nodes is possible in different networks through this layer.

Packet delivery can be accomplished by using either a connection-oriented or a connectionless network service. In a connection-oriented protocol, the connection is established before sending the packets so route is established before and all the packets have to follow that route. Example: Frame relay and ATM uses this service.

In connectionless protocols, the network layer protocol treats each packet independently. The packets in a message may or may not follow the same path to their destination. For example, Internet uses this type of service.

Switching can be broadly divided into three categories: circuit switching, packet switching and message switching, as shown in Table 10.2.

Table 10.2 Comparison of circuit, message and packet switching

Attribute	**Circuit**	**Message**	**Packet**
Dedicated Physical Path	Yes	No	No
Bandwidth Available	Fixed		Dynamic
Route Selection	Static	Dynamic (per message)	Dynamic (per packet)
Potentially Wasted Bandwidth	Yes	No	No
Stored and Forward Transmission	No	Stored	Queued not stored
Transmission Length	Unlimited	Maximum length	No maximum length
Same Route Follows	Yes	No	No

(*Continued*)

Table 10.2 Continued

Attribute	Circuit	Message	Packet
Packets Arrive in Order	Yes		No
Call Setup	Required	Not Required	Not required
Congestion Route Blocking	At setup time, if user busy	No message blocking	On every packet
Possible Reordering	No	No	Yes
Response to Link Failure	Data loss	Rerouting/Retransmission	Rerouting/Retrans-mission
Message Delivery	Guaranteed	Depends on the network	Depends on the network
Delivery Time	Negligible	Long	Short
Path Establishment	Switch path for entire connection time	For each message	For each packet
Charging	Per minute	Per message	Per packet

The **Internet** is a global system of computer networks that are interconnected worldwide and all use the standard Internet protocol suite (TCP/IP) for linking.

1. **Internet as a Datagram Network:** The Internet, at the network layer, is a packet-switched network. Switching can be generally divided into three broad categories: circuit switching, packet switching, and message switching. Packet switching uses either the virtual circuit approach or the datagram approach.

   The Internet chooses the datagram approach to switching in the network layer, and uses the universal addresses defined in the network layer to route packets from the source to the destination. Switching at the network layer in the Internet uses the datagram approach to packet switching.
2. **Why Internet Uses Connectionless Network?** Delivery of a packet can be accomplished by using either a connection-oriented through TCP or a connectionless network service through UDP (see Table 10.3). In a connection-oriented service, the source has to make a connection with the destination before sending a data packet. Only after establishing connection, a sequence of packets from source to the destination can be sent on the same path that is established before in a sequential order. The connection is terminated only when all the packets of a particular message have been successfully. But the communication at the network layer in the Internet is connectionless. The reason is that Internet is made of so many heterogeneous networks that it is almost impossible to create a connection between every source and destination pair without knowing the nature of the networks in advance.

Table 10.3 Comparison between TCP and UDP

Attribute	TCP	UDP
Connection Management	Connection oriented	Connectionless
End-to-End Connection	Dedicated connection	No dedicated connection
Reliability	Reliable	Unreliable
Transmission	Byte oriented	Message oriented
Acknowledgement	Yes	No
Retransmission	Automatically	If needed
Congestion Control	Yes	No
Flow Control	Yes	No
Fault Tolerance	No	No
Data Delivery	Strictly ordered	Unordered
Security	Yes	Yes
Overhead	Low	Very low
Transmission Speed	High	Very high
Data Quantity Suitability	Small to very large amount of data	Small to moderate amount of data

10.5 ROUTING ALGORITHMS

When the router receives a packet, which route this packet should follow to reach to destination is an important concern. This is one of the major responsibilities of network layer. This decision is taken by router on the basis of the routing table maintained by it. The algorithm which decides the suitable route is known as routing algorithm. The desirable properties of any routing algorithm are correctness, fairness, stability, robustness and optimality.

These algorithms can be broadly divided into two categories:

1. **Adaptive algorithms:** The dynamic routing decision depends on the topology and traffic. Furthermore, adaptive algorithms have been divided into three forms:
   - ***Centralised:*** The decision is taken on the basis of global information and this is performed by a centralised node.
   - ***Isolated:*** The routing decision is taken based on local information. Generally, routers do not share information with their neighbours.
   - ***Distributed:*** A combination of local and global information.
2. **Non-adaptive algorithms:** The static routing decision is taken in advance and it is downloaded by the routers, that is, never change once initial route has been selected.

The properties of non-adaptive routing algorithms are as follows:

1. **Optimality principal:** It states that "if router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route". In other words, suppose r_1 is the route from I to J and r_2 is rest of the route. Then, if any route is better than r_2, it could have improved the overall optimal route. Hence, r_1r_2 is optimal.
2. **Sink tree:** A set of all optimal routes from any source to a fixed destination form a tree called sink tree. Sink trees may be more than one with the same path length. The concern of sink tree here is to help routers find the best path.

In addition to adaptive and non-adaptive categorisation, routing algorithm can be simply categorised into the following:

1. Static routing (shortest path, flooding)
2. Flow-based routing
3. Dynamic routing (distance vector, link state routing)
4. Hierarchical routing
5. Routing for mobile hosts
6. Broadcast routing
7. Multicast routing

These are explained as follows:

1. **Shortest path routing:** This non-adaptive approach is based on the simplest and most widely used principle. Each node is treated as a router and each arc as communication link. To find a path between a pair of routers, the shortest path is chosen. The shortest path is chosen based on the number of hops or geographical distance in kilometres. Dijkstra's and Bellman–Ford's algorithm are the most famous shortest path algorithms.

Example 10.1

Flooding: This is again a non-adaptive algorithm and it sends a copy of the packet to every outgoing line except the line on which it was received. This guarantees the packet delivery to destination but a large number of packet copies will be generated. Sometimes this count approaches to infinity and the only solution is to discard the packet using any of the following approach:

(a) Using a hop counter to avoid forwarding of packets as number of hops reaches the diameter of the network.
(b) Keeping track of flooded packets.
(c) Selective forwarding by forwarding only those relevant packets which approaches to the right destination.

To avoid looping, a sequence number may be added to each packet's header. This helps in discarding those packets whose sequence number is lower than the one already received.

Note: Although flooding is inefficient in most applications, an exception may be in the case of military application where large number of routers are placed and robustness is highly desirable.

2. **Flow-based routing:** This is a non-adaptive algorithm which uses topology and traffic condition for deciding the route. If traffic on some route is more than average, then the route should be avoided to achieve optimal path.

   If line capacity and flow is given, delay can be determined easily using the following formula:

$$T = \frac{1}{\mu C - \lambda}$$

where $1/\mu$ is mean packet size in bits, λ is the mean number of packets arrived per second and C is the line capacity in Kbps.

Example 10.2

Distance Vector/Distributed Bellman–Ford/ Ford–Fulk-erson Routing Algorithm: This is one of the popular examples of dynamic routing algorithm. Each router maintains a table (called vector) which helps in finding the best-known distance to any destination. It also indicates preferred outgoing line to be used to reach the destination. The performance metric can be the number of hops, time delay or number of packets in the queue.

Issues:

(a) Slowness in converging to the right answer and this problem is well known as count to infinity (can be solved using split-horizon algorithm).

(b) Line bandwidth should be a metric when choosing root.

Whenever a packet comes to a router, the neighbouring router will give their routing table and a new vector table is created at that router.

Example 10.3

Consider a network shown in Fig. 10.4:

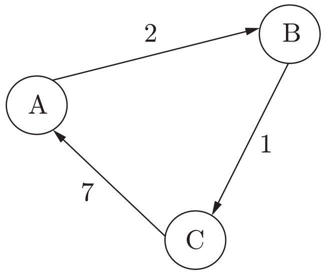


Figure 10.4

Step 1: Initialise cost of direct links and set to °cost from neighbours.

Node A table

		Cost to		
		A		C
From	A	0	2	7
	B	∞	∞	∞
	C	∞	∞	∞

Node B table

		Cost to		
		A	B	C
From	A	∞	∞	∞
	B	2	0	1
	C	∞	∞	∞

Node C table

		Cost to		
		A	B	C
From	A	∞	∞	∞
	B	∞	∞	∞
	C	7	1	0

Step 2: Each node periodically sends its own distance vector (DV) to neighbours. When node A receives DV from neighbour B, it keeps it and updates its own DV as follows:

$$D_A(\mathrm{B}) = \min_v\{\, C(x,v) + D_v(\mathrm{B})\}$$

Node A updated table after receiving DV from node B and node C is:

$$D_A(\mathrm{B}) = \min\,\{2 + 0{,}7 + 1\} = 2 \text{ and}$$
$$D_A(\mathrm{C}) = \min\,\{2 + 1{,}7 + 0\} = 3$$

Node A table

		Cost to		
		A	B	C
From	A	0	2	3
	B	2	0	1
	C	7	1	0

Node B table

		Cost to		
		A	B	C
From	A	0	2	7
	B	2	0	1
	C	7	1	0

Node C table

		Cost to		
		A	B	C
From	A	0	2	7
	B	3	0	1
	C	3	1	0

Step 3: In similar fashion, algorithm proceeds until all nodes have updated tables.

Node A table

		Cost to		
		A	B	C
From	A	0	2	3
	B	2	0	1
	C	3	1	0

Node B table

		Cost to		
		A	B	C
From	A	0	2	3
	B	2	0	1
	C	3	1	0

Node C table

		Cost to		
		A	B	C
From	A	0	2	3
	B	2	0	1
	C	3	1	0

3. **Link state routing:** This is simply a modern replacement of distance vector routing. The steps of the algorithm are as follows:
   - Each router discovers the neighbours for their network addresses.
   - Measure delay or cost to each of these neighbours.
   - Construct a packet including network address and delays of all neighbours.
   - Send it to all routers.
   - Find the shortest path to all routers (Dijkstra's algorithm can be used).

Example 10.4

OSPF protocol (used in Internet) uses the link state algorithm: IS-IS (intermediate system–intermediate system) is another example used in Internet backbones and in some digital cellular systems.

4. **Hierarchical routing:** In the situation of telephone networks, all above routing algorithms fail because the size of the routing table is too large here. In this routing, routers are divided (placed) into different regions. A router will have the knowledge of other routers in its own region but unaware about the internal structure of other regions. This will reduce the size of the routing table.
5. **Broadcast and multicasting routing:** Broadcasting means sending packets to all other hosts in the network, whereas multicasting refers to sending packets to a specific group or a fixed number of hosts.

10.6 NETWORK LAYER PROTOCOLS

In the Internet model, or the TCP/IP suite, there are five main network layer protocols: ARP, RARP, IP, ICMP and IGMP (Fig. 10.5).

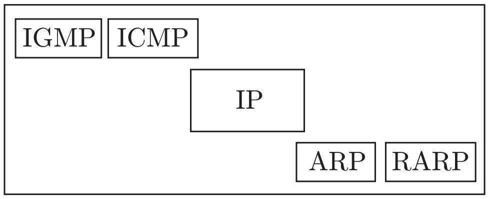


Figure 10.5 Network protocols.

The main protocol in this layer is IP. It is responsible for host to host delivery of packets from a source to destination. IP needs services of other protocols for better network performance. IP needs ARP to find MAC address of the next hop. As IP is an unreliable protocol, it needs ICMP (Internet Control Message Protocol) to handle unusual situations and errors. IGMP (Internet Group Message Protocol) is used for multicast delivery.

10.6.1 Internet Protocol (IPv4)

Internet Protocol is a layer-3 protocol of network layer of OSI model. It takes data segments from layer-4 transport layer and divides it into packets. Thus, it encapsulates data units received from the above layer and adds its own header information (Fig. 10.6).

Maximum size of IP header = 60 bytes

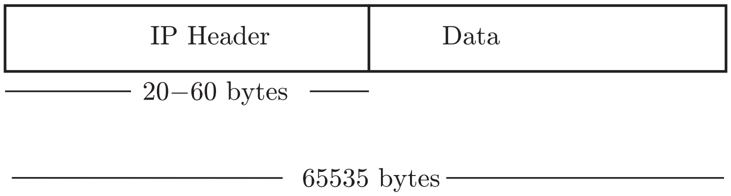
Minimum size of IP header = 20 bytes

If the size of header is 32 bytes in IP, calculate the number of option bytes.

Option bytes = 32 − 20 = 12 bytes

IP header details are as follows:

1. **Version:** Version number of Internet Protocol is 4 (e.g. IPv4).
2. **IHL:** Internet header length indicates the size of header available in the packet.
3. **TOS:** Type of service that is provided by the router to the packets such as minimum delay or cost.
4. **Total length:** Length of the entire IP packet (including IP header and IP payload).
5. **Identification:** If IP packet size is greater than the maximum transmission unit (MTU), it has to be fragmented during the transmission by the router, then all the fragments of the packet contain same identification number to identify original IP packet they belong to.
6. **Flags:** If IP packet is too large, these 'flags' tell if they can be fragmented or not. In 3-bit flag, the MSB is



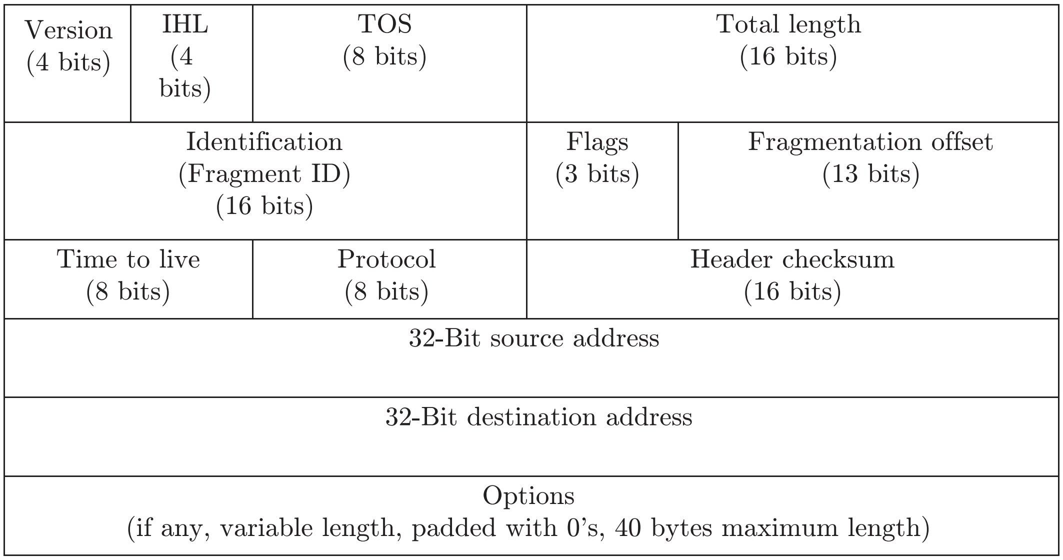

<table>
<tr><td>Version
(4 bits)</td><td>IHL
(4
bits)</td><td>TOS
(8 bits)</td><td colspan="2">Total length
(16 bits)</td></tr>
<tr><td colspan="3">Identification
(Fragment ID)
(16 bits)</td><td>Flags
(3 bits)</td><td>Fragmentation offset
(13 bits)</td></tr>
<tr><td colspan="2">Time to live
(8 bits)</td><td>Protocol
(8 bits)</td><td colspan="2">Header checksum
(16 bits)</td></tr>
<tr><td colspan="5">32-Bit source address</td></tr>
<tr><td colspan="5">32-Bit destination address</td></tr>
<tr><td colspan="5">Options
(if any, variable length, padded with 0's, 40 bytes maximum length)</td></tr>
</table>

Figure 10.6 IPv4 packet header.

always set to '0'. The next bit is DF (do not fragment). If DF = 0, fragmentation can be done if required and buffered at the router of the receiver until all fragment comes, and if DF = 1, fragmentation should not be done. The third bit is MF (more fragment). If MF = 1, then datagram is not the last fragment. If MF = 0, then datagram is the last fragment.

7. **Fragment offset:** This offset tells the exact position of the fragment in the original IP packet.
8. **Time to live:** It controls the maximum number of routers visited by the datagram. To avoid looping in the network, every datagram is sent with some-time-to-live(TTL) value set. Each router receiving the datagram decreases TTL by 1 and when it becomes 0, the datagram is discarded.
9. **Protocol:** It tells the higher-level protocol which uses the service of the IP layer. For example, protocol number for ICMP is 1, IGMP is 2, TCP is 6 and UDP is 17.
10. **Header checksum:** This field stores checksum value of the entire header excluding data which is used to check if the packet has been received error-free.
11. **Source address:** 32-Bit address of the sender of the packet.
12. **Destination address:** 32-Bit address of the receiver of the packet.
13. **Options:** These options may contain values for various options such as strict source routing, security, record route, time stamp, etc.

Problem 10.4: If the total length bits are 0000000000 111111 and header length is 1001, calculate the size of the packet, header and payload.

Solution:

(a) Packet size = Decimal equivalent of 00000000 00111111 = 63 bytes
(b) Header size (1001) = 9 × 4 = 36 bytes
(c) Packet size = Header + Payload

Payload = Packet size − Header = 63 − 36
= 27 bytes

Problem 10.5: Suppose a router receives an IP packet containing 600 data bytes and has to follow a packet maximum transferable unit of 200 bytes. Assume that the IP header is 20 bytes long; specify the relevant values in each fragment header.

Solution: IP packet = 600 bytes
MTU = 200 bytes

IP header = 20 bytes

Maximum possible data length per fragment = 200 − 20 = 180 bytes

Data length of each fragment must be a multiple of 8 bytes so 22 × 8 = 176 bytes

Data packet must be divided into the following four frames:

(176 + 20) + (176 + 20) + (176 + 20) + (72 + 20) = 680

	Length	ID	MF	Fragment Offset
Original Packet	620 bytes	X	0	0
Fragment 1	196 bytes	Z	1	0
Fragment 2	196 bytes	Z	1	22
Fragment 3	196 bytes	Z	1	44
Fragment 4	92 bytes	Z	0	66

10.6.1.1 IPv4 Addresses

An **IPv4** address is a 32-bit address that can find the device on the Internet uniquely and universally. These addresses are unique, that is, these define only one connection by the device to the Internet. The total number of addresses used by this protocol which is called as address space is 2^n where n is the total number of bits. As IPv4 uses 32-bit addresses, the address space of this protocol is 2^{32}.

In IPv4, addresses are 32-bit binary numbers. However, for ease of use of people, theses binary patters are represented as dotted decimals. Therefore, there are two notations available to denote IPv4 addresses: binary and dotted decimal.

1. **Binary notation:** In this notation, IPv4 addresses are represented as 32 bits where each octet is said to be a byte. Therefore, the IPv4 address is usually said to be the 4-byte address. The example for this notation is as follows:

   01111101 10000011 00000110 00000001

2. **Dotted-decimal notation:** In this notation, each byte (8 bits) of 32-bit binary address, known as octet is separated with a dot, and then the binary number is converted into its decimal equivalent. The example for this notation is as follows:

   11000000 10101000 00001010 00001010
   is equivalent to 192.168.10.10

Classful Addressing

The architecture used by IPv4 is classful addressing where the addresses are divided into five classes: A, B, C, D and E. The first few bits reveal the class of address when the address is in binary and first byte reveals the class of address when the address is in dotted decimal notation. This architecture is called classful addressing (Table 10.4).

Table 10.4 Classful addressing architecture of IPv4

	Leading Bits	Value Range	Starting Address	Ending Address
Class A	0	0–126	0.0.0.0	127.255.255.255
Class B	10	128–191	128.0.0.0	191.255.255.255
Class C	110	192–223	192.0.0.0	223.255.255.255
Class D	1110	224–239	224.0.0.0	239.255.255.255
Class E	1111	240+	240.0.0.0	255.255.255.255

Note: 127 reserved for loopback address.

The addresses of class A, B, C are unicast, class D addresses are multicast and class E addresses are reserved. The IP address in class A, B and C is divided into **netid** and **hostid**. In class A, one byte defines the netid and the other three bytes define the hostid. In class B, two bytes define the netid, while the other two bytes define the hostid. In class C, three bytes define the netid and one byte defines the hostid.

Mask: The length of the netid and hostid (in bits) is predetermined in classful addressing but we can also use a mask (also called the default mask) which is a 32-bit number made of contiguous 1's. The subnet mask is compared to the IP address from left to right, bit for bit. The 1's in the subnet mask represent the network portion and 0's represent the host portion. The subnet mask is created by placing a binary 1 in each bit position that represents the network portion and placing a binary 0 in each bit position that represents the host portion.

The subnet mask assigned along with IP address signifies which part of the IP address is network and which part is host.

There is a flaw in this type of architecture, that is, each class is divided into fixed number of blocks, and a lot of addresses are wasted as blocks A and B addresses are too large to consume, block C addresses are small in number, class D addresses are reserved for multicasting and class E addresses are reserved for future, which is another wastage of addresses. Therefore, these address scheme architecture is almost obsolete and leads to an introduction of classless addressing scheme as if there are no address classes.

Classless Addressing

In classless addressing, the addresses are granted in a block and the number of addresses in the block depends upon the addresses needed by the entity. The addresses in the block must be contiguous, the first address must be evenly divisible by the total number of addresses and the number of addresses in a block must be power of 2. Mask in classless addressing can take the value in the range of 0 to 32. Classless Inter-Domain Routing provides the flexibility of borrowing bits of host part of the IP address and using them as smaller sub-networks called subnet. This process is known as **subnetting**. The addresses can be defined in IPv4 as a.b.c.d/n, where a.b.c.d defines one address of the block.

1. The first address in the block can be found by setting the rightmost 32 $-n$ bits to Os in the binary notation.
2. The last address in the block can be found by setting the 32 $-n$ rightmost bits in the binary notation of the address to Is.
3. The number of addresses in the block is the difference between the first and the last address,that is, 2^{32-n}.

Problem 10.6: Determine the subnet identifier, broadcast address and number of valid host addresses having a host with IP address of 196.142.4.29/24.

Solution:

Subnet identifier: 196.142.4.0

Broadcast address: 196.142.4.255

Number of valid host addresses: 254

The subnet mask of /24 in CIDR notation corresponds to 255.255.255.0. The above subnet mask has last eight bits set to zero (32 − 24 = 8), which means that we have 2^8 IP addresses available. Total number of valid hosts in a network is obtained by subtracting two addresses: subnet address and broadcast address $2^8 - 2 = 256 - 2 = 254$.

Problem 10.7: If the IP address of a system is 131.121.61.189, calculate the netid, first host, last host and directed broadcast address.

Solution:

IP address (class B)	131.121.61.189
Net mask	255.255.0.0
Net id	131.121.0.0
First host	131.121.0.1
Directed broadcast address	131.121.255.255
(All host bits 1)	

Problem 10.8: In class B, if subnet mask is 255.255.240.0, find the number of subnets and host in each subnet.

Solution:

11111111 11111111	1111	0000000
	0000	
Netid	Subnet bits	Host bits

Number of subnets = $2^4 - 2 = 14$ subnets

Number of hosts in each subnet = $2^{12} - 2$

= 4094 subnets

Problem 10.9: If IP address of a system is 199.11.171.189 and subnet mask is 255.255.255.224, calculate the subnet id.

Solution:

IP address	199.11.171.189	
Subnet mask	255.255.255.224	
Subnet id	199.11.171.160	
160 is equivalent to	101	00000
	Subnet Bits	Host Bits

Number of subnets = $2^3 - 2 = 6$

Number of hosts = $2^5 - 2 = 30$

First subnet id is 199.11.171.32 (001 00000).

Second subnet id is 199.11.171.64 (010 00000).

Last subnet id is 199.11.171.192 (110 00000).

First host of first subnet is 199.11.171.33 (001 00001).

Last host of last subnet is 199.11.171.222 (110 11110).

Supernetting is a part of classless addressing. In classless addressing, the addresses should be contiguous in a block. The first address should be exactly divisible by the number of addresses in a block. In supernet, bits are borrowed from netid.

Rules of supernetting:

1. The number of blocks must be power of 2.
2. The blocks must be continuous in the address space.
3. The third octet of the first address in the superblock must be exactly divisible by the number of blocks.

Problem 10.10: A company needs 1000 addresses, which of the following set of class C block can be used to form a supernet?

(a) 198.47.32.0, 198.47.33.0, 198.47.34.0
(b) 198.47.31.0, 198.47.32.0, 198.47.33.0, 198.47.34.0
(c) 198.47.32.0, 198.47.42.0, 198.47.52.0, 198.47.62.0
(d) 198.47.32.0, 198.47.33.0, 198.47.34.0, 198.47.35.0

Solution: (d)

(a) Not in power of 2 or blocks are 3.
(b) Third octet of first address is not divisible by 4.
(c) Blocks are not contiguous.

Problem 10.11: Which of the following can be the beginning address of a block that contains 16 addresses?

(a) 205.16.37.32
(b) 190.16.42.44
(c) 17.17.33.82
(d) 123.45.24.52

Solution:

(a) 32 is divisible by 16.

10.6.2 ICMP

As IP is a connectionless and unreliable protocol, it cannot report errors, so it takes help of ICMP to communicate updates or error information to other intermediate routers, devices or hosts.

Each ICMP message contains three fields: Type, Code and Checksum (Fig. 10.7). The Type field identifies the ICMP message, the Code field provides further information about the associated Type field and the Checksum field verifies the integrity of the message.

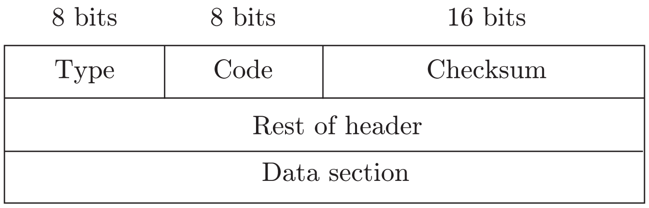


Figure 10.7 ICMP message.

Extra options are specified in the rest of the header. ICMP places the message to be sent in the data section. Different types defined for ICMP messages are shown in Table 10.5.

Table 10.5 Different types defined for ICMP messages

Category	Type	Message
Error Reporting Message	3	Destination Unreachable
	4	Source Quench
	11	Time Exceeded
	12	Parameter Problem
	5	Redirect Message
Query Message	0	Echo Reply
	8	Echo Request
	9	Router solicitation
	10	Router advertisement
	13	Timestamp Request
	14	Timestamp Reply
	17	Address Mask Request
	18	Address Mask Reply

10.7 LAYER 4: TRANSPORT LAYER

Although the reliability of the network layer is undoubtful, the transport layer has many responsibilities to carry out the following:

1. Managing connections and timers
2. Allowing reliable, connection-oriented byte stream from one end to other end
3. Multiplexing
4. Addressing
5. Performing segmentation
6. Packetizing
7. Handling error control and variable sized sliding window for flow control
8. Allocating bandwidth with congestion control

Issues:

(a) Headers, error detection, reliable communication
(b) Communication between processes (running on machines on possibly different networks)

10.8 CONGESTION

Congestion occurs when packets overload the subnet (means the number of packets sent to the network is greater than the capacity of the network), which results in performance degrade. The following are causes for congestion:

1. A sudden stream of packet from various sources reaching the same destination.
2. Slow links.
3. Slow processor.
4. Suppose an intermediate router has no free buffer, it will discard the packet. As the ender will not receive any acknowledgement, it will resend the packet and hence congestion will be there.

Congestion is unavoidable, but it is necessary to control it. It can be handled with the following approach:

1. Congestion information may be forwarded to the concerned node so that it is possible to limit senders (prevent one sender from overflowing the receiver) or reroute the packets.
2. Resource availability may reduce the congestion.
3. Prevent additional packets from entering the congestion region.

To control congestion in network traffic, leaky bucket algorithm is used. This algorithm shapes the bursty traffic into fixed rate traffic. It does so by averaging the data rate and dropping the packets if bucket is full.

10.8.1 Congestion Versus Flow Control

Congestion control ensures the ability to carry the offered traffic by any subnet. The major entities that effect the congestion are behaviour of hosts, routers as well as the factors that reduce the carrying capacity.

Flow control makes it sure that there is not much difference between the sending and receiving packet rate, that is, a fast sender does not send at a rate faster than the rate at which the receiver receives.

10.9 USER DATAGRAM PROTOCOL AND TRANSMISSION CONTROL PROTOCOL

10.9.1 User Datagram Protocol

UDP, or User Datagram Protocol, is an unreliable and connectionless protocol used by applications that transmit small amount of data at one time and do not require receipt of acknowledgement of data, for example, audio or video broadcasting (Fig. 10.8).

<table>
<tr><td>Source port (16 bits)</td><td>Destination port (16 bits)</td></tr>
<tr><td>Length (16 bits)</td><td>Checksum (16 bits)</td></tr>
<tr><td colspan="2">Data</td></tr>
</table>

Figure 10.8 UDP packet.

10.9.2 Transmission Control Protocol

TCP, or Transmission Control Protocol, provides a connection-oriented, reliable stream delivery through the use of sequenced acknowledgement with retransmission of packets when necessary. The TCP header structure is shown in Fig. 10.9.

<table>
<tr><td colspan="8">Source port (16 bits)</td><td>Destination port (16 bits)</td></tr>
<tr><td colspan="9">Sequence number(32 bits)</td></tr>
<tr><td colspan="9">Acknowledgement number (32 bits)</td></tr>
<tr><td>HLEN
(4 bits)</td><td>Reserved
(6 bits)</td><td>U</td><td>A</td><td>P</td><td>R</td><td>S</td><td>F</td><td>Window (16 bits)</td></tr>
<tr><td colspan="8">Checksum (16 bits)</td><td>Urgent pointer (16 bits)</td></tr>
<tr><td colspan="9">Option + padding</td></tr>
<tr><td colspan="9">Data</td></tr>
<tr><td colspan="9"><i>TCP - header structure</i></td></tr>
</table>

Figure 10.9 TCP packet.

In TCP, sequence number is attached for every byte in the segment. The initial sequence number for the first data byte will be a random number that is generated by a random number generator in the range of 0 to $2^{32} - 1$. Acknowledgement number will always be the sequence number of the next expected data. The control bits and their description is given in Table 10.6.

Table 10.6 Control bits

Control Bits	Description
U (URG)	Urgent pointer field significant
A (ACK)	Acknowledgement field significant
P (PSH)	Push function
R (RST)	Reset the connection
S (SYN)	Synchronize sequence numbers
F (FIN)	No more data from sender

10.10 SOCKETS

A **socket** is an endpoint for communication between client process and server process across a network. A **socket address** is the combination of an IP address and a port number. This address is used to send data packet to a particular process running on a machine. When two programs are executed, a client process and a server process are created, and these processes communicate with each other by reading from, and writing to, sockets.

There are few system calls, such as those given below:

1. Socket() returns a socket descriptor (like file descriptor), which is an integer value.
2. Bind() binds an address to a socket descriptor created by socket.
3. Listen() announces willingness to accept connections.
4. Accept() blocks the caller until a connection attempt arrives.
5. Connect() actively attempts to establish a connection.
6. Send() sends some data over the connection.
7. Receive() receives some data from the connection.
8. Close() releases the connection.

The following steps (as shown in Figure 10.10) occur when establishing a TCP connection between two computers using sockets:

1. The server instantiates a ServerSocket object, which denotes the port number on which the communication is to take place.
2. The server invokes accept() method of the ServerSocket class, which waits until a client connects to the server at the given port.
3. While the server is waiting, a client instantiates a Socket object, which specifies the server name and port number for the connection.
4. The constructor of the Socket class attempts to connect client to the specified server and port number. Once the communication is established, the client has a Socket object for communicating with the server.
5. The accept() method returns a reference to a new socket on the server that is connected to the socket of the client.

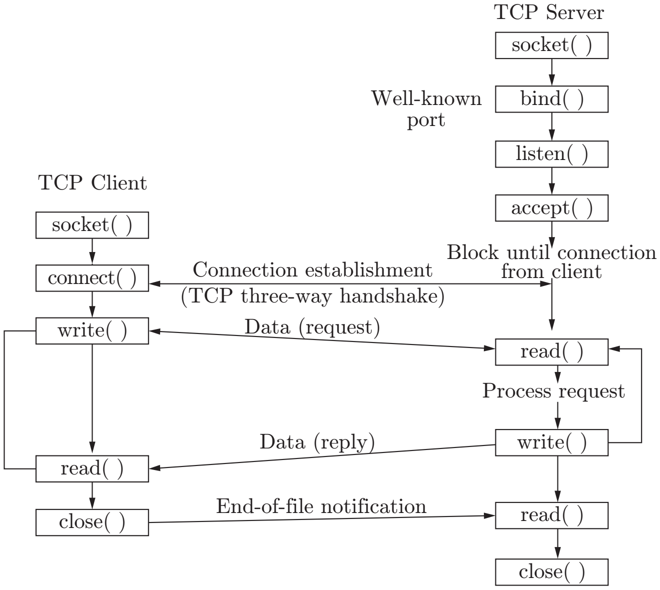


Figure 10.10 Socket creation.

10.11 LAYER 5: SESSION LAYER

The services provided by session layer are as follows:

1. Establishes, manages and terminates a communication session with remote systems
2. Allows two machines to enter into a dialog (communication may be half duplex or full duplex)
3. Adds checkpoints or synchronisation points to a data stream.
4. Groups several user-level connections into a single ‘session’.

Some protocol suites do not include the session layer.

Checkpoint: If we need a file of 1000 pages, it is suggested to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. Meanwhile, if a crash occurs during the transmission of page 324, the only pages that need to be resent after system recovery are pages 301 to 324. The pages from 1 to 300 need not be resent.

10.12 LAYER 6: PRESENTATION LAYER

The following are the major concerns of presentation layer:

1. Syntax and semantics of the information exchanged between two systems.
2. Support to different encoding schemes used by different machines. It converts the sender-dependent format into a common format and the receiver converts it back to the receiver-dependent format.
3. Data encryption—to ensure privacy, sensitive information should be encrypted. Information encrypted to some other form is unreadable to others.
4. Data compression—to reduce the size of information to carry. In the case of multimedia, for example, text, audio and video, compression is a very useful tool.

Note: Many protocol suites do not include a presentation layer.

10.13 LAYER 7: APPLICATION LAYER

The major responsibility of application layer is to implement communication between two applications of the same type. There is a common misconception that every user application runs on application layer, but it runs only on those applications which interact with the communication system. For example, FTP, HTTP, SMTP/POP3/IMAP (email)are all application layer protocols, but a designing software or text editor cannot be considered as an application layer protocol.The following are popular application layer protocols:

1. **Internet Control Message Protocol (ICMP):** ICMP provides error reporting and query management mechanism for host, which lacks in IP. ICMP messages are of two types: error-reporting messages and query messages. The header of ICMP is of 8 bytes and data section is of variable size. First byte is ICMP type, second byte is code, the next two bytes specify the checksum field and rest of the header is specific for each message type.
2. **Domain Name System (DNS):** DNS is a supporting program, which is used by other programs used by the users, such as email. DNS is a client–server program used to find the IP address of an email recipient. In DNS client–server application, a sender sends an email to the email address of the receiver. The DNS client sends the request to the DNS server to map the email address to IP address. DNS has three different domains: generic, country and inverse domains. Generic domain specifies the registered hosts according to their generic behaviour. For example, com, org, edu, gov, net, etc. The country domain uses two character country abbreviations, for example, 'in' for India. The inverse domain is used to map an address to a name (Fig. 10.11).

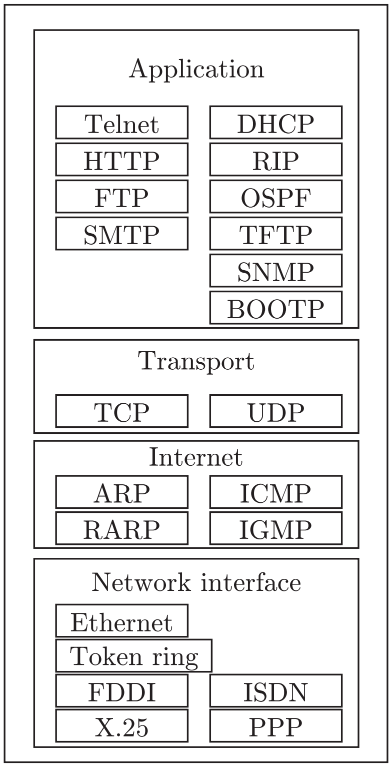


Figure 10.11 | TCP/IP protocols.

3. **Simple Mail Transfer Protocol (SMTP):** SMTP is a message transfer agent (MTA), which is used to transfer mail. A system should have client MTA for sending a mail and server MTA for receiving a mail. SMTP is used two times, between the sender and the sender's mail server and between the two mail servers. The main task of SMTP is to push the message from the client to the server.
4. **Post Office Protocol (POP):** POP3 (version 3) is a message access protocol which is used to extract the message for client to the server. It has two modes: the delete mode and the keep mode. In the delete mode, the mail is deleted from the mailbox after each retrieval, while in the keep mode, the mail remains in the mailbox after retrieval.
5. **File Transfer Protocol (FTP):** It is used for transferring files from one system to another. FTP establishes two connections between hosts, one for data transfer and the other for control information. FTP uses TCP Port 21 for the control connection

and TCP Port 20 for the data connection. The FTP client has three components: user interface, client control process and the client data transfer process. The server has two components: the server control process and the server data transfer process. The control connection remains open for entire FTP session, whereasthe data connection remains open for each file transfer.

6. **Hypertext Transfer Protocol (HTTP):** HTTP is used to access data on the World Wide Web (WWW). It works as a combination of FTP and SMTP. Unlike FTP, HTTP does not have any control connection and uses only one TCP connection. Unlike SMTP, it does not store and then forward the messages. It immediately sends the messages. HTTP uses a TCP Port 80.

10.14 DEVICES

The devices used for internetworking at different layers are specified in Fig. 10.12.

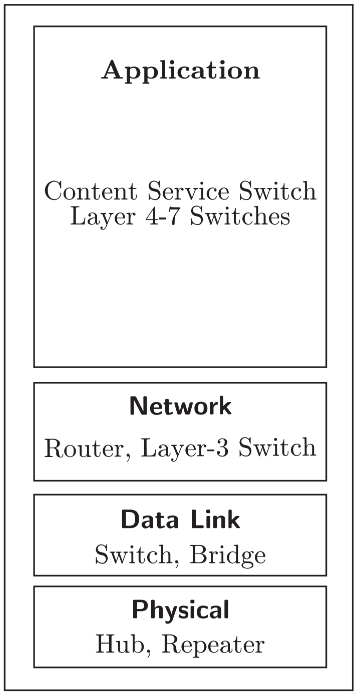


Figure 10.12 | Devices.

1. **Repeaters:** It is an electronic device which can receive the weak signal and retransmit it with higher speed. For example, if the LAN is connected for a long distance, then to cover a distance the signal is sent to a repeater which is attached with two LANs and retransmits the signal to a higher level. Thus, repeater connects two segments of network cable. It works at the physical layer of the OSI model.
2. **Bridge:** There is a limited number of stations that can be connected with a single LAN. So, a bridge is used to connect multiple LANs of the same type. It operates on a physical layer and data link layer. A bridge checks the physical address contained in the frame. It also uses table for filtering frames. It partitions the collision so that performance increases.
3. **Hub:** It is a network device which is used to connect various computers together. Hub is the central connection for all the computers, which connect through Ethernet. Hub can receive and send the information but cannot perform both tasks at the same time. This makes it slower than a switch. It is less expensive and less complex.
4. **Switch:** Switch is a small network device used to connect one or more computers through LAN. It is mostly used in home networks. Switches and hubs are used in the same network. Hubs increase the network by providing more ports, and switches divide the whole network into smaller networks. Switching reduces the amount of unnecessary traffic when every port sends the same information. Switch also reduces the possibility of collision in network.
5. **Router:** A router is a networking device which takes packets from one network and after analysis sends that packet to another network. When a data packet comes to the router, the router reads the destination address of the packet and sends it to the respective router which contains that destination address (listed in its routing table). A router is more intelligent than a hub because a hub only sends the information between the devices but the router analyses the packet and then forwards it to the other network. It controls the traffic on the network.
6. **Gateway:** It is a networking system capable of interconnecting one or more networks that has different base protocol. Gateway serves as an entry and exit point. Gateway, sometime called as protocol converter, is used in different layers. For example, a gateway can be used to convert a TCP/IP packet to a NetWare IPX packet.

Problem 10.12: If the capacity of router is 1 MB, data output rate is 8 Mbps. Tokens are generated at 6 Mbps. Calculate the time burst traffic is routed.

Solution: $C + \rho \cdot S = M \cdot S$

where M = output rate, C = capacity of router, S = time of burst traffic and ρ = token rate

$$10^6 + 6\times 10^6 \cdot S = (8\times 10^6)\cdot S$$
$$10^6(1+6S) = (8\times 10^6)\cdot S$$
$$1+6S = 8S$$
$$2S = 1 \Rightarrow S = 0.5 \text{ s}$$

10.15 NETWORK SECURITY

Network security is an activity designed to protect the network and data in terms of usability, reliability, integrity and safety. It targets a variety of threats and prevents them to enter into the network. It is accomplished through hardware and software, for example, firewall, anti-virus and anti-spyware, cryptography, intrusion prevention systems (IPS) used to identify fast-spreading threats such as zero-day attacks,and virtualprivate networks (VPNs) used to provide secure remote access.

Network security components are as follows:

1. **Confidentiality:** It ensures the concealment of data to unauthorised individuals.
2. **Integrity:** It ensures that information is changed in a specified and authorised manner. There is no change in content or source by an unintended user.
3. **Availability:** It ensures that systems are available for the authorised users.

The trio form the term CIA (Confidentiality, Integrity and Availability).

10.15.1 Basic Concepts in Cryptography

Cryptography is the science of providing secure communication over insecure channels. Cryptography consists of two operations: encryption and decryption.

Encryption is the process in which data is ciphering so that only the intended recipient can know the message. Decryption is the process of deciphering the message.

Basically, encryption and decryption are two functions of a cryptographic algorithm mathematically related to each other. A cryptographic algorithm is widely known, but a key, which is used for the encryption/decryption, is kept secret.

Cryptography is of two types: symmetric cryptography and asymmetric cryptography. Both the approaches have their own pros and cons.

10.15.1.1 Symmetric Cryptography

In symmetric key cryptography, a single shared key is used for both encryption and decryption as shown in Fig. 10.13. Symmetric cryptographic primitives use block ciphers, stream ciphers, cryptographic hash functions, and message authentication codes (MACs). Block cipher uses a deterministic algorithm and operates on a block (fixed length of bits) with unaltered transformation. Stream cipher encrypts each bit individually to generate cipher text. Hash functions or one-way hash functions are used to map an arbitrary-length message string to fixed-size message string. The final value is called hash value.

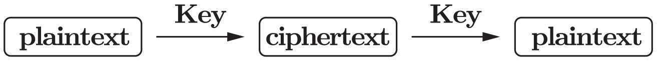


Figure 10.13 | Symmetric key cryptography.

The security of the symmetric key cryptography lies in the secrecy of the shared symmetric key. If the adversary captures the shared secret key, then it affects both confidentiality and authentication of the message. Examples of symmetric key cryptography are Twofish, Serpent, AES (Rijndael), Blowfish, RC4, RC5, 3DES, IDEA, SEAL, SNOW, etc.

10.15.1.2 Asymmetric Cryptography

In asymmetric cryptography, a private key is used for the decryption of a message while a public key is used for the encryption of the message as shown in Fig. 10.14. The private key needs to be kept confidential while the public key can be published freely. Asymmetric cryptography is also known as public key cryptography (PKC). PKC was introduced first by Diffie and Hellman in 1976. Public key algorithms are based on mathematical functions rather than substitution and transposition as in symmetric key cryptography. Examples of asymmetric key cryptography are Diffie-Hellman, RSA, Merkle-Hellman, Rabin, McEliece, El Gamal, Ellliptic curves, etc.

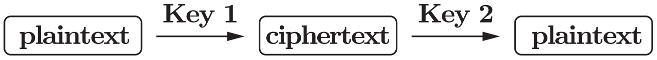


Figure 10.14 | Asymmetric key cryptography.

RSA Algorithm

The RSA algorithm, developed in 1977 by Riverst, Shamir, Adelman at MIT, provides encryption and digital signatures. RSA is based on factoring of large numbers, which is not known to be NP-complete.

Encryption and decryption are as follows for a plaintext block M and cipher textblock:

$$C = M^e \bmod n$$
$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

where n and e are known to both the sender and receiver, but$d$ is only known to the receiver. Thus, the public key is $P = \{n, e\}$ and the private key $S = \{d, n\}$. It is impossible to find d given e and n.

The detailed RSA algorithm is as follows:

1. **Key generation:**
   - Choose two prime numbers p and q, keep them secret.
   - Compute $n = pq$, n is public.
   - Calculate $\phi(n) = (p - 1)(q - 1)$
   - Choose e with $gcd(\phi(n), e) = 1;1 < e < \phi(n)$, which is also public.
   - Compute $d = e^{-1} \bmod \phi(n)$ and keep it private.
   - The private key consists of $S = \{d, n\}$ and public key consists of $P = \{n, e\}$.
2. **Encryption:**
   Plaintext $M < n$
   Cipher text $C = M^e \bmod n$
3. **Decryption:**
   Ciphertext C
   Plaintext $M = C^d \bmod n$

10.15.2 Digital Signature

It provides the authenticity of the origin of information to the user and verifies the information is intact. Hence, it provides authentication and data integrity. It also provides non-repudiation, which ensures that the sender cannot deny the origin of information. It is based on the public key cryptography concept.

10.15.2.1 Digital Standard Algorithm

Digital Standard Algorithm (DSA) is based on the difficulty of discrete logarithm problem (DLP). It is also based on Elgamal and Schnorr system.

DSA involves the following four steps:

1. **Key generation:**
   - *Global Public Components:* p is a prime number with 512-1024 bits, q is a prime divisor of $(p - 1)$ with 160 bits, g is an integer $g = h^{(q-1)/q} \bmod p$.
   - *Users Private Key:* x is random integer less than q.
   - *Users Public Key:* $y = g^x \bmod p$
2. **Signature:**
   - For each message M, generates random k
   - Computes $r = (g^k \bmod p) \bmod q$
   - Computes $s = k^{-1}(H(M) + xr) \bmod q$
   - Signature is (r, s)
3. **Verification:**
   - Computes $w = s^{-1} \bmod q$, $u_1 = H(M)w \bmod q$
   - Computes $u_2 = rw \bmod q$, $v = (g^{u_1} y^{u_2} \bmod p) \bmod q$
   - Verify if $v = r$
4. **Correctness:**

$$\begin{aligned}
v &= (g^{u_1} y^{u_2} \bmod p) \bmod q \\
&= (g^{H(M)w \bmod q} y^{rw \bmod q} \bmod p) \bmod q \\
&= (g^{H(M)w \bmod q} y^{xrw \bmod q} \bmod p) \bmod q \\
&= (g^{(H(M)w + xrw) \bmod q} \bmod p) \bmod q \\
&= (g^{(H(M)+xr)w \bmod q} \bmod p) \bmod q \\
&= (g^{(H(M)+xr)k(H(M)+xr)^{-1} \bmod q} \bmod p) \bmod q \\
&= (g^k \bmod p) \bmod q \\
&= r
\end{aligned}$$

10.15.3 Firewall

It is a device that filters access to the protected network from the outsider network. It is an integrated collection of security measures, which are designed to prevent unauthorised electronic access to a network system. It has a predefined set of rules, which can protect private network from unauthorised access by filtering incoming or outgoing traffic. These predefined set of rules are called firewall policies.

10.15.3.1 Functions of Firewalls

1. Examining the packet header and filtering
2. Verifying the IP address or the port
3. Granting and denying access

Packets can be filtered on the basis of the following criteria:

1. Source IP address
2. Destination IP address
3. TCP/UDP source port
4. TCP/UDP destination port

10.15.3.2 Types of Firewalls

1. **Packet Filter (stateless):** It is router-based filters. It does not use any context for filtering the packets. Individual packets are accepted or rejected. It has following limitations: (i) filter rules are hard to set up, (ii) inadequate primitives, (iii) hard to manage access to RPC-based services.
2. **Stateful Filter:** It maintains records of all connections passing through it. It determines if a packet is either the start of a new connection, a part of an existing connection or is an invalid packet. It maintains tables of each active connection, including the IP addresses, ports and sequence numbers of packets.
3. **Application gateway:** It works as a proxy. It is made up of bastion hosts, which run special software to act as a proxy server. It inspects the contents of the traffic, blocking inappropriate contents, such as websites, viruses, vulnerabilities, etc.

10.15.3.3 Limitations of Firewall

1. It cannot protect against attacks that bypass the firewall.
2. It cannot protect fully against internal threats.
3. It cannot provide protection against malicious code problems such as viruses and Trojan horses, although some are capable of scanning the code.

IMPORTANT FORMULAS

Application	Protocol
SMTP	TCP
TELNET	TCP
SSH	TCP
HTTP	TCP
DNS	TCP & UDP
PING	ICMP

1. In early token release,

   Throughput for single station or N stations

   $$= \frac{\text{Data}}{\text{Transmission time} + (\text{Ring latency}/\text{Number of stations})}$$

2. In delayed token release,

   Throughput for single station

   $$= \frac{\text{Data}}{\text{Transmission time} + \text{Ring latency} + (\text{Ring latency}/\text{Number of stations})}$$

   Throughput for N stations

   $$= \frac{\text{Data}}{\text{Ring latency} + (\text{Ring latency}/\text{Number of stations})}$$

3. $\text{Transmission time} = \dfrac{\text{Message size (bits)}}{\text{Bandwidth (bits/s)}}$

4. $\text{Propagation time} = \dfrac{\text{Distance}}{\text{Velocity}}$

5. For stop-and-wait ARQ

   $$\text{Transmission time} = \frac{\text{Message size}}{\text{Bandwidth}}$$

   $$\text{Propagation delay} = \frac{\text{Distance of the link}}{\text{Velocity}}$$

   Link utilisation of sender or throughput is given by

   $$\eta = \frac{\text{Transmission time}}{\text{Transmission time} + 2 \times \text{Propagation delay}}$$

6. Pure ALOHA

   Throughput $(S) = G \times e^{-2G}$

   Vulnerable time $= 2 \times T_{fr}$

   $S_{\max} = 18.4\%$

7. Slotted ALOHA

   Throughput $(S) = G \times e^{-G}$

   Vulnerable time $= T_{fr}$

   $S_{\max} = 36.8\%$

SOLVED EXAMPLES

1. Which layer is responsible for delivery from process to process?

(a) Network (b) Transport
(c) Physical (d) Data link

Solution: Transport layer is responsible for end-to-end process communication.

Ans. (b)

2. The minimum size of an Ethernet frame is

(a) 18 bytes (b) 64 bytes
(c) 46 bytes (d) 56 bytes

Solution: Ethernet header = 18 Bytes [Dest. Mac (6) + Source Mac (6) + Length (2) + CRC (4)] Minimum Data Portion = 46 Bytes and Minimum Ethernet Frame Size = 64 Bytes

Ans. (b)

3. Using a 7-bit sequence number, what is the maximum size (in bits) of the sender and receiver window using stop-and-wait protocols?

(a) 1 and 1 (b) 1 and 7
(c) 7 and 1 (d) 7 and 7

Solution: Stop-and-wait protocol uses 1 bit sequence.

Ans. (a)

4. Vulnerable time in CSMA is

(a) It is double the transmission time as it includes the sensing time
(b) Half of average frame transmission time as collisions are less
(c) 2 × (the average frame transmission time)
(d) None of the above

Solution: Vulnerable time for CSMA is the propagation time T_p needed for a signal to propagate from one end of the medium to the other.

Ans. (d)

5. Which class of IP addresses is used for multicasting?

(a) Class E (b) Class C
(c) Class A (d) Class D

Solution: Class D is used for multicasting. Refer Table 10.4.

Ans. (a)

6. The options field of IPv4 is used for?

(a) Time stamping, strict source routing
(b) Loose source routing, strict source routing
(c) Time stamping only
(d) Time stamping, loose source routing, strict source routing

Solution: It may contain values for various options, such as strict source routing, security, record route, time stamp, etc.

Ans. (d)

7. In a fully connected mesh network with d devices and c connections, there are ________ physical channels to link all devices.

(a) $d^*(d-1)/2$ (b) $d^*(d+1)/2^*c$
(c) $(2^*d + 2^*c)/2$ (d) $2^*(d+1)+c$

Solution: The total number of wired links required to establish a fully connected mesh network of d nodes can be calculated as $c = d\,(d-1)/2$.

Ans. (a)

8. In IPv4 header, the ______ field is used to determine to which datagram a newly arrived fragment belongs to.

(a) Identification (b) Datagram_id
(c) Fragment offset (d) Time to live

Solution: Identification field is used to identify original IP packet the fragments belong to.

Ans. (a)

9. Encryption and decryption is the responsibility of _______ layer.

(a) Session (b) Data link
(c) Application (d) Network

Solution: Application layer is responsible for the encryption and decryption processes.

Ans. (c)

10. Maximum throughput of an ALOHA network is

(a) 18.4% (b) 35.8% (c) 36.8% (d) 50%

Solution: When $G = 1$, the throughput is increased to the maximum value of 36.8%.

Ans. (c)

11. A terminal multiplexer has six 1200 bps terminals and 'N' 300 bps terminals connected to it. The outgoing line is 9600 bps. What is the maximum value of N?

(a) 4 (b) 6 (c) 8 (d) 12

Solution: Since, there are six 1200 bps terminals. So, $6 \times 1200 + n \times 300 = 9600 \Rightarrow n = 8$

Ans. (c)

12. The total number of wired links required to establish a fully connected mesh network of 9 nodes will be

(a) 36 (b) 56 (c) 72 (d) 64

Solution: Total number of wired links required to establish a fully connected mesh network of n nodes can be calculated as $c = n(n-1)/2$
So, for 9 nodes, total links are 36.

Ans. (a)

13. Considering a classful addressing, the IP address 128.252.144.84 denotes

(a) 0.0.0.0 as network ID and 128.252.252.84 as node ID
(b) 128.0.0.0 as network ID and 128.252.127.84 as node ID
(c) 128.252.0.0 as network ID and 128.252.144.84 as node ID
(d) 128.252.144.0 as network ID and 128.252.144.84 as node ID

Solution: The IP belongs to class B. The network id is 128.252.0.0 and the node id is 128.252.144.84.

Ans. (c)

14. In Ethernet CSMA/CD, the special bit sequence transmitted by media access management for collision handling is called

(a) Hamming code (b) CRC
(c) Jam (d) Preamble

Solution: Hamming code is a set of error-correction code, which is used to detect and correct bit errors. CRC (cyclic redundancy check) is used to detect data transmission errors.
Preamble is used in network communications for synchronizing transmission time between systems.

Ans. (c)

15. A Gateway operates at ______________ layers.

(a) All layers except physical and application layer
(b) All the seven layers
(c) Only on session, transport and network layers
(d) Same layers on which the switch and bridge operates

Solution: Hubs, repeaters(or active hubs) operate at physical layer (1); Bridges, Switches operates at data link layer (2) ; Routers operate at network layer (3) ; content-switches (or web-switches or application-switches) operates at layers 4 to 7. Gateway operates at all the seven layers–physical, data link, network, transport (4), session (5), presentation (6), application (7).

Ans. (b)

16. Consider the network with subnet mask 153.224.0.0/13. Determine the last host address in the network.

(a) 153.208.255.255
(b) 153.224.255.254
(c) 153.231.255.255
(d) 153.231.255.254

Solution: Network id: 153.224.0.0
Subnet mask: 255.248.0.0
By doing XOR between network id and subnet mask, one gets network id= 153.231.0.0
We have 19 bits for host, so first host address is 153.224.0.1 and last host is 153.231.255.254

Ans. (d)

17. Consider a token ring LAN of length 12 km and having 40 stations, signal propagation speed is 8 ns/m and data rate is 100 Mbps. An average frame contains 220 bytes. Delay occurred at each station is equivalent to 10-bit delay. What is the utilisation of token ring approximately?

(a) 10% (b) 12%
(c) 15% (d) 21%

Solution: Transmission time of a frame

$$T_t = \frac{220 \times 8}{100 \times 10^6}\ \text{s} = 17.6\ \mu\text{s}$$

Propagation time around the ring

$$T_p = 10 \times 12000\ \text{ns} = 120\ \mu\text{s}$$

Delay at each station = 1/10 µs, so delay at 40 stations will be = 4 µs
Now,

$$\text{Utilisation (U)} = \frac{T_t}{T_t + T_p + \text{delay at all stations}}$$
$$= \frac{17.6}{17.6 + 120 + 4} = \frac{17.6}{141.6}$$
$$= 0.124 \times 100 = 12.4\% \approx 12\%$$

Ans. (b)

18. A CSMA/CD-based network has transmission rate 120 Mbps, length 1 km and speed of signal is 10^9 m/s. What should be the minimum frame size?

(a) 120 B (b) 240 B
(c) 400 B (d) 440 B

Solution: We know that

$$\frac{\text{Length of packet}}{\text{Bandwidth}} = 2 \times \frac{\text{Distance}}{\text{Velocity}}$$

So, packet length $= 2 \times \frac{1000}{10^9} \times 120 \times 10^6 \Rightarrow 240$ bytes

Ans. (b)