ਜਗਤ ਗੁਰੂ ਨਾਨਕ ਦੇਵ
ਪੰਜਾਬ ਸਟੇਟ ਓਪਨ ਯੂਨੀਵਰਸਿਟੀ
ਪਟਿਆਲਾ

JAGAT GURU NANAK DEV
PUNJAB STATE OPEN UNIVERSITY, PATIALA

(Established by Act No. 19 of 2019 of the Legislature of State of Punjab)

The Motto of the University

(SEWA)

SKILL ENHANCEMENT EMPLOYABILITY ACCESSIBILITY WISDOM



M.SC. (COMPUTER SCIENCE)
SEMESTER-III
COURSE: COMPUTER NETWORKS

COURSE CODE: MSCS-3-03T

ADDRESS: C/28, THE LOWER MALL, PATIALA-147001
WEBSITE: www.psou.ac.in

SELF-INSTRUCTIONAL STUDY MATERIAL FOR JGND PSOU, ALL COPYRIGHTS WITH JGND PSOU, PATIALA

M.Sc. (Computer Science)
Semester-3
MSCS-3-03T: Computer Networks

Total Marks: 100
External Marks: 70
Internal Marks: 30
Credits: 4
Pass Percentage: 40%

INSTRUCTIONS FOR THE PAPER SETTER/EXAMINER

1. The syllabus prescribed should be strictly adhered to.
2. The question paper will consist of three sections: A, B, and C. Sections A and B will have four questions from the respective sections of the syllabus and will carry 10 marks each. The candidates will attempt two questions from each section.
3. Section C will have fifteen short answer questions covering the entire syllabus. Each question will carry 3 marks. Candidates will attempt any ten questions from this section.
4. The examiner shall give a clear instruction to the candidates to attempt questions only at one place and only once. Second or subsequent attempts, unless the earlier ones have been crossed out, shall not be evaluated.
5. The duration of each paper will be three hours.

INSTRUCTIONS FOR THE CANDIDATES

Candidates are required to attempt any two questions each from the sections A and B of the question paper and any ten short q questions from Section C. They have to attempt questions only at one place and only once. Second or subsequent attempts, unless the earlier ones have been crossed out, shall not be evaluated.

Course: Computer Networks	
Course Code: MSCS-3-03T	
Course Outcomes (COs) After the completion of this course, the students will be able to:	
CO1	Build an understanding of the fundamental concepts of computer networking.
CO2	Identify the different types of network topologies and protocols.
CO3	Understand the concept of networking models, protocols, functionality of each layer
CO4	Identify the function of a firewall, and how it keeps a computer secure and safe from viruses.
CO5	Understand the concept of Firewalls for Network Security.

SECTION-A

Unit I: Basic Concepts: Components of data communication, modes of communication, standards and organizations, Network Classification, Network Topologies; Transmission media, network protocol; layered network architecture.

Unit II: Models: Overview of OSI reference model; TCP/IP protocol suite. Physical Layer: Cabling, Network Interface Card, Transmission Media Devices- Repeater, Hub, Bridge, Switch, Router, Gateway; Transmission impairments.

Unit III: Data Link Layer: Framing techniques; Error Control; Flow Control Protocols; Shared media protocols - CSMA/CD and CSMA/CA.

Unit IV: Network Layer: Virtual Circuits and Datagram approach, IP addressing methods - Sub netting; Routing Algorithms (adaptive and non-adaptive)

SECTION-B

Unit V: Transport Layer: Elements of transport protocols – Addressing, Connection establishment and release, Flow control and buffering, Transport services, Transport Layer protocol of TCP and UDP.

Unit VI: Session and Presentation Layer: Session Layer – Design issues, remote procedure call. Presentation Layer – Design issues, Data compression techniques, Cryptography.

Unit VII: Application Layer: Application layer protocols and services – Domain name system, HTTP, E-mail, WWW, telnet, FTP, SMTP.

Unit VIII: Network Security: Common Terms, Firewalls, Virtual Private Networks

Reference Books:

- B.A. Forouzan, “Data Communication and Networking”, 4th Edition, Tata McGraw Hill, 2017.
- A. S. Tanenbaum, “Computer Networks”, 5th Edition, Pearson, 2011
- D.E. Comer, “Internetworking with TCP/IP”, Vol. I, Prentice Hall of India, 2015
- W. Stalling, Data & Computer Communication, 8th edition, Prentice Hall of India, 2013
- D. Bertsekas, R. Gallager, Data Networks, 2nd edition, Prentice Hall of India. 1992.

UNIT I: INTRODUCTION TO COMPUTER NETWORKS

STRUCTURE

1.0 Objectives

1.1 Introduction

1.2 Components of Data Communication System

1.3 Modes of Communication

1.3.1 Simplex

1.3.2 Half Duplex

1.3.3 Full Duplex

1.4 Standards and Organizations

1.5 Network Classification

1.5.1 Personal Area Network

1.5.2 Local Area Network

1.5.3 Metropolitan Area Network

1.5.4 Wide Area Network

1.6 Network Topologies

1.6.1 Bus Topology

1.6.2 Ring Topology

1.6.3 Star Topology

1.6.4 Mesh Topology

1.6.5 Tree Topology

1.6.6 Hybrid Topology

1.7 Transmission Media

1.7.1 Guided Media

1.7.2 Unguided Media

1.8 Network Protocols

1.9 Layered Network Architecture

1.10 Summary

1.11 Review Questions

1.0 OBJECTIVES

- Understanding the data communication system
- Analyzing the differences among modes of communication
- Identifying the organizations for standardization in data communications
- Understanding the network types and topologies
- Understanding the type of transmission media
- Knowing the concept of protocols

1.1 INTRODUCTION

Data communication is the exchange of data between two devices using any communication media. The main objective of data communication is to enable exchange of data between any two points in the network. For data communication both hardware and software are required. This module provides an introduction to data communication and covers fundamental topics related to the actual transmission of data.

1.2 COMPONENTS OF DATA COMMUNICATION SYSTEM

There are five constituent components of data communication.

- Data
- Sender
- Receiver
- Communication media
- Communication protocols

Data: It is piece of information that may be a text, images, audio or video file.
Sender: It is a source from where communication will start. It can be any device eg. computer, mobile, laptop etc.
Receiver: It is a destination where the data communicated will be received. Like sender, receiver can also be a computer, mobile, laptop etc.
Communication media: It is a medium through which data will be physically communicated from sender to receiver. It can be air, cable or optical fibre.
Communication protocols: It is the set of rules agreed upon by sender and receiver for communicating, in absence of rules the communication cannot happen.
A typical data communication system can be represented in fig. 1, where the data communication has been depicted.
In case of communication among two persons, at any time one is sender, other is receiver and voice or sound is communicated through air. The protocol in this communication is to choose a common language for communication and preferably one should speak i.e. send, at any time and other should listen i.e. receive.

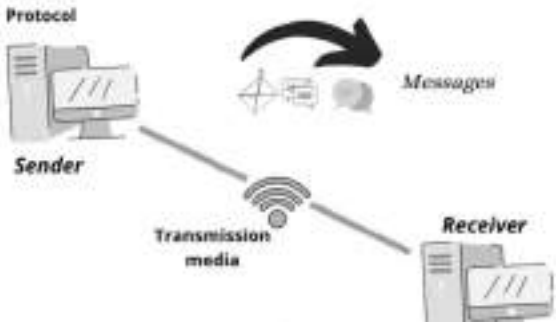


Fig 1: Data Communication System

Similar example in case of email communication will have sender, receiver, data to be emailed, communication through internet, which is wireless/ wired connectivity and the most commonly used protocol - simple mail transfer protocol (SMTP).

1.3 MODES OF COMMUNICATION

The direction of exchange of data between two devices connected through communication media is termed as communication mode. Many such interconnected devices are termed as data communication network.

There are three types of modes of communication:

- Simplex
- Half duplex
- Full duplex

1.3.1 Simplex

This is the simplest mode of communication. The communication is this mode is unidirectional. The sender can only send data and receiver can only receive it.. It is like a one-way traffic, where the vehicles use the full road width to commute and there are no traffic issues. Similarly in mode, the full capacity of communication media channel is used by sender for transmission so it is fast. Radio and television transmission are its examples.This mode suffers from drawback that even the acknowledgement of data received cannot be communicated to sender.

1.3.2 Half Duplex

In this mode communication can be either way between two devices. Both the connected devices can act as sender or as receiver. At any point of time one device will act as sender and other as receiver and vice-versa. This mode also has an advantage of using full capacity of communication media channel. Walkie-Talkie, internet browsers are the examples of this mode of communication.

This mode suffers a drawback that there may be delay in communication as at any given point of time it is still unidirectional.

1.3.3 Full Duplex

The communication in this mode is bidirectional. Both the connected devices can send and receive data simultaneously at any point of time. It is like two-way traffic where vehicles can move in both the directions but the width of road is divided into two paths for accommodating traffic moving in both directions. Mobile phones is example of this mode.

The communication is fast as compared to other two modes but it suffers from a drawback that the communication channel in any direction is half as compared to simplex or half duplex mode.

Amongst the three modes, simplex uses full channel capacity whereas half duplex uses full capacity for communication along one direction at a time and full duplex uses half the capacity for communication in any direction. The bandwidth which is the amount of data communicated between the connected devices in given amount of time is more in full duplex as compared to other two modes. Fig. 2 depicts these three modes of communication.



Fig 2a: Simplex Mode

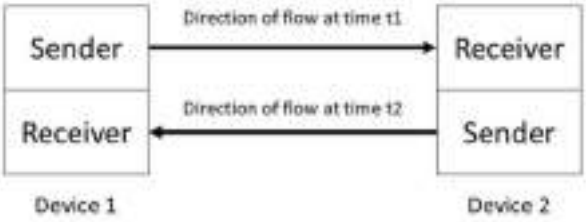


Fig 2b: Half Duplex Mode

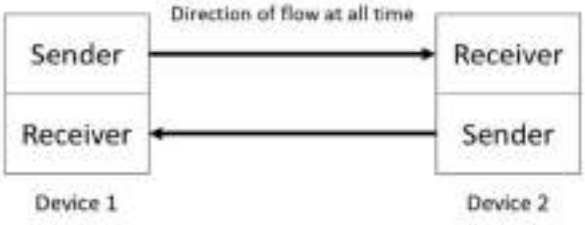


Fig 2c: Full Duplex Mode

1.4 STANDARDS AND ORGANIZATIONS

Standards are required for interoperability at national and international levels. There are organizations that define and maintain the set of rules, guidelines or protocols termed as standards for data communication.These organizations work in association with government

and manufacturers to develop, coordinate and maintain the standards for data communication. The major standards organizations for data communication are:

International Organization for Standardization (ISO): It is a non-governmental, international organization having membership from national standards body. The major functionality of ISO is to develop models and standardsensuring system compatibility, quality and cost effectivity. Open System Interconnection (OSI), a basic reference model for networks was developed by ISO.

International Telecommunications Union-Telecommunication Sector (ITU-T): The most significant role of ITU-T is to set and regulate the standards for telecommunications worldwide. The major functionalities of ITU-T are standardizing telecommunication technologies, services, tariffs etc. The play a vital role in allocating radio frequency bands for different services. ITU-T defines series of communication standards, most popular among them are V-series that define methodologies for data exchange over public telephone system by using modems etc.

Institute of Electrical and Electronics Engineers (IEEE): It is an international professional body with members from different nations. IEEE standards association (IEEE-SA) is the global standard developer for industry including telecommunications and information technology. IEEE 802 family of network standards is the most commonly adopted by the industry for wireless and wired communications.

American National Standards Institute (ANSI): It a private organization in United States (US) that manages voluntary standards system in the country. The major role of ANSI is to being together the various stakeholders i.e. government, industry and users to voluntarily develop and consensus standards. American Standard Code for Information Exchange (ASCII), a character encoding standard widely used for data communication is a contribution of ANSI.

Electronic Industries Association (EIA): It is aligned with ANSI to promote standards for electronic equipment manufacturing for data communication. The physical connection interface standards were defined by EIA. The most commonly used connector Recommended Jack #45 (RJ-45) in networking is the result of standards by EIA.

Internet Engineering Task Force (IETF): It is the leading internet standards organization. With a focus to develop open standards using open processes for enhancing quality of internet. It is large open community of network designers, internet vendors, researchers and users. IETF is responsible for managing internet protocol stack, including Transmission Control Protocol and Internet Protocol (TCP/IP).

Telecommunications Industry Association (TIA): It is an association of global manufacturers and suppliers of hi-tech Information and Communication Technology (ICT) products. It prepares guidelines to produce equipment like cellular towers, telephone terminal equipment, satellites, healthcare ICT, mobile device communication etc.

Internet Architecture Board (IAB): It is a committee of IETF to provide architectural oversight on internet protocols and to give technical direction for internet development and evolvement as a global communication platform. The major functions of IAB are to provide foundation for privacy and security in internet, to provide direction to integrate internet with IoT and mobile networks and to promote open internet without special controls.

Internet Research Task Force (IRTF): It focuses on promoting research for evolution of internet by creating research groups for long term research goals. It works in parallel to IETF that focuses on short term goals. Primary task of these groups are to work on topics related to internet protocols, applications and architecture.

Internet Society (ISOC): It manages internet standards and works for policy development. ISOC has mission to ensure open internet development.

Internet Corporation for Assigned Names and Numbers (ICANN): It majorly works for IP address space allocation and top-level domain name system management.

These organizations work to create uniformity across all the stakeholders i.e. manufacturers, consumers and government agencies. Their primary function is doing research, coordinating with other organizations, developing and maintaining technical standards for communication and networks.

1.5 NETWORK CLASSIFICATION

There are different network architectures and protocols with different characteristics and application domains. Networks can be classified according to scale or scope and transmission technology.The networks can be classified based on their scale i.e. the physical size as:

1.5.1 Personal Area Network (PAN)

As the name suggests, it is for a person. It is generally a wireless communication network where personal devices autonomously detect and connect with each other in the range of upto 30 fts. Smartphones, laptops and other wearables devices comes under this category.

1.5.2 Local Area Network (LAN)

It is a small size network covering few buildings or a campus upto few kilometers. It is mainly used to connect workstations or personal computers in offices, factories, institutes for resource sharing. Sharing of printer is a typical example of LAN.

1.5.3 Metropolitan Area Network (MAN)

It is a network covering buildings to cities. It generally covers the geographical area of about100km andusually interconnects LANs using optical fibre links. A cable television network, fiber distributed data interface (FDDI) and asynchronous transfer mode(ATM) are examples of MAN.

1.5.4 Wide Area Network (WAN)

It covers large geographical area such as country or entire continent for long-distance transmission of data. The communication speed is slow as compared to LAN and MAN. WAN is not owned by any organization like LAN. Internet, 3G and 4G mobile broadband services are examples of WAN.

Fig 3 diagrammatically represents the networks based on physical size.

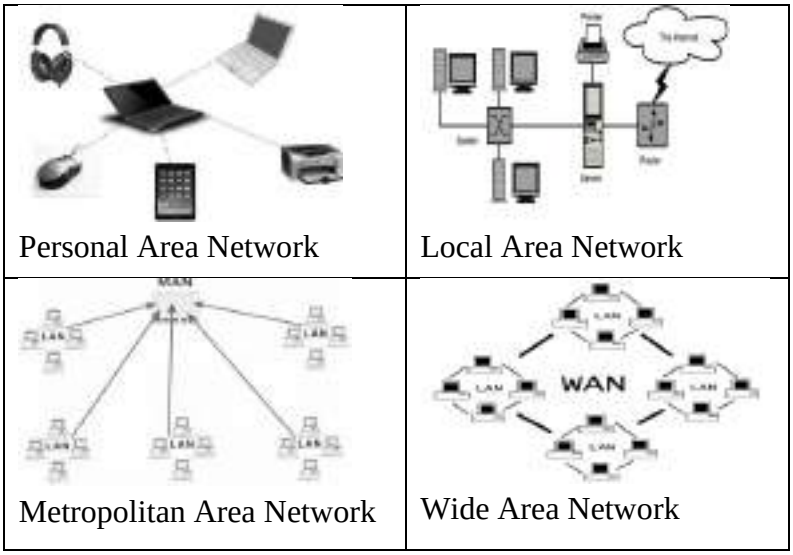


Fig 3: Types of networks based on physical size

1.6 NETWORK TOPOLOGIES

Network topology refers to the arrangement of network comprising of placement of nodes and links and also the physical or logical description of links. The physical description focuses on how the network is actually laid whereas the logical description defines the actual flow of data in the network.

Network topology plays an import role in functionality of the network. Choosing right topology helps in enhancing performance, ease in finding faults, effective resource sharing.

The two main categories of network topology are

- **Point-to-point topology:** A simplest topology where two devices or nodes are connected through a common link. It has the advantages of providing highest bandwidth, simple and fast, low latency and easy to maintain. Personal computer connected to printer is an example of point-to-point topology.
- **Multipoint topology:** In this topology three or more nodes are connected through a single transmission media. Buss or mesh type topologies are example of multipoint topologies.

Multipoint topologies can be categorized as:

1.6.1 Bus Topology

It consists of multiple nodes connected through a single cable with a terminator at each end. One of the nodes act as sender or server and transmits data form one end to another. As the data reaches to each node, it checks the address (MAC or IP address), if it matches with its address, the node processes that data otherwise it just passes the data. Once the data reaches to other extreme, the terminator removes the data from the line. The main line or cable acts as the backbone for the complete network in this topology. It is also known as multi-drop, linear or horizontal bus and is mainly suited for small networks like LAN. Fig 4 represents the bus topology.

Advantages:

- Simple to install and easy to add nodes

- In case of failure of node, other nodes will not be affected
- Less cabling cost as compared to other topologies
- Cost efficient

Disadvantages:

- Strength of signal decreases when nodes are more, leading to low efficiency
- The entire network fails if the backbone cable is damaged
- Congestion and traffic on bus as there is single communication media connecting nodes
- Due to limited bus length, limited number of nodes can be connected
- Broadcasting of message leads to security issues and risks

1.6.2 Ring Topology

In this topology, each node is connected to two neighbouring nodes on either side and form the shape of ring. The message by the sender travels through the ring in the same direction either clockwise or anticlockwise, till it reaches to receiver.

The data flow in this topology is based on token passing, only the node having token can transmit. The receiver takes data from token and sends back it to sender as an acknowledgement. The token is regenerated and is passed in the ring again.

Advantages:

- Chance of packet collision is minimal
- Transmission speed is high due to Unidirectional data
- Easy installation, robust and easy maintenance
- Less costly as compared to other topologies eg. mesh, hybrid, tree

Disadvantages:

- Failure of any load will lead to network failure
- Data packet has to pass through all the nodes causing security issues
- Addition and removal of any node is difficult

1.6.3 Mesh Topology

It's a kind of point to point topology where each node is connected to every other node.

The total number of simplex links in a mesh topology is n(n-1) and for full duplex it will be n(n-1)/2; where n is total number of nodes in the topology.

Mesh can be full mesh or partial mesh depending on all the nodes of partial nodes are connected with each other.

Advantages:

- Dedicated links to facilitate one to one communication
- No congestion on the channel
- In case of failure of single node the communication can be routed through other nodes
- Privacy and security is maintained

Disadvantages:

- Cost of cabling is high

- Installation is difficult

1.6.4 Star Topology

It is point to point connection topology where every node is connected to the centralized hub. Nodes are not directly connected to each other like in mesh. Sender sends the data to hub, which further sends it to receiver. The hub finds the addresses of all the receiver node and communicates the data.

Advantages:

- Easy to maintain and troubleshoot
- Failure in cable will affect only a single node
- Less expensive, easy to add new nodes

Disadvantages:

- Failure of central hub will lead to network failure
- Central hub has limitation to handle the nodes due to limited input-output ports

1.6.5 Tree Topology

In this topology nodes are connected directly or indirectly to main bus cable like branches of a tree. It is a combination of star and bus topologies where number of star networks are connected using bus.

Advantages:

- Large distance networks can be covered
- Failure of any node will not affect any other nodes
- It is scalable topology with easy in fault finding

Disadvantages:

- More cabling required as compared to star and bus topologies
- Configuration is complex
- The failure of main bus will lead to failure of complete network

1.6.6 Hybrid Topology

It is a combination of two or more topologies. It can be combination of any of the five topologies discussed above. It can be star-ring or star bus topologies.

Advantages:

- It has the advantages of multiple topologies being used
- It is flexible, reliable and scalable

Disadvantages:

- Design is complex
- It is expensive as compared to other topologies

Fig. 4 represents the type of network topologies commonly used in data communications.

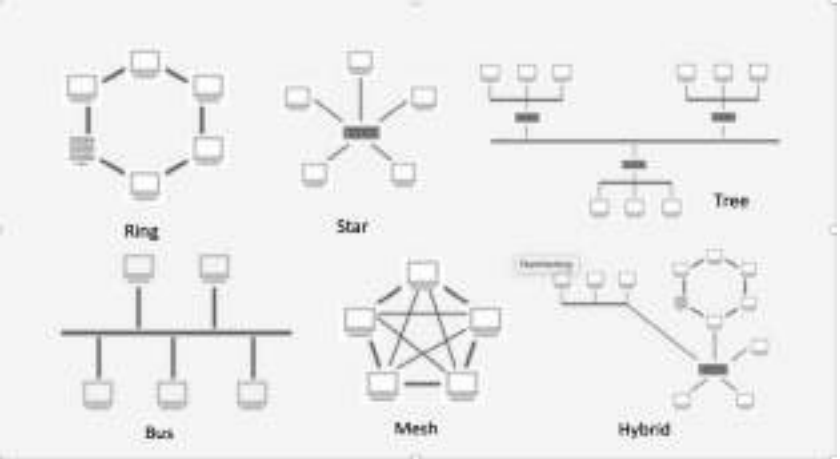


Fig 4: Types of network topologies

1.7 TRANSMISSION MEDIA

A physical path or channel between sender and receiver for transmitting data is a transmission media. Data can be in the form of electromagnets signals or light pulses depending of the type of media.

Transmission media can be categorized into guided and unguided media.

1.7.1 Guided Media

It is a wired or bounded media. The signals are transmitted through a defined path by using physical link. This type of media is secure, provides high speed. It is used for small distances. There are three kinds of guided media used for communication.

- **Twisted pair cable:** The two insulated conductor wires are wound about each other to make twisted pair. Several such pairs are bundled together in a proactive sheath. It is most commonly used transmission media. It is of two types:
    - **Unshielded Twisted Pair (UTP):** During transmission, this cable can block the external interference without using protective shield. It is majorly used for telephonic applications, LAN networks. It is used for short distance transmission due to attenuation. It is easy to install, comparatively cheap and has high speed.
    - **Shielded Twisted Pair (STP):** The cable consists of protective shield to block external interference. It is used where fast data rate is required. Its performance is better than UTP as it eliminates cross talk and provided better communication speed. STP is more expensive and bulky than UTP. It is mainly used in the network with high electro-magnetic interference. IBM token ring topology uses STP.

- **Coaxial Cable:** It is commonly termed as “coax”. It consists of inner conductor surrounded by concentric conducting shield with insulator in-between the two. Coax has outer jacket of insulating material. Its major application is in cable TV network. It is less expensive as compared to twisted pair cable. It is easy to install and provides

high bandwidth which is the maximum amount of data transmitted in given amount of time. Any failure in the cable disrupts the entire network.

- **Optical Fibre Cable:** Optical fibre is a thin strand of glass acting as a wave guide for transmitting light signals over long distance. It works on the principal of total internal reflection. It has two layers; core and cladding surrounding core. Cladding helps in attaining total internal reflection of light in core. Based on the size of core, it can be categorized into single-mode or multi-mode. It is used in telecommunications, medical use etc. It provides very high bandwidth, fast speed and negligible interference.It is light weight. Handling of optical fibres is difficult as it is very fragile. Its cost is high as compared to other two cables.

Fig. 5 shows the guided transmission media i.e. two types of twisted cables, UTP & STP, coaxial cable and optical fibre

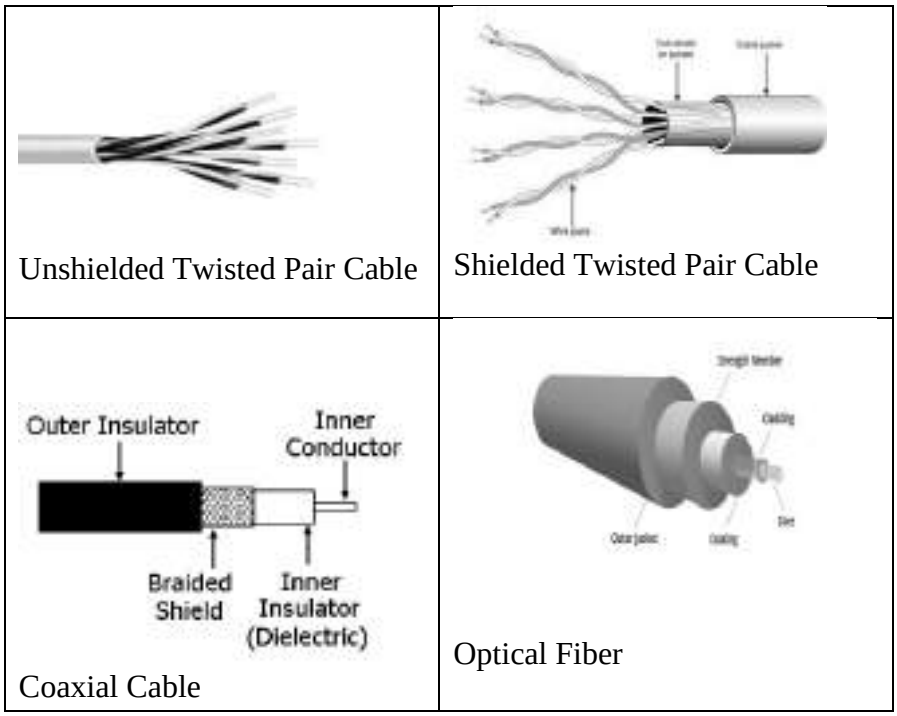


Fig 5: Types of guided transmission media

1.7.2 Unguided Media

It is wireless or unbounded media. There is no physical media for transmitting data but air. Antennas are used to send and receive data. It is generally used for covering large distances. Data is not secure while transmitting through this media. It is of three types.

- **Radio Transmission:** The radio waves used for this type of transmission are low frequency electro-magnetic waves ranging from 3KHz to 1GHz. It can penetrate walls so the signal can be received in-side the building as well. It uses a sky propagation thus the signal can be broadcasted over large distance. It is used for AM and FM

radios. Simultaneous data transmission is a problem in radio waves due to its low frequency.

- **Microwave Transmission:** These waves have frequency ranging from 1 to 300GHz. These waves are unidirectional so it propagates in line-of sight mode. The sender and receiver antennas should be aligned for transmission and receival. They are used for mobile phone communication or for television distribution. It supports large bandwidth so large amount of data can be transmitted. The equipment cost and the installation cost is very high.

  It can be categorized into Terrestrial microwave transmission, where both sender and receiver antenna are on ground and Satellite microwave transmission, where sender antenna signal is received by satellite, amplified and transmitted to receiver antenna on ground. Terrestrial is used for telecommunication and satellite is used for global positioning system.
- **Infrared Transmission:** It has a frequency range from 300GHz to 400GHz. It is used for short range communication. It cannot penetrate the walls so the interference is less. It is mainly used for remote operations of home appliances and devices.

1.8 NETWORK PROTOCOLS

Set of rules determining how the data will be communicated between two nodes connected in a network. With the well-defined protocols the nodes can communicate with each other irrespective of differences in their structure, design or internal processes. Protocols are designed as per industry standard by the various organizations mentioned in the earlier section.

The major functionality of network protocols is to provide:

- means to identify the nodes in the network and to make wired or wireless connection for communication
- formatting rules for packaging, sending and receiving the data
- process for establishing and terminating connection
- acknowledgement services wherever required
- data compression and encryption features for faster and secure communication

Based on the primary actions performed by protocols they can be broadly categorized as:

- **Communication Protocols:** They allow the nodes or devices to communicate, transfer files or accessing the internet.
- **Network Management Protocols:** They specify the procedures for the optimal performance of the network.
- **Security Protocols:** They ensure the protection of data communicated through channels is protected from unauthorized users.

There are few common protocols which we come across on daily basis, hypertext transfer protocol (HTTP), short message service (SMS), simple mail transfer protocol (SMTP), internet message access protocol (IMAP) are few among those.

Protocols are designed based on a layered architecture of the networks such as the OSI reference model or TCP/IP model.

1.9 LAYERED NETWORK ARCHITECTURE

The whole process of data communication is divided into tasks whereas data communication network is divided into layers. Each layer handles the specified tasks. The purpose of layered architecture of network is to reduce the design complexity. Network is organized as layers build upon one another. Each layer get the services of layers below it and provides services to layers above it.

Advantages:

- It reduces the design complexity of network
- Due to the modular approach of the layered architecture, the changes made in one layer will not affect the other layers
- It provides the abstraction of logical communication with corresponding layer in another node. Logically nodes can communicate at different layers with another node.

The layered architecture has several advantages but layering add to overheads in communication.

There are three basic elements of layered network architecture :

- Service: set of actions that a layer offers to higher layers. Service and be accessed through service access point (SAP) by sending service data unit (SDU).
- Protocols: set of rules to exchange information among layers. It is done by exchanging protocol data unit (PDU).
- Interfaces: to communicate the data or message to another layer. Interface data unit (IDU) is shared among layers.

The two models in the networking based on the layered architecture are Open Source Interconnection (OSI) model and Transmission Control Protocol/ Internet Protocol (TCP/IP) model. OSI model is a generic model based on the functionality of layers while the TCP/IP model layout the standards on which internet was developed.

1.10 SUMMARY

This module covers the basics of data communications system, its components, modes of data communication i.e. simplex, half duplex and full duplex. The organizations that define standards for data communication for the global network or internet have been detailed. Various network topologies that are being used while setting the network have been discussed along with their advantages and disadvantages. In general hybrid topology is the most preferred that has the advantages of multiple topologies. According to the size and scope of the network, its categorization in PAN, LAN, MAN and WAN have been detailed. Transmission media plays a very major role in data communication. Characteristics of media decides the bandwidth it can manage also the reliability with which it can communicate data. Both the guided and unguided media have
their own utility in wired or wireless types of networks. Another important part in the data communication is the protocols following which the sender can communicate with receiver.

Broad categories of the protocols have been discussed in the chapter. Also, the concept of layering in the networks have been introduced in this module.

1.11 REVIEW QUESTIONS

a. List the five major components of data communication system.
b. Differentiate between half duplex and full duplex transmission mode. Give one example each for these modes.
c. What is the need for defining standards for data communication? List any three organizations that design the standards.
d. According to the physical size, how can we classify networks? The most commonly used internet will come under which specific category.
e. Differentiate between point to point and multipoint topologies.
f. Which topologies will result in network break down if only the single device goes down?
g. What are the advantages of mesh topology over ring topology?
h. Which topology is practically used in setting up a network? Specify the reasons for your answer.
i. Compare the types of guided transmission media. Which one would be preferred for transmitting data to a large distance?
j. Why the infrared waves used in the remotes of home devices do not interfere with each other?
k. Enumerate the importance of protocols in data communication.

REFERENCES & FURTHER READING

- Data Communication & Networking by BehrouzForouzan
- Computer Networks by Andrew Tannenbaum
- Data and computer Communications by William Stallings
- Computer Networking: A Top-Down Approach by James Kurose and K.W. Ross

UNIT 2: MODELS AND TRANSMISSION MEDIA

STRUCTURE

2.0 Objectives

2.1 Introduction

2.2 Network Architecture

2.3 Concept of Layered Network Architecture

2.4 Open Source Interconnection (OSI) Reference Model

- **2.4.1 Introduction to OSI Model**
- **2.4.2 Layered Architecture of OSI Model**
- **2.4.3 Description of Layers in the OSI Model**

2.5 TCP/IP Model

- **2.5.1 Introduction to TCP/IP Model & its layers**
- **2.5.2 OSI and TCP/IP Model Comparison**
- **2.5.3 Description of Layers in the TCP/IP Model**
- **2.5.4 Addressing in TCP/IP**

2.6 Physical Layer

- **2.6.1 Cabling**
- **2.6.2 Comparison of Optical Fiber and Copper Wire**
- **2.6.3 Network Interface Card**
- **2.6.4 Transmission Media Devices**

2.7 Transmission Impairments

2.8 Summary

2.9 Review Questions

2.10 References & Further Reading

2.0 OBJECTIVES

a. Understanding the concept and importance of layered network architecture
b. Understanding the OSI and TCP/IP model
c. Difference between OSI and TCP/IP model
d. Understanding the functioning of physical layer
e. Get introduced to commonly used devices in networks
f. Understanding impairments in transmission of data

2.1 INTRODUCTION

Data Communication Network has basic underlying reference model known as Open System Interconnection (OSI) model and practically working model which is TCP/IP model. The physical layer common in both of these models has specific functionality including the functioning of network devices for transmission of data. This module gives an understanding of OSI and TCP/IP model and covers the introduction of physical layer focusing on devices and flaws in transmission.

2.2 NETWORK ARCHITECTURE

Network architecture is the design of a computer network. It is a framework for the specification of a network's physical components and their functional organization and configuration, its operational principles and procedures, as well as communication protocols used.

2.3 CONCEPT OF LAYERED ARCHITECTURE

The major task of the network is to transfer data from sender node to the receiver node. This major task can be divided into well-defined smaller sub-tasks.

Each of this sub-task consist of various processes. Each sub-task will accept the output from previous sub-task as input for its processes and will provide its output as an input to next sub-task. Each of these sub-tasks is performed by a separate layer in the network architecture. Hence the name layered architecture. In general the complex tasks are assigned to higher layers and simpler ones are assigned to middle or the lower layers.

2.4 OPEN SOURCE INTERCONNECTION (OSI) REFERENCE MODEL

2.4.1 Introduction to OSI Model

OSI model was developed by International Organization for Standardization (ISO) as a reference model for computer networks. The objective to have a reference model was to allow different platforms which is a combination of hardware, software and operating system to communicate with each other. The protocols are well defined for each layer in OSI reference model.

The sub-tasks in OSI model are divided into seven categories, each one is taken up by each of the seven layers in the model. These layers from lower to higher are listed below:

1. Physical Layer
2. Data Link Layer
3. Network Layer
4. Transport Layer

5. Session Layer
6. Presentation Layer
7. Application Layer

The functionality of each of these layers is explained in next section.

2.4.2 Layered Architecture of OSI Model

The OSI model has seven layers with well-defined tasks. The layered architecture of the model has following features:

- Message or data from sender node to the receiver node pass through all the layers from top to bottom i.e. application to physical layer and then it is delivered to receiver through the wired or wireless transmission media.
- Message or data received by receiver node passes from physical layer to the application layer.
- While transmitting the data, the sender and receiver may not have point-to-point connection i.e. the data may have to pass from intermediary nodes to reach to the receiver node. In that case intermediary nodes only involve the first three layers i.e. physical, data link and network layer for passing the data to next node.
- The data link layer decides about the next intermediary node in the communication path whereas the network layer identifies the end receiver.

Fig. 1 depicts the data flow from sender to receiver through various layers of OSI model showing the intermediary nodes between sender and receiver for transmitting the data.

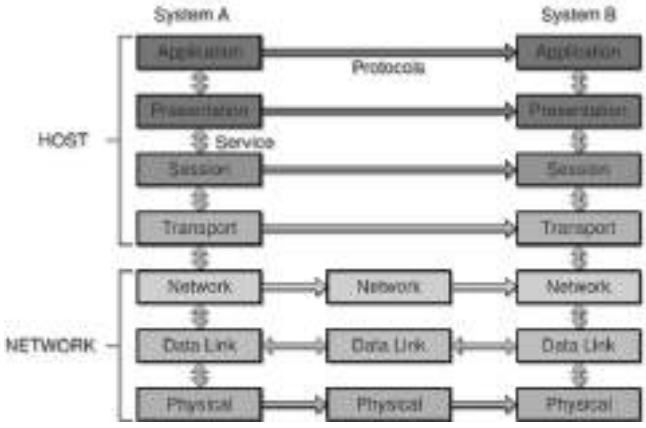


Fig 1: Open System Interconnection (OSI) model

- For the communication, at sender node, each layer add its information known as header, to the data or message received from the layer above. After adding the header information layer passes the message to the layer below it. This is known as data encapsulation and is represented in fig. 2 below.

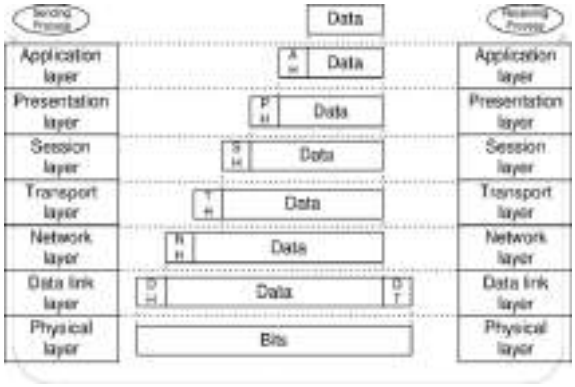


Fig 2: Header information at layers of OSI model

- At receiver node, the reverse process happens, each layer removes the information pertaining to the specific layer from the message received from lower layer and pass the message to the layer above.
- The function at each layer is specific and different from that of other layers.
- At sender, each layer calls for the service offered by layer below it but at receiver each layer call for the service offered by layer above it.
- The abstraction at the layer level provides the communication between the respective layers at sender and receiver nodes.
- The message among layers is communicated through the interface which defines the service that must be provided by that layer.

Fig 3., depicts the layered architecture with interfaces between layers.

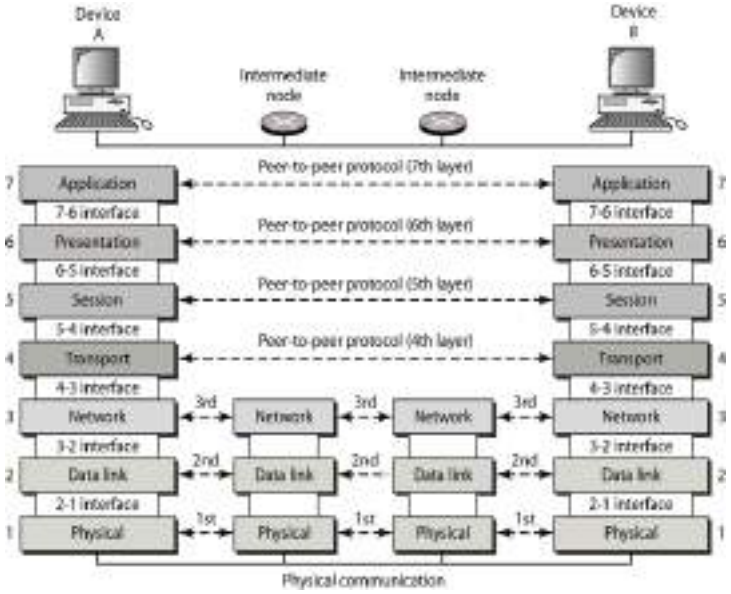


Fig 3: Interfaces and abstraction at layers in OSI model

2.4.3 Description of Layers in the OSI Model

The functions of each layer in the OSI model can be described as:

1. **Physical Layer:** The function of physical layer is to transmit the data through transmission media from sender to receiver.
    - **a.** A standard interface is provided by the layer to transmission media. This interface defines type of transmission media, its specifications and bit by bit delivery of message.
    - **b.** At sender, physical layer receives encapsulated message from data link layer and encodes the message into signal according to the transmission media.
    - **c.** At receiver side, the signal is received by receiver through the transmission media, after decoding, signal is sent to data link layer. Fig 4. Depicts the physical communication of data from sender to receiver.

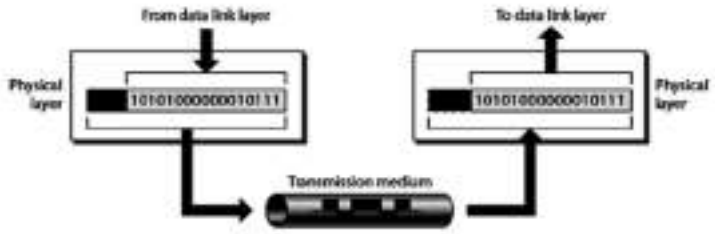


Fig. 4: Transmission to and from Physical layer

    - d. Physical layer converts the data from combination of 1's and 0's into signals from communication at sender and vice-versa at receiver.
    - e. The data transmission rate i.e. bits/ sec is decided and managed by physical layer.
    - f. The synchronization between sender and receiver is maintained by this layer. The same bit rate for both the nodes is pre-decided and data is transmitted in accordance to synchronized clocks.
    - g. The type of connection i.e. point-to-point or multi-point and accordingly the type of topology discussed in previous module, is defined by physical layer.
    - h. The transmission mode i.e. simplex, half-duplex or full duplex between sender and receiver is defined by this layer.

In totality, all the specifications and processes related to actual transmission of data from sender to receiver is defined by the first layer i.e. physical layer in OSI model.

2. **Data Link Layer:** The reliability in communication of data is provided by data link layer through error detection and correction process.
    - **a.** Data link layer on the sender side receives message from network layer and divides into smaller fixed size units termed as **frames,** which is further sent to physical layer for communication as a bit stream.

b. At receiver side this layer receives bit stream of message from physical layer and does the re-grouping to form a frame for sharing with the network layer. This process of re-grouping the bit stream into frames is known as **framing**.

c. The header information added by data link layer in frame includes the physical address of the sender and receiver in case the sender and receiver are on the same physical network otherwise instead of receiver it stores the physical address of the intermediary node to whom data would be communicated next.

d. Data link layer manages the flow control mechanism so as to avoid any loss or overflow of data due to speed mismatch of sender and receiver.

e. This layer offers error control by identifying lost or damaged frames and resending the duplicate frames.

f. This layer ensures hop-to-hop transmission of frames from sender to receiver.

3. **Network Layer:** The main function of network layer is to deliver data packets from sender to receiver across multiple hops.
    a. At sender side, network layer receives message from transport layer and divides it into packets.
    b. At receiver side the network layer receives frames from data link layer and converts it into packets and communicates to transport layer.
    c. To deliver the data from sender to receiver, the data is routed through various intermediary nodes.
    d. To recognize the devices in the network, this layer uses internet protocol (IP) addresses, which is a unique logical address for the nodes connected in network.
    e. Network layer's main function is routing, the packets are numbered and transmitted from sender to receiver through multiple paths based on congestion, number of hops etc. At receiver the numbered packets are combined in a sequence to receive the full message.
    f. During the packet transmission the intermediate nodes check about the best possible path for next hop.
4. **Transport Layer:** The message transmitted from sender to receiver may be for a specific process whereas multiple processes are running on sender and receiver nodes. This layer ensures the data to be delivered to the correct process at the receiver node and hence achieve process to process delivery.
    a. At sender the transport layer receives data from session layer and divides it into segments which are numbered and delivers it to the network layer.
    b. At receiver side the packets received from network layer are combined to make a segment to be further delivered to session layer.
    c. Every process is assigned a port number on the node, transport layer checks the port number at the receiver node to ensure the delivery to correct process.
    d. Transport layer provides connection oriented and connection less services for communication.
    e. Like data link layer, transport layer also provides flow control and error control but for end to end delivery.

5. **Session Layer:** This layer handles a task for establishing and synchronizing session generally termed as dialog between sender and receiver.
    a. At sender side, it accepts data from presentation layer add checkpoints, synchronization bits and delivers data to transport layer.
    b. At receiver, session layer receives data from transport layer removes checkpoints and delivers data to presentation layer.
    c. Checkpoints ensure that the data before checkpoint has been successfully received and will not be re-transmitted.

6. **Presentation Layer:** This layer helps overcoming the challenge of data transmission to devices on different platforms. It performs translation, encryption and data compression.
    a. At sender side it receives the data from application layer adds header containing information related to compression and encryption and delivers it to presentation layer.
    b. At receiver side it receives data from session layer, decompresses and decrypts the data as per the encoding scheme agreed upon by both sender and receiver and delivers it to application layer.

7. **Application Layer:** This layer provides the user interface to communicate data from sender to receiver by providing specific services, eg. email transfer which is known as X400 service, file transfer known as FTAM service etc.

The major functions of these seven layers can be summarized as:
Physical Layer: Transmitting data as a bit stream from one to another hop and providing specification of communication media and type of communication.
Data Link Layer: Hop to Hop delivery of frames to receiver
Network Layer: Delivery of packets from sender to receiver by deciding best routing path.
Transport Layer: Providing process to process delivery of message.
Session Layer: Establishing, managing and terminating the sessions.
Presentation Layer: Providing translation, encryption and compression for disparity in platforms of sender and receiver.
Application Layer: Providing the access of network to the user.

2.5 TCP/IP Model

2.5.1 Introduction to TCP/IP Model & its layers

The TCP/IP model is the practical model, that is currently being used in networks and internet. It was developed by Department of Defence, United States. This model is a collection of protocols termed as protocol suite. Based on the two major protocols i.e. Transmission Control Protocol (TCP) and Internet Protocol (IP), this model got its name as TCP/IP model. There are no standard specifications followed universally for describing TCP/IP model but it has fewer layers as compared to OSI model. For practical purpose this model divides the tasks in four layers, termed from lower to higher layer as:

1. Physical and Data Link Layer

2. Network Layer
3. Transport Layer
4. Application Layer

2.5.2 OSI and TCP/IP Model Comparison

Though the structure of OSI model and TCP/IP model is very similar but there are few differences in the two

- OSI model has seven layers but TCP/IP models has four layered architecture.
- The first layer of TCP/IP model performs the functionality as done by physical layer and data link layer of OSI model.
- The network layer and transport layers of TCP/IP performs the similar functionality as network layer and transport layers of OSI model.
- Application layer in TCP/IP models performs the functionality of application as well as session and presentation layers of OSI model.

2.5.3 Description of Layers in the TCP/IP Model

The functions performed by each layer in TCP/IP model is described below:

1. **Physical and Data Link Layer:** It is combinations of physical and data link layer of OSI model, all the functions performed by these two layers in OSI model is taken up by layer 1 of TCP/IP model. This layer supports all the standard protocols defined for these two layers.
2. **Network Layer:** This layer is also known as internetwork layer. The major protocol defined for this layer is IP. This layer is responsible for source to destination transmission of data.
    a. IP protocol is a connection-less and unreliable protocol. There is no error checking in this protocol. It simply sends the data and rely on the layers below to get the data transmitted to receiver node.
    b. The data is divided into same sized packets known as datagrams. Each datagram is numbered and is transmitted through different routes. At receiver side packets or datagrams are received out of order.
    c. No single connection is established between sender and receiver, and the packets are delivered through best possible route, so the services provided by network layer are connection-less services.
    d. There is no guarantee in IP for transmission of all the packets. During transmission some packets may get dropped due to congestion or other reasons. This makes IP an unreliable protocol.
    e. IP provides the maximum efficiency in data delivery as no error detection and error correction services are provided by this protocol, making the delivery fast.
    f. IP protocols takes the services of four protocols, ARP, RARP, ICMP and IGMP.

- **Address Resolution Protocol (ARP):** This protocol is used to resolve the physical address of the device in the network when the logical address of the node is known.
  The physical address or media access control (MAC) address is imprinted on the network interface card (NIC) or LAN card. Logical address is the IP address that uniquely identifies the node in the internet.
- **Reverse Address Resolution Protocol (RARP):** This protocol helps in identifying the logical or IP address when physical address is known.
- **Internet Control Message Protocol (ICMP):** This protocol provides the signalling mechanism. The intermediate devices eg. gateway sends the message to sender node if the datagram received by it from sender gets corrupted during transmission.
- **Internet Group Message Protocol:** This protocol has mechanism to deliver the message of corrupted datagram to a particular group of recipients.

3. **Transport Layer:** The protocols at this layer are responsible for process to process communication of data, similar to the transport layer of OSI model. The three major protocols in this layer are TCP, UDP and SCTP.
   - **Transmission Control Protocol (TCP):** It is a connection-oriented protocol, thus a reliable protocol. Before the actual transmission starts, it establishes the connection between sender and receiver.
     The data received from application layer is divided into segments and each segment is numbered.
   - **User Datagram Protocol (UDP):** Contrary to TCP protocol, it is connection-less and unreliable protocol. It does not provide flow control and error control mechanism.
   - **Stream Control Transmission Protocol (SCTP):** It is a relatively new protocol that combines the functionality of TCP and UDP protocols. Its major feature is to provide multi-homing, which is multiple connection paths establishment between sender and receiver nodes.
4. **Application Layer:** This layer in TCP/IP model is providing functionality of Session, presentation and application layers of OSI model. The high level protocols eg. file transfer protocol (FTP), virtual terminal protocol (TELNET), domain name service (DNS) are defined in this layer.

2.5.4 Addressing in TCP/IP

There are four different types of addresses defined in TCP/IP model and protocols are defined to resolve the addressing. These addresses are referred to as:

- **Physical Address:** It is the lowest level of addressing and is local to the network to which the node is connected. This address is unique. Media access control (MAC) address is 48 bits physical address printed on the network interface card (NIC) of the node. The size of the address may vary according to the type of network. In ethernet, MAC address is 48 bits (6 bytes).

- **Logical Address:** This logical address of IP address is used for uniquely identifying the node in the internet. The physical address is local to the network so while multiple networks are connected, there are chances of duplication of address. For sender to receiver communication, which are on different networks, physical address is not adequate to uniquely identify the nodes.

All the devices in the internet are uniquely identified by IP address. Two versions of IP addresses are commonly used to identify devices:

IPv4: It's a 32 bit logical address. It can identify maximum of 2^{32} unique devices in the internet. As the number of devices on internet is growing, IPv4 may not be able to allocate unique addresses to the devices.

IPv6: It's a 128 bit address and can support 2^{128} unique devices connected to internet.

- **Port Address:** The logical address helps in identifying the sender and receiver nodes connected to the internet but there may be multiple processes running on these sender and receiver nodes. Google meet, email and other processes are running on the devices, identifying the process for which the data is sent or received, is important. Port address uniquely identifies the process on the devices for which data is transmitted. It is a 16 bit address given to the process. The universal processes have been assigned same port numbers on all the devices, eg. http uses port address 80, telnet uses port address 23, smtp has port address of 25.

- **Application Specific Address:** For the same process there may be multiple windows or tables open for eg. multiple tabs on web browser. These multiple tabs for same process cannot be uniquely identified by port address, so to identify the unique instances of such processes, application specific address has been defined. Such addresses are specific to user so these are user friendly addresses. Eg. e-mail addresses (abc@gmail.com) and Universal Resource Locator (URL) (www.sggswu.edu.in) are user friendly specific addresses.

TCP/IP model is the practical model that has been evolved for internet.

2.6 PHYSICAL LAYER

Physical layer plays a major role in interaction of actual hardware and the signalling mechanism. The hardware equipment, cabling, transmission frequency, synchronization, pulses representing binary signal are all defined by the physical layer. This layer receives the data from data link layer in the form of frames, converts into a sequence of bits, transmits it to physical layer of receiver or of the intermediary node.

Information can be transmitted through cables in the form of electromagnetic signals. Data to be transmitted can either be digital or analog Digital signals like in file transfer can be represented as voltage pulses which is discrete in nature. Analog signals like voice is represented as a continuous electromagnetic waves.

The few terminologies are important for understanding the functioning of physical layer.

- **Channel Capacity:** The data transmission speed is termed as channel capacity. It depends on bandwidth, encoding and error rate.
  Bandwidth is limited by the transmission media, encoding depends on the levels used for signalling and error rate depends on the noise in the channel.
- **Multiplexing:** It is a technique to transmit multiple data streams meant for different receiving nodes over a single medium. Multiplexer is used to combine the data streams whereas de-multiplexer is required at receiver's end to separate out the information.
- **Switching:** It is a mechanism by which data is transmitted from sender to receiver that are not directly connected. In this case the data is transmitted to intermediary devices and forwarded to the next device closest to the receiver. The three different types of switching is circuit switching, message switching and packet switching.
  - **Circuit Switching:** In this technique, a dedicated link is established between the sender and receiver that includes fixed intermediary nodes eg.telephone network.
  - **Message Switching:** A dedicated link in circuit switching will be underutilized if no data is there for transmission between sender and receiver. To overcome this drawback, message- a logical unit of data was communicated through store and forward technique by dynamically deciding the route for communication.
  - **Packet Switching:** It is a type of message switching in which the data is divided in fixed size packets. It is also based on store and forward technique eg. internet.
- **Transmission modes:** The data in the form of 0s and s can be transmitted from sender to receiver in serial or parallel mode.
  - **Serial Transmission:** In this transmissionbits are sent in a queue one after the other. Single communication channel is required for such mode.
    It can be synchronous or asynchronous in nature. Synchronous mode is timing based transmission whereas in asynchronous bit pattern itself identifies the start and end of the transmission.
  - **Parallel Transmission:** Multiple bits are transmitted at the same time through multiple data lines. It is a high speed communication as compared to serial transmission.

2.6.1 Cabling

In the previous module, we discussed about two types of transmission media, guided and unguided. Guided media consists of twisted pair, coaxial and optical fiber cables. We will discuss the types of cables used for networking in this section.

- **Twisted Pair:** It is one of the oldest and the most common type of cable. Twisted pair of cables help in reducing noise by forming antenna by twisting the two cables.
  It can be used for transmitting either analog or digital data. These cables are commonly used due to its optimal performance and low cost.
  For LAN, type of coaxial cable termed as category 5 or Cat 5 is used. It consists of two insulated wires gently twisted together. Cat 5 has four such twisted pairs put together by insulation. The cables are categorized based on number of twists per meter. Cat 5 cables can be used for speed of 100mega bits per second (Mbps),

cat 5e can be used for speed of 1000Mbps etc. Fig. 5 shows the different categories of coaxial cables.

Cat 6 type of cables can support transmission at rate of 10 giga bits per second (Gbps). All these categories are of unshielded twisted pair (UTP) cables. Cables of Cat 7 category comes under shielded twisted pair (STP) cables.

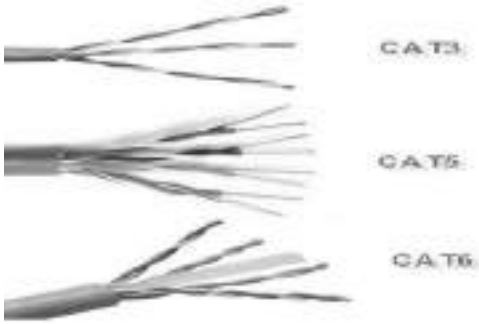


Fig 5: Coaxial cables : cat 3, cat 5 and cat 6

- **Optical fiber:** These are used for long-distance networking backbones with the high speed of transmission. The three basic components of optical fiberare : source of light, transmission media and the light detector. The transmission media is ultra-thin fiber of pure glass. A light source is attached to one end of the optical fiber send the light signals through transmission media. The source can be light emitting diode (LED) or semiconductor laser. A light pulse is considered as 1 bit and absence of light is considered as 0 bit. On the receiver end of the fiber, a detector, which is a photodiode, on receiving light pulse converts it to electrical signal.

  The transmission through optical fiber follows the physics principle of total internal reflection as shown in figure below.

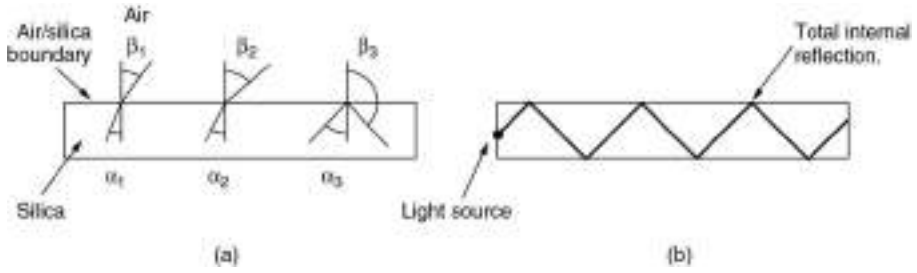


Fig 6: a) Refraction at boundary of two materials b) Total internal reflection

Fiber Cables are similar to coax, excluding the braid. The figure shows the single fiber and view of the sheath with three fibers. Cladding shown in the figure is made up of glass but with lower index of refraction than the core. It helps in keeping the light only in the core.

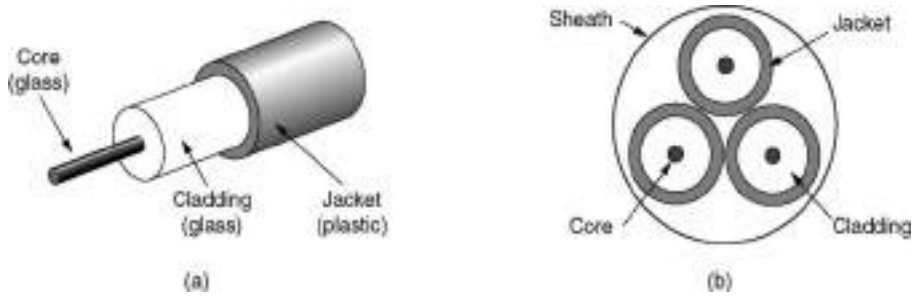


Fig 7: a) Optical fibre b) sheath with three optical fibers

The optical fiber can support high data rate of transmission but is limited to 100Gbps due to response time of the detector to convert light to electrical signal. The transmission through optical fiber suffer from thermal noise.

2.6.2Comparison of Optical fiber and Copper Wire

Both the transmission medias are being used for communication, but it is important to compare these two medias. The table below gives the comparison of the two based on important parameters:

Parameter	Optical Fiber	Copper Wire
Data transmission	Data is transmitted in the form of light pulses.	Data is transmitted in the form of electrical signals.
Bandwidth	Offers high bandwidth as light communicates faster than electrical pulses.	Low bandwidth in comparison to optical fiber.
Attenuation	The loss in signal strength i.e. attenuation is low.	Attenuation is comparatively high.
Interference	Resistant to electromagnetic interference or lightening.	Vulnerable to electromagnetic interference and lightening.
Durability	High tensile strength, so it is durable.	It is less durable as compared to copper wire.
Installation cost	High	Low
Security	Signals cannot be tapped, hence it is secure.	It can be easily tapped.

The optical fibre has many advantages over copper wire, hence it is replacing the twisted pair cables very fast.

2.6.3Network Interface Card (NIC)

NIC is a hardware component and is inbuilt in the computer. It connects the computer with other devices on the network. It is also known as network or ethernet adapter, LAN card, network controller etc.

Functions of NIC

- It converts data into digital signal like a translator.
- It offers both wired and wireless communications.
- NIC operates on layer 1 i.e. physical and layer 2 i.e. data link layer of OSI model.

Components of NIC

The main components of NIC are as follows:

- External Memory: It is used to store data temporarily and uses the data while processing the communication.
- Connectors: They are used to make the physical link between cables and ethernet port.
- Processor: It converts the data into signal format for communication.
- Jumpers: They are used to control the communication by turning on or turning off the switch.
- Routers: It is a NIC device to provide wireless connectivity.
- MAC address: It is unique address given to NIC on which the message is communicated. It is also known as physical address.

Types of NIC

NIC are of two types: Ethernet and Wireless Networks.

Ethernet NIC: It is a slot for the ethernet cable. The other end of the cable is connected to modem. It is most widely used in LAN, MAN and WAN. 5-Base T, 10-Base T, 100-Base T and gigabit Ethernet are commonly used NICs.

Wireless NIC: It consists of small antenna integrated onto NIC that helps in communication among devices which are connected wirelessly. Fiber data digital interface (FDDI) is an example of wireless NIC.

NIC has certain advantages:

- It provides reliable connection.
- It allows multiple devices to be connected using different ports of NIC.
- It facilitates bulk data sharing.

NIC has the following disadvantages:

- Wired NIC is not portable as compared to wireless NIC
- Proper configurated is required for getting better communication.
- It does not provide data security.

NIC is used for data exchange over the internet. It is applicable for devices like bridges and repeaters and also for hubs, switches and routers in case of wired communication device.

2.6.4 Transmission Media Devices

The common transmission media devices used in networks are as follows:

- **Repeater:** This device operates at physical layer. Its function is to regenerate the signal in the same network before it becomes too weak. The purpose of regeneration is to extend the length over which the signal can be transmitted in the same network. There is no amplification involved but the regenerating of signal with original strength. It has two ports.
- **Hub:** It is a repeater with multiple ports. It is a connector like in star topology, which connects the wires coming from various branches. Hub cannot filter data and sends packets to all connected devices. Hubs are not designed to find the best path for data packet delivery.
  Based on their features they are categorized into active, passive and intelligent hubs.
- **Bridge:** It is a repeater operating at data link layer. It can filter data by reading MAC addresses of sender and receiver. It can also be used to connect two LANs. It is a two port device.
  It is of two type: transparent and source routing.
- **Switch:** It is a multiport bridge working on data link layer. It has capability of performing error checking before sending the packets.
- **Router:** It is a kind of switch that routes the data packets based on the IP address. It works on the network layer. It connects LANs and WANs. Router maintains the routing table based on which decision is taken for routing data packets.
- **Gateway:** It acts like a passage to connect two networks which may be working on different models. It act as data interpreter, that accepts data from one network, interprets it and send it to another network. It works on the network layer as protocol converter. It is more complex in design then switch or router.
  Fig. 8 shows the media devices.

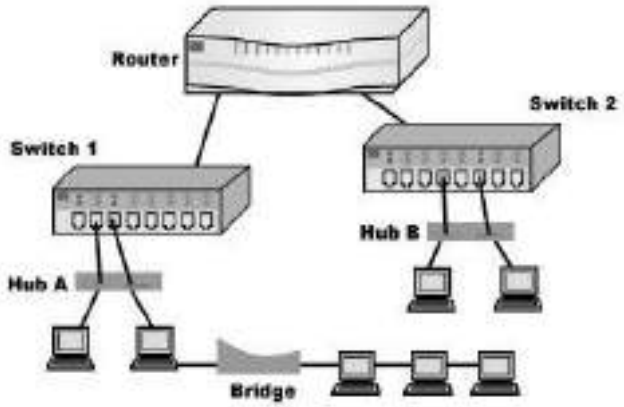


Fig 8: Transmission media devices

2.7 TRANSMISSION IMPAIRMENTS

The data communicated through transmission media deteriorates in the quality at the end of the media. The imperfection in transmission media causes transmission impairments.

There are three major causes of transmission impairments:

Attenuation: It is loss of signal strength. This is due to loss of signal energy in overcoming the resistance of transmission media. Amplifiers are used to amplify signals and compensate for the loss. Fig below shows the attenuation and amplified signal.

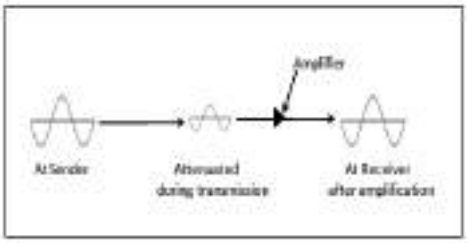


Fig 9: Attenuation in signal

Distortion: It is the change in the form or in the shape of signal. In a composite signal by combination of signals of multiple frequencies, each frequency component has different propagation speed. Due to this they reach at different timings at the receiver thus leading to distortion.

Fig below shows the change in phase of the received signal due to distortion.

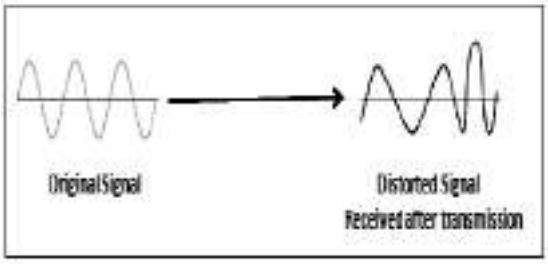


Fig 10: Distortion in signal

Noise: It is the unwanted signal that gets mixed with the original signal during transmission. It can be one of the following:

Induced noise: It comes from sources like appliances or motors.

Thermal noise: It is the movement of electrons in wires creating additional signal.

Crosstalk: It is due to one wire affecting another.

Impulse noise: It is high energy signal due to lightening or power lines.

These transmission impairments result in poor signal quality and resulting in low signal-to-noise ratio.

2.8 SUMMARY

The module covers the layered description of OSI model and TCP/IP model. OSI model is conceptual model whereas TCP/IP model in a practical model specifying how nodes can communicate in network and can be termed as compact version of OSI model. OSI model has

seven layers whereas TCP/IP model has four layers the functionality of some of the layers of OSI model has been combined into a single layer in TCP/IP model.

Physical layer covered in this module is the basis of all networks as it deals with the data transmission among two nodes. The functions major to the physical layer are multiplexing, switching, deciding transmission mode. The transmission media used for communication can be guided or unguided. It play an important role in defining channel bandwidth. The most commonly used guided media are twisted pair cables which are copper cables or optical fiber. Optical fiber are used for high speed communication over long distance as it has less attenuation, less prone to interference and durable as compared to copper wire.

For the physical connection of the node to network, network interface card is used. It can be active, passive or intelligent NIC card based on the functions performed.

Transmission media devices, repeater and hubs works on physical layer for regenerating signal. Hub is multiport repeater. Switches and bridges work on data link layer and routers work on network layer. Gateway acts as a translator between two networks deploying different protocols.

Transmission media used for communication has various imperfections resulting in signal attenuation, distortion and crosstalk.

2.9 REVIEW QUESTIONS

a. In OSI model, how the data encapsulation occurs from top to bottom layer?
b. In which layers the term frames and packets are used?
c. Name some well-known protocols of application layer?
d. Enumerate the differences between OSI model and TCP/IP model.
e. List the major functions of all the seven layers of OSI model.
f. Which layer in TCP/IP model performs the functions of session and presentation layers of OSI model?
g. Illustrate the principle used in optical fiber for communication of data.
h. Which features of optical fiber give it preference over copper wire for data transmission?
i. How circuit switching is different from packet switching?
j. What are the application areas of ethernet and wireless NIC?
k. It is important in the network to uniquely identify the nodes for communication. How addressing helps in identifying the nodes and what kinds of addressing TCP/IP offers?
l. Name the transmission media devices working on physical, data link and network layers respectively.

2.10 REFERENCES & FURTHER READING

- Data Communication & Networking by BehrouzForouzan
- Computer Networks by Andrew Tannenbaum
- Data and computer Communications by William Stallings
- Computer Networking: A Top-Down Approach by James Kurose and K.W. Ross

UNIT 3: DATA LINK LAYER

STRUCTURE

3.0 Objectives

3.1 Data Link Layer: An Introduction

3.2 Introduction

3.3 Services by Data Link Layer

3.4 Framing

3.5Error Control

3.5.1 Error Detection and Correction

3.5.2 Error Correcting Codes

3.5.2.1 Hamming Codes

3.5.2.2 Binary Convolution Codes

3.5.2.3 Reed-Solomon Codes

3.5.2.4 Low Density Parity Check Codes (LDPC)

3.5.3 Error Detection Codes

3.5.3.1 Parity Check

3.5.3.2 Checksum

3.5.3.3 Cyclic Redundancy Check (CRC)

3.6 Flow Control Protocols

3.6.1Simplex Protocols

3.6.1.1 Utopian Simplex Protocol

3.6.1.2 Simplex Stop-and-Wait Protocol for error free channel

3.6.1.3 Simplex Stop-and-Wait Protocol for a noisy channel

3.6.2 Sliding Window Protocols

3.6.2.1 One-bit Sliding Window Protocol

3.6.2.2 Sliding Window Protocol using go-Back-N

3.6.2.3 Sliding Window Protocol using Selective Repeat

3.7 Shared Media Protocols

3.7.1 Aloha

3.7.2 Carrier Sense Multiple Access (CSMA)

3.7.2.1 Non-Persistent CSMA

3.7.2.2 Persistent CSMA

3.7.2.3 CSMA with Collision Detection (CSMA/CD)

3.7.2.4 CSMA with Collision Avoidance (CSMA/CA)

3.8 Summary

3.9 Review Questions

3.10 References & Further Reading

3.0 OBJECTIVES

- Understanding the type of services provided by data link layer
- Understanding the framing techniques and the challenges in the schemes
- Getting introduced to error detection and error correction algorithms at data link layer
- Understanding the flow control mechanisms
- Understanding the media access protocols

3.1 INTRODUCTION

The data link layer is layer 2 in the OSI model. It uses the services of layer 1 i.e. physical layer for data transmission from sender to receiver. The functions of data link layer are:

- To provide service interface to network layer
- Dealing with the errors in the transmission
- Controlling the transmission flow to overcome the issues due to speed mismatch of sender and receiver

In this module we will focus on the concepts of frames, framing in data link layer, error control and flow control in data link layer. Also, protocols to share media for data transmission will be covered.

3.2 DATA LINK LAYER: AN INTRODUCTION

The data link layer is concerned with local delivery of frames between nodes on the same level of the network. Data-link frames, as these protocol data units are called, do not cross the boundaries of a local area network. Inter-network routing and global addressing are higher-layer functions, allowing data-link protocols to focus on local delivery, addressing, and media arbitration. In this way, the data link layer is analogous to a neighborhood traffic cop; it endeavors to arbitrate between parties contending for access to a medium, without concern for their ultimate destination. When devices attempt to use a medium simultaneously, frame collisions occur. Data-link protocols specify how devices detect and recover from such collisions, and may provide mechanisms to reduce or prevent them.

3.3 SERVICES BY DATA LINK LAYER

Data link layer offers the following services:

- **Unacknowledged connectionless service:** Frames are transmitted by data link layer to receiver without establishing any logical connection between sender and receiver. Frames received by receiver are not acknowledged. Ethernet is an example of this kind of service.
- **Acknowledged connectionless service:** In this service no logical connection is established but frames are acknowledged by the receiver. This service is mainly used in wireless systems where channels are unreliable.
- **Acknowledged connection-oriented service:** Sender and receiver establishes connection before transmitting the data. The frames sent are acknowledged by the receiver.

3.4 FRAMING

The data link layer uses the services of physical layer in transmitting data to the receiver in the form of bit stream of 1s and 0s. Due to noise in the wireless channel the error rate is high. Physical layer in order to bring the error rate within the permissible limits add redundant bits. It does not guarantee that the data received by data link layer will be error free. Data link layer detects and corrects the error.

To achieve this, layer breaks the bit stream into frames of fixed size, the checksum is performed for the frame. The checksum is also transmitted along with the frame data. At receiver side the checksum is computed. If checksum is same the frame is error free otherwise error is reported and data link layer takes measures to deal with the error.

It is important to understand how the breaking-up of bit stream is done to form the frames. It is important the start and the end of the frame is clearly known to the receiver. There are four methods for framing:

- **Byte Count:** It is the simplest way of framing. In the header information, count of bytes are specified which will make a frame. Receiver on receiving the frame checks the header information and count the number of bytes mentioned and consider that as one frame.

  This is the simplest way of forming frames but it has a major drawback. In case the header information about count of bytes in a frame is transmitted erroneously, the receiver will not be able to identify the frame boundaries and will not be able to send message to the sender to retransmit as receiver cannot know that how many frames are to be asked for the retransmission.

  The figure below specifies the frame sizes of 5, 5, 8 and 8 bytes in the data stream and shows the error in identifying the start and end of frame if byte count is garbled.

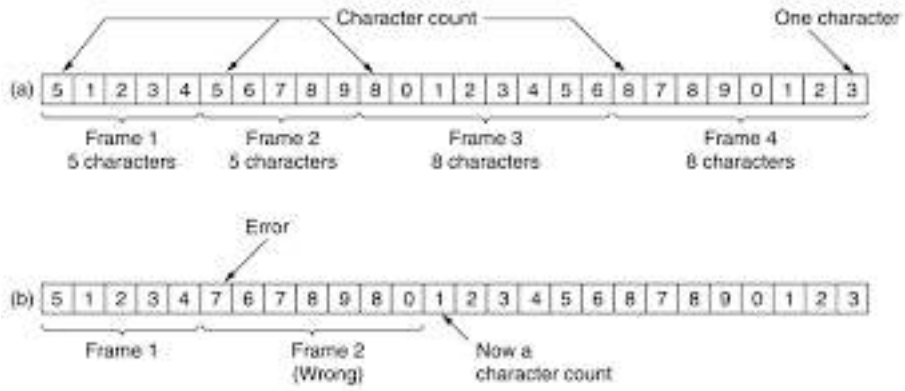


Fig1: Byte count framing a) byte stream without error b) byte stream with error

The figure clearly depicts that in case of error in the byte count value the start and end of frame as perceived by the receiver will be incorrect. This will result in erroneous data.

- **Flag byte with bit stuffing:** In this technique the start and end of the frame is identified by the adding special byte at the start and at the end of frames. This

special byte is known as flag byte. The flag byte is always same. Two consecutive flag bytes will tell that one frame has ended and another has started.

This will help receiver finding the start of new frame. The techniques faces a problem:

a. If the flag byte occurs in the data itself then it would be wrongly interpreted. The solution to this problem is that data link layer of sender on identifying the same flag byte in the data should put special escape byte before the flag byte. Data link layer at the receiver on finding escape byte should remove it from the data. Adding escape byte is known as byte stuffing.
b. Another problem arises when escape byte is also there in the data stream itself, in that case an additional escape byte is put by the sender. Receiver on getting the two escape bytes removes one from the data stream.

Fig 2 shows some of the examples of flag byte with byte stuffing.

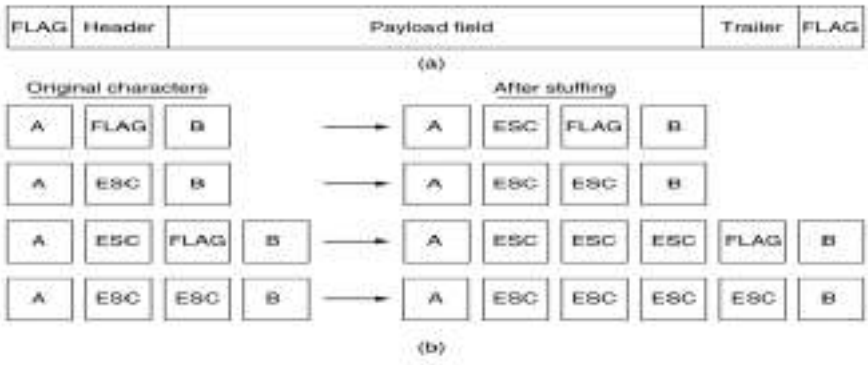


Fig 2: a) Frame format b) Byte stuffing with esc and flag bytes in data

- **Flag bits with bit stuffing:** In flag byte, it becomes restriction to have one byte i.e. 8 bits for flag and for stuffing. This technique overcomes this drawback. This technique allows the frame to have multiple bits which may not be multiple of 8 to convert it into bytes. In this case each frame starts and ends with flag bits 01111110. If in the data five times digit 1 is identified by sender it automatically stuffs 0 in the bit stream. Receiver on identifying five 1s followed by 0, simply de-stuff the 0 and keep the remaining data. Fig 3 shows this technique. Fig. 3a shows the actual data at sender, 3b shows the bits stuffed by sender and fig. 3c shows the data at the receiver after removing stuffed bits.

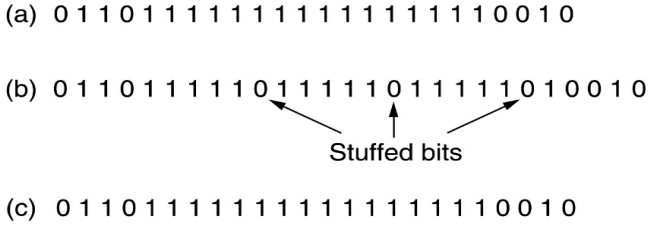


Fig 3: Flag bits with bit stuffing a) original message b) bit stuffing c) decoded message

Both the flag byte and flag bit techniques increase the size of the data stream due to byte and bit stuffing.

- **Physical layer coding violations:** This technique is applicable where during the physical transmission encoding is done in the network by adding redundant bits. In some cases 1 bit data is encoded into two bits, for eg. 1 is represented as high-low pair i.e. 10 and 0 is represented as low-high pair i.e. 01. This ensures that during transmission any bit there is transition from low to high or vice versa. Receiver can identify the bit boundaries in this case. The high-high and low-low combination is used for delimiting frames in some cases. In this technique no additional bit or byte stuffing is required.
  Fig 4 shows this technique.

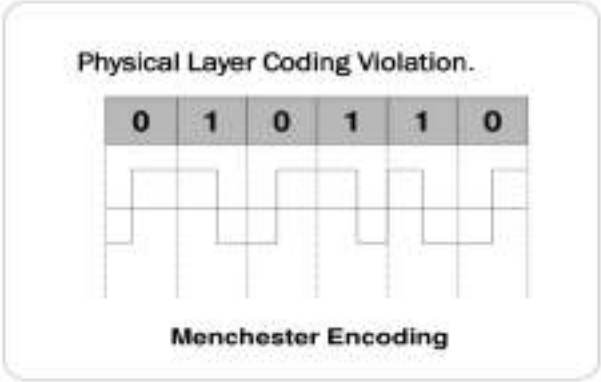


Fig 4: Technique for representing a bit as pair of two

All these techniques have their advantages and disadvantages. Many data link layer protocols uses combination of these methods.

3.5 ERROR CONTROL

Once the framing is done the next challenge for the design link layer is to ensure that the frames are delivered to the network layer of the receiver in proper order and error free. Error corrections and error detections codes at data link layer ensures this.

Error control is not required for unacknowledged connection-less service but for acknowledged connection-oriented service it is required.

The positive or negative acknowledgement received by sender gives an indication for retransmission of frames in case of error. The problem occurs when the frame is lost and the acknowledgement is also lost. The sender will not receive the acknowledgement and hence keep on waiting infinitely. This can be handled by starting a timer once the frame is transmitted, if no acknowledgement is received before timer expires, sender re-transmits it.

3.5.1Error detection and correction

Error correction schemes are used in data link layer to rectify the errors in the transmission due to noise or interference. Two basis strategies have been devised for error detection and error correction:

- To include enough redundant bits in the transmission so as to enable the data link layer at receiver to identify error and deduce the actual transmitted data from it. This strategy uses error-correction codes for detecting and correcting errors.
- To include redundant bits only to indicate receiver that error has occurred so as to send request for re-transmission. This strategy uses error detection codes.

The use of error correcting codes is referred to as forward error correction (FEC). The error correction codes are used in case the transmission media is unreliable but for fiber optics where the chances of errors in transmission are very low, error detection and retransmission of frames is comparatively cheap.

To understand the error handling technique it is important to understand the how error is identified. A frame consists of **m** bits of data and **r** redundant bits. Total message bits will be **m+r,** which can be denoted by **n**. This n-bit data unit with message bits and redundant bits for checking is known as codeword.

By applying logical operation exclusive OR (XOR) it can be identified that how many bits differ in the actual codeword and received codeword. Eg.

Original codeword	10100101
Received codeword	10111001
XOR Operation	00011100

XOR gives the count of bits that have been erroneously received. In this example 3 bits are incorrect. Number of bit positions in which the two given codewords vary is termed as hamming distance. Hamming distance in this example is 3. The hamming distance tells that how many single bit errors are required to exactly match the one codeword to another. The error-detecting and error correcting properties of a code depend on its Hamming distance. To detect the errors in **d** data bits a code with distance **d+1** is required as with such code there is no chance that d single bit errors can change valid codeword to any other valid codeword. To correct errors in d data bits, a code with distance **2d+1** is required.

3.5.2Error Correcting Codes

There are four error correcting codes that are commonly deployed for data link layer.

3.5.2.1 Hamming Codes: It is a block code capable of correcting single bit errors and errors upto two simultaneous bits. In this method, encoding by sender is done by adding redundant bits in the message. These are the extra bits added at specific positions in the message. At receiver, the logic is applied to detect errors and find the erroneous bit positions. The process of encoding the message using Hamming code involves the following basic steps:

- **Calculating the count of redundant bits:** For **m** bit message, **r** redundant bits are added. So total bits in encoded message will be **m+r**.Considering error can be in any of the **m+r** bits, there can be **m+r** different stated of code words and one state with no error. In total different stats of code word will be **m+r+1**. In a binary number system **r** bits can have 2^r
  states so the following equation holds $\mathbf{2^r \geq m+r+1}$.
- **Positioning of redundant bits:** The **r** redundant bits are placed at positions r_1, r_2, r_4 .....r_n, where these positions are defined as power of 2, i.e. 1, 2, 4, 8… so, $r_1 = 1$, $r_2 = 2$, and so on.
- **Calculating the values of redundant bits:** These redundant bits are parity bits, generally even parity is taken. Even parity is when the count of 1s in binary number is even. The redundant bits are calculated by checking the even parity.
  r_1 is the parity bit for all the data bits whose binary representation includes 1 in the least significant position i.e. 1,3,5,7,9,11…etc.
  r_2 is the parity bitfor all the data bits who binary representation includes 1 in the second position from the least significant bit i.e 2, 3, 6, 7, 10, 11…etc
  r_4 is the parity bitfor all the data bits who binary representation includes 1 in the third position from the least significant bit i.e. 4-7, 12-15, 20-23…etc

  r_8 is the parity bitfor all the data bits who binary representation includes 1 in the fourth position from the least significant bit i.e. 8-15, 24-31, 40-47…etc

  The encoded message by adding redundant bits is transmitted by sender. At receiver the message is decoded. The following are the decoding steps:
  - Count and position of redundant bits is ascertained using the same logic as in encoding.
  - **Parity Checking:** Parity bits are checked based on the data and redundant bits using the same rule as for generating parity bits.
    $c_1$is parity for bits at position 1,3,5,7,9,11….
    $c_2$is parity for bits at position 2,3,6,7,10,11….
    $c_3$is parity for bits at position 4-7,12-15, 20-23….
    $c_4$is parity for bits at position 8-15, 24-31,40-47….

  Decimal equivalent of c_1 c_2 c_3 c_4 is calculated. This value represents the location of erroneous bit in the message. To correct the same, the bit is flipped.

  Following example explains this method:

  Eg.

  Data bits to be transmitted are 1011010. Using the equation $\mathbf{2^r \geq m+r+1}$, **r**is calculated as 4. So four redundant or parity bits are added at specific locations

11	10	9	8	7	6	5	4	3	2	1
1	0	1	**r_8**	1	0	1	**r_4**	0	**r_2**	**r_1**

r_1 =parity of1,3,5,7,9,11 bits, count of 1s at these locations is odd excluding r_1, so make it an even parity value of r_1 has to be 0.
r_2 =parity of2,3,6,7,10,11 bits, count of 1s at these locations is even excluding r_2, so to make it an even parity value of r_1 has to be 0.
r_4 =parity of4-7bits, count of 1s at these locations is even excluding r_4, so to make it an even parity value of r_4 has to be 0.
r_8 =parity of8-11bits, count of 1s at these locations is even excluding r_8, so to make it an even parity value of r_8 has to be 0.
The encoded data will become

11	10	9	8	7	6	5	4	3	2	1
1	0	1	**0**	1	0	1	**0**	0	**0**	**0**

Let the transmitted data be, where 9th bit is erroneous bit

11	10	9	8	7	6	5	4	3	2	1
1	0	**0**	**0**	1	0	1	**0**	0	**0**	**0**

Receiver will decode the data and calculate

$c_1$is parity for bits at position 1,3,5,7,9,11 , which is odd so c_1=1
$c_2$is parity for bits at position 2,3,6,7,10,11, which is even so c_2=0
$c_3$is parity for bits at position 4-7, which is even so c_3=0
$c_4$is parity for bits at position 8-11, which is odd so c_4=1

$c_1\ c_2\ c_3\ c_4$ is 1001, decimal equivalent of it is 9. Hence error at 9th bit is detected, value of 9th bit is changed from 0 to 1.

Hamming code can correct one bit error but can detect two bits error.

3.5.2.2 Binary Convolution Codes : It is different from the block code. Encoder processes a input bits sequence and generates output bits sequence. These codes are commonly used for mobile phone system, satellite communications etc.
The encoder is collection of shift registers through which the input bits are sequentially passed.
Convolution code is represented by (n, k, K), i.e. for every k bits, there are n output bits and K being the constraint length. K is the count of shift registers in the encoder.
Shift registers in this encoder has memory, so the output of n bits not only depend on the k input bits but also on previous K-1 blocks each of k input bits, giving output bits **n** as **K x k**. Generally n and k are small numbers. It is easy to implement and is effective in communication where the error rate probability is high and channels are noisy.

3.5.2.3Reed-Solomon Codes: It is also a kind of linear block codes like hamming codes. Instead of operating on individual bits, it works on m bit symbols. These codes work on the fact that n+1 points are required to define any polynomial of degree n. For eg.polynomial ax^2+bx+c, of degree two can be defined by three points a, b and c. Extra points are redundant and can be utilized for error correction. To correct the error in addition to the data points additional points satisfying the polynomial equation are transmitted. If there is error in any point, still the polynomial equation can be recovered by using other points. For m bit symbol, the codeword will be $2^m - 1$ symbols long. If m=8 bytes, then the codeword will be of 255 bytes. The most widely used code is (255, 233) code, i.e. 32 redundant symbols are added to 233 symbols of data. It can correct errors upto 16 symbols or 128 bits. Decoding with error correction is done with an algorithm. In general on adding 2n redundant bytes it can correct n bytes of errors.

It is used for correcting burst errors, which is error in the contiguous sequence of symbols. These codes are used for error correction in satellite communications, data over cables, CDs, blu-ray discs etc.

3.5.2.4Low Density Parity Check Codes (LDPC): These are linear block codes. Every single output bit is formed from a fraction of input bits. These codes are specified by parity matrix, containing mostly 0s and low density of 1s, hence the name LDPC. It is useful for error corrections in large blocks of data. It is mainly used in power line networks, digital video broadcasting etc.

3.5.3 Error Detection Codes

For wireless transmission media where the channel reliability is low and transmission is prone to errors, the error correction is preferred. In case of reliable channel, eg.using optical fiber or copper cable as transmission media, error detection and retransmission is efficient mechanism. There are three linear systematic error detecting block codes:

3.5.3.1 Parity Check: This check is done by adding extra bit, termed as parity bit to make count of 1s even in data in case of even parity. Sender count 1s in the data and set parity bit as 1 if no of 1s counted are odd. Receiver on receiving the data counts 1s in the received data, if count is even it is accepted else rejected. It can only detect single bit error.

Eg. data to be transmitted is 1001010, the parity of the data is odd so to make it even parity bit is added with value 1, the data 1001010 **1**, including parity bit is transmitted. At receiver end parity is again checked and in case there is change in parity from even to odd, the error is detected and data is discarded.

Parity check can be done in matrix of data, where parity is calculated for all columns and rows and parity bits are also transmitted along with data.

Eg.

The following data is to be transmitted

1000101	1110001	1010110	0111001

Parity is checked

	Row Parity
1000101	**1**
1110001	**0**
1010110	**0**
0111001	**0**
1011011	**1**

Column Parity

The following data is transmitted by sender, where the underlined digits are the parity bits. Any change in parity at the receiver side is detected as an error.

1000101**1**	1110001**0**	1010110**0**	0111001**0**	**10110111**

3.5.3.2 Checksum

This scheme uses group of parity bits to detect errors. In this case data is divided into fixed size blocks. At sender side, segments are added using 1's complement arithmetic to get the sum. Complement of sum gives the checksum.

The checksum segment is transmitted along with the data. At the receiver all the segments are added using 1's complement and further the sum is complemented. All zeros tells no error hence the data is accepted, otherwise the data is rejected and request for retransmission is sent.

Eg.the following data is to be transmitted

10110011	10101011	01011010	11010101

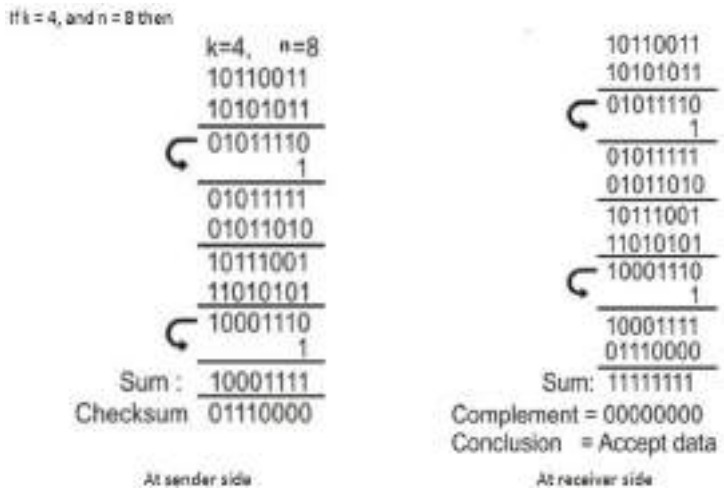


Fig 5: Calculation of checksum at sender and at receiver side

At sender the sum is performed for the k blocks of data with n bits, the one's complement is calculated, which is termed as checksum. Checksum is also transmitted along with the data. At receiver side sum of data received including checksum and further one's complement of the sum is calculated. In case of no error the complement will be all zeros.

3.5.3.3 Cyclic Redundancy Check (CRC)

CRC is also known as polynomial code. Unlike checksum it is based on binary division. In this case the data to be transmitted is represented in the form of polynomial. Eg. 1101 has four bits so it can be represented as four-term polynomial as $1x^3+1x^2+ 0x^1+1x^0$. The polynomial arithmetic is done by modulo 2, hence division is carried out. The CRC bits are added to message so that the resulting data units are divisible by any predetermined binary number.

Eg. The frame bits are 1101011011, and the generator polynomial is $1x^4+0x^3+0x^2+ 1x^1+1x^0$ with bits represented as 10011. Number of zero bits equal to the degree of polynomial is appended at the end of frame bits so the frame bits become 1101011011**0000**. For modulo 2 division (i.e. XOR) is performed.

After module 2 as shown in fig. 6 the remainder is added to the frame bits to the data completely divisible by generator polynomial bits. So transmitted bits become 1101011011**1110**, which is exactly divisible by generator polynomial bits.

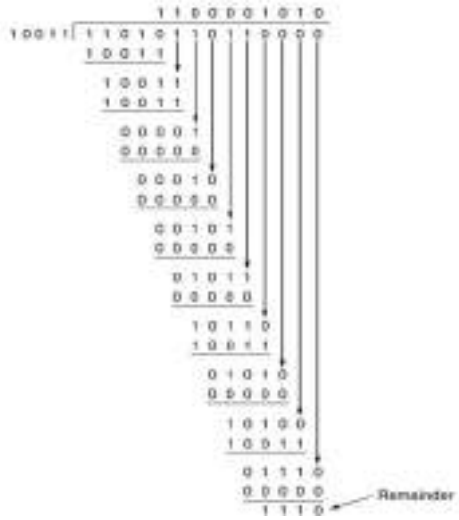


Fig 6: Modulo 2 division for Cyclic Redundancy Check

At the receiver the modulo 2 division is performed with the same generator polynomial, non-zero remainder implies that error has occurred.

CRC codes applied in practice are standardized, eg. CRC-32, a polynomial with degree 32 is a standard CRC. It can detect burst error of length 32 or less.

3.6 FLOW CONTROL PROTOCOLS

Flow control is technique to synchronize the speeds of sender and receiver. In data link layer, flow control restricts the number of frames the sender can transmit before it gets the acknowledgement from the receiver. Flow control protocols in data link layer uses two techniques:

- **Stop-and-wait:** In this technique, sender transmits the frame and wait for acknowledgement. Once it gets the acknowledgement from receiver, sender sends the next frame.
- **Sliding Window:** In this technique, both the sender and receiver have buffer of finite size called window. Buffer size is fixed on mutual consent of sender and receiver. Sender transmits multiple frames in a sequence without waiting for acknowledgement till the time window is filled. Then it waits for acknowledgement. On getting acknowledgements it slides the window based on the number of acknowledgements received and start transmitting the next frames in the sequence.

3.6.1 Simplex Protocols

3.6.1.1 UtopianSimplex Protocol

This protocol assumes that there is infinite buffer size on both sender and receiver side. This is an ideal protocol and not realistic that's why the name Utopian has been given. It assumes a single direction transmission with no transmission error.

Sender transmits the data at a fast speed and the receiver receives it. This protocol assumes that there is no requirement of error control and flow control.

3.6.1.2 Simplex Stop-and-Wait Protocol for error free channel

Simplex protocol suffers from drawback of flooding receiver with frames at a rate faster than it can process. It takes into account the limitation on processing speed of receiver. Due to this limitation the frames will be dropped at receiver end. In this protocol as error free channel is considered so no retransmissions are considered. The sender transmits the frame, waits for acknowledgement, once received it transmits next frame.

3.6.1.3Simplex Stop-and-Wait Protocol for a Noisy Channel

This protocol considers the realistic scenario where error control and flow control are required. It assumes that the processing speed of receiver is limited and frames may get corrupted while transmission. If the sender speed is high and receiver is not able to process, the frames get dropped. Also, the frames with error are discarded by data link layer of the receiver. Receiver sends acknowledgement only for the valid frames. After getting acknowledgement sender transmits other frames and also resends the frames for which no acknowledgment has been received.

Sender add the sequence number to frame and transmits it. After receiving positive acknowledgement it transmits next frame. In case of no acknowledgement it waits for fixed amount of time and resends the frame.

Receiver maintains the sequence numbers of frames. It matches the sequence number of the frame received with the expected sequence number and sends acknowledgement which also has a sequence number.

This protocol is also known as simplex stop-and-wait Automatic Repeat Request protocol for noisy channel. Fig. 7 represents the various scenario of this protocol.

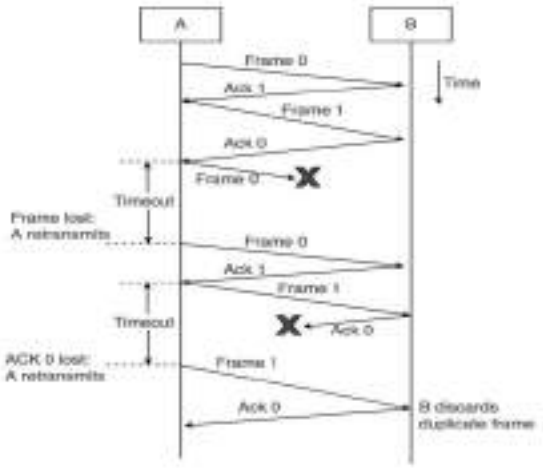


Fig 7: Simplex stop-and-wait Protocol for nosy channel

3.6.2Sliding window Protocols

Simplex protocols have one way transmission mode. The bidirectional transmission which is fullduplex transmission adds complexity in the protocols. Piggy-backing technique helps is achieving full duplex transmission whereas acknowledgement is sent along with the frame to be transmitted. Sliding window protocols are bi-directional protocols. Following are the types of sliding window protocols for data link layer:

3.6.2.1One-bit sliding window Protocol

Referring to the sliding window technique, in this protocol the window size is one bit. In this protocol sender transmits frame, wait for acknowledgement, once received it moves its window to next and transmits next frame. It is similar to stop-and wait protocol.

For full duplex transmission, the acknowledgement is piggy-backed with the frame. The data frame to be transmitted has additional field for acknowledgement (ack). The ack field contains sequence number of last frame received without error. If this sequence number matches with the sequence number of next frame to be transmitted, then no error is reported otherwise the it is assumed that there was error in earlier transmission.

Fig 8 a) represents ideal case of the protocol where notation (seq no., ack, frame number*) has been used. The asterisk specifies that the frame has been received. Fig. 8 b) represents the simultaneous transmissions resulting in half of the frames transmitted are duplicates. It can also result due to early timeout, causing same frame to be transmitted multiple times.

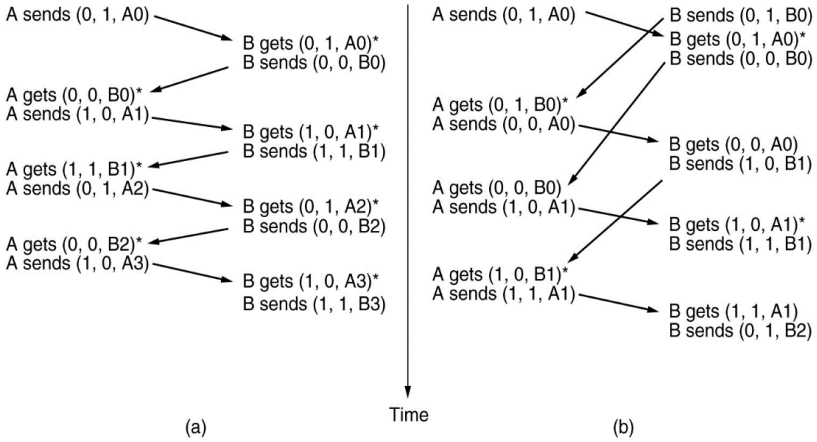


Fig 8: a) Ideal case b) Simultaneous transmission

3.6.2.2Sliding Window Protocol using go-Back-N

This protocol assumes sender window of size **N** and receiver window of size **1**. It is also known as go-back-n ARQ protocol. The frames in this case are numbered sequentially. Sender transmits frames depending on the window size without waiting for acknowledgement. It sender does not get the acknowledgement of frame within specified time, so all the frames starting from that frame are re-transmitted. If the sending window size is represented as 2^n-1, then n bit sequence number can be applied for numbering frames where sequence number will be assigned from 0 to 2^n-1.

Fig. 9 represents the protocol. It shows that sender is transmitting frames and receiver window size is 1. If the acknowledgement is not received for any frame due to error then all subsequent frames are discarded by receiver and is resend by the sender.

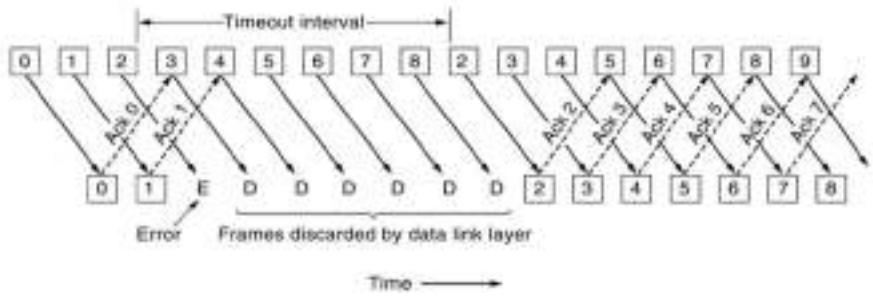


Fig. 9: Sliding window protocol – go-back-N with receiver window size of 1

3.6.2.3 Sliding Window Protocol Using Selective Repeat

Go-back-n protocol has drawback of retransmitting all the frames after the erroneous frame. In case errors are rare and channel is reliable, this protocol has acceptable performance but in case the errors are more it will result in retransmitting multiple frames, which will take lot of bandwidth.

In selective repeat protocol sender and receiver has specific window size and receiver can accept frames out of order. The frames are buffered at receiver and is shared by data link layer to upper layer.

In case of damaged frame a non-acknowledgement of the frame only the selected frame is to be retransmitted.

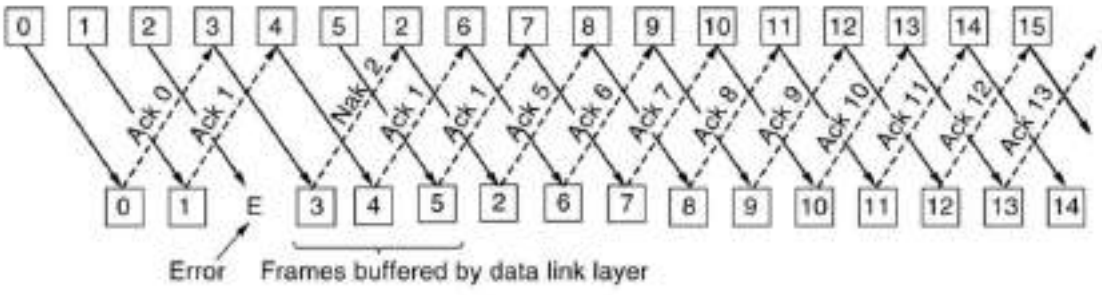


Fig 10: Sliding Window elective Repeat with large window size

Fig. 10 shows the working of selective repeat protocol. The window size both at receiver and sender side is large. The non-acknowledgement of frame 2 tells sender to retransmit only that frame.

Sliding window protocols have advantage over stop-and -wait as these are more efficient. Moreover, selective repeat protocol avoid unnecessary retransmissions of frames.

3.7 SHARED MEDIA PROTOCOLS

The sublayer of data link layer, termed as Medium Access Control (MAC) layer. One of the major functions of MAC layer is to perform multiple access resolutions and to determine the channel access method for data transmission. In case of multipoint communication, multiple devices can access the transmission media or channel simultaneously. To avoid the collision and crosstalk, multiple access protocols are required to define rules for the sharing of communication media.

A category of shard media protocols is random access protocols. There is an assumption that all the devices have equal priority to send the data over communication media. The main protocols under this category are as follows.

3.7.1Aloha

There are two categories of aloha protocol, pure aloha and slotted aloha.

Pure aloha protocol works on the following rules:

- Any node/device can transmit data through channel at any time without considering the status data transmission by other devices.
- Data frames lost in collision are to be retransmitted based on the acknowledgement mechanism. If no acknowledgement is received within the specified time, the node waits for random amount of time termed as backoff time and retransmits the frame.

The efficiency of this protocol is very low.

Slotted aloha protocol was designed to enhance the efficiency of pure aloha. In this protocol, shared channel is divided into slots of fixed time duration. Node can only transmit at the beginning of time slot and only one frame can be transmitted per slot. This helps in avoiding collision to some extent.

3.7.2 Carrier Sense Multiple Access (CSMA)

It is possible in LAN to sense the carrier i.e. transmission media and act accordingly. Protocols using the mechanism of carrier sensing are CSMA category protocols. If node wants to transmit data, it senses the media and if channel is find idle, it transmits. There are various access modes in CSMA.

3.7.2.1 Non-persistent CSMA

The node senses the media, if not idle then it waits for a random time and rechecks the media. This process continues till the node finds the media idle and then it transmits.

3.7.2.2 Persistent CSMA

1-persistent: The node senses the media, if not idle then it keeps on checking continuously till it finds the media idle and then transmits will full probability of 1.

P-persistent: This is different from 1-persistent, in a way, that when the media is find idle, node transmits the frame with p probability and not with 1 probability. With probability 1-p it defers the transmission for the next time slot.

3.7.2.3 CSMA with Collision Detection (CSMA/CD)

Persistent and non-persistent CSMA protocols have better performance as compared to aloha but if the two nodes begin transmission simultaneously then there will be collision.

CSMA/CD protocol implements a mechanism of collision detection. If any collision is detected, the node sends stop signal to the shared media channel to terminate transmission. After that, the node waits for a random period of time before next transmission. CSMA/CD consists of transmission, contention and idle period. After transmission node will contend for the shared media, once it finds the media idle it will transmit. If there in no data to transmit it will be in idle state. Fig. 11.shows the three status of nodes in CSMA/CD.

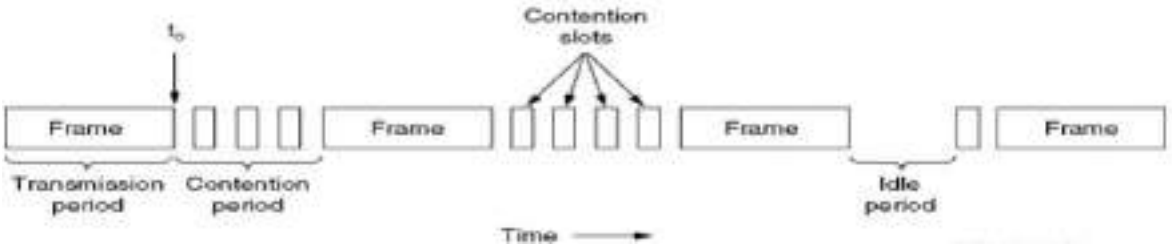


Fig 11: Status of nodes in CSMA/CD

3.7.2.4 CSMA with Collision Avoidance (CSMA/CA)

The protocol focuses on collision avoidance by using the signalling mechanism. In this protocol, before transmitting the actual frame, sender node transmits request to send (RTS) frame indicating the duration of transmission using network allocation vector (NAV), which is broadcasted to all the channels.

Receiver node replies with clear to send (CTS) frame. It indicates the hidden nodes as well. Once CTS is received, sender node will start transmitting.

NAV is used for virtual carrier sensing and used in IEEE 802.11 standard of wireless LAN. The duration of transmission is specified so the other nodes know for how much time they need to defer accessing the shared media. Zero value of NAV indicates the media is idle.

Fig. 12 shows the working of CSMA/CA protocol

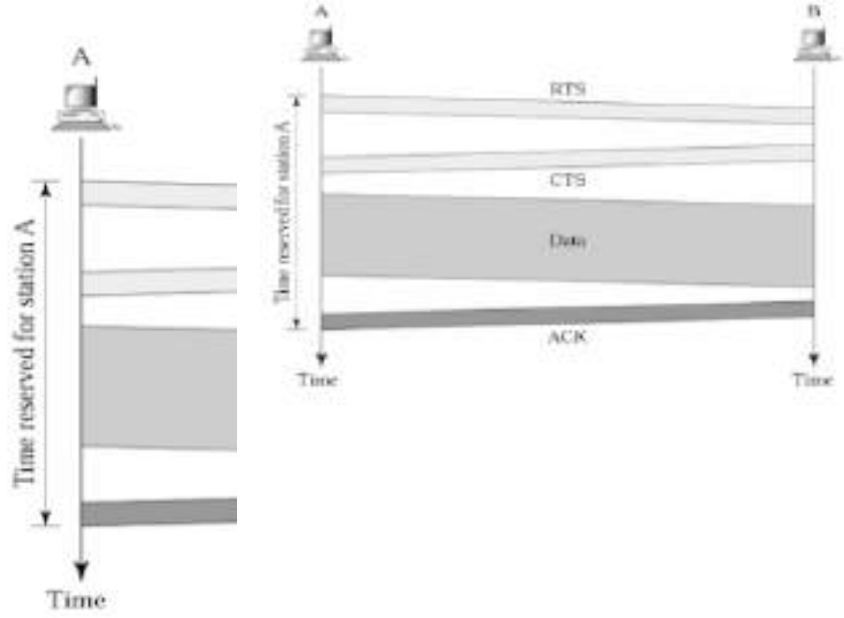


Fig 12: Working of CSMA/CA

3.8 SUMMARY

This module focuses on the functionality of data link layer. The data unit is frame and various techniques can be applied to help identifying the frame boundary at receiver. The error detection and error correction mechanism in data link layer help in attaining reliable communication. Wireless media is more prone to errors so error correction is not an efficient option, instead, error is detected and then retransmission request is send.

Flow control is important to avoid receiver being choked with data which then is discarded by receiver. Stop-and-wait protocols following half duplex communication mode has low efficiency as compared to sliding window protocols.

MAC sublayer, which is the lower among the twosublayers of data link layer deals with the access of the share media for communication. MAC protocols help node attaining control over the shared media to transmit frame with less probability of collision. CSMA/ CA is preferred over CSMA/CD though the overheads are more.

3.9 REVIEW QUESTIONS

1.12 What are the major functions of data link layer?

1.13 Specify the reasons for using error correction codes instead of error detection and retransmission.

1.14 Mention the two broad categories of flow control in data link layer and discuss about their efficiencies.

1.15 Which techniques of framing can result in the most number of stuffing bits in the frame?

1.16 Define Hamming distance. How many bits of errors can hamming code correct?

1.17 With an example, explain how checksum and CRC works.

1.18 If the generator polynomial is $1x^4+0x^3+0x^2+ 1x^1+1x^0$, identify if the message transmitted 11010111111110 has error or not.

1.19 Compare the performance of sliding window go-back-n and selective repeat protocols.

1.20 In parity check technique, find the row and the column parity of data, when both sender and receiver agree for odd parity.

1000111	1110101	1000110	0101001

1.21 Identify the suitable media access protocol for wireless transmission media and specify the reason for the choice.

3.10 REFERENCES & FURTHER READING

- Data Communication & Networking by BehrouzForouzan
- Computer Networks by Andrew Tannenbaum
- Data and computer Communications by William Stallings
- Computer Networking: A Top-Down Approach by James Kurose and K.W. Ross

In CRC assume the generator polynomial $G(X)=x^5+x^4+x^2+1$ is to be used. Determine whether the received message 111000110101110 has detectable errors or not. Show all workings.

In CRC assume the generator polynomial $G(X)=x^5+x^4+x^2+1$ is to be used. Determine whether the received message 111000110101110 has detectable errors or not. Show all workings.

UNIT 4: NETWORK LAYER

STRUCTURE

4.0 Objectives

4.1 Introduction

4.2 Network Layer: An Introduction

4.3 Services Provided by Network Layer

4.3.1 Connection Oriented Service – Virtual Circuits

4.3.2 Connectionless Service – Datagram Service

4.3.3 Comparison of Virtual Circuit and Datagram Approach

4.4 IP Address

4.4.1 Understanding IP Addressing

4.4.2 Classes of IP Addresses

4.4.3 Network Masking

4.5 Subnetting

4.6 Routing Algorithms

4.6.1 Type of Routing Algorithms

4.6.2 Non-adaptive Algorithms

4.6.2.1 Flooding

4.6.2.1.1 Controlled Flooding

4.6.2.1.2 Selective Flooding

4.6.2.2 Random Walk

4.6.3 Adaptive Algorithms

4.6.3.1 Centralized Algorithm

4.6.3.2 Isolated Algorithm

4.6.3.3 Distributed Algorithm

4.6.3.4 Distance Vector and Link State Packet Routing Algorithms

4.6.3.5 Comparison of Distance Vector Routing and Link State Routing

4.7 Summary

4.8 Review Questions

4.0 OBJECTIVES

a. Understanding the virtual circuit and datagram services provided by network layer
b. Understanding IP addressing technique
c. Identifying the subnet masks is IP addressing
d. Understanding the protocol functioning in network layer
e. Analysing the difference between various protocols

4.1 INTRODUCTION

Network layer is responsible for end-to-end delivery of packets from sender to receiver. The transmission may be single hop or multi-hop where intermediary nodes termed as routers are involved for the transmission. The major function of the network layer is to decide the transmission route. For this network layer should be aware of the network topology for choosing the right path. Network layer deals with the problems in defining routes for communication when both sender and receiver are on same network or on different networks. This module focuses on two kind of services provided by network layer connection-oriented i.e. virtual circuit and connectionless i.e. datagram. Also, IP addressing and routing protocols will be discussed in the module.

4.2 NETWORK LAYER : AN INTRODUCTION

The network layer is a portion of online communications that allows for the connection and transfer of data packets between different devices or networks.

The network layer is the third level (Layer 3) of the Open Systems Interconnection Model (OSI Model) and the layer that provides data routing paths for network communication. Data is transferred to the receiving device in the form of packets via logical network paths in an ordered format controlled by the network layer.

4.3 SERVICES PROVIDED BY NETWORK LAYER

Network layer provides the services to transport layer with the goal that transport layer should be independent of router technology and network topology. Also, the network addresses provided by network layer should be unique and uniform. Data link layer deal with frames and network layer deals with packets as a data unit for transmission.

Network layer is the lowest layer in OSI model to provide end-to-end transmission. It provides the following two types of services:

4.3.1 Connection-Oriented Service – Virtual Circuit

In this type of communication service a path from the source router to the destination router is established before sending the data packets. The connection establishes is known as Virtual Circuit (VC). Eg.the telephone system is a VC network.

The purpose of virtual circuit is to avoid the process of choosing new path every time the packet is to be transmitted. Every intermediary node i.e. router store the path established between source router and destination router in its memory in the form of routing tables. Same route is followed for transmitting all the packets from the defined source to defined destination. Once the transmission is complete VC is terminated.

In this service, each packet carrier a identifier specifying the virtual circuit to which it belongs.

Fig 1, details the concept of virtual circuit and routing tables.
Sender host H1 wants to transmit data to receive host H2. Connection from H1 to H2 is established and given an identifier 1.
Data packet is transmitted from H1 to connected router i.e. A. A maintains a routing table which checks that if packet from H1 comes with a identifier 1, then transmit to router C.
C in its table checks that any packet coming from A with identifier 1 is to be routed to E.
E in its table checks that any packet coming from C with identifier 1 has to be transmitted to F.
Router F is connected to receiver host H2, so transmits data. Identifier 1 tells H2 that data has been transmitted from sender host H1.

In case another sender host H3 wants to transmit data to H2, it sends a packet to router A keeping the identifier as 1. Router A on checking the packet adds an entry in routing table to transmit this packet to router C but with identifier 2 as identifier 1 has already been assigned to packet received from other host.
C checks in the table that any packet received from A with identifier 2 is to be routed to router E with identifier 2 and further E in its table checks and routes the packet to F with identifier 2.

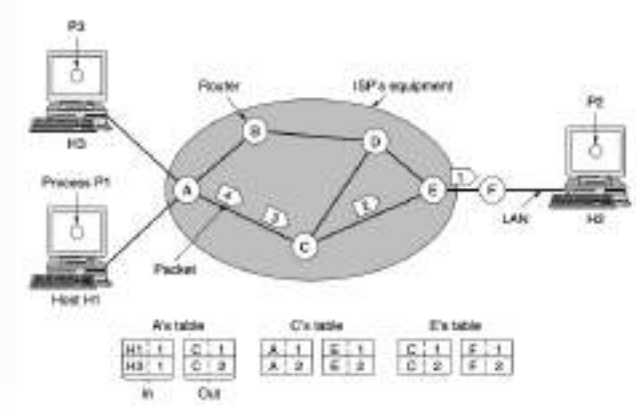


Fig 1: Defining path in Virtual Circuit

Router F transmits it to receiver host H2 and the identifier specifies that packet has been transmitted from sender host H3.
Router decides the path further based on the congestion and traffic in the network.

4.3.2 Connectionless Service – Datagram Service

In a connectionless service packets from sender host is numbered and transmitted through the available route. There is no advance connection setup like in virtual circuit. In this type of service packets are called **datagrams** and the network is termed as **datagram network**.
Fig 2 shows that the process in sender host H1 wants to communicate to process p2 at receiver host H2. The complete message shared by transport layer of H1 is divided into 4 packets of equal size.

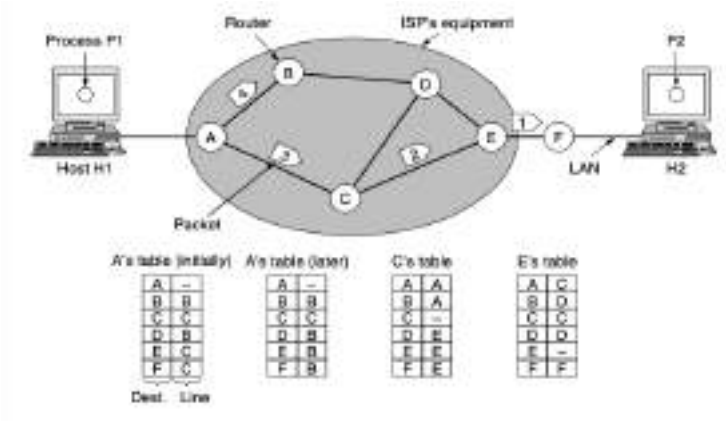


Fig 2: Datagram Service

H1 transmits each of the four packets to router A. A has internal routing table where the next router in the path i.e. line is specified based on the destination. In routing table of A there are only two outgoing paths i.e. through B and through C which are directly connected to A through point-to-point protocol.

Every packet received by A can be further transmitted to either B or C. After verifying the checksum for errors, A sends packet 1,2 and 3 to C, which further sends them to E and E sends packets to F which is further delivered to H2.

After transmitting three packets from same route, router A probably learned from the network that the path is congested, so routed it through another path. This decision is taken by routing algorithms.

4.3.3 Comparison of Virtual Circuit and Datagram Approach

Both virtual circuit and datagram approach has its benefits and drawbacks. Their comparison on the key points is detailed as below:

Key Point	Virtual Circuit Approach	Datagram Approach
Setting up circuit	Required	Not Required
Routing	Routing path is fixed during the set-up of VC and packets follow this path	Packets are routed independently, no fixed path is there
Packet delivery	All packets reach successfully as dedicated circuit is provided	Reliability of packet delivery is less as compared to VC
Effect of router failure	All VCs that pass through the failed router are terminated	No effect as packets can be routed from different path
Cost and Complexity	VC involves additional cost	Datagram network is

	of circuit set-up so it is more expensive but it is less complex	comparatively cheaper but is more complex
Example	Asynchronous Transfer Mode (ATM) used for telephone network	IP network for Internet services

4.4 IP ADDRESSES

An IP address is a unique address assigned to the device on internet. There are two versions of IP address; IPv4 and IPv6. IPv4 is a 32 bit address that can uniquely identify 2^{32} devices on the internet, whereas IPv6 is a 128 bit address and can uniquely identify 2^{128} devices.

4.4.1Understanding IP addressing

Considering IPv4, 32 bits are divided into two parts i.e. host address and network address with the help of subnet mask. 32 bits address is written as four octet address, where 1 octet is 8 bits. Each octet is converted into decimal number and is separately by dot from other octet. The decimal value of octet is from 0 to 255 and in binary it will be from 00000000 to 11111111, where both values are inclusive. Eg. 192.168.52.100 is a representation of an IP address.

4.4.2 Classes of IP Address

IP addresses are divided into five classes from class A to class E but now a days the thisclassful addressing is rarely used. Fig. 3 specify the how this addressing works. In class A, first octet is for network address and remaining three octetsi.e 24 bits for host addresses. These bits can be divided into subnets by the network administrator. Class A can have 2^7 i.e. 128 networks and 2^{24} i.e. 16 million hosts. The address range as given in figure is 1.0.0.0 to 127.255.255.255.

Class B can have 2^{14}i.e 16384 networks and 2^{16} i.e. 65536 hosts. Class C has 2^{21} networks and 2^8 i.e. 256 hosts. Class D is for multicast and class E is for future use. The address range for each class has been specified in the figure.



Fig. 3: Classful Addressing of IP Addresses

Considering the need for allocation of only required or limited number of Ips classless interdomain routing (CIDR) concept was introduced. Whenever user asks for specific number of IP addresses, CIDR dynamically assigns the block of IP addresses which are contiguous addresses and are in power of 2. The notation used for CIDR is for eg.

192.168/.52.100 / 28. Number 28 signifies that 28 bits are being used for identifying network and remaining 4 bits are for identifying the host.

2.4.3Network Masking

Network mask helps in identifying as to which part of address bits identifies network and which part identifies host. Class A, B and C has default masks as their network bits are fixed.

Following are the masks for the classes:

Class A : 255. 0. 0. 0

Class B : 255. 255. 0. 0

Class C : 255. 255. 255. 0

Masking helps in identifying network address. For eg.

Address 21.10.30.16 is a class A address with a subnet mask 255.0.0.0

In binary form it can be written as

Address : 00010101. 00001010 .00011110 . 00010000

Mask : 11111111. 00000000 .00000000 . 00000000

Octet of IP address corresponding to Mask with all 1s is network address and other is host address. We get network id in binary as 00010101 and host id as 00001010. 00011110. 00010000 in decimal network address is 21.0.0.0 and host address is 0.10.30.16.

4.5 SUBNETTING

Subnetting allows to create multiple logical networks within the specified class A,B or class C networks. In any organization the large network adds complexity and result in traffic congestion. Creating a subnet allows to put limit on number of routers for communication hence results in faster communication. The subnetting provides the following benefits:

- Broadcast domain can be divided resulting in efficient data communication and improved network performance.
- Data Packets need to travel a shorter distance to reach to the destination leading to efficient communication.
- Network security can be improved, the sensitive data can be transmitted through a specific subnet only. It gives more control to the administrators to have control over data transmission.

If the major networks in class A, B or C is divided into smaller subnetworks, it allows to create smaller interconnecting networks. Each device will have unique network/ subnetwork id.

Subnetting can be achieved by extending the standard masking with some bits of host id part. Eg. Consider class C network with address 205.17.13.0, the standard mask of class c is 255.255.255.0. Some of the bits of the last octet of mask can be used to create

subnets. If we take 3 high order bits of the last octet i.e. from host id portion, we get a subnet mask of 255.255.255.224. Let us see the binary representation of this:

Address: 11001101 .00010001 .00001101 . 00000000
Mask: 11111111 .11111111 .11111111 .**111**00000

The underlined three bits in the mask are used for subnetting and remaining five for hosts in specific subnet. With three bits 2^3 i.e. 8 subnets can be created each having host ids 2^5 i.e. 32. The first and last id of any subnet is kept reserved and is not allotted to any device. So in this example following subnets can be created each with subnet mask of 255 .255 .255 .224 :

Subnet No.	Subnet Network Id	Range of Host Ids
1	205 .17 .13 . 0	205.17.13.1 to 205.17.13.30
2	205 .17 .13 . 32	205.17.13.33 to 205.17.13.62
3	205 .17 .13 . 64	205.17.13.65 to 205.17.13.94
4	205 .17 .13 . 96	205.17.13.97 to 205.17.13.126
5	205 .17 .13 . 128	205.17.13.129 to 205.17.13.158
6	205 .17 .13 . 160	205.17.13.161 to 205.17.13.190
7	205 .17 .13 . 192	205.17.13.193 to 205.17.13.222
8	205 .17 .13 . 224	205.17.13.225 to 205.17.13.254

Mask of 255 .255 .255 . 224 can also be denoted as / 27 as 27 bits are being used for sub networking. In CIDR notation it can be represented as 205. 17. 13. 32/ 27

4.6 ROUTING ALGORITHMS

Routing is a technique of forwarding packets from sender to receiver. Routing algorithm is a technique to decide the route or path to transfer data packets from sender to the receiver. It is part of network layer software that decides the output line for the incoming packet. For datagrams this decision is made for every packet but for virtual circuit the decision is made only once and path is fixed. Routing algorithm computes the best path for transmission which is a least cost path.

The goals of routing are:

- **Simplicity:** Routing algorithm should be simple with minimum overheads.
- **Robustness:** The algorithm must perform correctly even during hardware failures or during high load in the network.
- **Stability:** It should be stable to handle any issues in the network. Also the convergence should be fast.
- **Flexibility:** Routing algorithm should adapt to changes due to network bandwidth, delay or failure of any links.
- **Fairness:** The algorithm must provide equal opportunity to all the nodes to transmit packets according to quality of service requirements.

- **Optimality:** The algorithm should be able to find the best route depending on the metrics like throughput, delay etc.

4.6.1Types of Routing Algorithms

Routing Algorithms can be broadly categorized into non-adaptive and adaptive routing algorithms.

Non-adaptive algorithms do not make their routing decisions on any estimates of the traffic or the current network topology. Choice of the route from router A to router B is computed in advance and in offline mode. The same information is downloaded by routers when the network is booted. It is a static routing and does not respond to failures.

Adaptive algorithms change their routing decision based on the topology and traffic. It is a kind of dynamic routing algorithm.

4.6.2 Non-adaptive Algorithms

The static routing tables are constructed for packet transmission and router stores these routing tables when the network is up. Once the static paths are available with the routers in the network the packet transmission starts. The change in topology or traffic conditions do not have any effect on the previously constructed routing tables. There are majorly two types of non-adaptive algorithms.

4.6.2.1 Flooding

In a simple technique of flooding when the data packet arrives at any router it is transmitted to all the outgoing links except the one from which it arrived on. Each router make decisions based on the local knowledge and not with the complete picture of the network.

Large number of duplicate packets are generated in flooding if it is uncontrolled.

4.6.2.1.1 Controlled flooding

To control the flooding, hop counter is added in the header of the packet, it is initialized to the length of the path from sender to receiver. Hop counter is decremented at each hop and packet is discarded if hop counter reaches to zero. It the length of path is unknown the hop counter is put as the full diameter of the network. In this technique lot of duplicate packets will be generated if the hop count is large.

Sequence Number Controlled Flooding: Another technique to control the flooding is to avoid transmitting them again by keeping the track of packets at router. A sequence number is put by the source router for every transmitted packet. Each router maintains a list of the sequence numbers of packets transmitted by it per source. If the incoming packet to the router is in the list then it is not flooded.

Reverse Path Forwarding: It works on the technique that any broadcasted packet arriving at the router is checked by the router to see if the packet has arrived on the link that is generally used for transmitting packets towards sender for broadcast. It it is so, then it is assumed that broadcasted packet followed the best route from the router and is the first copy of the packet that has arrived to the router. If the packet has

arrived from any other link apart from that, it is considered as duplicate and is discarded. It is an efficient technique and is simple to implement.

4.6.2.1.2 Selective Flooding

Some applications like online games, do not require the packets to be delivered to all but only to the selected group. In this technique, packets are transmitted only to a specific group. It is also known as multicast routing. Group is created of the routers, each group is identified by multicast address and routers know to which group they belong. It is done by the techniques known as multicast spanning tree.

Flooding has the following advantages:

- It ensures that packet is delivered to every node which is effective for broadcasting.
- It is robust. Even if several routers go down, the path from sender to receiver will be available.
- In flooding router needs to know its neighbours so it can be used as basic technique for other routing algorithms as well.
- It chooses the shortest possible path for transmission.

4.6.2.2 Random Walk

n this algorithm, a packet is sent by node to one of its neighbours randomly. It suffers from the drawbacks of dead-end when no out-link is there. Also, when there is no route to another network then it becomes kind of infinite cycle or a spider-trap.

4.6.3 Adaptive Algorithms

The routing decisions in this category of algorithms changes due to changes in topology and network traffic. These are the dynamic algorithms, the router get the information either from the adjacent routers or from all the routers. These algorithms differ in the way of getting network information, time of change of routes, metrics for deciding optimal routes.

The adaptive algorithms are categorized into centralized, isolated and distributes algorithms.

4.6.3.1 Centralized Algorithm

As the name suggests, a central node has the information of entire network and it takes the routing decisions. It has an advantage that the resources are required only for the central node to store and process complete data but on the other hand it has a disadvantage that in case of failure of central node the entire network will go down.

4.6.3.2 Isolated Algorithm

In this algorithm the node takes a decision in isolation i.e. without seeking information from other nodes. The status of the link to which packet is transmitted is not known to the node. It has a major disadvantage of delay in transmission in case of congestion in the link. Examples of this type of algorithm are:

- **Hot Potato:** In this algorithm as and when packet arrives at node, it tries to transmit it to the shortest output queue, without considering whether the link is correct link to the destination or not.
- **Backward Learning:** In this algorithm, routing tables at each node is modified based on the information received through incoming packets. It is generally implemented by adding identity of source node together with hop counter. At the node, number of hops the packet has taken to reach to the node is noted. If the current hop count is better than the previously stored value in the node, it is replace with the new value. This algorithm does not store the previous records, so in case the best route is down, it does not have the information about second best route. The process is started all over again.

4.6.3.3 Distributed Algorithm

It is a decentralized algorithm, every node receives information from the neighbouring node and takes its own decision. The least-cost path is calculated in a distributed manner. It has an advantage that each node can dynamically change the decision and choose the best path for transmitting packets. But, this approach may result in delay due to time required by each node to take decision.

4.6.3.4 Distance Vector and Link State Packet Routing Algorithms

The two main algorithms under category of adaptive algorithms are distance vector routing and link state packet routing.

4.6.3.4.1 Distance Vector Routing

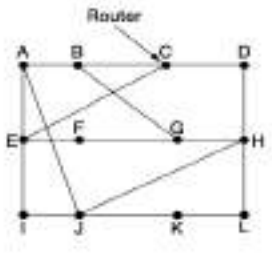
It is a dynamic routing algorithm in which each router maintains a table containing information about the best route to each destination and the link for transmitting packet to the destination. Routers exchange these tables with the neighbours and accordingly every router gets the information about best link to reach to the destination.

Every router maintains routing table containing one entry corresponding to each router in the network. Every table entry has two fields, one for preferred outgoing link to reach to destination and distance to the destination either in terms of hops or any other metric. The routing table is shared with all the neighbours and they update their tables accordingly.

Fig. 4 below shows the distance calculation using this algorithm. Every router in the network has its routing table specifying the distance to reach to another router. The metric used for calculating distance is the delay. Consider a case where distance from router J to G has to be calculated. The neighbours of J have its own routing table to reach to any other router as specified in figure.

Router J is also maintaining its routing table but based on the information from neighbours it will update its routing table which is explained below.

The information maintained currently by J to reach to its neighbours are:



Current distance from J to A : 10, J to I : 12, J to H : 8, J to K : 7

To	From			
	A	**I**	**H**	**K**
A	**0**	**24**	**20**	**21**
B	**12**	**36**	**31**	**28**
C	**25**	**18**	**19**	**36**
D	**40**	**27**	**8**	**24**
E	**14**	**7**	**30**	**22**
F	**23**	**20**	**19**	**40**
G	**18**	**31**	**6**	**31**
H	**17**	**20**	**0**	**19**
I	**21**	**0**	**14**	**22**
J	**9**	**11**	**7**	**10**
K	**24**	**22**	**22**	**0**
L	**29**	**33**	**9**	**9**

Fig 4:Example ofDistance Vector Routing

Based on the tables distance from J to G is calculated which is
J to G through A : Distance (J to A + A to G) = 10+18 = 28
J to G through I : Distance (J to I + I to G) = 12+31 = 43
J to G through H : Distance (J to H + H to G) = 8+6 = 14
J to G through K : Distance (J to K + K to G) = 10+31 = 41

From the above data it is evident that J to G distance is 14 which is the minimum distance and it is through I, so the specific entry in the table of J is updated and similarly other entries.

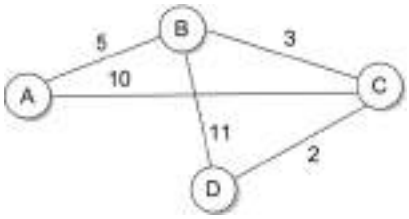
This algorithm suffers from a drawback termed as count to infinity problem. It is a routing loop that is formed due to failure in a link.

4.6.3.4.2 Link State Routing

The link state routing algorithms each router follows the steps:

- Discover the neighbours and get their network address.
- Set a distance or any metric to reach to each neighbour.
- Construct a packet with this information and send packets to all the routers.
- Receive the packets from other routers and calculate the shortest path to other routers.

Fig 5 below shows the link packets formed by each router by putting information about the distance to its neighbour. Any specific metric can be used to find distance.



Link Packets by

A		B		C		D	
B	5	C	3	A	10	B	11
C	10	D	11	B	3	C	2
				D	2		

Fig. 5 Link State Packets formed by routers

The link packets prepared by routers are shared with all and accordingly routers make their own routing table to reach to the destination. Sequence number of link state packets and age of packets help in rejecting the obsolete packets.

4.6.4 Comparison of Distance Vector Routing and Link State Routing

Based on the key points in the algorithms these two major adaptive algorithms can be compared as below:

Key Points	Distance Vector Routing	Link State Packet Routing
Underlying algorithm	Bellman Ford Algorithm	Dijkstra Algorithm
Metric for distance Calculation	Hop Count	Link Cost
Information Sharing	Router shares the information to its neighbours about its distance from every other router in the network	Router shares the information to every router about its distance from its neghbours
Updates in the tables	Frequent	Event triggered
Convergence	Slow	Fast
Issues	Count to infinity (routing loop) and creates more traffic as compared to link state	High processing requirements and memory requirements

4.7 SUMMARY

This module covers the functionality of network layer. The main function of the network layer is to provide end-to end delivery of packets between source to destination. It provides the routing of the packets in multi-hop networks. The two approaches of packet transmission – virtual circuit which is connection oriented and datagram which is connectionless service have been discussed. As the network layer deals with the packet transmission in the network, so unique identification or the addressing of nodes is required. IP addressing is the technique

used for uniquely identifying nodes in the network. Two versions IPv4 and IPv6 are used for this identification. IPv4 though most commonly used has a limitation to uniquely identify only 2^{32} nodes as it is a 32 bit address. With the increase of number of nodes in the internet addressing has been moved towards IPv6 which is 128 bit addressing scheme.

The earlier classful addressing for the networks is not very beneficial for the organization that require limited number of IP addresses. If the organization has requirement of 120 IP address then getting one Class C network address, which can address 256 nodes will be a wate of IP addresses as remaining 136 (256-120) addresses will not be ever used by the organization but at the same time it cannot be allocated to any other organization. Considering this subnetting was introduced, through which any number of IP addresses required by the organization was allocated thus avoiding unused IP addresses.

Routing another major function of network layer has underlying algorithms, the adaptive algorithms which are dynamic in nature are preferred inspite of the high complexity than static or non-adaptive algorithm. Distance vector routing and link start routing are examples of adaptive algorithms being used for routing. Each of these two has its advantages and disadvantages. Routing information protocol (RIP) and interior gateway routing protocol (IGRP) are examples of distance vector routing whereas an interior gateway routing protocols open shortest path first (OSPF) and intermediate-system to intermediate-system (IS-IS)are examples of link state routing algorithm.

4.8 REVIEW QUESTIONS

a. List the functions of network layer and the services it provides to upper layers.

b. Which are the factors that make datagram service better than the virtual circuit? Also, give the example of datagram service.

c. What was the requirement for IPv6 and how many unique addresses it can assign to the devices connected to internet?

d. Convert these IP addresses into decimal notation and identify in which class these addresses lie.

- 00011111.11110000.00110011.00000100
- 10111111.00110000.00000100.00110000
- 11000001.01111111.11111100.11110000
- 11111110.00001100.11111111.00001110

e. How the subnetting helps in improving network performance? What does notation 192.68.12.0/27 signifies.

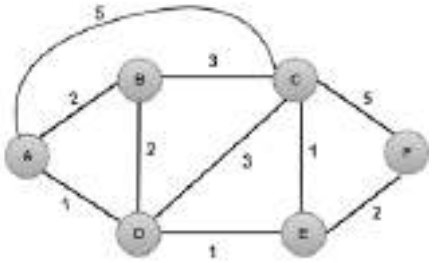
f. Determine the subnetwork address for 172.16.18.19/20.

g. Consider class C network with address 204.18.4.0, with subnet mask of 255.255.255.240 , identify the subnetwork ids and host range in each subnet.

h. Enumerate the differences between adaptive and non-adaptive routing algorithms.

i. Explain how flooding helps in broadcasting. What are disadvantages of flooding?

j. In the below figure, calculate the shortest path from A to F by showing all complete process of calculating distances.



REFERENCES & FURTHER READING

- Data Communication & Networking by BehrouzForouzan
- Computer Networks by Andrew Tannenbaum
- Data and computer Communications by William Stallings
- Computer Networking: A Top-Down Approach by James Kurose and K.W. Ross

UNIT 5: TRANSPORT LAYER

STRUCTURE

5.0 Objectives

5.1 Introduction

5.2 Transport Layer's Services

5.3 Elements of Transport Layer Protocols

- **5.3.1 Addressing**
- **5.3.2 TCP Connection Establishment**
- **5.3.3 TCP Connection Termination**

5.4 Flow Control and Buffering

5.5 Transport Layer Protocols

- **5.5.1 Introduction to UDP**
- **5.5.2 UDP Segment Structure**
- **5.5.3 Introduction to TCP**
- **5.5.4 TCP Segment Structure**

5. 6 Review Questions

5.7 Summary

5.8 References for Further Readings

5.0OBJECTIVES

At the end of this unit, you will be able to:

- Understand the issues related with transport layer.
- Discuss different services of transport layer.
- Discuss in detail the various elements of transport layer protocols.
- Understand the concept flow control and buffering.
- Discuss frame formats of TCP and UDP.
- Discuss steps for connection establishment and connection release.

5.1 INTRODUCTION

The transport layer is the OSI model's link pin. It is not just another layer but the heart of the whole protocol hierarchy. Protocols on this layer monitor the transmission on another device of data from one application programme into an application programme. This is the first OSI end-to-end layer. It provides services on the upper layer to use the network layer services and other protocols on the lower layer.

We should be aware that the Internet is made up of many physical networks such as LANs, MANs, and WANs that are linked together using a variety of media (wired and wireless) to enable data to be transferred from one network to another. The data in a transmission may be transformed as it travels from one network to another, i.e., it may be encapsulated in various forms and lengths of packets.The complexity of the physical networks at the transport layer is hidden from upper-layer protocols. Individual physical networks appear to the upper layers as a simple homogeneous network capable of efficiently transporting data. And if an Ethernet is replaced by an FDDI (Fiber Distributed Data Interface) in the LAN portion of the internet, the upper layers are unaffected. For them, the Internet is a single, mostly static network. This is provided by the transport layer.

Examples of transport layer protocols are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

5.2 Transport Layer's Services

Let's start with the services that a transport protocol can or should provide to upper layer protocols. The world for transportation services is depicted in below Figure. A transport object, which may be an application mechanism like http, telnet, etc., offers services to upper layer users. Depending on the network layer's facilities, this local transport entity interacts with any remote transport entity. As stated, a transport protocol's general service is to provide end-to-end data transport while shielding the upper layer user from the specifics of the underlying network infrastructure.

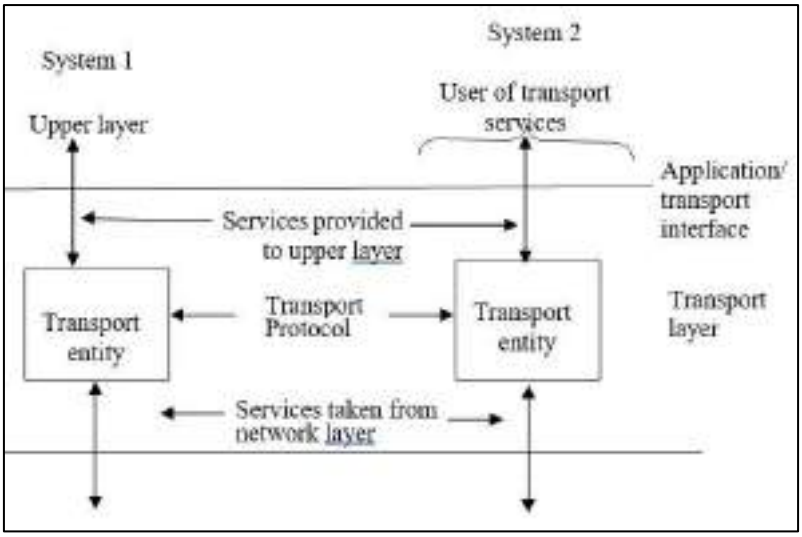


Figure: Transport Entity Environment

The services offered by the transport layers are typically classified into the following categories.

Types of services

1. Quality of service
2. Data transfer
3. Connection management
4. Expedited delivery

The Internet provides two basic types of services:

(i)Connection-oriented services and (ii) Connectionless, or datagram services.

Connection-Oriented

The most of protocol service is a connection-oriented service. It allows users to create, maintain, and terminate a logical link between them. Handshake procedures are another name for this. TCP is used to provide Internet connection-oriented services (Transmission Control Protocol). TCP provides a variety of services to an application, including reliable transportation, flow management, and congestion control. These programmes have been recognised. To preserve throughput, the TCP's congestion management function is used.

Connectionless Service

In connectionless service, before sending the packet, there is no handshaking. There are still no systems in place to manage traffic flow and congestion. Data can be transmitted more quickly since there is no handshaking protocol. However, although these services are acknowledged, there is no secure data transfer. UDP is the name of the Internet's connectionless operation (User Datagram Protocol). Online telephony and video conferencing are two examples of connectionless service applications.

1. Quality of Service

The upper layer protocol users should be able to define the types of transmission service quality that will be given by the transport protocol entity. Based on these requirements, the transport agency tries to make the best use of the underlying link, network, and other resources in order to provide the collectively requested services to the best of its capacity. However, these networks are constrained by the network layer services' internet capabilities.

Examples of services that might be requested are:

- Expedited delivery of packet
- Acceptable error and loss levels of packet
- Desired average and maximum delay in transmission of a packet
- Desired average and minimum throughout
- Priority levels.

You are aware that IP is a network layer standard protocol. IP does include a priority parameter, as well as binary specifications for normal or low latency, normal or high throughput, and normal or high reliability. As a result, the transport entity will communicate with the internetwork entity. To achieve desired throughput, the network can change flow control parameters and the amount of network resources allocated to a virtual circuit.

You must be aware that different applications may require different quality of services. For example:

- A file transfer protocol might require high throughput. It may also require high

  reliability to avoid retransmissions at the file transfer level.
- A transaction protocol (e.g., web browser-web server) may require low delay.
- An electronic mail protocol may require multiple priority levels.

The inclusion of a quality-of-service facility within the protocol is one approach to providing a variety of service qualities; transport protocols typically follow the same approach. Another option is to have different transport protocols for different types of traffic; the ISO-standard family of transport protocols takes this approach.

.2. Data Transfer

The major purpose of a transport protocol is to transfer data between two transport entities. User data and control data must be transferred from one end to the other on the same channel or on separate channels. Full-duplex service is required. Half-duplex and simplex modes may also be available to accommodate the needs of specific transport users.

If the network layer performs a similar function, why is it necessary at the transport layer? The network layer manages the hop-by-hop delivery of individual packets but sees no

relationship between them, even if they are part of a single message. Each packet is treated as a separate entity. The transport layer, on the other hand, ensures that not just one packet but the entire sequence of packets is delivered.

3. Connection Management

When providing connection-oriented service, the transport entity is in charge of establishing and terminating connections. It is possible to provide both symmetric and asymmetric procedures for connecting establishments.

Connections can be terminated abruptly or gracefully. Data in transmission may be lost if the connection is abruptly terminated. A graceful termination prevents either party from terminating until all data has been delivered.

4. Expedited Delivery

The expedited delivery of data is a service similar to that provided by priority classes. Some data submitted to the transportation service may supersede previously submitted data. The transport entity will make every effort to have the transmission facility send the data as quickly as possible. The transport entity will interrupt the transport service user at the receiving end to notify it of the receipt of urgent data. Thus, the expedited data service is an interrupted mechanism that is used to transfer occasional urgent data, such as a terminal break character or an alarm condition. A priority service, on the other hand, may devote resources and adjust parameters in such a way that, In contrast, a priority service might dedicate resources and adjust parameters such that, on average, higher priority data are delivered more quickly.

5.3 ELEMENTS OF TRANSPORT LAYER PROTOCOLS

Transport layer protocols such as TCP and UDP are used to implement the services provided by the transport layer. We'll go over them in the next unit. In this section, we will look at some of the most important components of the transport layer. A transport layer is in charge of hop-by-hop processes.

5.3.1 Addressing

The issue at hand is simply this: how will a user process establish a connection to a remote user process? How should the address of a remote process be specified? The most common method is to specify a transport address to which the process can listen for connection requests. These end points are known as ports or TSAPs in Internet jargon (Transport Services Access Points). Similarly, these network layer end points are referred to as NSAPs (Network Services Access Points). The diagram below depicts the connection between a user process as host 1 and a remote process (server) as host 2.

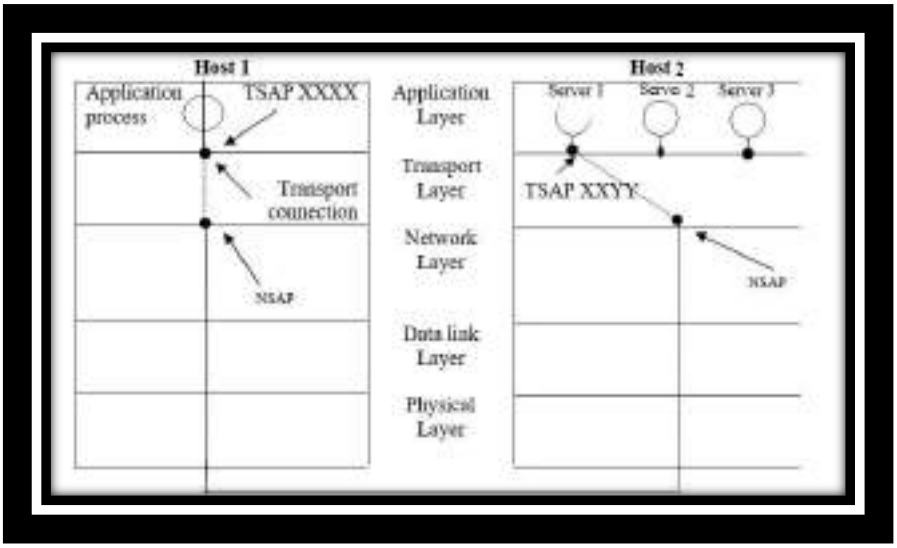


Figure: Transport Connections between a Client and a Server

The following steps may be needed for establishing a transport connection:

Assume that an application process running on host-1 wants to know the weather forecast for the next day, so it sends a ***CONNECT*** request specifying its **transport connection point number** TSAP XXXX as well as the weather forecasting server's TSAP number XXYY, which will establish a transport connection with the destination. This action results in a transport connection being established between the application process on host 1 and a server 1 on host 2.

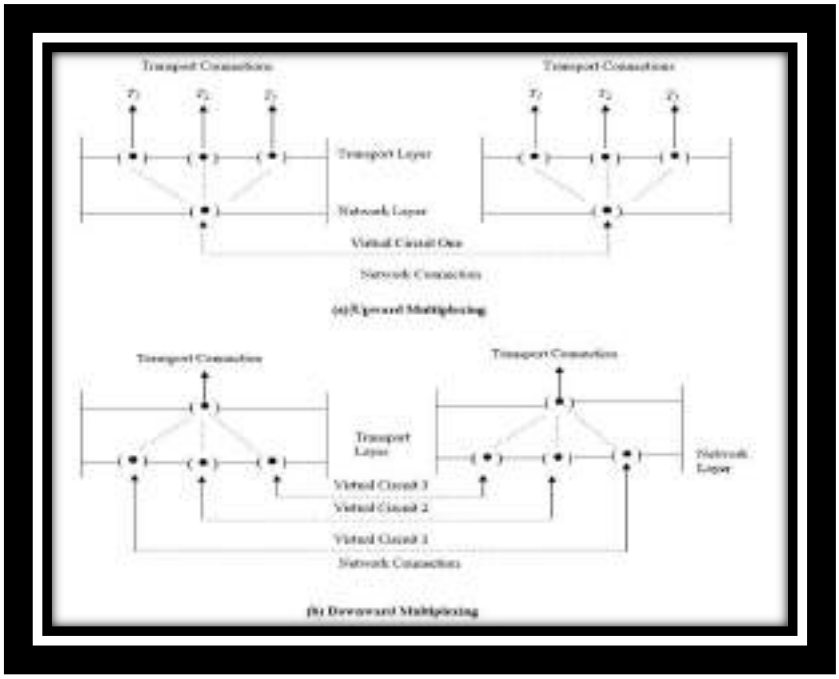
The following methods are used to know thatthe address of the destination server process is attached to a particular transport connection by the host user

Initial connection protocol

A process server acts as a proxy server for less heavily used servers in the scheme. Instead of each server listening at a well-known port address for a connection request, a process server listens to groups of ports at the same time and notifies a specific server to act upon receiving a connecting request to perform the required work. The new server then handles the request, while the process server returns to waiting for new requests. As a result, a rarely used server should not be listening to ports all of the time via a process server.

Some commonly used services are assigned "well-known address" (for example, time sharing and word processing).

Multiplexing: The transport layer can also perform multiplexing. Upward multiplexing, defined as the multiplexing of multiple transport connections on a single network link, and downward multiplexing, defined as the splitting of a single transport connection among multiple lower-level connections, are the two forms of multiplexing techniques. Figure below shows how this works.



Downward multiplexing or splitting could be used to provide more bandwidth than can be managed by a single virtual circuit. One way out of this is to open several network connections and round robin traffic, as shown in the above figure (b). The effective bandwidth is increased by a factor of k when the network connections are opened.

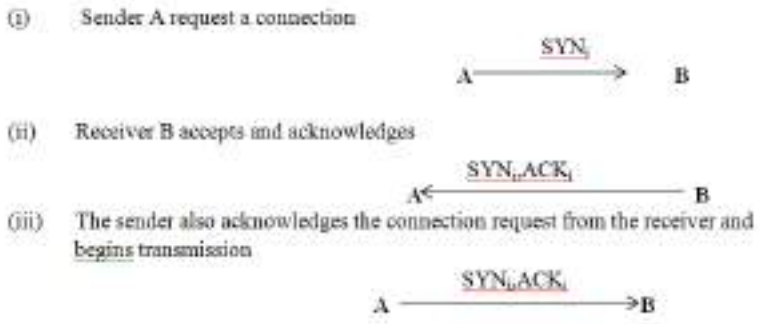
5.3.2 TCP Connection Establishment

A link must be formed before data can be transmitted. Connection establishment must account for a network service's unreliability, which can result in the loss of ACK, data, and SYN. As a result, these packets would need to be retransmitted. Remember that establishing a relation necessitates the exchange of SYNs.

Steps of typical three-way handshake operations are:-

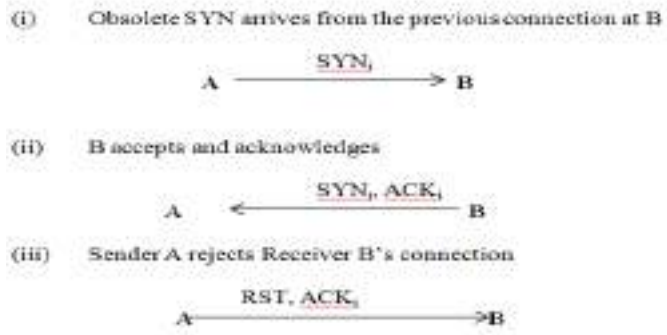
1) At the sender side transport entity A, initiates the connection to transport to entity B by setting SYN bit.
2) At the sender side transport entity B, acknowledges the request and also initiates a connection request.
3) An acknowledges to connection request of A and B sends a retransmission and B retransmits.

Normal Transmission



Delayed arrived of SYN packet:

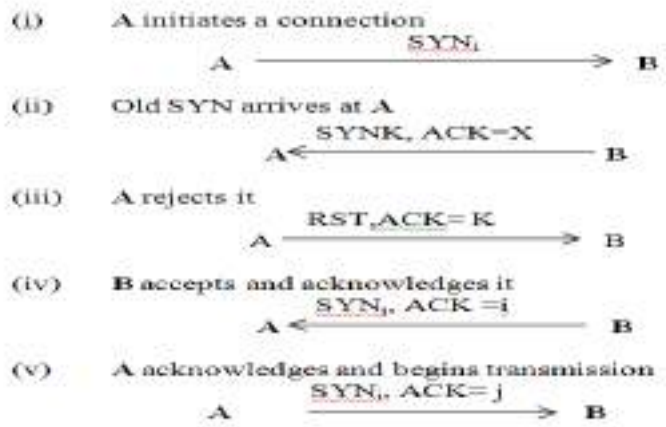
Now, let us test how this mechanism handles delayed SYN and ACK packets. It is shown that old SYN X arrives at B after the close of the relevant connection as shown below. B assumes that this is a fresh request and responds with SYN $_j$, ACK$_i$. When A receives this message, it realizes that it has not requested a connection and therefore, sends an RST, ACK $_j$. Note that the ACK $_j$ portion of the RST message is essential so that an old duplicate RST does not abort a legitimate connection establishment.



Delayed SYN and ACK

The final example shows a case in which an old SYN, ACK arrives in the middle of a new connection establishment. Because of the use of sequence numbers in the acknowledgements, this event causes no harm.

Both transport entities must agree on their initial sequence number, which must be different, as part of the three-way handshake protocol. What is the aim of having different sequence numbers?



5.3.3 TCP Connection Termination

TCP uses a method close to that of relation establishment. TCP allows for a graceful close that requires the closure of each link path independently. When an application informs TCP that it has no more data to send, a termination occurs. To be accepted, each side must clearly recognise the other's FIN. A graceful close occurs successfully with the following moves.

The sender must send FIN_j and receive an ACK_j

- It must receive a FIN_j and send an ACK_j
- It must wait for an interval of time, to twice the maximum expected segment lifetime.

Acknowledgement Policy:

When a data segment arrives that is in sequence, the receiving TCP entity has two options concerning the timing of acknowledgment:

•**Immediate:** When data are accepted, immediately transmit an empty (no data) segment containing the appropriate acknowledgment number.

•**Cumulative:** When data are accepted, record the need for acknowledgment, but wait for an outbound segment with data on which to piggyback the acknowledgement. To avoid a long delay, set a window timer. If the timer expires before an acknowledgement is sent, transmit an empty segment with the appropriate acknowledgement number.

5.4 Flow Control And Buffering

In order to support a flux control mechanism, similarities and differences exist between the data link layer and the transport layer. Whereas flow control is a relatively simple mechanism at the data link layer, it is a rather complex mechanism at the transport layer, for two main reasons:

- Flow control at the transport level involves the interaction of three components: TS users, transport entities, and the network service.
- The transmission delay between transport entities is variable and generally long compared to actual transmission time.

The following delay may arise during the interaction of the two transport entities A and B:

(i) Waiting time to get permission from its own transport entity (interface flow control).

(ii) Waiting time by A to have permission to send the data to B.

(iii) Waiting time as network layer services.

In any case, once the transport entity has accepted the data, it sends out a segment. Later on, it receives an acknowledgement that the data has been received at the remote end. It then sends a confirmation to the sender, about the same.

On the others side, in case of flow control problem, firstly we present two ways of coping with the flow control requirement by using a fixed sliding-window protocol and a credit scheme.

The performance of the transport connection can be reduced in long-term cases by a conservative flow management scheme. By optimistically granting credit for space, the recipient could potentially increase the performance. The buffer allocation scheme is called dynamic. E.g. a recipient's buffer is full but expects to space for two segments in a time of roundtrip propagation, a loan of 2 can be sent immediately. This scheme can increase outcomes and do no harm if the receiver can keep up with the sender. However, if the sender is faster than the receiver, segments that require a re-transmission may be discarded. Because retransmissions are not otherwise necessary with a reliable network service, an optimistic flow control scheme will complicate the protocol.

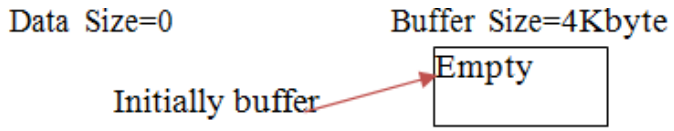
The optimal compromise between source and target buffer depends on the type of traffic that the connection performs. It is better not to dedicate buffers for low bandwidth, bursty traffic such as those produced by an interactive terminal, but to acquire buffers in both directions. The recipient must keep a copy of the TPDU (Transport Protocol Data Unit) until acknowledged since the sender cannot be certain that the recipient will be able to acquire a buffer. On the other hand, it is better to allow data to flow at high speed for file transfer and other high bandwidth traffic if the recipient has a whole buffer window. Therefore, for low bandwidth explosive traffic, the buffer on the sender is better, and a buffer on the recipient is better for high bandwidth smooth traffic.

TCP Flow Control

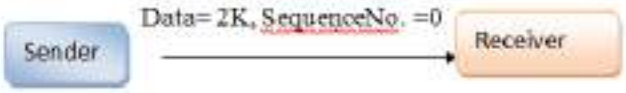
The method of controlling traffic between two end points is known as flow control, and it is used to prevent the sender from sending more data as compare to the recipient capacity to process it. To prevent the sender from overflowing, TCP offers a flow control service. TCP handles flow control at the receiver's buffer by using a sliding window with a credit scheme. The scheme allows the receiver to have more control over the data flow. A part of a credit scheme can be recognized without the guarantee of new credit, and vice versa. The two are intertwined in a fixed sliding window control (used at the data link layer).

E.g. assume that the sender wants to send application data to the receiver.

i) The receiver has 4 K byte buffer which is empty as shown below:



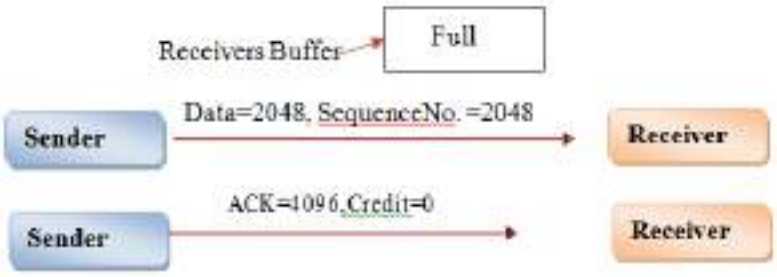
ii) Sender transmits 2 K byte segment (data) with sequence number 0 as shown below:



iii) The packet is examined at the receiver, after that it will be acknowledged by it. It will also specify the credit value (window size) of the receiver. Until the application process running at the receiver side removes some data from buffer, its buffer size remains fixed at 2048. Therefore, credit value (window size) is 2048 byte.



iv) Now the sender transmits another 2048 bytes, which is acknowledged, but now the advertised window (credit) is 0.



v) The sender must stop sending data until the application process on the receiving host has removed some data from the buffer, at which time TCP can advertise, a large window (credit value).



When the credit size is zero, normally there is no transmission from the sender side except in two situations:

a). Urgent data requires to be sent

b). Sender wants to know the credit size and the next byte expected by the receiver.

Both senders and receivers can delay transmission to save money on their end. If a sender knows that a receiver window's buffer capacity is 8 kilobytes and it has only received 2 kilobytes, it can buffer it at the sender side before it receives more data from the application phase. Similarly, if the receiver has to send any data to the sender, the acknowledgment can be delayed until the data is ready, and the acknowledgement can then be piggybacked. As a result, the primary motivation for delaying acknowledgment is to save bandwidth.

5.5 TRANSPORT LAYER PROTOCOLS

The protocol layer just above the Internet Layer is the *Host-to-Host Transport Layer*. This name is usually known to *Transport Layer*. The two most important protocols in the Transport Layer are *Transmission Control Protocol* (TCP) and *User Datagram Protocol* (UDP). TCP provides reliable data delivery service with end-to-end error detection and correction. UDP provides low-overhead, connectionless datagram delivery service. Both protocols deliver data between the Application Layer and the Internet Layer. Applications programmers can choose whichever service is more appropriate for their specific applications.

5.5.1 Introduction to UDP

As, you know that UDP and the transport protocol are unreliable. Apart from multiplexing, UDP adds little to the IP protocol and some error correction.

If we choose UDP for application development instead of TCP, then the application speaks nearly directly to the IP layer. For the multiplexing/demultiplexing services, UDP takes messages from the application process, attaches the source and destination port numbers to the fields, adds two small fields and passes the resulting segment on to the network layer. The network layer encapsulates the segment into an IP datagram and makes best efforts for transferring the segment to the receiving host without making a swipe between sending and receiving transmission layer entities. . If the segment arrives at the receiving host, UDP uses the destination port number to deliver the segment’s data to the desired application process.

DNS and many other applicationsare better suited for UDP for the following reasons:

No connection establishment: Since UDP does not delay the establishment of a connection,henceUDP-DNS is faster as compare to TCP. But, HTTP uses TCP instead of UDP because reliability in text Web pages is critical.

More support for the client: In the end systems, TCP maintains the connection state. This connection state includes the reception and sending of buffers, congestion control parameters and parameters for sequence and number recognition. This state information is necessary for the implementation and congestion control of reliable data transfer service by TCP. On the other hand, UDP has no connecting status and does not follow any of these parameters. When the application runs via UDP instead of TCP, a server dedicated to a particular application typically supports more active clients. Due to the absence of reliable data service and congestion control, UDP does not have to maintain the receipt or send of buffers (connection state, congestion control parameters, sequence and recognition number configurations), and therefore does not have to monitor the sending and sending of buffers.

Small packet header overhead: The TCP segment has 20 bytes of header over- head in every segment, whereas UDP has only eight bytes of overhead.

Can a reliable UDP application be developed? Yes, it can be done at the application level through the addition of recognition and retransmission mechanisms. Many proprietary streaming applications today are only possible this is they run over UDP, but they have built acknowledgements and retransmissions into the application in order to reduce packet loss.

5.5.2 UDP Segment Structure

UDP is an end to end transport level protocol that adds only port addresses, checksum error control and length information to the data from the upper layer.

The data for the application covers the UDP segment data field. For example, for DNS, a query or response message is contained in the data field. Audio samples fill in the data field for a streaming audio application. A user datagram is called the packet generated by UDP.

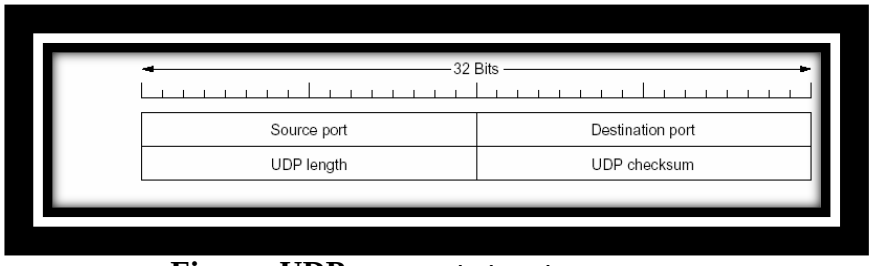


Figure: UDP segment structure

The UDP header has only four fields, each consisting of two bytes. Let us discuss each field separately:

Source port address:

It is the address of the application program that has created the message.

Destination port address:

It is the address of the application program that will receive the message.

Total length:

It specifies the length of UDP segment including the header in bytes.

Checksum:

The checksum is used by the receiving host to check whether errors have been introduced into the segment. In truth, the checksum is also calculated over a few of the fields in the IP header in addition to the UDP segment.

5.5.3 Introduction to TCP

TCP is a protocol that allows two processes (TCP users) to communicate reliably over a range of networks and Internets, both stable and unreliable. TCP is a Connection Oriented protocol. Any link is treated as if it were a stream of bytes. In contrast to UDP, the user application does not need to bundle data in individual datagrams. A connection between the sender and the receiver is needed in TCP. TCP requires a VC (Virtual Circuit) at the IP layers to establish this connection, which must be active for the duration of the transmission. The data is placed in allocated buffers and sent in segments via TCP. In addition, TCP provides two useful facilities for labelingdata;**push (PSH)** and **urgent (URG)**.

When the PSH (data push) bit is set, this is an indication that the receiver should pass the data to the upper layer immediately. The URG (Urgent) bit is used to indicate that there is data in this segment that the sending side upper layer entity has marked as urgent. The location of the last byte of this urgent data is indicated by the 16 bit urgent data pointer field.

Both IP and UDP treat multiple datagrams belongings to a single transmission as entirely separate units, un-related to each other. TCP on the other hand is responsible for the reliable delivery of entire segments. Every segment must be received and acknowledged before the VC is terminated.

5.5.4 TCP Segment Structure

TCP uses only a single type of protocol data unit, called a TCP segment. The header is shown in Figure below. Because one header must perform all protocol mechanisms, it is rather large, with a minimum length of 20 octets. A segment beginning with 9 fixed format 20 byte headers may be followed by header option. After the options, if any, up to 65,535 – 20 (IP header) – (TCP header) = 65,445 data bytes may follow. A Segment with no data is used, for controlling messages and acknowledgements.

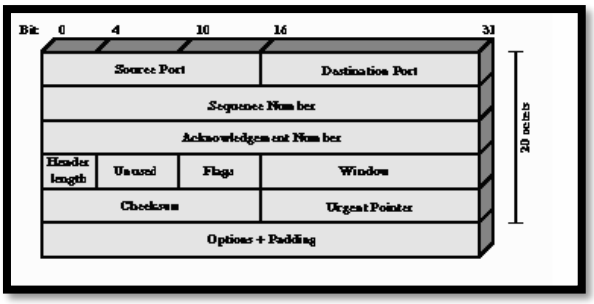


Figure: TCP header format

Source port (16 bits):
Source service access points and identify local end points of connection.

Sequence number (32 bits):
Sequence number of the first data octet in this segment except when SYN is present. If SYN is present, it is the initial sequence number (ISN), and the first data octet is ISN +1.

Acknowledgement number (32 bits):
A piggybacked acknowledgement contains the sequence number of the next byte that the TCP entity expects to receive and not for last byte correctly received. Separate number field and acknowledgement number fields are used by the TCP sender and the receiver has to implement a reliable data service.

Flag:
Since each flag is one bit long, and as there are 6 flags:***URG****ent Pointer,* ***ACK****nowledgement, Push(****PSH****), Reset (****RST****),****SYN****chronisation and****FIN****ish.* This makes the Flags section 6 bits in total. The most popular flags are the "SYN", "ACK" and "FIN", used to establish connections, acknowledge successful segment transfers and, lastly, terminate connections.

URG: Used to indicate that there is data in this segment which sends the at the upper layer has marked urgent. The location of the last byte of this urgent data is indicated by the 16 bit urgent data pointer field.

ACK: Acknowledgement field indicates that the value carried in the ACK field is valid.

PSH: Push function. The receiver is represented to deliver the data to the application upon arrival, and not buffer it until a full buffer has been received.

RST: Reset the connection due to host crash or some other reason. SYN: Synchronise the sequence numbers.

FIN: No more data from sender.

Window (16 bits):
The number of data bytes that the receiver is willing to accept, starting with the one indicated in the acknowledgement field.

Checksum (16 bits)

It adds extra assurance. It calculates the checksums for the header, data, and conceptual pseudo header. The Checksum algorithm simply adds all of the 16-bit words in one's complement and then subtracts the total. As a result, the result should be 0 when the receiver performs calculations on the entire segment, including the checksum field.

Urgent Pointer (16 bits)

Points to the octet following the urgent data; this allows the receiver to know how much urgent data is coming.

Options (Variable):

At present, only one option is defined, which specifies the maximum segment size that will be accepted.

5.6REVIEW QUESTIONS

1. Why User datagram protocol is called connectionless?

2. Explain the three types of address used by TCP/IP?

3. Why TCP services are also called stream delivery services?

4. When a process server acts as a proxy server?

5.7 SUMMARY

The two transport layer protocols (TCP and UDP) were thoroughly discussed in this unit. The transport port can be small (very simple) and provide minimal services. In such instances, the programme communicates directly with the IP. UDP is an example of a transport protocol that offers the bare minimum of features, including no power, link establishment, acknowledgment, or congestion handling. As a result, if the application is to be built around UDP, certain features must be supported inside the application. TCP, on the other hand, is a protocol that offers many benefits, including reliability, flow control, and link establishment to various application services. Nonetheless, the network layer protocol is often constrained by the services provided by the transport layer.

TCP link creation, UDP and TCP header formats, TCP flow control and finallycongestion control were all discussed.

5.8 REFERENCES

1. A.S Tanebaum, Computer Networks, 4th Edition, PHI, New Delhi
2. J.F. Kurose & K.W. Ross, Computer Networking, A top down approach featuring theInternet, Pearson Edition, New Delhi
3. William Stallings, Data and Computer Communication, Pearson Education, New Delhi
4. BehrouzForouzan, Data communications and Networking, Tata Mc-Graw Hill.

UNIT 6: SESSION AND PRSENTAION LAYER

STRUCTURE

6.0 Objectives

6.1 Introduction

6.2 Concepts

6.3 Session Layer Functions & Services

6.4 Session Layer Design Issues

6.4.1 Remote Procedure Call (RPC)

6.5 Presentation Layer

6.5.1 Functions: Presentation Layer

6.5.2 Design Issues: Presentation Layer

6.6 Data Compression

6.6.1 Data Compression Methods

6.7 Run Length Encoding

6.8 Cryptography

6.8.1 Public Key Cryptography

6.8.2 Cryptography Features

6.8.3 Types of Cryptography

6.8.4 Cryptology Terminologies

6.9 Review Questions

6.10 Summary

6.11 References and Further Readings

6.0 OBJECTIVES

At the end of this unit, you will be able to:

- Understand the significance of session and presentation layer in data transmission.
- Understand the functions & services of session and presentation layer.
- Discuss in detail the various features of session layer and presentation
- .Explore different encoding, compression and encryption techniques.

6.1 INTRODUCTION

Session Layer is the 5th layer of the Open Systems Interconnection (OSI) reference model, which enables sessions between computers on a network to be established and terminated.. The session layer allows users on different machines to establish sessions between them. A session allows ordinary data transport, as does the transport layer, but it also provides enhanced services useful in some applications. A session might be used to allow a user to log into a remote timesharing system or to transfer a file between two machines.

The presentation layer is located at the sixth level of the OSI model; it is responsible for the delivery and formatting of information to the application layer for further processing or display. This type of service is needed because different computer architectures use different data representations. In this contrast, the presentation layer handles all issues related to data presentation and transport, including translation, encryption, and compression.

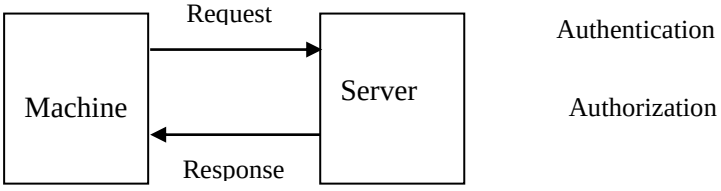
6.2 CONCEPTS

One of the services of the session layer is to manage dialogue control. Sessions can allow traffic to go in both directions (full duplex mode) at the same time, or in only one direction (half duplex mode) at a time. If traffic can only go one way at a time, the session layer can help keep track of whose turn it is through token management. For some protocols, it is essential that both sides do not attempt the same operation at the same time. To manage these activities, the session layer provides tokens that can be exchanged. Only the side holding the token may perform the data transmission.

Another session service is synchronization. Consider the problems that might occur when trying to do a 2-hour file transfer between two machines with a 1-hour mean time between crashes. After each transfer was aborted, the whole transfer would have to start over again and would probably fail again the next time as well. To eliminate this problem, the session layer provides a way to insert checkpoints into the data stream, so that after a crash, only the data transferred after the last checkpoint have to be repeated.

Hence responsibilities of session layer are:

1. Creating a session through **authentication**e.g login into the remote system by using username and password.
2. The remote service responds through **authorization**.



3. The Session restoration through tokens and the same depends on type of server used. Whether server authorizes restoration on basis of token. The

restoration of session is not allowed in banking transactions. But same is allowed by email servers like gmail.

4. The session layer also helps in flow control and synchronization.
5. The creation of session depends on application interface instead of operating system used.

6.3 SESSION LAYER FUNCTIONS & SERVICES

The session layer provides the following services/functions:

1. Dialog Management: The session layer allows two systems to enter into a dialog. It allows the communication between two processes to take place in either half- duplex (one way at a time) or full-duplex (two ways at a time) mode. Or can be said whose turn it is to talk.

In the ISO protocols, dialog management is implemented through the use of a data token. The token is sent back and forth, and a user may transmit only when it possesses the token.

2. Synchronization: The session layer allows a process to add checkpoints, or synchronization points, to a stream of data. For example, if a system is sending a file of 5000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. In this case, if a crash happens during the transmission of page 568, the only pages that need to be resent after system recovery are pages 501 to 568. Pages previous to 501 need not be retransmission.

The transport layer handles only communication errors, synchronization deals with upper layer errors. In a file transfer, for instance, the transport layer might deliver data correctly, but the application layer might be unable to write the file because the file system is full.

Users can split the data stream into pages, inserting synchronization points between each page. When an error occurs, the receiver can resynchronize the state of the session to a previous synchronization point. This requires that the sender hold data as long as may be needed.

Synchronization is achieved through the use of sequence numbers. The ISO protocols provide both major and minor synchronization points. When re-synchronizing, one can only go-back as far as the previous major synchronization point. In addition, major synchronization points are acknowledged through explicit messages. In contrast, minor synchronization points are just markers.

3. Activity Management: The session layer allows the user to delimit data in to logical units called activities. Each activity is independent of activities that come before and after it, and an activity can be processed on its own. Activities might be used to delimit files of a multi-file transfer.

Activities are also used for quarantining, collecting all the messages of a multi-message exchange together before processing them. The receiving application

would begin processing messages only after all the messages had arrived. This provides a way of helping insure that all or none of a set of operations is performed.

For example, a bank transaction may consist of locking a record, updating a value, and then unlocking the record. If an application processed the first operation, but never received the remaining operations (due to client or network failures),the record would remain locked forever.

6.4 SESSION LAYER DESIGN ISSUES

1. Establish sessions between machines

The establishment of session between machines is an important service provided by session layer. This session is responsible for creating a dialog between connected machines. The Session Layer provides mechanism for opening, closing and managing a session between end-user application processes, i.e. a semi-permanent dialogue. This session consists of requests and responses that occur between applications.

2. Enhanced Services

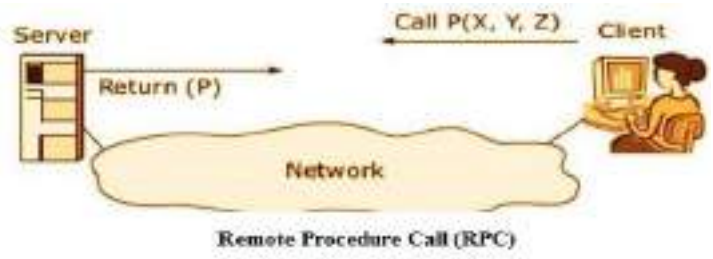
Certain services such as checkpoints and management of tokens are the key features of session layer and thus it becomes necessary to keep enhancing these features during the layer's design.

3. To help in Token management and Synchronization

The session layer plays an important role in preventing collision of several critical operation as well as ensuring better data transfer over network by establishing synchronization points at specific intervals. Thus it becomes highly important to ensure proper execution of these services.

Remote Procedure Call (RPC)

RPC is a protocol that one program use to request a service from a program located in another computer on a network without having to understand the network details. A procedure call is also called as function call or sub-routine. RPC uses client server model as shown below in figure.



Remote Procedure Call (RPC)

It mainly includes five elements:

1. The client.
2. The client stub (stub: piece of code used for converting parameters)
3. The RPC runtime (RPC communication package).

4. The Server stub.
5. The Server.

The Client:
It is a user process which initiates RPC. The client makes a perfectly normal call that invokes a corresponding procedure in the client stub:

The Client Stub:
On receipt of a request it packs a requirement into a message and asks to RPC Runtime to send. On receipt of a result it unpacks the result and passes it to client.

RPC Runtime:
It handles transmission of messages between client and server.

The Server Stub:
It unpacks a call request and make a perfectly normal call to be invoke the appropriate in the server on receipt of result the procedure executes it and packs the result and ask the RPC runtime to send

The Server:
It executes an appropriate procedure and returns the result from a stub.

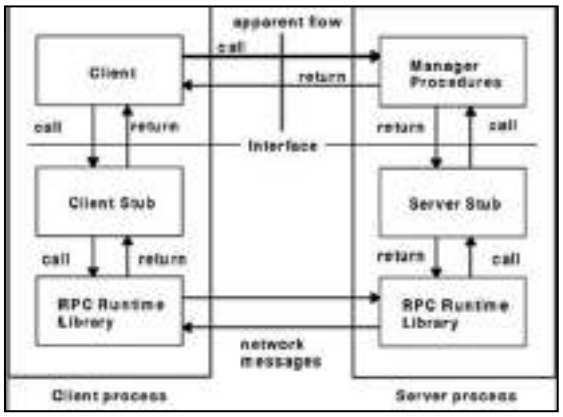


Figure: Remote Procedure call Flow

In a certain sense, sending a message to a remote host and getting a reply back is a lot like making a function call in a programming language. In both cases, you start with one or more parameters and you get back a result. This observation has led people to try to arrange request-reply interactions on networks to be cast in the form of procedure calls. Such an arrangement makes network applications much easier to program and more familiar to deal with. For example, just imagine a procedure named *get IP address* (*host name)* that works by sending a UDP packet to a DNS server and waiting for the reply, timing out and trying again if one is not forthcoming quickly enough. In this way, all the details of networking can be hidden from the programmer. The major work in this area was done by Birrell and Nelson

(1984), When a process on machine 1 calls a procedure on machine 2, the calling process on 1 is suspended and execution of the called procedure takes place on machine 2. Information can be transported from the caller to the callee in the parameters and can come back in the procedure result. No message passing is visible to the application programmer. This technique is known as **RPC** (**Remote Procedure Call**) and has become the basis for many networking applications. Traditionally, the calling procedure is known as the client and the called procedure is known as the server, and we will use those names here too.

The idea behind RPC is to make a remote procedure call look as much as possible like a local one. In the simplest form, to call a remote procedure, the client program must be bound with a small library procedure, called the **client stub** that represents the server procedure in the client's address space. Similarly, the server is bound with a procedure called the **server stub**. These procedures hide the fact that the procedure call from the client to the server is not local. The actual steps in making an RPC are shown in Figure.

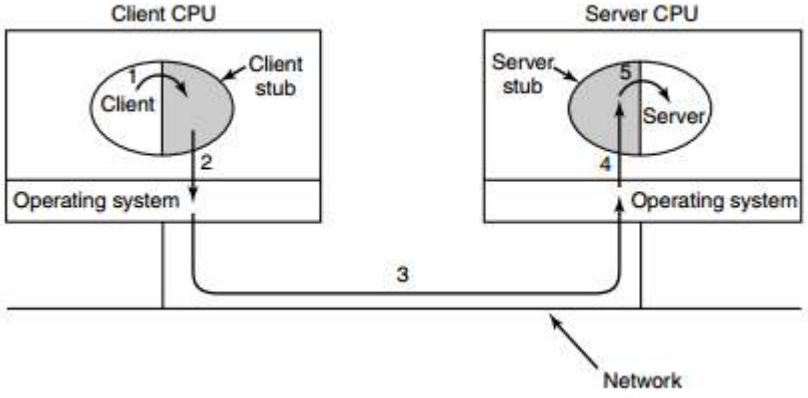


Figure: Steps in making a remote procedure call.

Step 1 is the client calling the client stub. This call is a local procedure call, with the parameters pushed onto the stack in the normal way.

Step 2 is the client stub packing the parameters into a message and making a system call to send the message. Packing the parameters is called **marshaling**.

Step 3 is the operating system sending the message from the client machine to the server machine.

Step 4 is the operating system passing the incoming packet to the server stub.

Step 5 is the server stub calling the server procedure with the un-marshaled parameters. The reply traces the same path in the other direction.

The key item to note here is that the client procedure, written by the user, just makes a normal (i.e., local) procedure call to the client stub, which has the same name as the server

procedure. Since the client procedure and client stub are in the same address space, the parameters are passed in the usual way. Similarly, the server procedure is called by a procedure in its address space with the parameters it expects. To the server procedure, nothing is unusual. In this way, instead of I/O being done on sockets, network communication is done by faking a normal procedure call. Despite the conceptual elegance of RPC, there are a few snakes hiding under the grass. A big one is the use of pointer parameters. Normally, passing a pointer to a procedure is not a problem. The called procedure can use the pointer in the same way the caller can because both procedures live in the same virtual address

Remote Procedure Call (RPC) provides a different paradigm for accessing network services. Instead of accessing remote services by sending and receiving messages, a client invokes services by making a local procedure call. The local procedure hides the details of the network communication.

When making a remote procedure call

The calling environment is suspended, procedure parameters are transferred across the network to the environment where the procedure is to execute, and the procedure is executed there.

When the procedure finishes and produces its results, its results are transferred back to the calling environment, where execution resumes as if returning from a regular procedure call.

The main goal of RPC is to hide the existence of the network from a program. As a result, RPC doesn't quite fit into the OSI model:

The message-passing nature of network communication is hidden from the user. The user doesn't first open a connection, read and write data, and then close the connection. Indeed, a client often doesn't even know they are using the network.

RPC often omits many of the protocol layers to improve performance. Even a small performance improvement is important because a program may invoke RPCs often. For example, on (diskless) Sun workstations, every file access is made via an RPC.

Advantages of RPC

1. RPC supports process oriented and thread oriented models
2. The internal message passing mechanism of RPC is hidden from User.
3. The Effort to re-write and re-develop the code is minimum in RPC.
4. RPC can be used in distributed environment as well as local environment.
5. Many of the protocol layers are omitted by RPC to improve the performance.
6. Supports multiple servers having same interface type.
7. Client requests can be spread evenly to balance the load.

Disadvantages of RPC

1. The overhead involved in binding clients to servers is large and becomes significant when many client processes are short lived.
2. A binding agent must be robust against failures and should not become performance bottleneck.

3. The RPC is a concept that can be implemented in different ways. It is not a standard.
4. There is no flexibility in RPC for hardware architecture, it is only interaction based.
5. There is increase in cost because of RPC.

RPC is especially well suited for client-server (e.g., query-response) interaction in which the flow of control alternates between the caller and callee. Conceptually, the client and server do not both execute at the same time. Instead, the thread of execution jumps from the caller to the callee and then back again. The following steps take place during an RPC:

A client invokes a client stub procedure, passing parameters in the usual way. The client stub resides within the client's own address space. The client stub marshals the parameters into a message. Marshalling includes converting the representation of the parameters into a standard format, and copying each parameter into the message.

The client stub passes the message to the transport layer, which sends it to the remote server machine. On the server, the transport layer passes the message to a server stub, which de-marshals the parameters and calls the desired server routine using the regular procedure call mechanism. When the server procedure completes, it returns to the server stub (e.g., via a normal procedure call return), which marshals the return values into a message. The server stub then hands the message to the transport layer.

The transport layer sends the result message back to the client transport layer, which hands the message back to the client stub. The client stub de-marshals the return parameters and execution returns to the caller.

6.5 PRESENTATION LAYER

The presentation layer is 6th layer of OSI reference model. The presentation layer mainly translates data between the application layer and the network format. Data can be communicated in different formats via different sources. Thus, the presentation layer is responsible for integrating all formats into a standard format for efficient and effective communication.

The presentation layer follows data programming structure schemes developed for different languages and provides the real-time syntax required for communication between two objects such as layers, systems or networks. The data format should be acceptable by the next layers; otherwise, the presentation layer may not perform correctly.

6.5.1 Design issues: Presentation Layer

I. Standard way of encoding data

The presentation layer follows a standard way to encode data when it needs to be transmitted. This encoded data is represented as character strings, integers, floating point numbers, and data structures composed of simple components. It is handled differently by different machines based on the encoding methods followed by them.

II. Maintaining the Syntax and Semantics of distributed information

The presentation layer manages and maintains the syntax as well as logic and meaning of the information that is distributed.

III. Standard Encoding on the wire

The data structures that are defined to be exchanged need to be abstract along with the standard encoding to be used "on the wire".

6.5.2 Functions: Presentation layer

The functions of the presentation layer include the following:

1. Syntax conversion - The abstract syntax is converted to the transfer syntax, and the other side to achieve the opposite conversion. Involved in the contents of the code conversion, character conversion, data format modification, as well as data structure operation adaptation, data compression, encryption and so on.

2. Semantic negotiation - According to the requirements of the application layer to negotiate the appropriate choice of context, that is, to determine the transmission syntax and transmission.

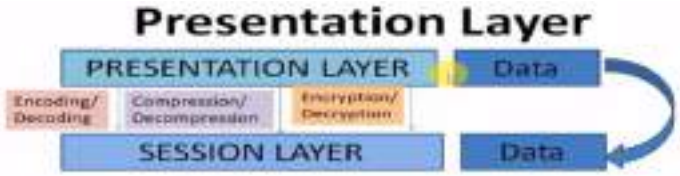
3. Network security and confidentiality management, text compression and packaging, virtual terminal protocol (VTP).

4. Connection management - Including the use of the session layer service to establish a connection, manage data transport and synchronization control over this connection (using the corresponding services at the session level), and terminate the connection either normally or absently.

In reference to above the presentation layer used to perform 3 actions on the data i.e. transmitted from application layer to presentation layer

1. Encoding/decoding
2. Compression/Decompression
3. Encryption/decryption

In short, presentation layer decides that how the data will be transmitted to session while sending or to application while receiving.



6.5.2 Encoding/Decoding

Computers use encoding schemes to store and retrieve meaningful information as data. This is known as data encoding.

Data Encoding /Decoding is a process performed to give a data a standard format like what type of file it is, e.g we have to send a pdf over Bluetooth to another device then unless encoding is not done properly the pdf reader will not open the file at the receiver end. Hence, this is needed run the program.Another e.g. if we have to send a picture of jpeg format then the presentation layer will encode it so that later at the receiver's end presentation layer will decode it and the receiver is receiving the jpeg format file.

On electronic devices like computers, data encoding involves certain coding schemes that are simply a series of electrical patterns representing each piece of information to be stored and retrieved. For instance, a series of electrical patterns represents the letter "A." Data encoding and decoding occur through electronic signals, or the electric or electromagnetic encoding of data In data encoding, all data is serialized, or converted into a string of ones and zeros, which is transmitted over a communication medium like a phone line. Serialization must be done in such a way that the computer receiving the data can convert the data back into its original format," according to Microsoft. "How serialization is accomplished is called a *communication protocol*, and is controlled by both software and data-transmission hardware. There are several levels at which the data is converted."

6.5.3 Compression/Decompression

Data compression reduces the size of a file by minimizing redundant data. In a text file, redundant data can be frequently occurring characters, such as the space character, or common vowels, such as the letters e and a; it can also be frequently occurring character strings. Data compression creates a compressed version of a file by minimizing this redundant data.

Each type of data-compression algorithm minimizes redundant data in a unique manner. For example, the ***Huffman encoding algorithm*** assigns a code to characters in a file based on how frequently those characters occur. Another algorithm, called **RPL** (*run-length encoding)*, generates a two-part value for repeated characters: the first part specifies the number of times the character is repeated, and the second part identifies the character.

6.5.4 Encryption/Decryption

This process is used for data protection while sending and receiving the data, used in high security transfers. Like passwords are used in files to open is also an encryption technique but in network encryption coding a data with specific code so that if someone goes through the data i.e Hack no one may not be able to access it **because of encryption.**

The receiver will receive exactly the same data that was transmitted by sender **because of decryption.**

The presentation layer is concerned with preserving the meaning of information sent across a network. The presentation layer may represent (encode) the data in various ways (e.g.,

data compression, or encryption), but the receiving peer will convert the encoding back into its original meaning. The syntax and the semantics of the information exchanged between two communication systems are managed by the presentation layer of the OSI Model.

6.6 DATA COMPRESSION

Data compression is a reduction in the number of bits needed to represent data. Compressing data can save storage capacity, speed file transfer, and decrease costs for storage hardware and network bandwidth. Data compression refers to the reducing the number of bits that need to be transmitted over communication channel.

Data compression reduces the number of bits sent. Data compression becomes particularly important when we send data with high size such as audio & video. Even with very fast transmission speed of data we need to send data in short time. Virtually all form of data contains redundancy i.e. it is the amount of wasted "space" used to transmit certain data. By making use of more efficient data representation methods, redundancy can be reduced. Even 7 bit ASCII code has some redundancy in it. The goal of data compression is to represent an information source (e.g. a data file, a speech signal, an image, or a video signal) as accurately as possible using the fewest number of bits.

Compression Ratio

Data compression ratio, also know as compression power, is a term used to quantify the reduction in data-representation size produced by a data compression algorithm. The data compression ratio is analogous to the physical compression ratio used to measure physical compression of substances.

Data compression ratio is defined as the ratio between the uncompressed size and compressed size.

$$CompressionRatio = \frac{UncompressedSize}{CompressedSize}$$

Thus a representation that compresses a 20 MB file to 2 MB has a compression ratio of 20/2 = 10, often noted as an exploit ratio, 10:1 (read "ten" to "one") or as an implicit ratio 10/1.

Note that this formulation applies equally for compression, where the uncompressed size is that of the original and for decompression, where the uncompressed size is that of the reproduction.

Sometimes the space saving is given instead, which is defined as the reduction in size relative to the uncompressed size.

$$SpaceSaving = 1 - \frac{CompressedSize}{UncompressedSize}$$

Thus a representation that compresses a 20 MB file to 2 MB would yield a space saving of 1-2/20 = 0.9 often notated as a percentage, 90%.

For signals of indefinite size, such as streaming audio and video the compression ratio is defined in terms of uncompressed and compressed data rates instead of data sizes.

$$CompressionRatio = \frac{Uncompressed\ DataRate}{CompressedDataRate}$$

Also, Data-rate saving, this is computed in reference to the compressed data rate and un-compressed data rate.

$$DataRateSaving = 1 - \frac{CompressedDataRate}{UncompressedDateRate}$$

6.6.1 Data Compression Methods

The data compression methods are of two types:

A. Lossy Compression: removes non-useful part of data that is undetectable. It decreases the size of file to a greater extent and in process quality of data is degraded.

B.Lossless Compression: reconstructs the exact data. It decreases size but at low extent and doesn't degrades the quality of data.

Lossy Compression

Lossy compression refers to data compression techniques in which some amount of data is lost. Lossy compression technologies attempt to eliminate redundant or unnecessary information. If the decompressed information need not be an exact replica of the original information but something very close, we can use a lossy data compression method.

Lossy compression reduces a file by permanently eliminating certain information, especially redundant information. When the file is uncompressed, only a part of the original information is still there(although the use may not notice it). Lossy compression is generally used for video and sound, where a certain amount of information loss will not be detected by most users.

Different Lossy Techniques: Methods Based on Lossy compression.

JPEG: Used for pictures and Graphics

MPEG: used for Video Compression.

Audio Compression

Comparatively, these methods are less time taking, cheaper as well as it can reduce more space.

The JPEG image file, commonly used for photographs and other complex still images on the web, is an image that has lossy compression. Using JPEG compression, the creator can decide how much loss to introduced and make a trade-off between file size and image quality. Most video compression technologies, such as MPEG, use Lossy technique.

The best example is a videoconference where there is an acceptable amount of frame loss in order to deliver the image in real time. People may appear jerky in their movements, but you still have a grasp for what is happening on the other end of the conference. In the case of

graphics files, some resolution may be lost in order to create a smaller file. The loss may be in the form of color depth or graphic detail. For example, high-resolution details can be lost if a picture is going to be displayed on a low-resolution device. Loss is also acceptable in voice and audio compression, depending on the desired quality

These methods are called lossy compression methods because we will lose some of the original data in the process. Several methods have been developed using lossy compression techniques. Joint Photographic experts group(JPEG) is used to compress graphics and pictures. Motion Picture Expert Group (MPEG) is used to compress video.

Lossless Compresssion

The lossless compresssion refers to data compression techniques in which no data is lost. For mos types of data, lossless compression techniques can reduce the space needed by only about 50%.

For greater compression, one must use a lossy compression technique. Note, however, that only certain types o fdata-graphics, audio, and video can tolerate lossy compression.

We must use a lossless compression technique when compressing data and programs. The PKZIP compression technology is an example of losslesss compression. With lossless compression, every single bit of data was originally in the file remains after the file is uncompressed. All the information is completely restored.

This is generally the technique of choice for text or spreadsheeet files, where losing words or financial data could pose a problem. The Graphics Interchagge File(GIF) is an image format used on the Web that provides lossless compression. In lossless Data Compression the compressing & decompressing algorithm are usually the inverse of each other. In other words after decompressing we will get exact data as they were before compressing, Nothing is lost. The following are some techniques used in lossless data compression.

1. Null compression
2. Run Length Encoding
3. Statistical Compression
    - ShannnonFano Encoding
    - Huffman Encoding
        - Adaptive compression(Adaptive Huffman)
    - Dictionary Based Compression
        - LZ&&, LZ8 Algorithms
        - Lempel ZIv Welch(LZW) encoding

Null Compression

It replaces a series of blank with a compression code, followed by a value that represents the number of spaces.

Example: hello friend how is your life?

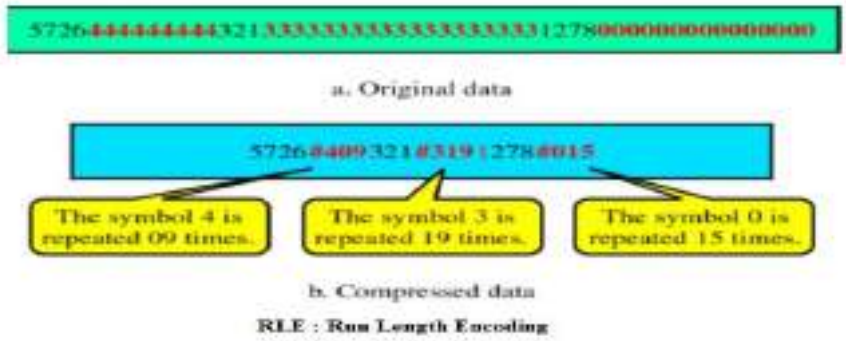
When data contain strings of repeated symbols (such as bits or characters), the strings can be replaced by special marker, followed by the repeated symbol, followed by the number of occurrences. For example, in given Figure, the symbol # is the marker. The symbol being repeated (the run symbol) follows the marker.

After the run symbol, the number of occurrences (length) is shown by a two digit number.
Example: This run-length encoding mentioned can be used in audio (silence is a run of 0s) and video (run of picture element having the same brightness and color).

6.7 RUN LENGTH ENCODING

Simple Compression technique Replace all consecutive numbers or alphabets by first the number of times an alphabet was used followed by the alphabet itself. This method becomes more effective with numbers especially when it is about only bits 1 and 0

When data contain strings of repeated symbols (such as bits or characters), the strings can be replaced by a special marker, followed by the repeated symbol, followed by the number of occurrences. For example, in given figure, the symbol # is the marker. The symbol being repeated (the run symbol) follows the marker. After the run symbol, the number of occurrences (length) is shown by a two-digit number.



RLE : Run Length Encoding

Statistical Compression

Another method of lossless compression is statistical compression. This method uses short codes for frequent symbols and long codes for infrequent symbols. In this way, the length of the toal data is reduced tremendously. The following methods are popular lossess statistical compression methods:

a. Shannon Fano Algorithm

Shannon Fano Algorithm is an entropy encoding technique for lossless data compression of multimedia. Named after Claude Shannon and Robert Fano, it assigns a code to each symbol based on their probabilities of occurrence. It is a variable length encoding scheme, that is, the codes assigned to the symbols will be of varying length.

HOW DOES IT WORK?

The steps of the algorithm are as follows:

1. Create a list of probabilities or frequency counts for the given set of symbols so that the relative frequency of occurrence of each symbol is known.
2. Sort the list of symbols in decreasing order of probability, the most probable ones to the left and least probable to the right.
3. Split the list into two parts, with the total probability of both the parts being as close to each other as possible.
4. Assign the value 0 to the left part and 1 to the right part.

5. Repeat the steps 3 and 4 for each part, until all the symbols are split into individual subgroups.

   The Shannon codes are considered accurate if the code of each symbol is unique.

b. Huffman Coding

Huffman coding is a lossless data compression algorithm. The idea is to assign variable-length codes to input characters; lengths of the assigned codes are based on the frequencies of corresponding characters. The most frequent character gets the smallest code and the least frequent character gets the largest code.

The variable-length codes assigned to input characters are prefix code means the codes (bit sequences) are assigned in such a way that the code assigned to one character is not the prefix of code assigned to any other character. This is how Huffman Coding makes sure that there is no ambiguity when decoding the generated bit-stream.

Let us understand prefix codes with a counter example. Let there be four characters a, b, c and d, and their corresponding variable length codes be 00, 01, 0 and 1. This coding leads to ambiguity because code assigned to c is the prefix of codes assigned to a and b. If the compressed bit stream is 0001, the de-compressed output may be "cccd" or "ccb" or "acd" or "ab".

There are mainly two major parts in Huffman Coding

1. Build a Huffman Tree from input characters.
2. Traverse the Huffman Tree and assign codes to characters.

Steps to build Huffman Tree

Input is an array of unique characters along with their frequency of occurrences and output is Huffman Tree.

1. Create a leaf node for each unique character and build a min heap of all leaf nodes (Min Heap is used as a priority queue. The value of frequency field is used to compare two nodes in min heap. Initially, the least frequent character is at root)
2. Extract two nodes with the minimum frequency from the min heap.
3. Create a new internal node with a frequency equal to the sum of the two nodes frequencies. Make the first extracted node as its left child and the other extracted node as its right child. Add this node to the min heap.
4. Repeat steps#2 and #3 until the heap contains only one node. The remaining node is the root node and the tree is complete.

   Let us understand the algorithm with an example:

for example we have a sentence like :- " Data Communication Application"

We have to assignnumbers to each alphabet and count occurrences:-

D	A	T	C	O	M	U	N	I	P	L
1	5	3	3	3	2	1	3	4	2	1

c. Dictionary compression

The compression techniques we have seen so far replace individual symbols with variable length code-words. In dictionary compression, variable length substrings are replaced by short, possibly even fixed length code-words. Compression is achieved by replacing long strings with shorter code-words. The general scheme is as follows: • The dictionary D is a collection of strings, often called phrases. For completeness, the dictionary includes all single symbols.

• The text T is parsed into a sequence of phrases:

$$T = T1T2 \ldots Tz, Ti \in D.$$

The sequence is called a parsing or a factorization of T with respect to D.

• The text is encoded by replacing each phrase Ti with a code that acts as a pointer to the dictionary.

Here is a simple static dictionary encoding for English text:

• The dictionary consists of some set of English words plus individual symbols.

• Compute the frequencies of the words in some corpus of English texts. Compute the frequencies of symbols in the corpus from which the dictionary words have been removed.

• Number the words and symbols in descending order of their frequencies.

• To encode a text, replace each dictionary word and each symbol that does not belong to a word with its corresponding number. Encode the sequence of number using γ coding.

Lempel-Ziv compression In 1977 and 1978, Abraham Lempel and Jacob Ziv published two adaptive dictionary compression algorithms are known as LZ77 and LZ78, and most related methods can be categorized as a variant of one or the other. The primary difference is the encoding of the phrases:

• LZ77 uses direct pointers to the preceding text.

• LZ78 uses pointers to a separate dictionary.

6.8 CRYPTOGRAPHY

The word Cryptography has been taken from greek words “kryptos” which means hidden and graphein which means to write. It simply means the design and analysis of codes and ciphers.

Cryptography is the process of transforming plain text or original information into an unintelligible form (cipher text) so that it may be sent over unsafe channels of communication.

6.8.1 Public Key Cryptography

Public Key Cryptography Both parties have a private key and a public key. The private keys are known only to their owners, but the public keys are available to anyone (like telephone numbers). The sending party encrypts the message with the receivers public key and the receiver decrypts with his own private key. This is possible due to the discovery by Diffie and Hellman (at Stanford University, autumn 1975) that algorithms can be developed which use one key for encryption and a different key for decryption. The public and private key constitute a key pair. The following public key crypto-systems are well known:

• The RSA (named after it's inventors Rivest, Shamir and Adleman) algorithm was

developed at MIT in 1977 and is the most common public key system is use today. A key minimum key length of 768 bits is recommended by RSA Inc. RSAREF is a library from RSA Inc. which is integrated into many commercial products and public domain products (such as the US version of PGP). International, public domain RSA compatible libraries (with large key sizes) do exist and are used in products such as SSH (SSH uses 1024 bit keys by default). RSA key generation is slower than verification. RSA is patented in the U.S. until 20.9.2000.

• The Diffie-Hellman (named also after it's inventors) key exchange protocol, published in 1976, produces shared secret keys from publicly known information over unsecured networks. These shared keys can be used 0 produce session keys. It's strength is based on the "discrete logarithm" problem. Since parties are not authenticated, it is vulnerable to "man in the middle" attacks, which can be prevented by use additional protocols or digital. Sun make extensive use of this algorithm in Secure RPC and- SKIP. This algorithm has the added advantage that it's patent expired in 1997.

• The EIGamal Public Key system (invented by TaherEIGamal) consists of both an encryption and signature algorithm. It is similar to the Diffie-Hellman key exchange and it's strength is based on the "discrete logarithm" problem. Key length strengths are similar to RSA. It is quite slow and requires very good random number generation. DSA is based on the signature algorithm.

DSS is the Digital Signature Standard that uses the DSA (Digital Signature Algorithm) approved in May 1994 by the U.S. government (NIST & NSA) as the standard for digital authentication. DSA is based on crypto algorithms from EIGamal and Schnorr. Signature generation is faster than verification (which is unusual, there are likely to be more verifications than generations). DSA lacks a key exchange mechanism, is very new and has been criticized because the NSA were heavily involved in it's selection and it was not subject to open peer review.

6.8.2 Cryptography Features

1. **Confidentiality:**
   Information can only be accessed by the person for whom it is intended and no other person except him can access it.
2. **Integrity:**
   Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.
3. **Non-repudiation:**
   The creator/sender of information cannot deny his or her intention to send information at later stage.
4. **Authentication:**
   The identities of sender and receiver are confirmed. As well as destination/origin of information is confirmed.

6.8.3 Types of Cryptography

In general there are three types of cryptography:

Symmetric Key Cryptography, Hash Functions and Asymmetric Key Cryptography

1. **Symmetric Key Cryptography**

   It is an encryption system where the sender and receiver of message use a single common key to encrypt and decrypt messages. Symmetric Key Systems are faster and simpler but the problem is that sender and receiver have to somehow exchange key in a secure manner. The most popular symmetric key cryptography system is Data Encryption System(DES).

   A system where one secret key shared is called symmetric or secret key Cryptography. An encryption system in which the sender and receiver of a message share a single, common key that is used to encrypt and decrypt the message. Contrast this with public-key cryptology, which utilizes two keys - a public key to encrypt messages and a private key to decrypt them. Symmetric-key systems are simpler and faster, but their main drawback is that the two parties must somehow exchange the key in a secure way. Public-key encryption avoids this problem because the public key can be distributed in a non-secure way and the private key is never transmitted. Symmetric-key cryptography is sometimes called secret-key cryptography. The most popular symmetric-key system is the Data Encryption Standard (DES).

   To put this in the simplest terms possible, **symmetric encryption** is a type of encryption that uses the same key to encrypt and decrypt data. Both the sender and the recipient have identical copies of the key, which they keep secret and don't share with anyone. This differs from **asymmetric encryption**, which uses two keys — a public key (that anyone can access) to encrypt information and a private key to decrypt information.

   Just in case it's helpful, let's do a quick review of how encryption works in general:

   1. The sender uses an encryption key (usually a string of letters and numbers) to encrypt their message.
   2. The encrypted message, called cipher-text, looks like scrambled letters and can't be read by anyone along the way.
   3. The recipient uses a decryption key to transform the cipher-text back into readable text.



Only these two parties (sender and recipient) can read and access the data. This is why it's also sometimes called **secret key encryption**, **secret key cryptography**, **private key cryptography**, **symmetric cryptography** and **symmetric key encryption**.



Having only one key to serve both the encryption and decryption functions simplifies the encryption process. After all, you're applying one key to turn plaintext, readable information into unreadable gibberish (ciphertext) and vice versa. One of the advantages of using symmetric encryption is that it provides data privacy and confidentiality without the extra complexity of multiple keys.

Symmetric key encryption does work on its own, for certain use cases. For example, it's useful for encrypting databases and files, where you're not exchanging data publicly between parties. But as with any technical process, there are other advantages and disadvantages of using symmetric key encryption, such as key distribution and management issues, and we'll talk about those a little later.

2. **Hash Functions**

There is no usage of any key in this algorithm. A hash value with fixed length is calculated as per the plain text which makes it impossible for contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.

A "Hash function" is a complex encryption algorithm used primarily in cryptography and is like a shortened version of full-scale encryption. A hash function H is a transformation that takes a variable-size input m and returns a fixed-size string, which is called the hash value h (that is, h = H(m)). Hash functions with just this property have a variety of general computational uses, but when employed in cryptography the hash functions are usually chosen to have some additional properties.

The basic requirements for a cryptographic hash function are:

- the input can be of any length,
- the output has a fixed length,
- H(x) is relatively easy to compute for any given x ,
- H(x) is one-way,
- H(x) is collision-free.

A hash function H is said to be one-way if it is hard to invert, where "hard to invert" means that given a hash value h, it is computationally infeasible to find some input x such that H(x) = h.

If, given a message x, it is computationally infeasible to find a message y not equal to x such that H(x) == H(y) then H is said to be a weakly collision-free hash function.

A strongly collision-free hash function H is one for which it is computationally infeasible to find any two messages x and y such that H(x) = H(y).

The hash value represents concisely the longer message or document from which it was computed; one can think of a message digest as a digital fingerprint of the larger document. Examples of well-known hash functions are MD2 and MD5 and SHA.

Perhaps the main role of a cryptographic hash function is in the provision of digital signatures. Since hash functions are generally faster than digital signature algorithms, it is typical to compute the digital signature to some document by computing the signature on the document's hash value, which is small compared to the document itself. Additionally, a digest can be made public without revealing the contents of the document from which it is derived. This is important in digital time stamping where, using hash functions, one can get a document time stamped without revealing its contents to the time stamping service.

Cryptographic hash functions have many information security applications, notably in digital signatures, message authentication codes (MACs) and other forms of authentication. They can also be used as ordinary hash functions, to index data in hash tables, for fingerprinting, to detect duplicate data or uniquely identify files and as checksums to detect accidental data corruption. Indeed, in information .security contexts, cryptographic hash values are sometimes called (digital) fingerprints, checksums or just hash values, even though all these terms stand for functions with rather different properties and purposes.

3. **Asymmetric Key Cryptography**

Under this system a pair of keys is used to encrypt and decrypt information. A public key is used for encryption and a private key is used for decryption. Public key and Private Key are different. Even if the public key is known by everyone the intended receiver can only decode it because he alone knows the private key.

In asymmetric cryptography (also known as public key cryptography), a public key and private key are used as a two-key system which allows for free distribution of the encryption key while making the decryption key public. This gives the benefit of allowing-a person to give away the encryption key to anyone while still keeping the data enciphered and sent to them secure. Even if someone intercepts the message and has a copy of the public key, they can't access the data. This is the major benefit of using the two-key system rather than one-key. A graphical explanation of this is shown to the right. Asymmetric key algorithms can also be used in the reverse- proving someone's identity to a large group of people. Suppose a big political leader wants to make announcements via the internet, but wants to ensure against imposters. He would give away the public key to everyone, encrypt data with the private key and send out the cipher-text via the internet. Since he has the only private key, only messages that were actually from him would be readable with his corresponding public key.

Mathematically, asymmetric encryption is considerably more complicated than single-key ciphers. The basic idea is to make computations which are incredibly hard without the key, but simple when you have the key available. With RSA (and several other 2-key ciphers) this is done with a pair of very large prime numbers. With the public key and these two large primes, the calculation of the private key (and therefore the decryption of the ciphertext) is simple. However, without the two primes, the calculation is incredibly complicated, with

millions or billions of possibilities. This renders the public key useless in decryption, but accessible to encryption functionality on its own. This use of two key encryption is the basis for digital signatures and certificates as well. Digital signatures are essentially an assurance of authentication and nonrepudiation by tying a public key to a private key. Just like it's physical counterpart (theoretically), a digital signature is an undeniable proof of identity- it can't be forged and therefore cannot be denied at a later time by it's owner. Digital signatures would be done by using both sets (sender and receiver) of publici private keys. The sender would encrypt a message using the receivers public key so only the receiver can read it. He would then encrypt the ciphertext "message" with his private key, so anyone who decrypts the message knows it comes from him. The message' is sent and decrypted twice, once for authentication/nonrepudiation and again for confidentiality. Certificates are similar, but use a certificate authority (CA) which acts as a trusted third party or voucher, to verify identity. Certificates have more details, however, giving the user's name, the CA, expiration date (if applicable) and any approved operations. This is a major aspect in modem e-commerce systems, where companies such as Verisign act as a reputable 3rd party, issuing certificates to companies.

Types of Asymmetric Cryptology

RSA asymmetric encryption

RSA is the best known asymmetric (public key) algorithm, named after its inventors: Rivest, Shamir and Adleman. RSA uses public and private keys that are functions of a pair of large prime numbers. Its security is based on the difficulty of factoring large integers. The RSA algorithm can be used for both public key encryption and digital signatures. The keys used for encryption and decryption in RSA algorithm, are generated using random data. The key used for encryption is a public key and the key used for decryption is a private key. Public keys are stored anywhere publicly accessible. The sender of message encrypts the data using public key and the receiver decrypts it using his/her own private key. That way, no one else can intercept the data except receiver.

Digital Signature Algorithm -DSA

The Digital Signature Algorithm (DSA) is a United States Federal Government standard or FIPS for digital signatures. It was proposed by the National Institute of Standards and Technology (NIST) in August 1991 for use in their Digital Signature Standard (DSS).

PGP

PGP (Pretty Good Privacy) is a public-private key cryptography system which allows for users to more easily integrate the use of encryption in their daily. tasks, such as electronic mail protection and authentication and protecting files stored on a computer. PGP was originally designed by Phil Zimmerman. It uses IDEA, CAST or Triple DES for actual data encryption and RSA (with up to 2048-bit key) or DHIDSS (with I024-bit signature key and 4096-bit encryption key) for key management and digital signatures. The RSA or DH public key is used to encrypt the IDEA secret key as part of the message.

6.8.4 Cryptology terminologies

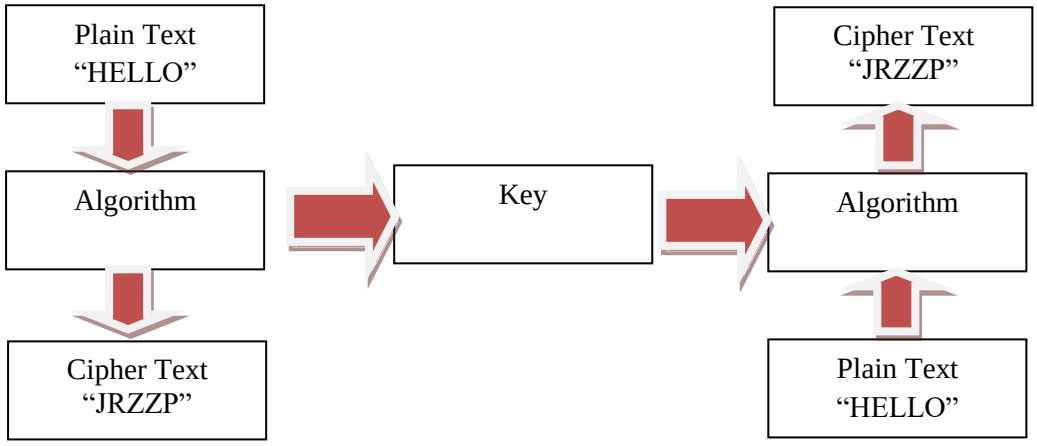
1) Crypto system - It is a system of a secure key pair consisting of a private key, for creating a digital signature and a public key to verify the digital signature.

2) Key pair - It means a private key and its mathematically related public key, which are so related that the public key can verify a digital signature created by the private key.

3) Private key - It means the key of a key pair used to create a digital signature.

4) Public key - It means the key of a key pair used to verify a digital signature and listed in the Digital Signature certificate.

5) Hash function - It means an algorithm mapping or translation of one sequence of bits into another, generally smaller, set known as "hash result" such that an electronic record yields the same hash result every time the algorithm is executed with the same electronic record as its input making it computationally infeasible (a) to derive or reconstruct the original electronic record from the hash result produced by the algorithm. (b) That two electronic records can produce the same hash result using the algorithm.
The second stage for cryptology is decryption (crypto analysis). Here, the cipher text back into the original form, when done by the authorized person.



Encryption **Decryption**

Modem cryptographic systems are implemented with computer programs that have two inputs- the plaintext message and key, both of which are represented as sequences of 0s and 1s.

6.9 REVIEW QUESTIONS

1. What are the challenges of RPC?

2. What is the significance of client stub and server stub?

3. Differentiate between Compression and Decompression.

4. How compression ratio for finite and indefinite data sizes is different?

5. Design a Huffman coding for "**JAGAT GURU NANAK DEV PUNJAB STATE OPEN UNIVERSITY".**

6. What is the difference between network security and cryptography?

6.10 SUMMARY

In this unit, we have given you a overview of various features and issues of session and presentation layers of OSI reference model. Our discussions are focused on various services like encoding, compression and encryption techniques with examples for better understanding.

REFERENCES FOR FURTHER READINGS

1. BehrouzForouzan, Data communications and Networking, Tata Mc-Graw Hill.
2. Andrew S. Tanenbaum, Computer Networks, Pearson Education.
3. William Stallings, Data and Computer Communications, Pearson education.

UNIT 7: APPLICATION LAYER

STRUCTURE

7.0 Introduction

7.1 Objectives

7.2 Concepts

7.3 Application Layer Functions

7.4 Application Layers Services

7.5 Application Layer Protocols

- **7.5.1 Domain Name System (DNS)**
- **7.5.2 HTTP**
- **7.5.3 Uniform Resource Locator (URL)**
- **7.5.4 E-mail**
- **7.5.5 WWW**
- **7.5.6 Telnet**
- **7.5.7 File Transfer Protocol (FTP)**
- **7.5.8 SMTP**

7. 6 Review Questions

7.7 Summary

7.8 References for Further Readings

7.0 OBJECTIVES

At the end of this unit, you will be able to:

- Understand the significance of application layer in data transmission.
- Understand the functions & services of application layer.
- Discuss in detail the various application layer protocols.

7.1 INTRODUCTION

Application layer is the top most layer in OSI and TCP/IP layered model. This layer exists in both layered Models because of its significance of interacting with user and user applications. This layer provides the better interface that helps to directly interacts along with software application and offers common web application services. This layer also fires a request to presentation layer, and it also delivers the network services to end-users.

7.2 CONCEPTS

A user may or may not directly interact with the applications. Application layer is where the actual communication is initiated and reflects. Because this layer is on the top of the OSI layer stack, it does not serve any other layers. Application layer takes the help of Transport and all layers below it to communicate or transfer its data to the remote host.



When an application layer protocol wants to communicate with its peer application layer protocol on remote host, it hands over the data or information to the transport layer. The transport layer does the rest with the help of all the layers below it.

Two remote application processes can communicate mainly in two different ways:

- **Peer-to-peer:** Both remote processes are executing at same level and they exchange data using some shared resource.
- **Client-Server:** One remote process acts as a Client and requests some resource from another application process acting as Server.

In client-server model, any process can act as Server or Client. It is not the type of machine, size of the machine, or its computing power which makes it server; it is the ability of serving request that makes a machine a server.

A system can act as Server and Client simultaneously. That is, one process is acting as Server and another is acting as a client. This may also happen that both client and server processes reside on the same machine.

7.3 APPLICATION LAYER FUNCTIONS

Application layer allows to users to implement the all services of network.

- This layer helps to design of network-based applications.
- It allows to users several services such as user login, transmission of data, e-mail, network devices configuration, and messages formatting.
- It is enabled with error handling and recovery of information.
- This layer offers various rights like as retrieve and handle files from remote terminal system.
- It provides the virtual terminal that take care abstract data structure, and it is controlled by keyboard and your computer system which helps to display your current state of data structure.
- It provides the authentication in between multiple network devices for delivering most security in the layer.
- Application layer also offers the mail services like as email forwarding and storage of e-mails.
- It also provides the directory services that help to access global information related to several objects and services.
- This layer monitors the sufficient availability of resources which are need in the network.

7.4 APPLICATION LAYERS SERVICES

- **Network Virtual terminal:** An application layer allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal, which in turn, talks to the host. The remote host thinks that it is communicating with one of its own terminals, so it allows the user to log on.
- **File Transfer, Access, and Management (FTAM):** An application allows a user to access files in a remote computer, to retrieve files from a computer and to manage files in a remote computer. FTAM defines a hierarchical virtual file in terms of file structure, file attributes and the kind of operations performed on the files and their attributes.
- **Addressing:** To obtain communication between client and server, there is a need for addressing. When a client made a request to the server, the request contains the server address and its own address. The server response to the client request, the request contains the destination address, i.e., client address. To achieve this kind of addressing, DNS is used.
- **Mail Services:** An application layer provides Email forwarding and storage.
- **Directory Services:** An application contains a distributed database that provides access for global information about various objects and services.

- **Authentication**: It authenticates the sender or receiver's message or both.

7.5 APPLICATION LAYER PROTOCOLS

An application layer protocol defines how application processes (clients and servers), running on different end systems, pass messages to each other. In particular, an application layer protocol defines:

- The types of messages, e.g., request messages and response messages.
- The syntax of the various message types, i.e., the fields in the message and how the fields are delineated.
- The semantics of the fields, i.e., the meaning of the information that the field is supposed to contain;
- Rules for determining when and how a process sends messages and responds to messages.

Application layer protocols can be broadly divided into two categories:

1) Protocols which are used by users. For example, email.
2) Protocols which help and support protocols used by users. For example DNS.

Some of Application layer protocols are DNS, HTTP, URL etc.

7.5.1 Domain Name System (DNS)

DNS (Domain Name System) is a decentralized naming system that converts domain names (such as logn.in) to its corresponding IP addresses. DNS is required for the functioning of the internet.

DNS is a service that translates the domain name into IP addresses. Each device connected to the internet has a unique IP address, which is a 32-bit number for example 132.32.0.1(in IPv4), and remembering such an IP address is almost impossible. The solution to this problem was using English letters, as we humans' beings are good at memorizing English words. Hence the concept of Domain Name came into the picture. Each domain name is a combination of English alphabets generally) and points to an IP address.

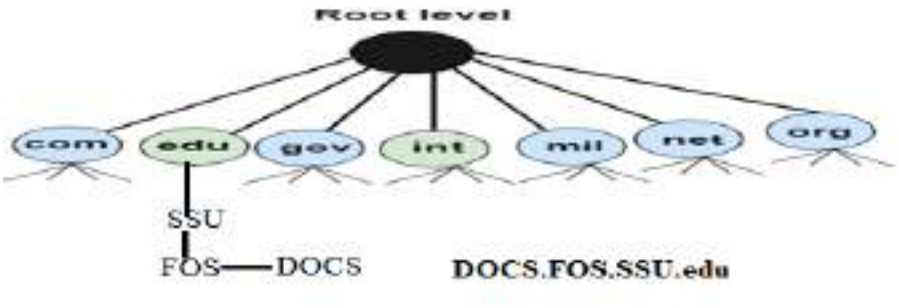
Label	Description
aero	Airlines and aerospace companies
biz	Businesses or firms
com	Commercial Organizations
coop	Cooperative business Organizations
edu	Educational institutions
gov	Government institutions

info	Information service providers
int	International Organizations
mil	Military groups
museum	Museum & other nonprofit organizations
name	Personal names
net	Network Support centers
org	Non-profit Organizations
pro	Professional individual Organizations

DNS is a TCP/IP protocol used on different platforms. The domain name space is divided into three different sections: generic domains, country domains, and inverse domain.

Generic Domains

It defines the registered hosts, according to their generic behaviour. Each node in a tree defines the domain name, which is an index to the DNS database. It uses three-character labels, and these labels describe the organization type.



Country Domain

The format of country domain is same as a generic domain, but it uses two-character country abbreviations (e.g., us for the United States) in place of three-character organizational abbreviations.

Inverse Domain

The inverse domain is used for mapping an address to a name. When the server has received a request from the client, and the server contains the files of only authorized clients. To determine whether the client is on the authorized list or not, it sends a query to the DNS server and ask for mapping an address to the name.

Working of DNS

- DNS is a client/server network communication protocol. DNS clients send requests to the. Server while DNS servers send responses to the client.

- Client requests contain a name which is converted into an IP address known as a forward DNS lookups while requests containing an IP address which is converted into a name known as reverse DNS lookups.
- DNS implements a distributed database to store the name of all the hosts available on the internet.
- If a client like a web browser sends a request containing a hostname, then a piece of software such as **DNS resolver** sends a request to the DNS server to obtain the IP address of a hostname. If DNS server does not contain the IP address associated with a hostname, then it forwards the request to another DNS server. If IP address has arrived at the resolver, which in turn completes the request over the internet protocol.

7.5.2 HTTP

HTTP stands for Hyper Text Transfer Protocol. It is a protocol used to access the data on the World Wide Web (www).It is the foundation of World Wide Web. The HTTP protocol can be used to transfer the data in the form of plain text, hypertext, audio, video, and so on.

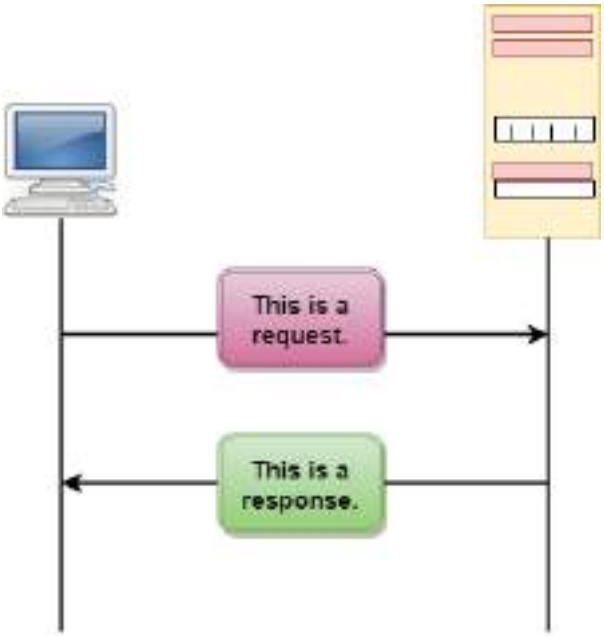
This protocol is known as HyperText Transfer Protocol because of its efficiency that allows us to use in a hypertext environment where there are rapid jumps from one document to another document. HTTP is similar to the FTP as it also transfers the files from one host to another host. But, HTTP is simpler than FTP as HTTP uses only one connection, i.e., no control connection to transfer the files. HTTP is used to carry the data in the form of MIME-like format. HTTP is similar to SMTP as the data is transferred between client and server. The HTTP differs from the SMTP in the way the messages are sent from the client to the server and from server to the client. SMTP messages are stored and forwarded while HTTP messages are delivered immediately.

Features of HTTP:

- **Connectionless protocol:** HTTP is a connectionless protocol. HTTP client initiates a request and waits for a response from the server. When the server receives the request, the server processes the request and sends back the response to the HTTP client after which the client disconnects the connection. The connection between client and server exist only during the current request and response time only.
- **Media independent:** HTTP protocol is a media independent as data can be sent as long as both the client and server know how to handle the data content. It is required for both the client and server to specify the content type in MIME-type header.
- **Stateless:** HTTP is a stateless protocol as both the client and server know each other only during the current request. Due to this nature of the protocol, both the client and server do not retain the information between various requests of the web pages.

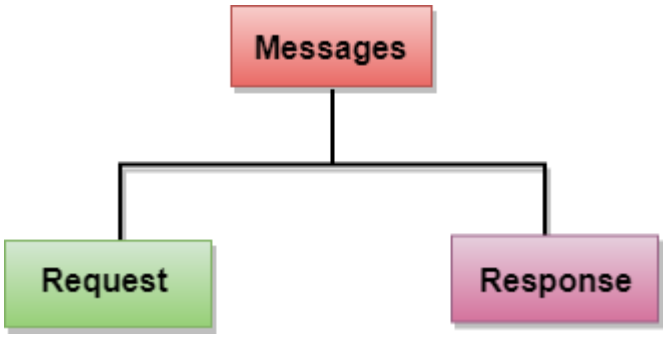
HTTP Transactions

The figure shows the HTTP transaction between client and server. The client initiates a transaction by sending a request message to the server. The server replies to the request message by sending a response message.

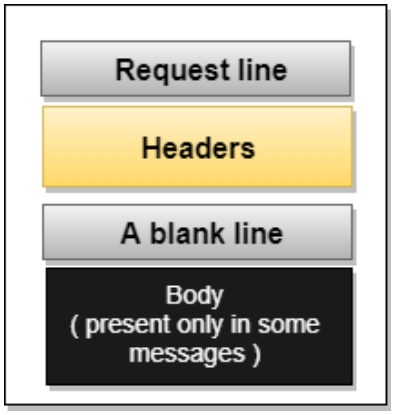


Messages

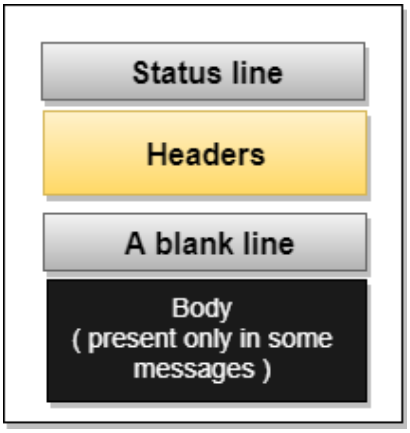
HTTP messages are of two types: request and response. Both the message types follow the same message format.



Request Message: The request message is sent by the client that consists of a request line, headers, and sometimes a body.



Response Message: The response message is sent by the server to the client that consists of a status line, headers, and sometimes a body.



7.5.3 Uniform Resource Locator (URL)

A client that wants to access the document in an internet needs an address and to facilitate the access of documents, the HTTP uses the concept of Uniform Resource Locator (URL). The Uniform Resource Locator (URL) is a standard way of specifying any kind of information on the internet.

The URL defines four parts: method, host computer, port, and path.



- **Method:** The method is the protocol used to retrieve the document from a server. For example, HTTP.

- **Host:** The host is the computer where the information is stored, and the computer is given an alias name. Web pages are mainly stored in the computers and the computers are given an alias name that begins with the characters "www". This field is not mandatory.
- **Port:** The URL can also contain the port number of the server, but it's an optional field. If the port number is included, then it must come between the host and path and it should be separated from the host by a colon.
- **Path:** Path is the pathname of the file where the information is stored. The paths itself contain slashes that separate the directories from the subdirectories and files.

HTTP versions

- HTTP 1.0 uses non persistent HTTP. At most one object can be sent over a single TCP connection.
- HTTP 1.1 uses persistent HTTP. In this version, multiple objects can be sent over a single TCP connection.

7.5.4 Email

Email is a service which allows us to send the message in electronic mode over the internet. It offers an efficient, inexpensive and real time mean of distributing information among people.

E-Mail Address

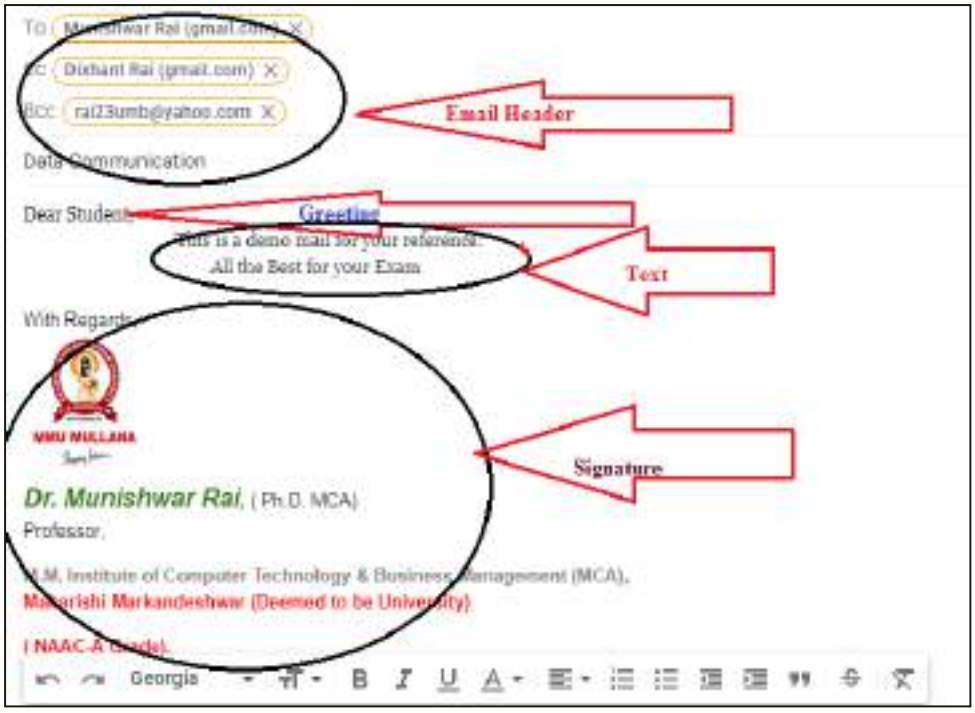
Each user of email is assigned a unique name for his email account. This name is known as E-mail address. Different users can send and receive messages according to the e-mail address.

E-mail is generally of the form username@domainname. For example, connect@psou.ac.in is an e-mail address where ***connect*** is username and ***psou.ac.in*** is domain name.

- The username and the domain name are separated by **@ (at)** symbol.
- E-mail addresses are not case sensitive.
- Spaces are not allowed in e-mail address.

E-mail Message Components

E-mail message comprises of different components: E-mail Header, Greeting, Text, and Signature. These components are described in the following diagram:



E-mail Header

The first five lines of an E-mail message is called E-mail header. The header part comprises of following fields:

From

The **From** field indicates the sender's address i.e. who sent the e-mail.

Date

The **Date** field indicates the date when the e-mail was sent.

To

The **To** field indicates the recipient's address i.e. to whom the e-mail is sent.

Subject

The **Subject** field indicates the purpose of e-mail. It should be precise and to the point.

CC

CC stands for Carbon copy. It includes those recipient addresses whom we want to keep informed but not exactly the intended recipient.

BCC

BCC stands for Blind Carbon Copy. It is used when we do not want one or more of the recipients to know that someone else was copied on the message.

Greeting

Greeting is the opening of the actual message. Eg. Hi Sir or Hi Guys etc.

Text

It represents the actual content of the message.

Signature

This is the final part of an e-mail message. It includes Name of Sender, Address, and Contact Number.

Advantages of Email

- ✓ ***Reliable***
  Many of the mail systems notify the sender if e-mail message was undeliverable.
- ✓ ***Convenience***
  There is no requirement of stationary and stamps. One does not have to go to post office. But all these things are not required for sending or receiving an mail.
- ✓ ***Speed***
  E-mail is very fast. However, the speed also depends upon the underlying network.
- ✓ ***Inexpensive***
  The cost of sending e-mail is very low.
- ✓ ***Printable***
  It is easy to obtain a hardcopy of an e-mail. Also an electronic copy of an e-mail can also be saved for records.
- ✓ ***Global***
  E-mail can be sent and received by a person sitting across the globe.
- ✓ ***Generality***

  It is also possible to send graphics, programs and sounds with an e-mail.

Disadvantages of Email

Apart from several benefits of E-mail, there also exist some disadvantages as discussed below:

- ➢ ***Forgery***
  E-mail doesn't prevent from forgery, that is, someone impersonating the sender, since sender is usually not authenticated in any way.
- ➢ ***Overload***
  Convenience of E-mail may result in a flood of mail.
- ➢ ***Misdirection***
  It is possible that you may send e-mail to an unintended recipient.
- ➢ ***Junk***

Junk emails are undesirable and inappropriate emails. Junk emails are sometimes referred to as spam.

- ***No Response***
  It may be frustrating when the recipient does not read the e-mail and respond on a regular basis.

7.5.5 World Wide Web (WWW)

World Wide Web, which is also known as a Web, is a collection of websites or web pages stored in web servers and connected to local computers through the internet. These websites contain text pages, digital images, audios, videos, etc. Users can access the content of these sites from any part of the world over the internet using their devices such as computers, laptops, cell phones, etc. The WWW, along with internet, enables the retrieval and display of text and media to your device.

History

Tim Berners-Lee, a British scientist, invented the World Wide Web (WWW) in 1989, while working at CERN. The Web was originally conceived and developed to meet the demand for automated information-sharing between scientists in universities and institutes around the world. CERN is not an isolated laboratory, but rather the focal point for an extensive community that includes more than 17 000 scientists from over 100 countries. Although they typically spend some time on the CERN site, the scientists usually work at universities and national laboratories in their home countries. Reliable communication tools are therefore essential.

The basic idea of the WWW was to merge the evolving technologies of computers, data networks and hypertext into a powerful and easy to use global information system.

Difference between World Wide Web and Internet:

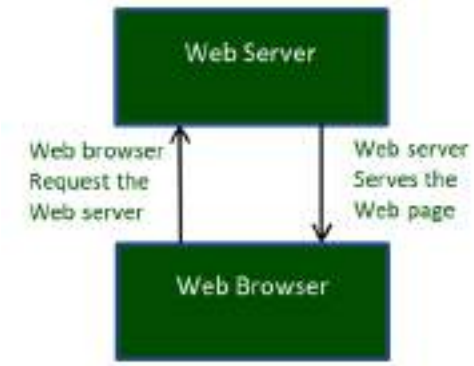
Some people use the terms 'internet' and 'World Wide Web' interchangeably. They think they are the same thing, but it is not so. Internet is entirely different from WWW. It is a worldwide network of devices like computers, laptops, tablets, etc. It enables users to send emails to other users and chat with them online. For example, when you send an email or chatting with someone online, you are using the internet.



But, when you have opened a website like google.com for information, you are using the World Wide Web; a network of servers over the internet. You request a webpage from your computer using a browser, and the server renders that page to your browser. Your computer is called a client who runs a program (web browser), and asks the other computer (server) for the information it needs.

World Wide Web Working

Now, we have understood that WWW is a collection of websites connected to the internet so that people can search and share information. Now, let us understand how it works ...The Web works as per the internet's basic client-server format as shown in the following image. The servers store and transfer web pages or information to user's computers on the network when requested by the users.



A web server is a software program which serves the web pages requested by web users using a browser. The computer of a user who requests documents from a server is known as a client. Browser, which is installed on the user' computer, allows users to view the retrieved documents.



All the websites are stored in web servers. Just as someone lives on rent in a house, a website occupies a space in a server and remains stored in it. The server hosts the website whenever a user requests its WebPages, and the website owner has to pay the hosting price for the same.

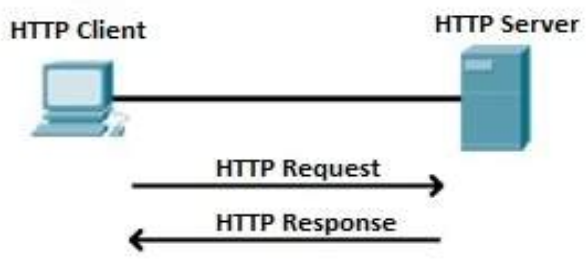
The moment you open the browser and type a URL in the address bar or search something on Google, the WWW starts working. There are three main technologies involved in transferring information (web pages) from servers to clients (computers of users). These technologies include Hypertext Markup Language (HTML), Hypertext Transfer Protocol (HTTP) and Web browsers.

Hypertext Markup Language (HTML):
HTML is a standard markup language which is used for creating web pages. It describes the structure of web pages through HTML elements or tags. These tags are used to organize the pieces of content such as 'heading,' 'paragraph,' 'table,' 'Image,' and more. You don't see HTML tags when you open a webpage as browsers don't display the tags and use them only to render the content of a web page. In simple words, HTML is used to display text, images, and other resources through a Web browser.

Web Browser:
A web browser, which is commonly known as a browser, is a program that displays text, data, pictures, videos, animation, and more. It provides a software interface that allows you to click hyperlinked resources on the World Wide Web. When you double click the Browser icon installed on your computer to launch it, you get connected to the World Wide Web and can search Google or type a URL into the address bar.

In the beginning, browsers were used only for browsing due to their limited potential. Today, they are more advanced; along with browsing you can use them for e-mailing, transferring multimedia files, using social media sites, and participating in online discussion groups and more. Some of the commonly used browsers include Google Chrome, Mozilla Firefox, Internet Explorer, Safari, and more.

Hypertext Transfer Protocol (HTTP):
Hyper Text Transfer Protocol (HTTP) is an application layer protocol which enables WWW to work smoothly and effectively. It is based on a client-server model. The client is a web browser which communicates with the web server which hosts the website. This protocol defines how messages are formatted and transmitted and what actions the Web Server and browser should take in response to different commands. As we enter a URL in the browser, an HTTP command is sent to the Web server, and transmits the requested Web Page.



When we open a website using a browser, a connection to the web server is opened, and the browser communicates with the server through HTTP and sends a request. HTTP is carried over TCP/IP to communicate with the server. The server processes the browser's request and sends a response, and then the connection is closed. Thus, the browser retrieves content from the server for the user.

7.5.6 Telnet

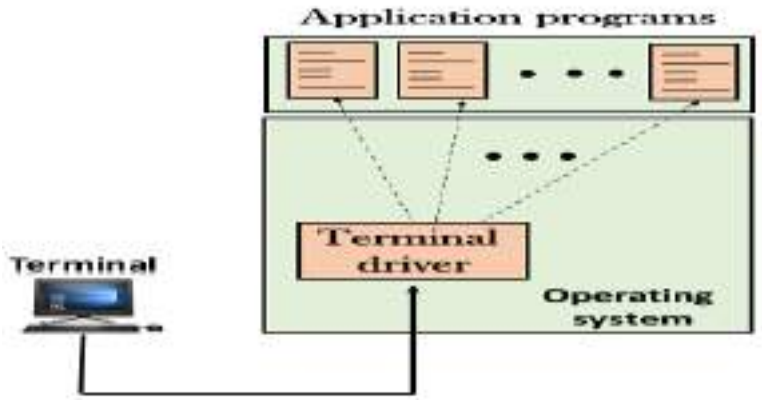
The main task of the internet is to provide services to users. For example, a user wants to run different application programs at the remote site and transfers a result to the local site. This requires a client-server program such as FTP, SMTP. But this would not allow us to create a specific program for each demand.

The better solution is to provide a general client-server program that lets the user access any application program on a remote computer. Therefore, a program that allows a user to log on to a remote computer. A popular client-server program Telnet is used to meet such demands. Telnet is an abbreviation for **Terminal Network**.

Telnet provides a connection to the remote computer in such a way that a local terminal appears to be at the remote side.

There are two types of login: Login Local and Remote login

Login Local

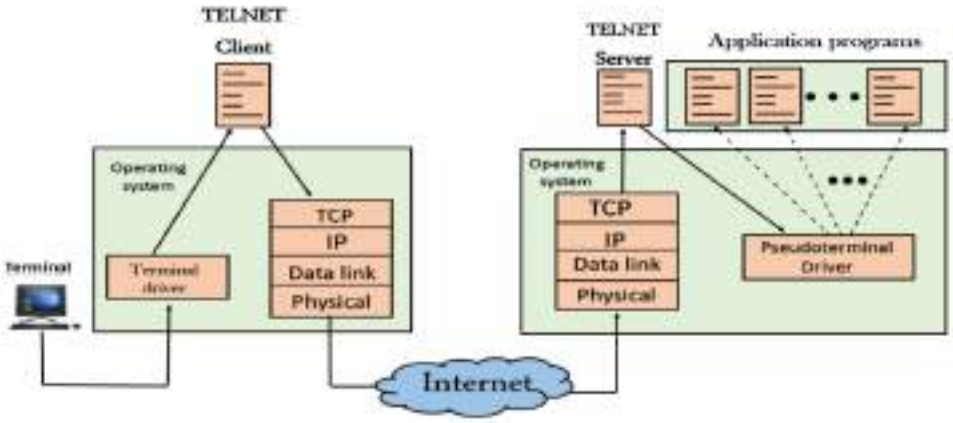


When a user logs into a local computer, then it is known as local login. When the workstation running terminal emulator, the keystrokes entered by the user are accepted by the terminal driver. The terminal driver then passes these characters to the operating system which in turn, invokes the desired application program.

However, the operating system has special meaning to special characters. For example, in UNIX some combination of characters has special meanings such as control character with "z" means suspend. Such situations do not create any problem as the terminal driver knows the meaning of such characters. But, it can cause the problems in remote login.

Remote login

When the user wants to access an application program on a remote computer, then the user must perform remote login.



How remote login occurs

At the local site

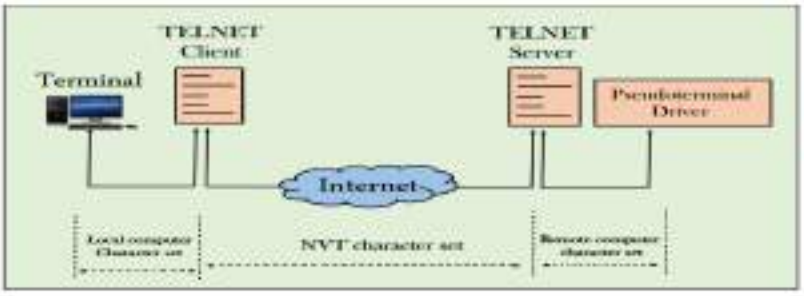
The user sends the keystrokes to the terminal driver, the characters are then sent to the TELNET client. The TELNET client which in turn, transforms the characters to a universal character set known as network virtual terminal characters and delivers them to the local TCP/IP stack

At the remote site

The commands in NVT forms are transmitted to the TCP/IP at the remote machine. Here, the characters are delivered to the operating system and then pass to the TELNET server. The TELNET server transforms the characters which can be understandable by a remote computer. However, the characters cannot be directly passed to the operating system as a remote operating system does not receive the characters from the TELNET server. Therefore it requires some piece of software that can accept the characters from the TELNET server. The operating system then passes these characters to the appropriate application program.

Network Virtual Terminal (NVT)

The network virtual terminal is an interface that defines how data and commands are sent across the network.



In today's world, systems are heterogeneous. For example, the operating system accepts a special combination of characters such as end-of-file token running a DOS operating system *ctrl+z* while the token running a UNIX operating system is *ctrl+d*.

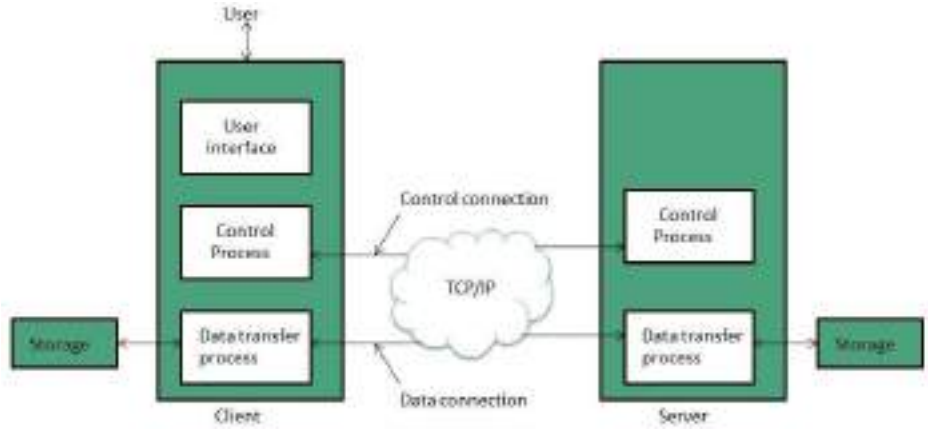
TELNET solves this issue by defining a universal interface known as network virtual interface.

The TELNET client translates the characters that come from the local terminal into NVT form and then delivers them to the network. The Telnet server then translates the data from NVT form into a form which can be understandable by a remote computer.

7.5.7 File Transfer Protocol (FTP)

FTP is used to copy files from one host to another. FTP offers the mechanism for the same in following manner:

- FTP creates two processes such as Control Process and Data Transfer Process at both ends i.e. at client as well as at server.
- FTP establishes two different connections: one is for data transfer and other is for control information.
    - **(a) Data Connection:** The Data Connection uses very complex rules as data types may vary. The data connection is made between data transfer processes. The data connection opens when a command comes for transferring the files and closes when the file is transferred.
    - **(b) Control Connection:** The control connection uses very simple rules for communication. Through control connection, we can transfer a line of command or line of response at a time. The control connection is made between the control processes. The control connection remains connected during the entire interactive FTP session.
- FTP uses port 21 for the control connection and Port 20 for the data connection.



Advantages of FTP

- **Speed:** One of the biggest advantages of FTP is speed. The FTP is one of the fastest ways to transfer the files from one computer to another computer.
- **Efficient:** It is more efficient as we do not need to complete all the operations to get the entire file.
- **Security:** To access the FTP server, we need to login with the username and password. Therefore, we can say that FTP is more secure.
- **Back & forth movement:** FTP allows us to transfer the files back and forth. Suppose you are a manager of the company, you send some information to all the employees, and they all send information back on the same server.

Disadvantages of FTP

- The standard requirement of the industry is that all the FTP transmissions should be encrypted. However, not all the FTP providers are equal and not all the providers offer encryption. So, we will have to look out for the FTP providers that provides encryption.
- FTP serves two operations, i.e., to send and receive large files on a network. However, the size limit of the file is 2GB that can be sent. It also doesn't allow you to run simultaneous transfers to multiple receivers.
- Passwords and file contents are sent in clear text that allows unwanted eavesdropping. So, it is quite possible that attackers can carry out the brute force attack by trying to guess the FTP password.
- It is not compatible with every system.

7.5.8 SMTP

SMTP stands for Simple Mail Transfer Protocol. SMTP is a set of communication guidelines that allow software to transmit an electronic mail over the internet is called **Simple Mail Transfer Protocol**. It is a program used for sending messages to other computer users based on e-mail addresses.

- It provides a mail exchange between users on the same or different computers, and it also supports:
    - send a single message to one or more recipients.
    - sending message can include text, voice, video or graphics.
    - It can also send the messages on networks outside the internet.

The main purpose of SMTP is used to set up communication rules between servers. The servers have a way of identifying themselves and announcing what kind of communication they are trying to perform. They also have a way of handling the errors such as incorrect email address. For example, if the recipient address is wrong, then the receiving server replies with an error message of some kind.

SMTP Components

1. **Mail User Agent (MUA)** – Mail User Agent or User-Agent is the client application for sending and receiving emails. Ex – Gmail, Microsoft Outlook, etc.
2. **Mail Submission Agent (MSA)** – Mail Submission Agent is a server that receives mail from MUA, checks for errors, and in case of no errors passes it to the SMTP server. There exist Mail transfer agents that work as a mail submission agent too.
3. **Mail Transfer Agent (MTA)** – Mail Transfer Agent receives the email from MSA or MUA(if MSA is not used) and passes the mail(using Simple Mail Transfer Protocol SMTP) to other MTA. An MTA works in the background, while the user can't interact directly with a mail transfer agent. It will also find the MX record from the receivers' DNS zone. Ex – Postfix, SendMail etc.
4. **Mail Delivery Agents (MDA)** – Mail Delivery agents are responsible for delivering emails to recipients' mailboxes.

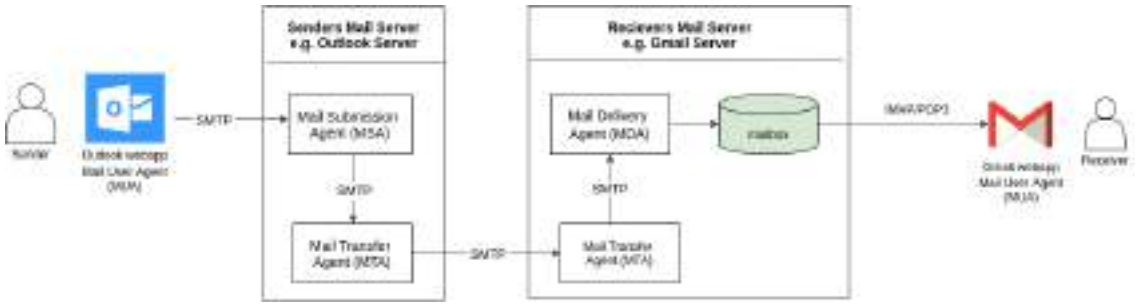
SMTP Working

An SMTP session is initiated when a client opens a connection to a server by issuing the EHLO command and the server responds with an opening message. The SMTP session consists of commands originated by the SMTP clients and the responses from the SMTP Server. A session can contain zero or more SMTP transactions. Three important commands in SMTP transaction-

1. **MAIL** – The transaction starts with the mail command. It does the sender identification.
2. **RCPT** – It gives the receiver information. If RCPT command exists before previous MAIL command the n server throws 503 "Bad Sequence of Commands" response. This command can be issued multiple times.
3. **DATA** – This command tells about the beginning of a message. Once accepted the SMTP server immediately replies with 354 responses and when the entire message is

received SMTP server again replies with **"250 OK"**. SMTP indicates the end of the mail data by sending a line containing only a "." (period or full stop).

The client who wants to send an email, the Mail client(or formally known as mail user agent or SMTP client) opens a full-duplex(two way) TCP connection with the Mail Server(SMTP Server) which is always in listening mode at the port number (25) and an SMTP session will be created. The Mail client uses TCP because it is a reliable ordered data stream channel for communication, hence the entire message will be delivered to the server without any data loss.



7. 6 REVIEW QUESTIONS

1. What is the significance of URL?

2. What is the significance of bcc and cc in email?

3. What are the two purpose of TELNET?

4. What are the similarities and differences between a domain name and its address**?**

7.7 SUMMARY

In this unit, we have given you a broad overview of various protocols of application layer of OSI reference model. Our discussions are focused on only some of the important protocols, but it doesn't mean that these are the only protocols associated with the application layer. Many other protocols are used by the network administrator for different purpose. You should also keep on exploring about other protocols.

7.8 REFERENCES FOR FURTHER READINGS

1. BehrouzForouzan, Data communications and Networking, Tata Mc-Graw Hill.

2. Andrew S. Tanenbaum, Computer Networks, Pearson Education.

3. William Stallings, Data and Computer Communications, Pearson education.

UNIT 8: NETWORK SECURITY

STRUCTURE

8.0 Objectives

8.1 Introduction

8.2 Common Terms

8.3 Network Security Issues

8.4 Goals of Network Security

8.5 Types of Network Attack

8.5.1 Difference between Active and Passive Attack

8.6 Firewall

8.6.1 Firewall types

8.6.2 Deployment of Firewall

8.7 Virtual Private Network (VPN)

8.7.1 VPN Types

8.7.2 VPN Protocols

8. 6 Review Questions

8.7 Summary

8.8 References for Further Readings

8.0 INTRODUCTION

The modern era is an age of information. Our life, these days are more digitally advanced than ever, and as technology improves, organization's security postures must be enhanced as well. Now, with many devices communicating with each other over wired, wireless, or cellular networks, network security is an important concept. In this unit, we will explore what are network security and its key features.

Network security is the process of taking preventative measures to protect the associating networking infrastructure from unauthorized access, misuse, malfunction, modification, destruction or improper disclosure.

Due to the frequency and variety of existing attacks as well as the threat of new and more destructive future attacks, network security has become a central topic in the field of cyber security. By implementing and learning about network security, a small business will make their employees more responsible, a law firm will be motivated to protect its data, and an interior designer will find more effective ways to control their heavy files.

8.1 OBJECTIVES

At the end of this unit, you will be able to:

- Understand the concept of network security in data transmission.
- Discuss different terms used in network security.
- Discuss in detail the various types of Network Attacks.
- Understand the concept of firewall with its types.
- Discuss various types of Virtual Private Network (VPN) and VPN Protocols.

8.2 COMMON TERMS

Advanced Persistent Threats:

When an unauthorized user invades a network, stays for an extended period of time, and steals data without harming the network.

Adware:

Software that automatically displays or downloads material, even user is in offline mode.

Black hat:

"Black hat" is used for the hackers that break into the network to steal information that will be used to harm the owner or the users without consent. It's entirely illegal.

Bitcoin:

Bitcoin is a cryptocurrency, a form of electronic cash created by Satoshi Nakamoto.

Brute force attack:

An attacker inputs many passwords in the hope that it is eventually guessed correctly.

Breach:
The moment a hacker successfully exploits a vulnerability in a computer or device, and gains access to its files and network.

Bot/Botnet:
It is a type of software application or script that performs tasks on command, allowing an attacker to take complete control remotely of an affected computer. A collection of these infected computers is known as a "botnet" and is controlled by the hacker or "bot-herder".

BYOD (Bring Your Own Device):
It refers to a company security policy that allows for employees' personal devices to be used in business. A BYOD policy sets limitations and restrictions on whether or not a personal phone or laptop can be connected over the corporate network.

Catfishing:
A fake identity on a social network account is creating, usually a dating website, to target a specific victim for deception.

Cloud:
Cloud is a technology that allows us to access our files and/or services through the internet from anywhere in the world. Technically speaking, it's a collection of computers with large storage capabilities that remotely serve requests.

Domain:
Domain is a group of computers, printers and devices that are interconnected and governed as a whole. For example, your computer is usually part of a domain at your workplace.

Deepfake:
It is an audio or video clip that has been edited and manipulated to seem real or believable. The most dangerous consequence of the popularity of deepfakes is that they can easily convince people into believing a certain story or theory that may result in user-behavior with a bigger impact as in political or financial.

DDoS:
An acronym that stands for distributed denial of service – a form of cyber attack. This attack aims to make a service such as a website unusable by "flooding" it with malicious traffic or data from multiple sources (often botnets).

Exploit:
Exploit is a malicious application or script that can be used to take advantage of a computer's vulnerability.

Firewall:
A defensive technology designed to keep the bad guys out. Firewalls can be hardware or software-based.

Hacker:
Hacker is a person who spends time learning the details of computer programming and operating systems, how to test the limits of their capabilities, and where their vulnerabilities lie.

IP Address:
An internet version of a home address for your computer, which is identified when it communicates over a network; e.g. connecting to the internet (a network of networks).

Keylogger:
Keylogger is a computer program that records keystrokes made by a user. This user is typically unaware that their actions are being monitored and that a hacker now has access to passwords and other confidential data.

Mitigation defense:
Mitigation defense is software that doesn't stop hacking from happening, but will mitigate the effects.

Malware:
A common term that describes all forms of malicious software designed to wreak havoc on a computer. Common forms include: viruses, trojans, worms and ransomware.

Phishing:
A technique used by hackers to obtain sensitive information. For example, using hand-crafted email messages designed to trick people into divulging personal or confidential data such as passwords and bank account information.

ReCAPTCHA:
ReCAPTCHA is a service from Google that works to protect websites from spam and abuse caused by robots. A user is presented with a Turing test to distinguish them from a robot.

Ransomware:
Ransomware is a form of malware that deliberately prevents you from accessing files on your computer – holding your data hostage. It will typically encrypt files and request that a ransom be paid in order to have them decrypted or recovered.

Rootkit:
Rootkit is a kind of malware that allows cybercriminals to remotely control your computer. Rootkits are especially damaging because they are hard to detect, making it likely that this type of malware could live on your computer for a long time.

Spyware:
It is a type of malware that functions by spying on user activity without their knowledge. The capabilities include activity monitoring, collecting keystrokes, data harvesting (account information, logins, financial data), and more.

Sniffer:
A program that captures data as it travels across a network. Also called a packet sniffer.

Trialware:
Software that can only be run for a limited amount of time before it expires.

Trojan horse:
It is a computer program that appears to perform a desirable function but contains hidden code that is intended to allow unauthorized collection, modification or destruction of data.

Technical vulnerability: A flaw or bug in the hardware or software components of a system that leaves it vulnerable to security breach.

Virtual Private Network (VPN):
VPN is a tool that allows the user to remain anonymous while using the internet by masking the location and encrypting traffic.

Virus:
A type of malware aimed to corrupt, erase or modify information on a computer before spreading to others.

Vulnerability:
A weakness in the hardware or software or security plan that leaves a system or network open to threat of unauthorized access or damage or destruction of data.

Worm:
A piece of malware that can replicate itself in order to spread the infection to other connected computers.

White hat:

It breaches the network to gain sensitive information with the owner's consent – making it completely legal. This method is usually employed to test infrastructure vulnerabilities.

8.3 NETWORK SECURITY ISSUES

The importance of network security in the modern business atmosphere has increased after a major portion of the workforce went remote due to COVID-19. Today, the office system is distributed in one huge network across multiple geographical locations. A centralized system

to protect network devices from being breached doesn't exist in the same capacity. This scenario leaves more vulnerable points that hackers can take advantage of.

In today's business infrastructure, network security is not limited to IT professionals and companies connected with it. Network security is for everyone — lawyers, interior decorators, musicians, investment bankers, etc., will all find a network security system beneficial for their work and business.

A *network attack* can be defined as any method, process, or means used to maliciously attempt to compromise network security. Network security is the process of preventing network attacks across a given network, but the techniques and methods used by the attacker further distinguish whether the attack is an active cyber attack, a passive type attack, or some combination of the two.

8.4 GOALS OF NETWORK SECURITY

As discussed in earlier sections, there exists large number of vulnerabilities in the network. Thus, during transmission, data is highly vulnerable to attacks. An attacker can target the communication channel, obtain the data, and read the same or re-insert a false message to achieve his nefarious aims.

Network security is not only concerned about the security of the computers at each end of the communication chain; however, it aims to ensure that the entire network is secure.

Network security entails protecting the usability, reliability, integrity, and safety of network and data. Effective network security defeats a variety of threats from entering or spreading on a network.

The primary goals of network security are Confidentiality, Integrity, and Availability. These three pillars of Network Security are often represented as CIA triangle.

- **Confidentiality:** The function of confidentiality is to protect precious business data from unauthorized persons. Confidentiality part of network security makes sure that the data is available only to the intended and authorized persons.

- **Integrity:** This goal means maintaining and assuring the accuracy and consistency of data. The function of integrity is to make sure that the data is reliable and is not changed by unauthorized persons.

- **Availability:** The function of availability in Network Security is to make sure that the data, network resources/services are continuously available to the legitimate users, whenever they require it.

Vulnerabilities & Attacks

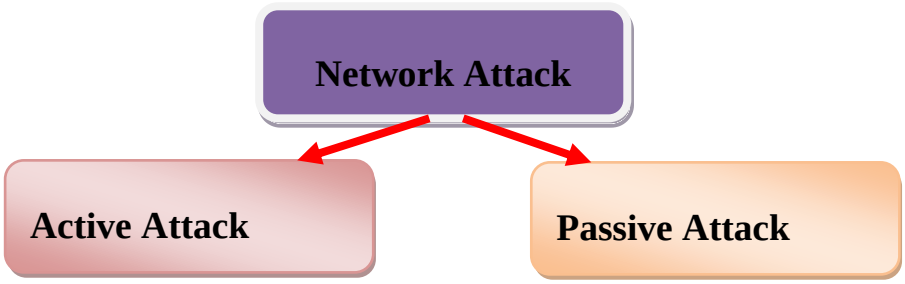
The common vulnerability that exists in both wired and wireless networks is an "unauthorized access" to a network. An attacker can connect his device to a network though unsecure hub/switch port. In this regard, wireless network are considered less secure than

wired network, because wireless network can be easily accessed without any physical connection.

After accessing, an attacker can exploit this vulnerability to launch attacks such as under:

- Sniffing the packet data to steal valuable information.
- Denial of service to legitimate users on a network by flooding the network medium with spurious packets.
- Spoofing physical identities (MAC) of legitimate hosts and then stealing data or further launching a 'man-in-the-middle' attack.

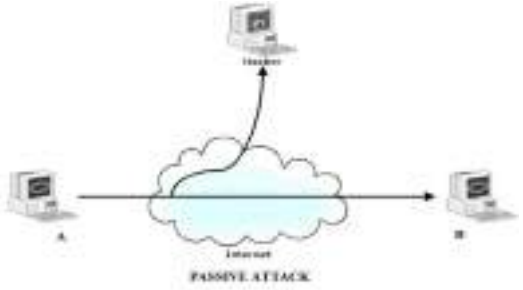
8.5 TYPES OF NETWORK ATTACK



Passive Attack:

In passive attacks, hackers monitor and scan systems for vulnerabilities or entry points that allow them to intercept information without changing any of it. In a passive attack, an intruder monitors a system and network communications and scans for open ports and other vulnerabilities. For example, they might exploit an unpatched system or take advantage of an expired certificate on a security device.

Once the intruder has infiltrated the network, they can collect information in a couple of ways. In a footprinting passive attack, the intruder will try to collect as much intelligence as they can to use it later to attack the target system or network in a later step. An example is when an intruder records network traffic using a packet analyzer tool, such aswireshark for later analysis. Installing a keylogger is another sort of passive attack, where an intruder waits for the user to enter their credentials and records them for later use.



The two most common use cases of passive attacks are:

1. **Traffic analysis:** In this type, an attacker monitors communication channels to collect a range of information, including human and machine identities, locations of these identities and types of encryption used, if applicable.
2. **Release of message contents:** In this type, an attacker will monitor an unprotected communication medium—like unencrypted email or telephone call—and intercept it for sensitive information.

Other types of passive attacks include "passive reconnaissance," where an attacker tries to gain important information about the target organization connected to the internet without sending any traffic (packets) to the target server or network. An example of such type of attack includes browsing a website contents for relevant information (such as employee contact information) that can be used in active attacks or finding files that have been left unprotected on a target server, such as meeting papers or intellectual property.

Detecting a passive attack is very difficult and impossible in many cases because it does not involve data alteration in any way. However, you can implement protective measures to stop it, including:

By using encryption techniques to scramble messages, making them unreadable for any unintended recipients.

Active Attack:

In an active attack, hackers attempt to modify the integrity and availability of the information they have intercepted—with the goal of gaining access or greater privileges. Simply put, hackers may use data they have gathered during passive attacks to compromise a target in an active attack.

The attacker can read and update the data without the information of any of the users. In Active Attack, the attacker tries to induce noise in the data transmission.

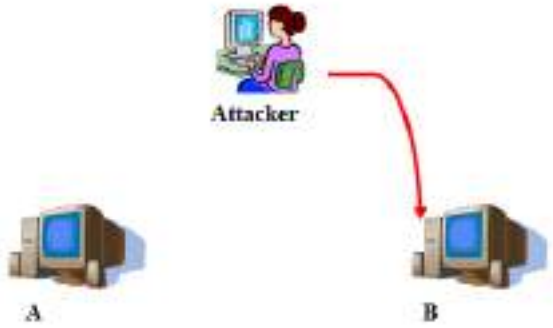


ACTIVE ATTACK - MESSAGE MODIFICATION

Active attack examples

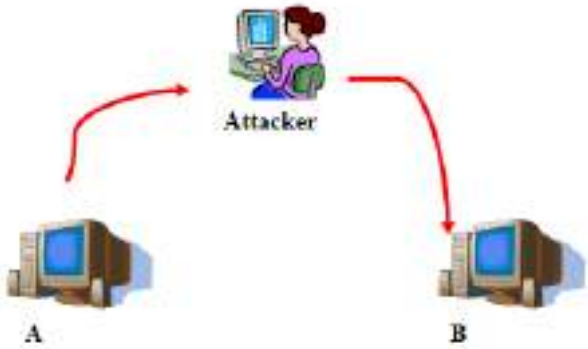
1) Masquerade

Assume that A and B are connected and they are transferring data to each other. A and B are genuine users. In the Masquerade attack, the attacker used the identity of the authentic users and he breaks into the communication and behaves like the authentic user and grabs all the data.



2) Relay:

Assume that A and B are connected and they are transferring data to each other. A is sending some message to B. The message is on its way but in between the attacker captures the message and now not only he can read the message but he can update and modify it too. He can create error bits in the message. Error bits are the bits that don't belong to the original message.



3) Denial of service:

In this attack, the attacker sends a lot of requests to the server to increase the traffic. If the server has a lot of requests then it will take a lot of time to respond to the genuine requests which are made by the authentic users. In this way, by increasing the traffic on the server, he can slow down the server. In this way, the authentic users will not get a response from the server. In this way, their service is denied.

Types of active attacks include:

- Denial of service (DoS)
- Distributed Denial of Service (DDoS)
- Session replay
- Masquerade
- Message modification
- Trojans

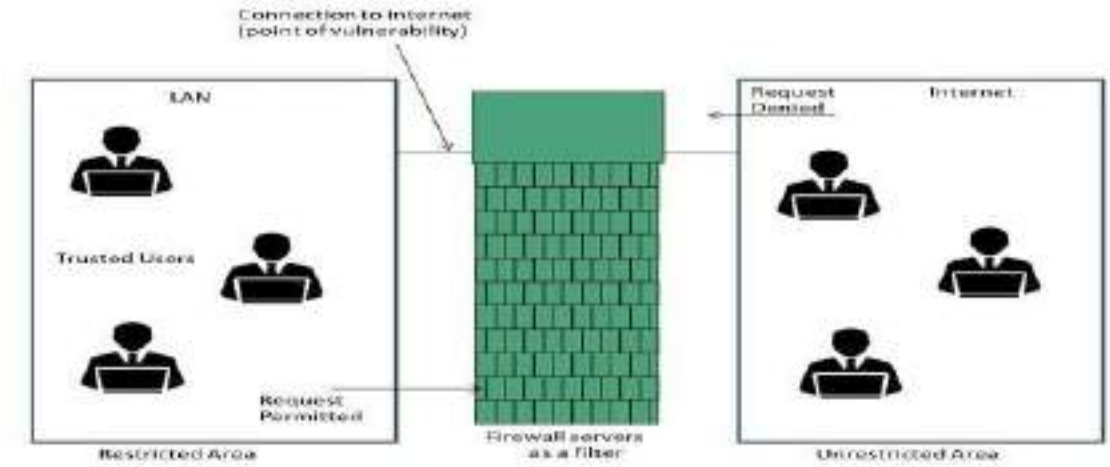
7.5.1 Difference between Active and Passive Attack

Active Attack	Passive Attack
Attacker needs to have physical control of the media or network.	Attacker merely needs to observe the communication in the media or network.
It can be easily detected.	It cannot be easily detected.
It affects the system.	It does not affect the system.
It involves a modification of data.	It involves the monitoring of data.
Types of active attacks are Masquerade, session replay, denial of service, distributed denial of service.	Types of passive attacks are the Release of a message, traffic analysis.
It does not check for loopholes or vulnerabilities.	It scans the ports and network in the search of loopholes and vulnerabilities.
It is difficult to prevent network from active attack.	Passive attacks can be prevented.

8.6FIREWALL

Firewall is a filter between Local Area Network (LAN) and the Internet. It allows keeping private resources confidential and maximizes the security. It controls network traffic, in both directions (to and from).

The following diagram depicts a sample firewall between LAN and the internet. The connection between the two is the point of vulnerability. Both hardware and the software can be used at this point to filter network traffic.



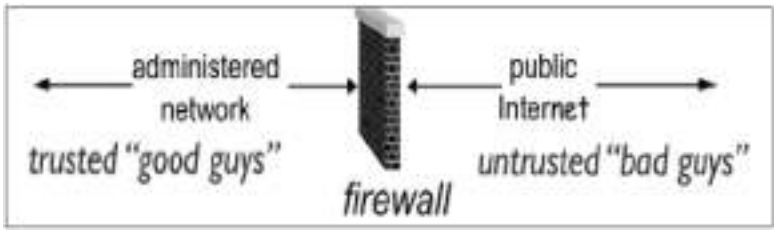
There are two types of Firewall system:

One works by using filters at the network layer and

The other works by using proxy servers at the user, application, or network layer.

Firewall management must be configured by both system managers and the network managers. The amount of filtering a firewall varies; the amount of filtering may be different in different directions for same firewall.

Firewall is a network device with inbuilt software that isolates organization's internal network from larger outside network/Internet. It can be hardware, software, or combined system that prevents unauthorized access to or from internal network. All data packets entering or leaving the internal network pass through the firewall, which examines each packet and blocks those that do not meet the predefined security criteria.



Deploying firewall at network boundary is like aggregating the security at a single point. It is analogous to deploying a security guard at the entrance in an apartment and not necessarily at each flat.

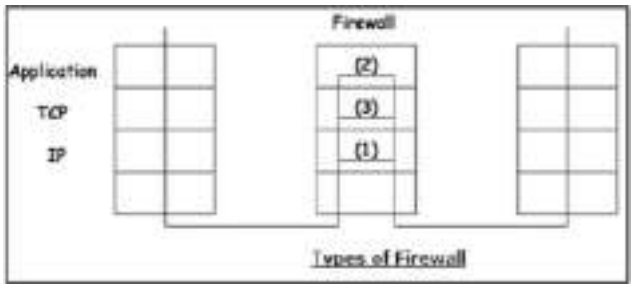
Firewall is considered as an essential to achieve network security for the following reasons −

1. Internal network and hosts are unlikely to be properly secured.
2. Internet is a dangerous place with criminals, users from competing companies, disgruntled ex-employees, spies from unfriendly countries, vandals, etc.
3. To prevent illegal modification/access to internal data by an outsider attacker.
4. To prevent an attacker from launching denial of service (DoS) attacks on network resource.

8.6.1 Firewall types

1. Packet filter (Stateless &Stateful)
2. Application-level gateway
3. Circuit-level gateway

These three categories, however, are not mutually exclusive. Modern firewalls have a mix of abilities that may place them in more than one of the three categories.



1. Packet Filtering Firewall

In this type of firewall deployment, the internal network is connected to the external network/Internet via a router firewall. The firewall filters data packet-by-packet. Packet filtering firewalls allow or block the packets mostly based on criteria such as source and/or

destination IP addresses, protocol, source and/or destination port numbers, and various other parameters within the IP header.

The decision can be based on factors other than IP header fields such as ICMP message type, TCP SYN and ACK bits, etc. Packet filter rule has two parts −

- ***Selection criteria*** − It is a used as a condition and pattern matching for decision making.
- ***Action field*** −This part specifies action to be taken if an IP packet meets the selection criteria. The action could be either block (deny) or permit (allow) the packet across the firewall.

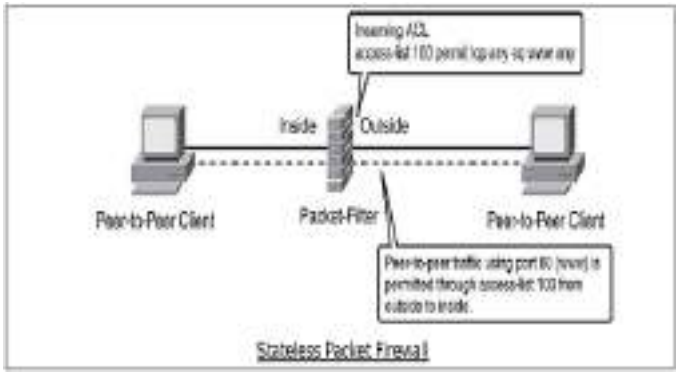
Packet filtering is generally accomplished by configuring Access Control Lists (ACL) on routers or switches. ACL is a table of packet filter rules.

As traffic enters or exits an interface, firewall applies ACLs from top to bottom to each incoming packet, finds matching criteria and either permits or denies the individual packets.

Stateless firewall

It is a kind of a rigid tool. It looks at packet and allows it if its meets the criteria even if it is not part of any established ongoing communication.

Hence, such firewalls are replaced by stateful firewalls in modern networks. This type of firewalls offer a more in-depth inspection method over the only ACL based packet inspection methods of stateless firewalls.



Stateless Packet Firewall

Stateful firewall

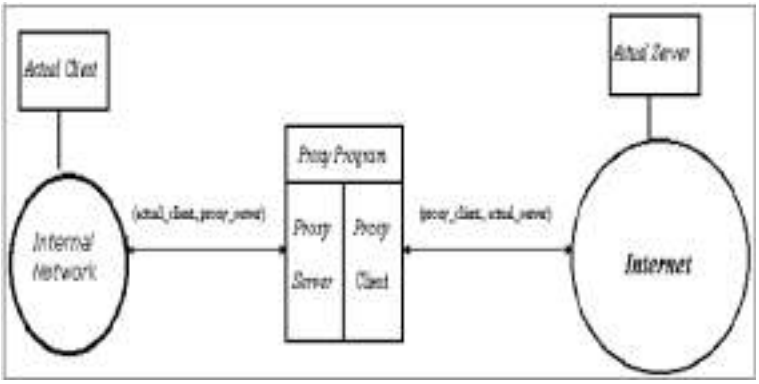
It monitors the connection setup and teardown process to keep a check on connections at the TCP/IP level. This allows them to keep track of connections state and determine which hosts have open, authorized connections at any given point in time.

They reference the rule base only when a new connection is requested. Packets belonging to existing connections are compared to the firewall's state table of open connections, and decision to allow or block is taken. This process saves time and provides added security as well. No packet is allowed to trespass the firewall unless it belongs to already established connection. It can timeout inactive connections at firewall after which it no longer admit packets for that connection.

2. Application Gateways:

An application-level gateway acts as a relay node for the application-level traffic. They intercept incoming and outgoing packets, run proxies that copy and forward information across the gateway, and function as a proxy server, preventing any direct connection between a trusted server or client and an untrusted host.

The proxies are application specific. They can filter packets at the application layer of the OSI model.



An application-specific proxy accepts packets generated by only specified application for which they are designed to copy, forward, and filter. For example, only a Telnet proxy can copy, forward, and filter Telnet traffic.

If a network relies only on an application-level gateway, incoming and outgoing packets cannot access services that have no proxies configured. For example, if a gateway runs FTP and Telnet proxies, only packets generated by these services can pass through the firewall, and all other services are blocked.

Application-level Filtering

An application-level proxy gateway, examines and filters individual packets, rather than simply copying them and blindly forwarding them across the gateway. Application-specific proxies check each packet that passes through the gateway, verifying the contents of the packet up through the application layer. These proxies can filter particular kinds of commands or information in the application protocols.

Application gateways can restrict specific actions from being performed. For example, the gateway could be configured to prevent users from performing the 'FTP put' command. This can prevent modification of the information stored on the server by an attacker.

Transparent

Although application-level gateways can be transparent, many implementations require user authentication before users can access an untrusted network, a process that reduces true transparency. Authentication may be different if the user is from the internal network or from the Internet. For an internal network, a simple list of IP addresses can be allowed to connect to external applications. But from the Internet side a strong authentication should be implemented.

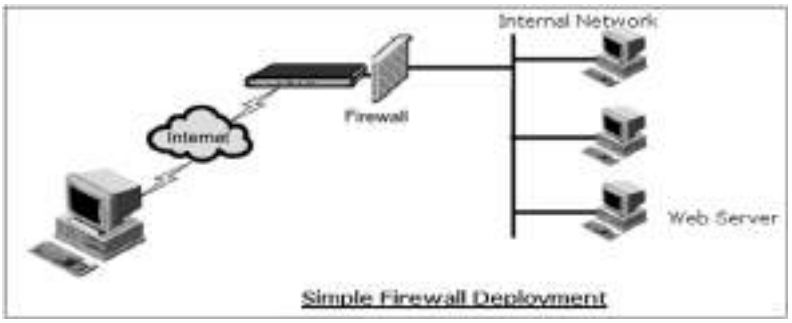
An application gateway actually relays TCP segments between the two TCP connections in the two directions (Client ↔ Proxy ↔ Server).
For outbound packets, the gateway may replace the source IP address by its own IP address. The process is referred to as ***Network Address Translation*** (NAT). It ensures that internal IP addresses are not exposed to the Internet.

3. Circuit-Level Gateway:
The circuit-level gateway is an intermediate solution between the packet filter and the application gateway. It runs at the transport layer and hence can act as proxy for any application.
Similar to an application gateway, the circuit-level gateway also does not permit an end-to-end TCP connection across the gateway. It sets up two TCP connections and relays the TCP segments from one network to the other. But it does not examine the application data like application gateway. Hence, sometime it is called as 'Pipe Proxy'.

SOCKS
SOCKS (RFC 1928) refer to a circuit-level gateway.OpenText™ SOCKS Client is a Windows–certified security solution that connects to hosts across firewalls. It is a networking proxy mechanism that enables hosts on one side of a SOCKS server to gain full access to hosts on the other side without requiring direct IP reachability. The client connects to the SOCKS server at the firewall. Then the client enters a negotiation for the authentication method to be used, and authenticates with the chosen method.

The client sends a connection relay request to the SOCKS server, containing the desired destination IP address and transport port. The server accepts the request after checking that the client meets the basic filtering criteria. Then, on behalf of the client, the gateway opens a connection to the requested untrusted host and then closely monitors the TCP handshaking that follows.
The SOCKS server informs the client, and in case of success, starts relaying the data between the two connections. Circuit level gateways are used when the organization trusts the internal users, and does not want to inspect the contents or application data sent on the Internet.

8.6.2 Deployment of Firewall
A firewall is a mechanism used to control network traffic 'into' and 'out' of an organizational internal network. In most cases these systems have two network interfaces, one for the external network such as the Internet and the other for the internal side.
The firewall process can tightly control what is allowed to traverse from one side to the other. An organization that wishes to provide external access to its web server can restrict all traffic arriving at firewall expect for port 80 (the standard http port). All other traffic such as mail traffic, FTP, SNMP, etc., is not allowed across the firewall into the internal network. An example of a simple firewall is shown in the following diagram.

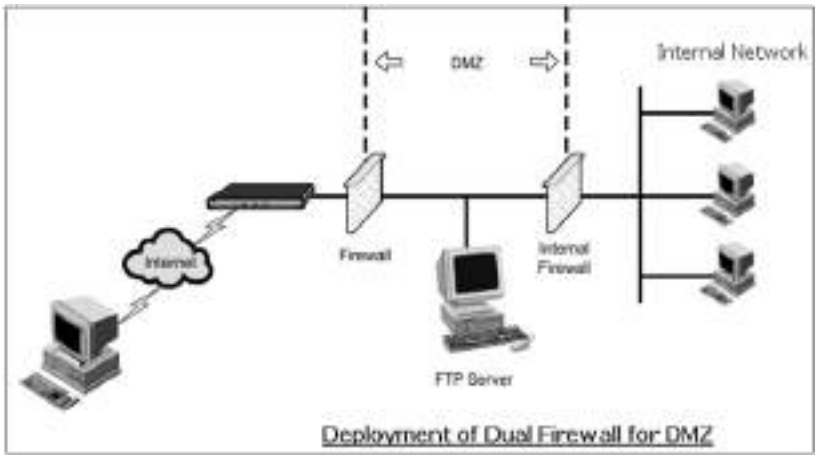


In the above simple deployment, though all other accesses from outside are blocked, it is possible for an attacker to contact not only a web server but any other host on internal network that has left port 80 open by accident or otherwise.

Hence, the problem most organizations face is how to enable legitimate access to public services such as web, FTP, and e-mail while maintaining tight security of the internal network. The typical approach is deploying firewalls to provide a Demilitarized Zone (DMZ) in the network.

In this setup (illustrated in following diagram), two firewalls are deployed; one between the external network and the DMZ, and another between the DMZ and the internal network. All public servers are placed in the DMZ.

With this setup, it is possible to have firewall rules which allow public access to the public servers but the interior firewall can restrict all incoming connections. By having the DMZ, the public servers are provided with adequate protection instead of placing them directly on external network.



Intrusion Detection / Prevention System

The packet filtering firewalls operate based on rules involving TCP/UDP/IP headers only. They do not attempt to establish correlation checks among different sessions.

Intrusion Detection/Prevention System (IDS/IPS) carry out Deep Packet Inspection (DPI) by looking at the packet contents. For example, checking character strings in packet against database of known virus, attack strings.

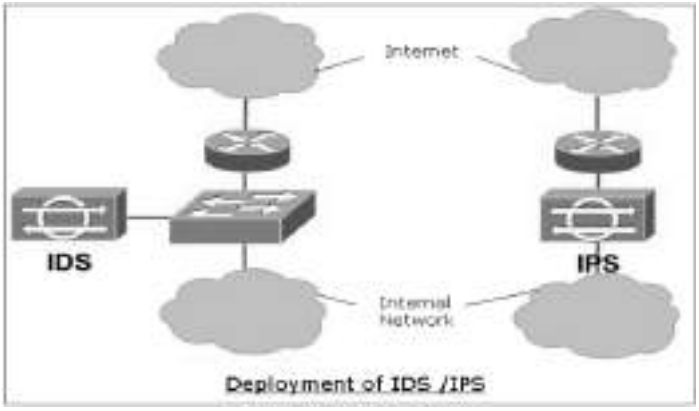
Application gateways do look at the packet contents but only for specific applications. They do not look for suspicious data in the packet. IDS/IPS looks for suspicious data contained in packets and tries to examine correlation among multiple packets to identify any attacks such as port scanning, network mapping, and denial of service and so on.

Difference between IDS and IPS

IDS and IPS are similar in detection of anomalies in the network. IDS is a 'visibility' tool whereas IPS is considered as a 'control' tool.

IPS sits off to the side of the network, monitoring traffic at many different points, and provide visibility into the security state of the network. In case of reporting of anomaly by IDS, the corrective actions are initiated by the network administrator or other device on the network.

IPS are like firewall and they sit in-line between two networks and control the traffic going through them. It enforces a specified policy on detection of anomaly in the network traffic. Generally, it drops all packets and blocks the entire network traffic on noticing an anomaly till such time an anomaly is addressed by the administrator.



Deployment of IDS /IPS

8.7 VIRTUAL PRIVATE NETWORK (VPN)

When you use a private WAN connection from a service provider, you trust them that they treat your data confidential. The service provider will make sure that they separate traffic from different customers and that nobody else is able to see your data. When you send traffic from A to B using the Internet, you have no control at all which networks are used to get from the source to the destination. Someone in between the traffic path might be capturing your packets and you wouldn't know.

VPNs (Virtual Private Network) help by establishing a secure connection over an insecure network, such as the Internet. This is a great alternative to private WAN connections since Internet access is usually cheaper and it's available pretty much everywhere.

VPNs provide a couple of features such as:

- **Confidentiality:** preventing anyone from reading your data. This is implemented with encryption.

- **Authentication:** verifying that the router/firewall or remote user that is sending VPN traffic is a legitimate device or router.
- **Integrity:** verifying that the VPN packet wasn't changed somehow during transit.
- **Anti-replay:** preventing someone from capturing traffic and resending it, trying to appear as a legitimate device/user.

8.7.1 VPN Types

There are two common VPN types that we use:

- **site-to-site VPN**
- **client-to-site VPN (remote user)**

Site-to-site VPN

With the site-to-site VPN, we have a network device at each site, between these two network devices we build a VPN tunnel. Each end of the VPN tunnel will encrypt the original IP packet, adds a VPN header, a new IP header and then forwards the encrypted packet to the other end of the tunnel.

Client-to-site VPN

The client-to-site VPN is also called the ***remote user VPN***. The user installs a VPN client on his/her computer, laptop, smartphone or tablet. The VPN tunnel is established between the user's device and the remote network device.

8.7.2 VPN Protocols

There are a number of VPN protocols, the most common ones are:

- **IPsec**
- **PPTP**
- **L2TP**
- **SSL VPN**

IPsec

The IP protocol itself doesn't have any security features at all, which is why IPSec was created. IPSec framework is responsible for confidentiality, integrity, authentication and anti-replay features on network layer of the OSI model. It uses a variety of protocols and the advantage of a framework is that the protocols it uses can change in the future. For example, encryption algorithms like DES, 3DES or AES are presently most favourable protocols but if a new algorithm is created, IPSec might use it in the future like:-

- ✓ Creating a site-to-site VPN tunnel.
- ✓ Creating a client-to-site (remote user) VPN tunnel.
- ✓ Between two servers to authenticate and/or encrypt traffic.

PPTP

PPTP (Point to Point Tunnelling Protocol) is one of the older VPN protocols, released in 1995. It uses a GRE tunnel for tunnelling and PPP for authentication. Encryption is done with the MPPE protocol. PPTP is supported on many clients and operating systems. Although, PPTP has been proven insecure, therefore this protocol is not used anymore.

L2TP

L2TP (Layer Two Tunnelling Protocol) is an extension of PPTP. L2TP can be used to "bridge" two remote LANs together and you want to use a single subnet on both sites. L2TP itself does not offer any encryption or anything, therefore use it together with IPSec and together, it's often referred to as L2TP/IPSec

SSL VPN

SSL (Secure Sockets Layer) is a protocol that is normally used to encrypt data traffic between a web browser and web server. Even though it's called SSL VPN, but nowadays network engineers uses TLS (Transport Layer Security) for HTTPS, which is the successor of SSL.
One of the advantages of SSL VPN is that since it uses HTTPS, its omni acceptable.

Most public wifi hotspots do permit HTTPS traffic while some might block other traffic like IPSec. Most SSL VPN solutions offer a "portal" through the web browser to access applications. For some advanced features, a software client has to install.

8. 6 REVIEW QUESTIONS

1. What are the potential outcomes of a network security attack?
2. What are the basic objectives of network security?
3. What is the importance of firewalls in network security?
4. What is the significance of Virtual Private Network?
5. Why there are different protocols associated with VPN?

8.7 SUMMARY

Therefore, you can notice that you have to cover a broad range of topics for network security. The network security topics presented here provide you an interaction with network security issues. Most important of all, the above-mentioned questions also reflect on the fundamentals of various aspects of network security. So, aspiring candidates should start their preparations immediately for network security for further reading and explore the same from the reference books mentioned in the next section here.

REFERENCES FOR FURTHER READINGS

1. BehrouzForouzan, Data communications and Networking, Tata Mc-Graw Hill.
2. Andrew S. Tanenbaum, Computer Networks, Pearson Education.
3. William Stallings, Data and Computer Communications, Pearson education.