**National Polytechnic School of Oran – Maurice Audin (Ex. ENSET)**

# Course Handbook (Lectures & Practical Sessions – Tutorials)

# Module: Enterprise Network (Réseaux d'Entreprise)

## IMSI (Management Engineering and Information Systems)

**Prepared by: Dr.BOUMEDJOUT AMAL**

**Academic Year: 2024/2025**

## Abstract

This course material entitled **"Computer Networks"** provides a structured overview of networking courses delivered to **first-year engineering students** specializing in **Management Engineering and Information Systems (IMSI)**.
This course aims to introduce the core principles of IP-based networks, describe their operation, and present the essential protocols required for building reliable and efficient network infrastructures.
It covers the principles of the **OSI and TCP/IP architectures**, compares different **network topologies**, and explains **network cabling practices** in accordance with **TIA/EIA standards**. The course also addresses **IP addressing mechanisms**, including configuration, validation, and troubleshooting.
Furthermore, this course introduces the basics of **network administration**, guiding students toward practical skills such as **resource sharing**, **management of core network protocols** (IP, DHCP, DNS, ARP), and **router configuration**.
The practical component of the course is supported by **hands-on laboratory sessions using Windows and Linux environments**, as well as **network simulation tools such as Cisco Packet Tracer**.
Finally, a set of **guided exercises and tutorial sessions** is included to enhance comprehension and ensure effective mastery of the covered topics.

# TABLE OF CONTENTS

# CHAPTER I: BASIC CONCEPTS OF COMPUTER NETWORKS

## INTRODUCTION

Computer networks were developed to enable the transmission of information from a central location to remote terminals. This need expanded rapidly with the progress of telecommunications technologies. These advances allowed the transition from simple data exchange to more advanced communications, including voice and video. The widespread adoption of the Internet has significantly enhanced communication and facilitated efficient information exchange across various fields.

## I. DEFINITION OF A COMPUTER NETWORK

A computer network is an essential means for exchanging information. A digital network consists of a group of hosts, such as computers and printers, connected through physical links. It enables data exchange between distant machines, either through a direct connection or via intermediate devices.

A computer network can also be defined as a set of hardware and software resources distributed across different locations, allowing users to share information and services. A network is made up of devices known as nodes. To communicate with one another, these nodes follow a set of communication rules called communication protocols.

## II. PURPOSES OF COMPUTER NETWORKS

1. Sharing of resources: Applications, data, and hardware resources can be accessed by authorized users, independent of their physical location.
2. Improved data management in organizations: In professional environments, the use of removable storage media such as disks, CDs, or USB drives is inefficient and costly. Any modification to a file would require repeated redistribution, making data consistency difficult to ensure.
3. Efficient communication: Computer networks provide rapid communication between users and promote cooperation and information exchange.
4. Cost efficiency: Data can be storedon one or more file servers and printed through shared network printers, reducing hardware expenses and operational costs.

## III. CLASSIFICATION OF COMPUTER NETWORKS

Computer networks can be classified based on two main criteria:

- **Network coverage (scope)**
- **Transmission technique**

## A. Classification According to Network Coverage

There are three main types of computer networks:

- **LAN (Local Area Network):**

A LAN connects hosts that are located within a limited geographical area, typically ranging from one meter to a few kilometers. It can support from a small number to several hundred users and operates at data rates between 10 Mbps and 1 Gbps. LANs are mainly used for local sharing of computing resources, including software and hardware.

LANs differ from other types of networks based on three key characteristics: their size, transmission techniques, and network topology.

As computer usage became widespread within organizations, local networks alone were no longer sufficient. This limitation led to the development of new technologies designed to support information sharing over larger areas, resulting in the emergence of MAN and WAN networks.

- **MAN (Metropolitan Area Network):**

  A MAN connects multiple LANs that are geographically close, typically within a city or metropolitan area, covering distances of several tens of kilometers.

- **WAN (Wide Area Network):**

  A WAN covers very large geographical areas, extending over hundreds or thousands of kilometers, and is capable of interconnecting a large number of users and networks.

| **Distance Between Processors** | Processor Location |
|---|---|
| **0,1m** | Printed circuit board (PAN) |
| **10m** | Single room (LAN) |
| **100m** | Building (LAN) |
| **1Km** | 1 Campus (LAN) |
| **10Km** | City (MAN) |
| **100Km** | Region (WAN) |
| **1000Km** | Continent (WAN) |

**Table 1-1: Classification of Networks According to Their Size**

## B. Classification of Networks According toThe Transmission Techniques

Networks can be classified based on their transmission techniques. Two main transmission modes are identified:

- **Broadcast Transmission:**

In broadcast-based networks, every device receives the data packet sent by the source host. Therefore, each packet carries a special destination address known as a broadcast address. In certain systems, only a selected group of devices is intended to process the packet; this form of transmission is referred to as multicast or restricted broadcast.

- **Point-to-Point Transmission:**

This transmission method relies on multiple direct links between pairs of hosts. To travel from the source to the destination, a data packet may follow one or more possible paths. Routing protocols determine the most appropriate path based on specific network criteria.

# IV. NETWORK TOPOLOGIES

Network topology defines the arrangement of network nodes and their interconnections. Two main categories of topology can be identified:

- **Physical topology:**

  Describes the physical structure of the network and how devices are physically connected.

- **Logical topology:**

  Describes the way data flows and how communication occurs between devices within the network.

## A. Physical Network Topologies

- **Bus Topology:**

  In a bus topology, all computers share a single transmission medium, typically a coaxial cable. The length of the cable is limited due to signal loss. Devices such as repeaters are used to extend the network and restore signal strength.

- **Star Topology:**

  In a star topology, all devices are connected to a central network component, such as a hub or a switch, using twisted-pair cables with RJ45 connectors. All data traffic passes through the central device, which represents a critical point in the network architecture.

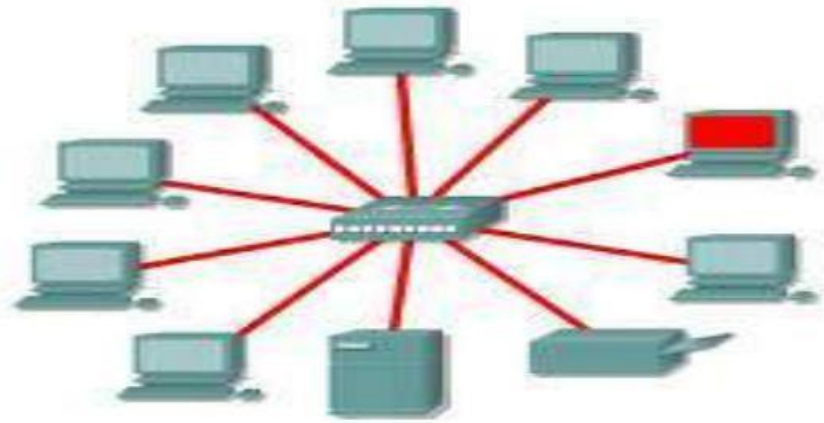


**Figure 1-2: Star Topology**

- **Hierarchical Topology**

  Derived from the star topology, It is composed of multiple star-based networks interconnected through central devices such as hubs. This structure organizes the network into levels, improving scalability and management.

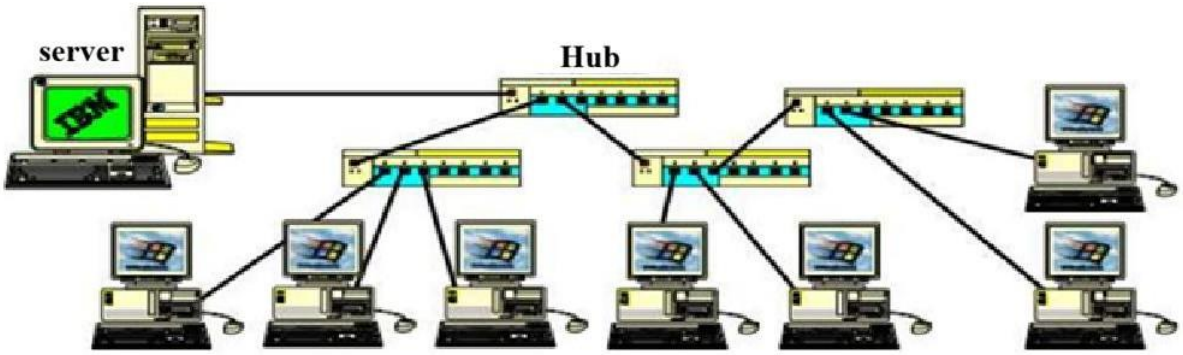



**Figure 1-3: Hierarchical Topology**

- **Ring Topology**

  In a ring topology, each workstation is connected to the next one, forming a closed loop. The transmission medium connects all stations, and data circulates in a single direction. Each station receives the data, regenerates it, and forwards it along the ring. If the data is addressed to a specific station, it is copied during transmission.

  This topology supports high data rates and is suitable for long distances. However, it is vulnerable to link failures, which can disrupt communication. To reduce this risk, a dual-ring structure is commonly implemented.

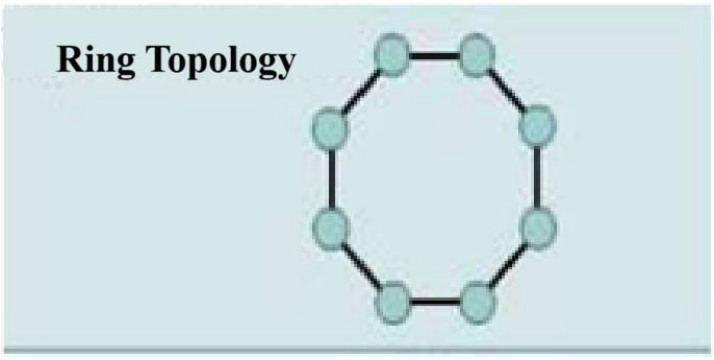



**Figure 1-4: Ring Topology**

- **Mesh Topology (Full and Partial Mesh)**

  In a mesh topology, network stations are interconnected through several alternative paths, allowing data to reach its destination using different routes. This structure provides a high level of reliability and fault tolerance.

  The total number of links required to connect all stations is determined by the following equation:

  **Number of links = N × (N − 1) / 2**

  Where **N** represents the number of stations.

  For instance, a network consisting of 10 stations would require 45 physical connections. Such a design is difficult to implement in practice, which highlights the importance of switching devices. To limit complexity, the number of links can be reduced by using a partial mesh configuration.

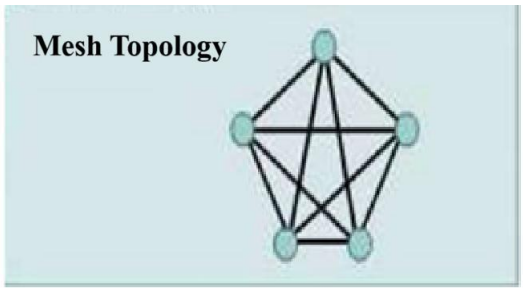



**Figure 1-5: Mesh Topology**

## B. Logical Network Topologies

Logical network topologies define the manner in which data is transmitted between devices within a network. Three main types of logical topologies can be identified:

- **Ethernet Topology:**

Ethernet is based on a logical bus structure and uses a medium access technique known as Carrier Sense Multiple Access (CSMA). Before sending data, a station monitors the transmission medium to ensure that no other station is currently transmitting. If the medium

is busy, data transmission is postponed. Although a station may attempt to transmit at any time, this method does not completely prevent simultaneous transmissions, which may lead to collisions. Ethernet networks standardized under IEEE 802.3 employ this access method, which reduces but does not fully eliminate collision risks.

- **Token Ring Topology:**

  Token Ring technology addresses collision issues by using a token-passing mechanism. A special frame, called a token, circulates through the network, and only the station holding the token is authorized to transmit data. One drawback of this approach is increased latency, as the token circulation time grows with the number of stations and the distance between them.

- **FDDI Topology:**

  FDDI (Fiber Distributed Data Interface) is a local area network technology that operates over optical fiber. It uses a token-based ring architecture combined with error detection and correction mechanisms to provide reliable and efficient data transmission.

# CHAPTER II: LAYERED COMMUNICATION MODEL

## I. PROTOCOL HIERARCHY

In the early development of computer networks, the primary focus was placed on hardware components, while software aspects received less attention. Today, this approach has changed significantly, as software architecture has become highly structured and essential to network operation.

- ### Hierarchical Organization of Protocols

  To simplify network design, communication systems are structured into multiple layers, with each layer built upon the services of the lower one. The number of layers, their names, and their functions differ depending on the network architecture. Each layer is designed to provide well-defined services to the layer above it.

  Layer *N* of a given system communicates logically with layer *N* in another system. The rules and conventions governing this exchange are known as the ***protocol of layer N***.

- ### Definition of a Protocol

  A protocol is a formal set of rules that controls communication between network entities. Any violation of these rules can significantly disrupt communication.

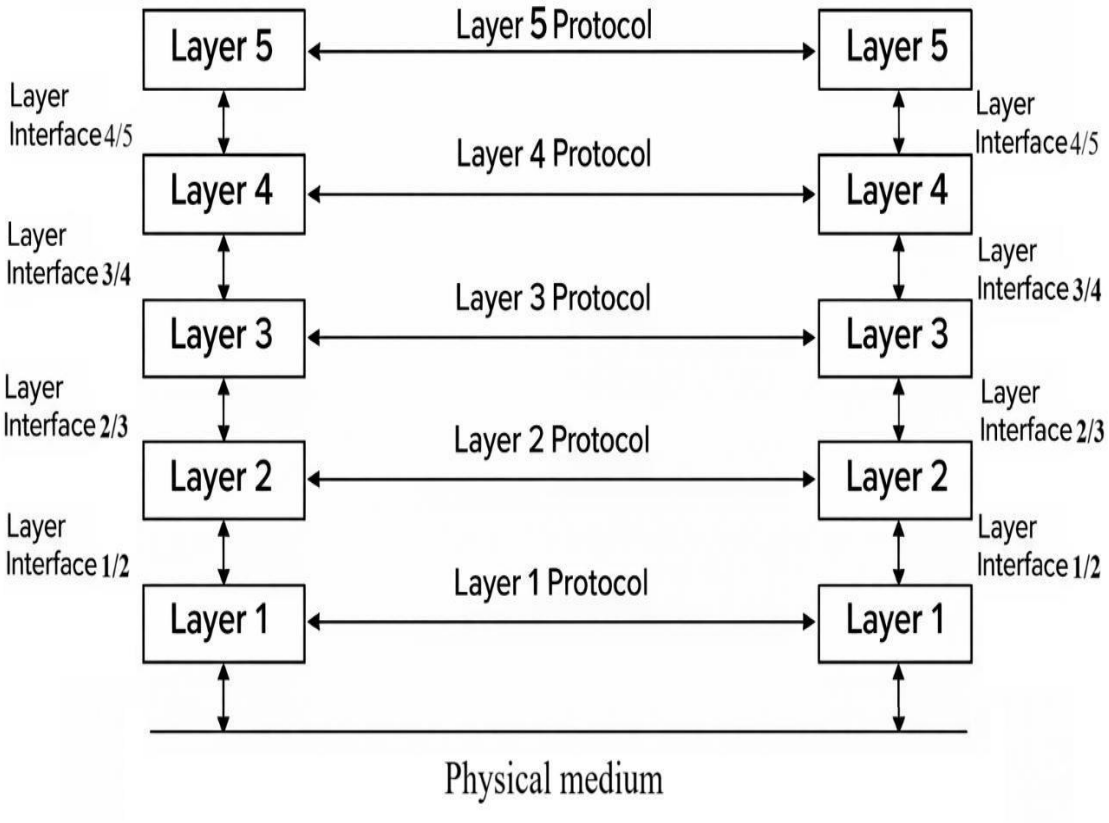



**Fig. 2.1 Communication between two hosts**

Entities that include corresponding layers on different machines are known as ***peer processes*.** These peer processes communicate with each other using protocols.

The layered communication model follows these fundamental principles:

- Data does not move directly from layer *N* of one system to layer *N* of another; instead, it is passed sequentially through each lower layer until it reaches the physical medium.
- The physical medium is responsible for transmitting the actual signals.
- An interface exists between each pair of adjacent layers.
- This interface specifies the operations and services offered by a lower layer to the layer above.
- The complete set of layers and protocols forms the *network architecture.*

The full collection of protocols used within a system is commonly referred to as the ***protocol stack*.**

An example of a layered communication model is illustrated below.

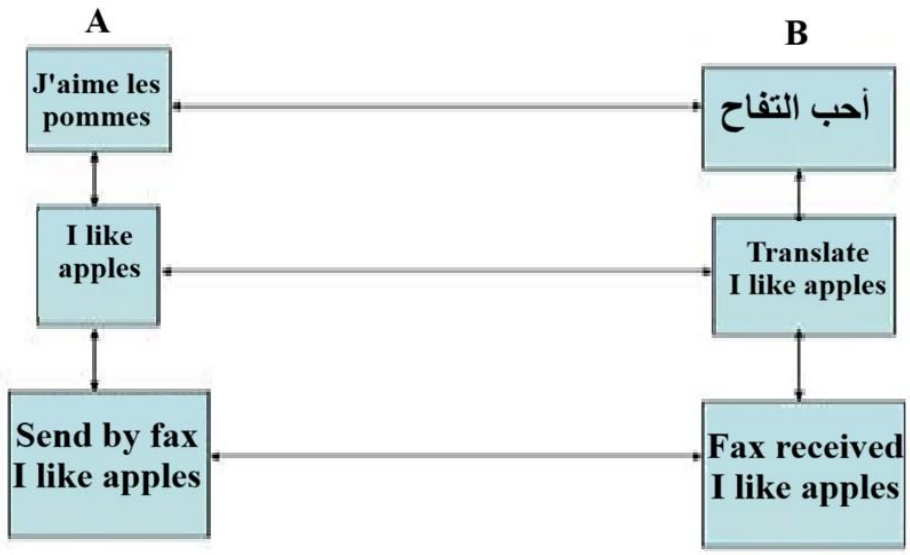



**Figure 2.2: Illustration of Layered Communication between Two Individuals**

A intends to convey an idea to B and transmits the message to a translator via the interface.

The translators first agree on a common reference language, namely English. The selection of this language is considered part of the layer-2 protocol. The translator then forwards the message to a secretary for transmission, for instance via fax, which represents a layer-1 protocol. The secretaries are free to choose electronic mail or any other transmission method without disturbing or notifying the other layers.

In addition, each process may attach information intended exclusively for its peer process. Such information is not transmitted to the upper layer.

A more concrete illustration is provided in the following figure.

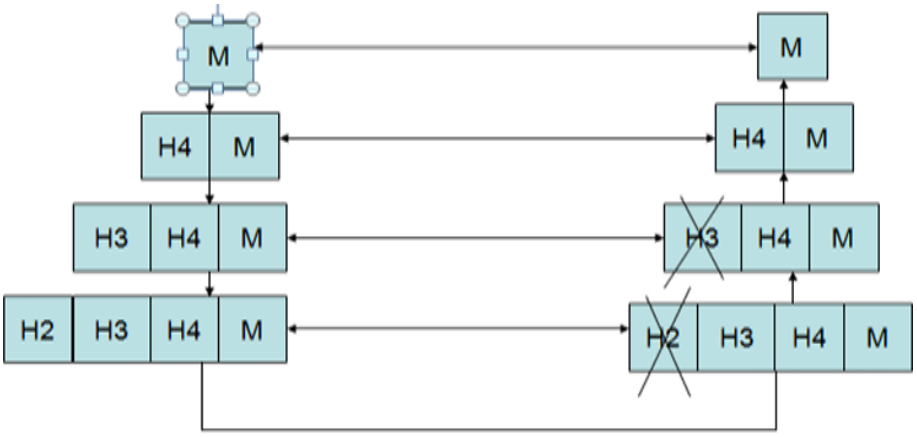



**Figure 2.3: Example of Communication between Two People Using a Layered Model**

# II. LAYER DESIGN PRINCIPLES

- Each layer must provide mechanisms to identify both the transmitting and receiving entities.
- Layers must also define the rules governing data transfer, including communication modes such as simplex and duplex.
- Error handling is a major concern because physical transmission media are inherently imperfect. Various error detection and correction techniques exist, and both ends of the communication must agree on the method to be applied.
- Communication channels do not always maintain the original order of transmitted messages. As a result, protocols must enable the receiver to reorder packets correctly, typically by using sequence numbers.
- A sender operating at a high speed may overload a slower receiver. Therefore, coordination between the sender and the receiver is necessary to control the transmission rate.

# III. PROTOCOLS AND SERVICES

Vertical interaction refers to the exchange of information between adjacent layers within the same system. This interaction is achieved through service primitives.

Horizontal interaction occurs through the exchange of protocol messages across the network between corresponding layers at the same level in different systems. This interaction defines the protocol of layer *N*.

# IV. DATA ENCAPSULATION AND ACCESS POINTS

A protocol data unit at level *N+1*, composed of data and a header, is transmitted within a protocol data unit of level *N*. In this context, level *N+1* data is said to be encapsulated by the level *N* protocol.

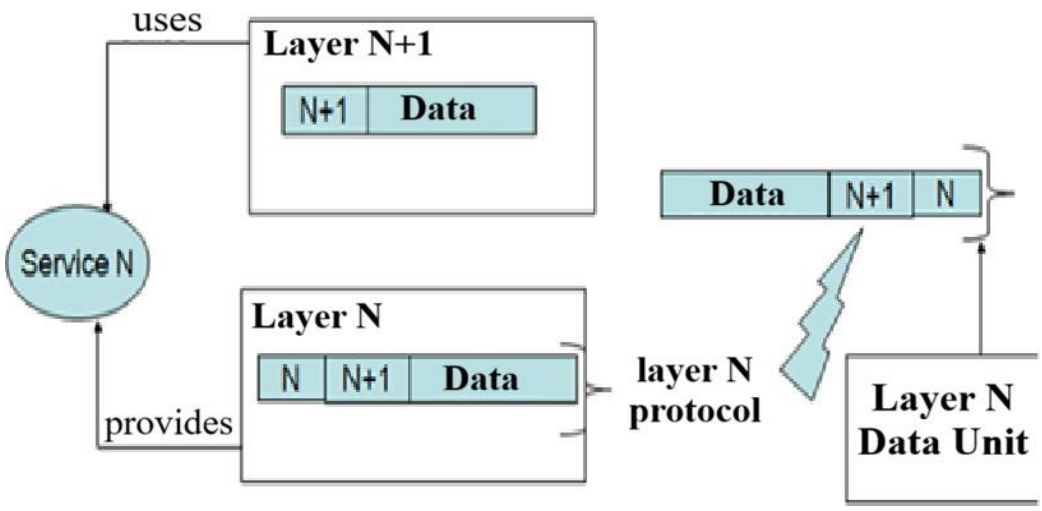



**Figure 2.4: Data Encapsulation and Service Access Point (SAP)**

The services of **layer N** are provided by a **layer-N** entity and are accessed through an interface identified as a **Service Access Point (SAP)**. **SAPs** represent the locations where **layer N+1** can request and use the services offered by **layer N**. Each **SAP** is associated with a unique address.

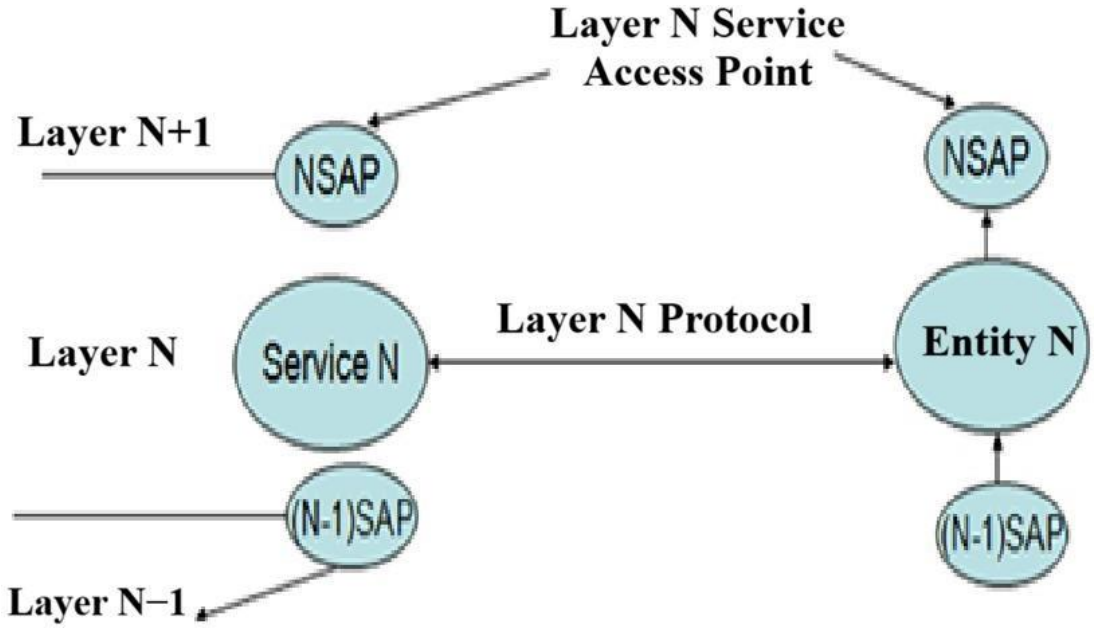



**Figure 2.5: Service Identification**

# V. DATA UNITS PROCESSED

The information handled by a given layer and delivered to its peer entity is referred to as a data unit. When a higher layer (*N+1*) makes use of the services provided by a lower layer (*N*), it submits service data units to that layer, known as **(N) SDUs (Service Data Units)**.

At layer *N*, the received data is considered to be using the services provided at level *(N)*.

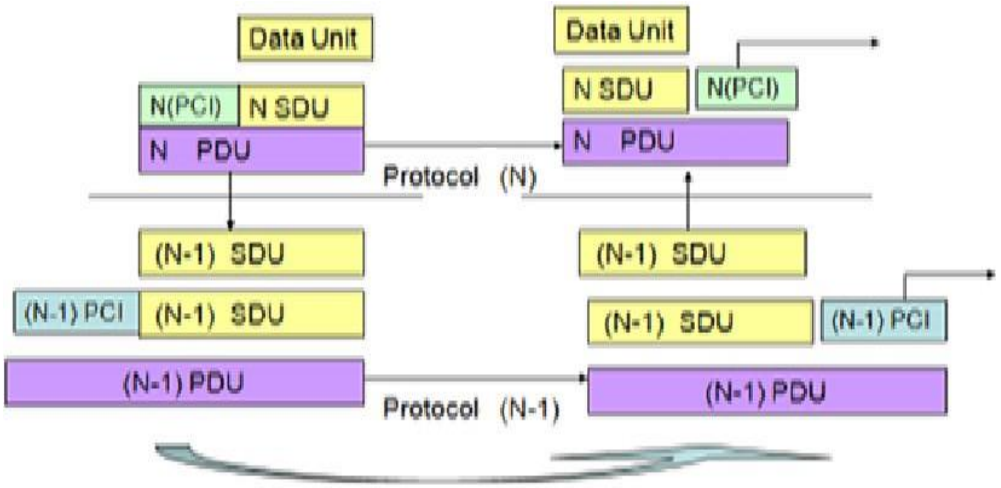



**Figure 2.6: Processed Data Units**

# VI. INTERFACE CONTROL

When a **layer (N)** service is requested, the higher **layer (N+1)** supplies the information necessary for proper handling of the data unit. A portion of this information represents the **PCI**, while the remaining portion is reserved exclusively for the **layer-N** entity. This information defines the local operations to be performed on the data and is known as **Interface Control Information (ICI)**.

The **ICI** is combined with the **SDU** to create an **Interface Data Unit (IDU)**. The **ICI**, which is intended solely for internal use by **layer N**, is not forwarded to the upper layers.

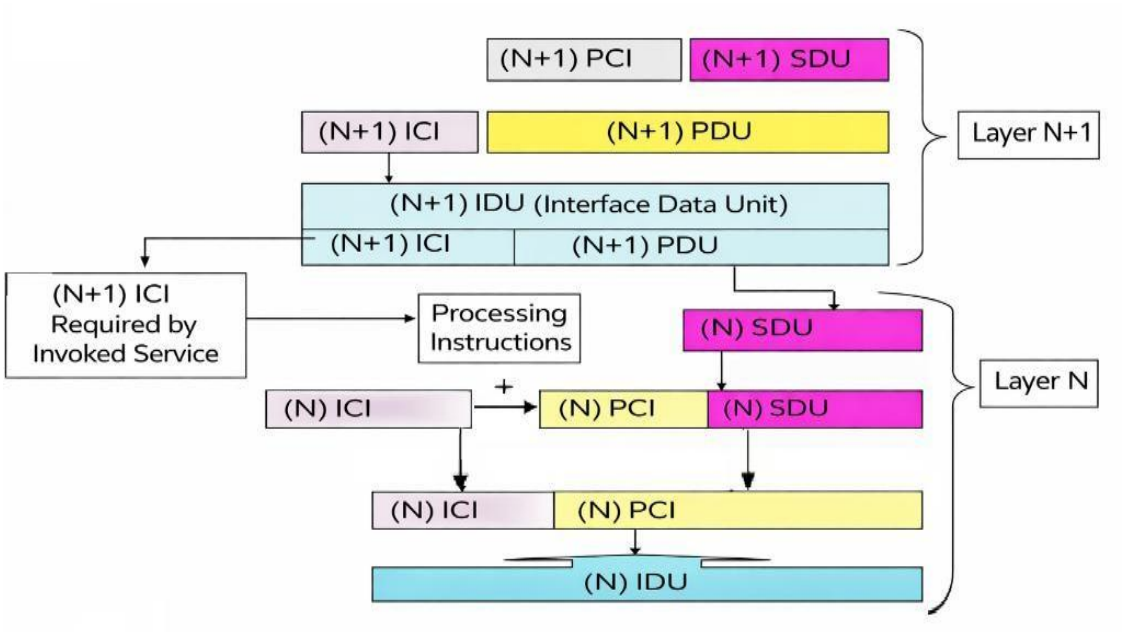



**Figure 2.7: Interface Control**

# CHAPTER III: OSI MODEL AND TCP/IP MODEL

The rapid advancement of computing technologies, combined with the increasing role of information systems, has resulted in a wide variety of networking techniques. As communication needs became more complex, the necessity for a complete and structured communication model, known as a network protocol architecture, became evident. Historically, each major manufacturer developed its own architecture, such as IBM's Systems Network Architecture (SNA) and Bull's Distributed System Architecture (DSA).

These proprietary architectures were incompatible with one another and therefore prevented system interoperability. To address this limitation, a standardized architecture was required. This standardization was achieved by the ISO (International Organization for Standardization) through the definition of a unified communication architecture known as the reference model, or the OSI (Open Systems Interconnection) model.

## I. THE OSI MODEL

The OSI reference model, introduced by the ISO in 1984, is a conceptual framework used to describe network communication. It provides a common set of standards that ensure improved compatibility and interoperability among diverse network technologies developed by various vendors worldwide.

The OSI model was designed to harmonize communication systems between computers within a network. Its primary objectives include:

- minimizing system complexity,
- standardizing communication interfaces,
- supporting modular system design,
- enabling faster technological development.

The OSI model follows a layered architecture, meaning that it is divided into multiple layers, each responsible for specific functions. Together, these layers enable structured and efficient communication between networked systems.


**Figure 3-1: The OSI Model Layers**

**The seven-layer structure allows a clear distinction between:**

- **Lower layers (1–4):** Dedicated to the transmission of information through transport-related mechanisms.
- **Upper layers (5–7):** Dedicated to information processing through application-oriented services.

# II. FUNCTIONS OF THE OSI MODEL LAYERS

## A. Physical Layer

The physical layer is responsible for the transmission of signals between networked systems. It ensures the direct transfer of raw bits over a communication medium. This layer specifies the mechanical, electrical, and operational characteristics necessary to establish and maintain physical connections for binary data exchange between interconnected entities.

The physical layer includes all hardware components required for reliable bit transmission, such as:

- physical connectors and interfaces, including voltage levels, data rates, and wiring specifications,
- modems used to perform signal modulation and demodulation,
- Hubs, which aggregate multiple communication channels from different devices into a single shared link toward a remote destination.

## B. Data Link Layer

The data link layer handles the delivery of information units called **frames** across the physical medium. It provides mechanisms to clearly determine the **frame boundaries (**the start and end of a frame), using a standard bit sequence **01111110**.This sequence serves as a **synchronization marker**, also referred to as a **frame flag** or **delimiter**.

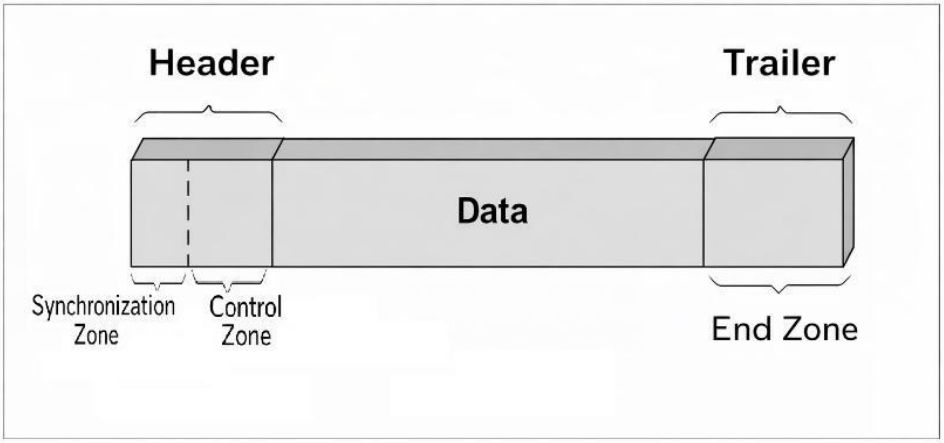



**Figure 3-2: Structure of a Standard Frame**

This layer also carries out several additional important functions, which can be summarized as follows:

- **Error detection and correction:**
  Since errors may arise at the physical layer, the data link layer incorporates mechanisms to detect errors when a data block is received and to correct most of them, , thereby ensuring that the residual error rate remains minimal.
- **Regulation of access to the shared physical medium:**
  The data link layer defines the rules that control access to a single physical medium shared by multiple stations. Typical access mechanisms include **CSMA/CD** for wired networks and **CSMA/CA** for wireless networks.
- **Use of MAC addressing:**
  This layer relies on **MAC (Medium Access Control) addresses** to identify network interfaces and manage access to network stations.

## C. Network Layer

The network layer ensures the proper delivery of data packets to their destination by selecting an efficient routing path. When the source and destination are located on different networks, this layer forwards the data to an appropriate gateway.

At the network layer, information is encapsulated into packets known as **datagrams**. IP packets consist of data from higher layers combined with an **IP header**.

The contents of this header are described in the following figure.

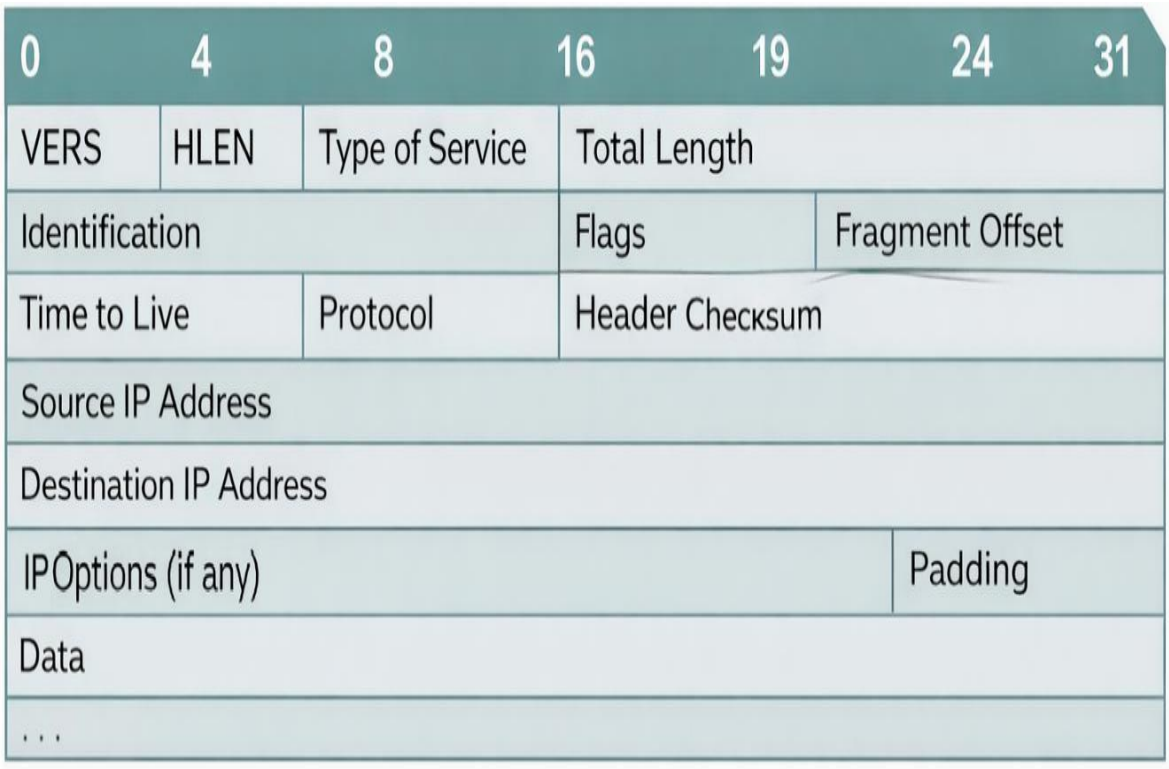



**Figure 3-3: IP Packet Structure**

## IP Header Fields Description

- **Version (VERS):**

  This field specifies the format of the datagram or the version of the IP protocol being used. It allows different protocol versions to coexist within intermediate systems. The currently deployed version is **IPv4**, while **IPv6** is gradually being adopted across the Internet.

- **IP Header Length (HLEN – 4 bits):**

  This field indicates the length of the IP header expressed in 32-bit words. It represents the total header size, including variable-length fields. When no options are present, the header length is set to 5, corresponding to 20 bytes.

- **Type of Service (ToS – 8 bits):**

  This field indicates the priority level assigned by a higher-layer protocol. such as best-effort service, error-free delivery, and real-time traffic.

- **Total Length (16 bits):**

  This field defines the total size of the IP packet in bytes, including both header and data. The payload size can be obtained by subtracting the header length from this value. The theoretical maximum size is 65,536 bytes, although the actual limit is constrained by the network's **MTU (Maximum Transmission Unit)**.

- **Identification (16 bits):**

  This field uniquely identifies each datagram and contains a value assigned by the source. In the event of fragmentation, the same identification value is included in all fragments to ensure correct reassembly.

- **Flags (3 bits):**

  The first bit is unused. The **DF (Don't Fragment)** bit, when set to 1, instructs intermediate systems not to fragment the datagram. This is used when the destination system cannot reassemble fragments. The **MF (More Fragments)** bit is set to 1 in all fragments except the last one, indicating that additional fragments follow.

- **Fragment Offset (13 bits):**

  This field indicates the position of a fragment within the original datagram, measured in units of 8 bytes. Consequently, all fragments except the final one have sizes that are multiples of 8 bytes.

- **Time To Live (TTL):**

  This field defines the maximum number of routers a packet may traverse. The value is reduced by one at each hop, and when it reaches zero, the packet is discarded to prevent endless circulation. The router that drops the packet sends an **ICMP** notification to the sender.

- **Protocol (8 bits):**

  This field identifies the upper-layer protocol, such as **TCP** or **UDP**, that should receive the packet once IP processing is complete.

- **Header Checksum (16 bits):**

  This field provides error detection for the IP header only, ensuring its integrity during transmission.

- **Source Address (32 bits):**

  This field specifies the IP address of the node that originated the packet.

- **Destination Address (32 bits):**

  This field contains the IP address of the receiving host.

- **Options:**

  This field supports optional IP features, including security-related functions. Its variable length requires the use of padding bits.

- **Padding:**

  Padding bits are added to ensure that the IP header length remains a multiple of 32 bits.

- **Data:**

  This field contains information from higher layers. Its length is variable.

**Functions of the Network Layer**

The network layer performs two main functions:

- **Routing:**

  Routing ensures that data packets are forwarded toward their destination through a network of routers.

- **Addressing:**

  The **Internet Protocol (IP)** is the primary routed protocol on the Internet. IP addresses enable packets to be delivered from a source to a destination by selecting the most efficient path.

## D. Transport Layer

The transport layer provides end-to-end data transmission, potentially across multiple underlying networks. It also optimizes the use of network resources by multiplexing several communication sessions over the same virtual circuit or route.

The primary function of the transport layer is to divide messages into packets and then reassemble them at the destination to reconstruct the original message. This process is known as **segmentation and reassembly**.

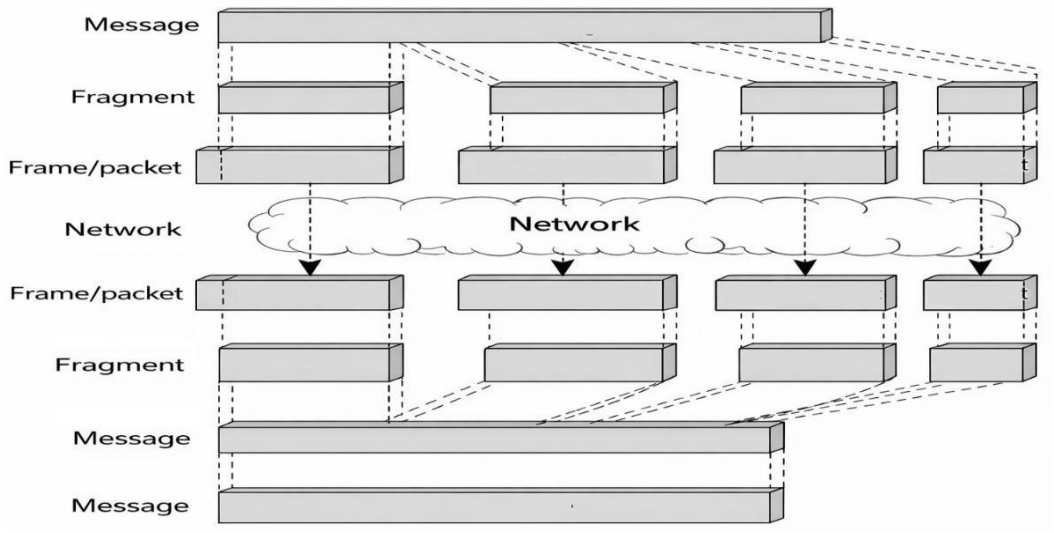



**Figure 3-4: Message Fragmentation and Reassembly at the Transport Layer**

In addition, layer-4 protocols incorporate flow control, congestion control, and resynchronization mechanisms.

## E. Session Layer

The session layer provides the mechanisms required to organize and synchronize the dialogue between communicating users. This layer allows sessions to be established, managed, and terminated between applications.

## F. Presentation Layer

- The presentation layer is responsible for the representation of data exchanged between application entities. It ensures that information is properly structured and formatted so that it can be correctly interpreted by the receiving system. Its main functions include:
- ensuring data readability for the destination system,
- defining data formats,
- organizing data structures,
- negotiating the data transfer syntax used by the application layer.

## G. Application Layer

The application layer is the highest layer of the OSI reference model. It provides services directly to application processes, including:

- **MHS (Message Handling System):** Electronic messaging services operating without a connection.
- **DS (Directory Service):** Directory services that organize and provide access to network resources and addressable entities.

- **FTAM (File Transfer, Access, and Management):** Services that support file transfer and remote file management.
- **VT (Virtual Terminal):** A virtual terminal service that allows users to interact with a remote system as if it were locally hosted.

# III. PROTOCOL DATA UNITS (PDUs) OF THE OSI MODEL

A **Protocol Data Unit (PDU)** is the information unit associated with the protocol of each OSI layer. The PDUs corresponding to the different layers of the OSI model are illustrated in the following figure.

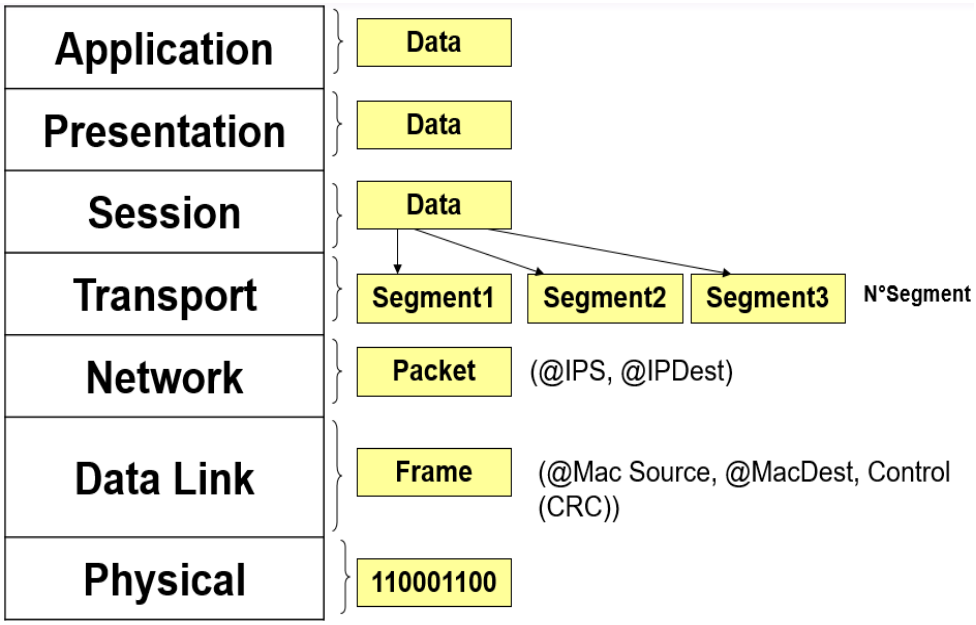



**Figure 3-5: PDUs Associated with Each OSI Layer**

# IV. TCP/IP MODEL

The TCP/IP reference model was introduced by the United States Department of Defense to design a network capable of remaining operational under all circumstances, including extreme conditions such as nuclear warfare. In a global environment relying on diverse communication media—such as copper wiring, microwave transmission, optical fiber, and satellite links—the Department of Defense required a packet-based communication system that could guarantee data delivery regardless of network conditions. This demanding requirement led to the development of the TCP/IP model.

TCP/IP was designed and implemented as an **open standard**.

The TCP/IP model is composed of the following four layers:

- Application layer
- Transport layer
- Internet layer
- Network access layer

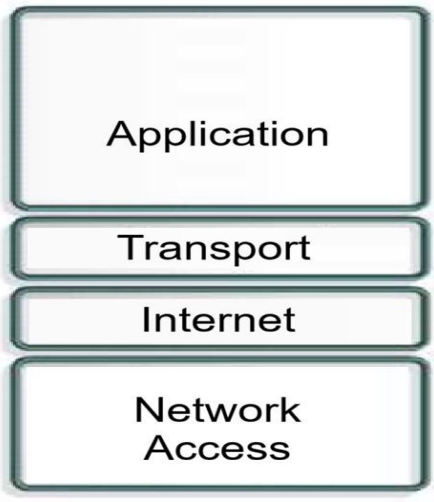



**Figure 3-6: The TCP/IP Model**

Although some layers of the TCP/IP model share the same names as those of the OSI model, they do not correspond exactly. It should be noted that the application layer performs different functions in each model.

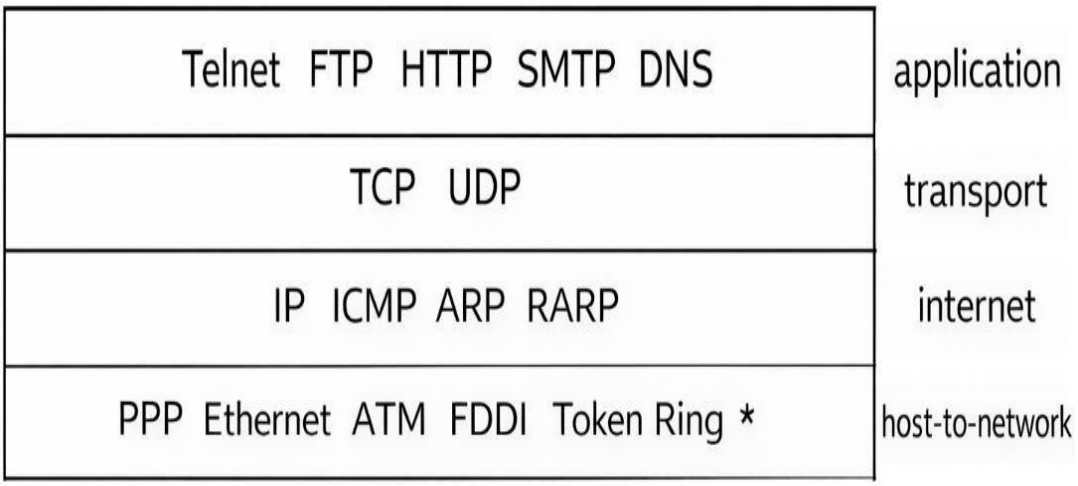



**Figure 3-7: Protocols Associated with the Layers of the TCP/IP Model**

## IV-1 Protocols of the Application Layer

- **FTP (File Transfer Protocol):** A protocol designed for the exchange of files between networked systems.
- **Telnet:** A protocol that allows remote access and command-line session control.
- **HTTP (HyperText Transfer Protocol):** A protocol used for the transfer of hypertext-based resources.
- **SMTP (Simple Mail Transfer Protocol):** A protocol responsible for the transmission of electronic mail.
- **DNS (Domain Name System):** A naming system that maps domain names to their corresponding IP addresses.

## IV-2 Transport Layer Protocols

The transport layer provides end-to-end data delivery. Transport protocols operate exclusively at the endpoints of communication to provide flow control functions, such as data segmentation, retransmission of segments, and management of segment sequence numbers. The most commonly used protocols at the transport layer are:

- **TCP (Transmission Control Protocol)**
- **UDP (User Datagram Protocol)**

➢ **TCP (Transmission Control Protocol):**

provides reliable and flexible network communication with efficient data flow and a low error rate. TCP is a connection-oriented protocol. It maintains a communication session between the source and destination hosts while organizing application-layer data into units known as **segments**.

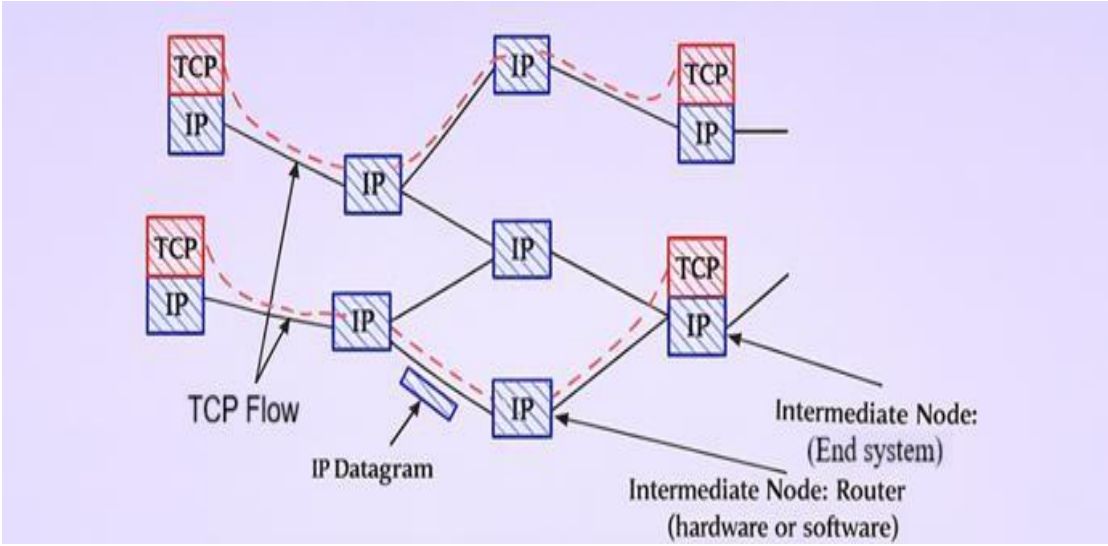



**Figure 3-8: Application Communication at the Transport Layer**

➢ **UDP (User Datagram Protocol):**

Similar to TCP, UDP performs data segmentation and reassembly. However, it does not offer additional reliability or control mechanisms. There is no flow control, no acknowledgment-based communication tracking, and no packet reordering. For these reasons, UDP is considered a connectionless transport protocol.

## IV-3 Internet Layer Protocols

The purpose of the Internet layer is to encapsulate TCP and UDP segments into IP packets and deliver them across heterogeneous networks. Packets may follow different routes and still reach the destination network successfully.

The protocol that defines this layer is the **Internet Protocol (IP)**. Functions such as route selection and packet forwarding are carried out at this level to ensure proper data delivery.

IP operates as a best-effort interconnection protocol. The figure below explains its behavior in the network.

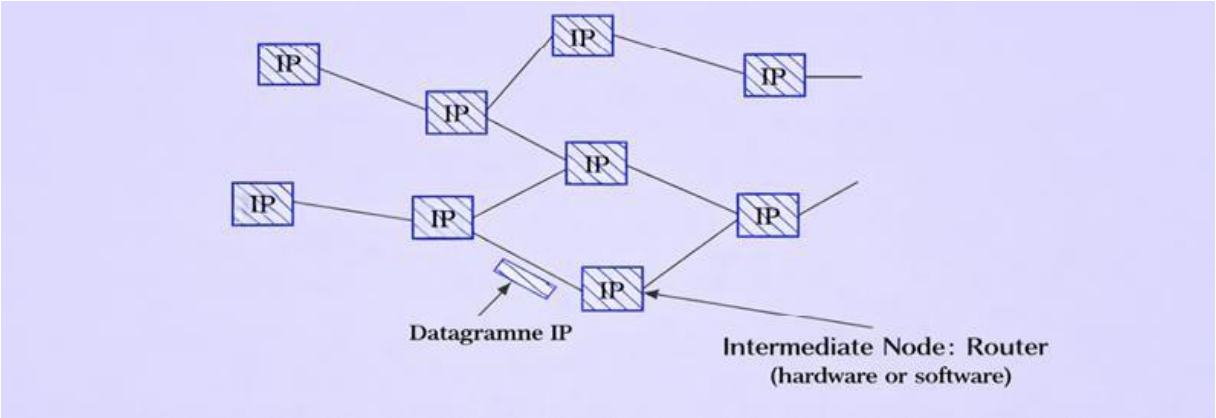



**Figure 3-9: Host-to-Host Communication at the Network Layer**

## IV-4 Network Access Layer Protocols

The network access layer, also known as the host-to-network layer, covers all the physical and logical components required to establish a physical connection. It includes details related to network technologies, as well as all functions associated with the physical and data link layers of the OSI model.

# V. TCP/IP MODEL PROTOCOLS AND CONNECTION PORTS

Within the TCP/IP protocol suite, message communication may occur across multiple concurrent sessions. Therefore, the TCP/IP model must be able to distinguish and manage messages originating from different sessions. This functionality is achieved through the use of connection ports.

A connection port, also called a service port, is a virtual identifier that uniquely represents an application at the software level. Port numbers are represented using 16 bits. During communication between two hosts, the source and destination port numbers must be clearly specified and associated with the corresponding source and destination IP addresses. The combination of an IP address and a port number is referred to as a **socket**.

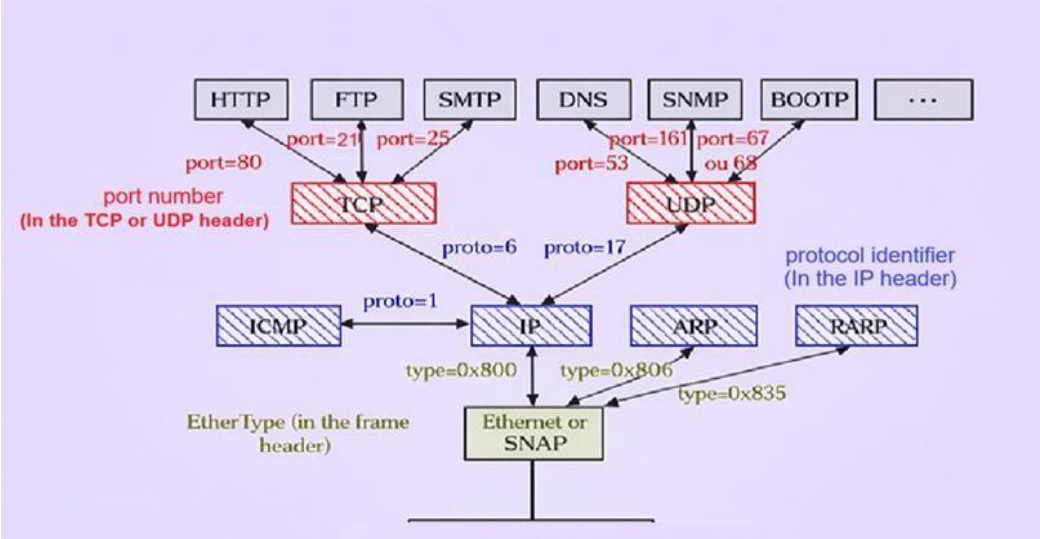



**Figure 3-10: TCP/IP Model Protocols and Connection Ports**

# VI. TCP/IP MODEL AND HOST-TO-HOST COMMUNICATION

**Homogeneous Hosts**

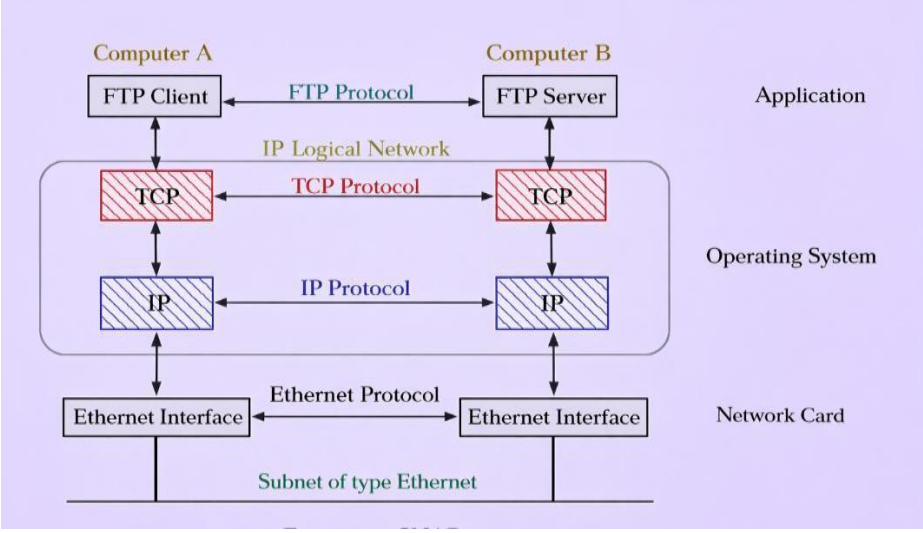



**Figure 3-11: Communication between Homogeneous Hosts**

**Heterogeneous Host**

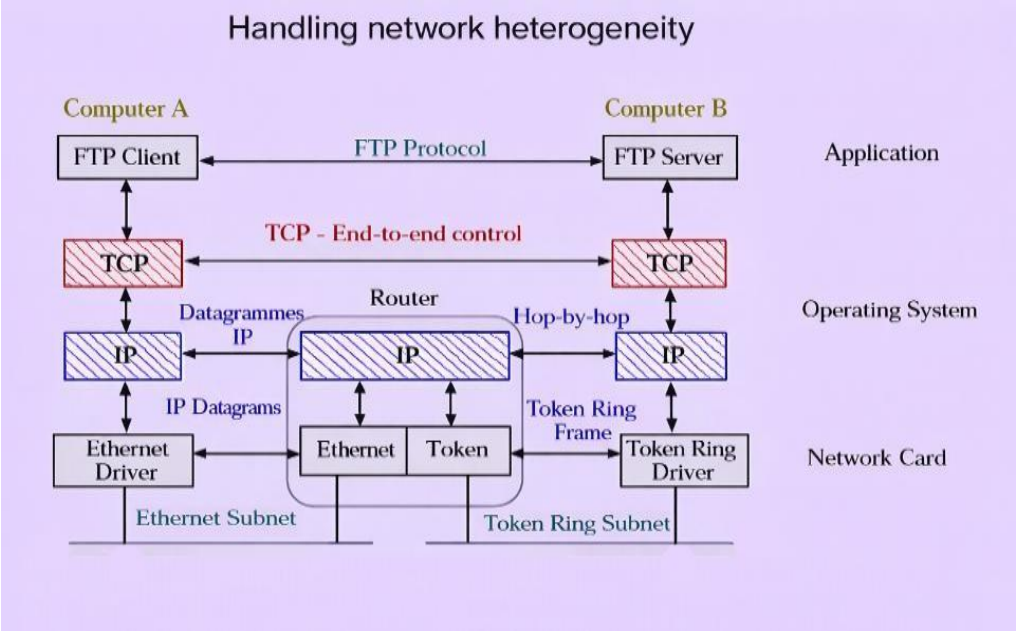



**Figure 3-12: Communication between Heterogeneous Hosts**

# VII. PRACTICAL NETWORK TESTING COMMANDS

The **ping** command operates by sending specific IP packets, called **ICMP (Internet Control Message Protocol) datagrams**, to send echo request messages to a specified destination. Each transmitted packet represents a request for a reply. The returned response indicates the success rate and the round-trip time between the source and destination devices.

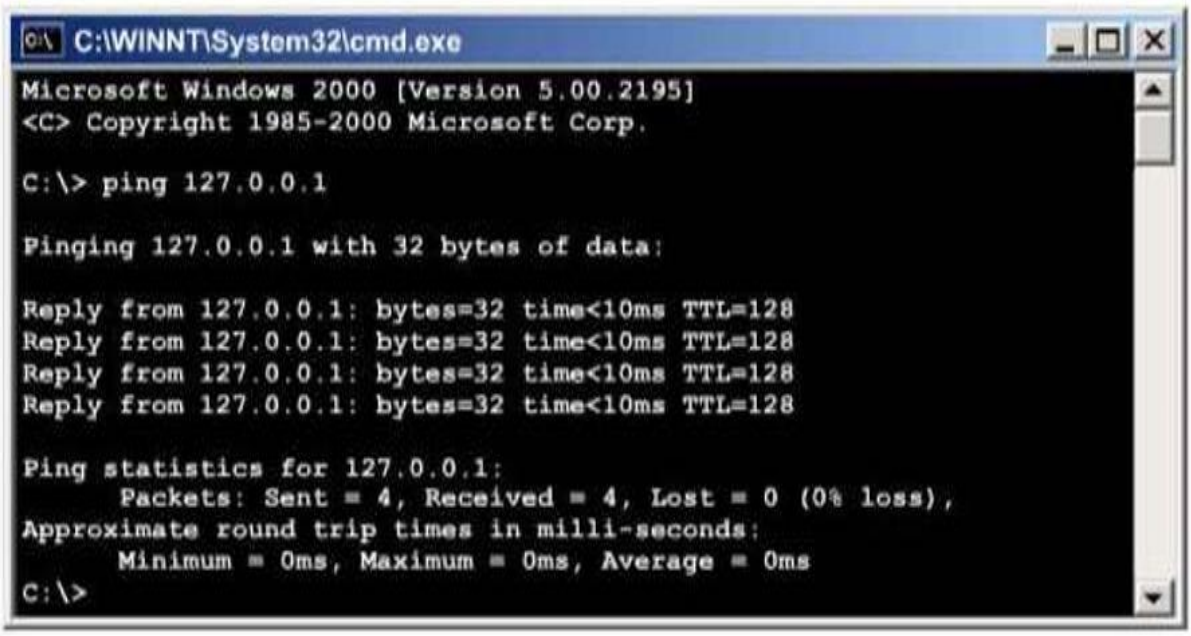



**Figure 3-13: Use of the Ping Command**

### Typical Uses of the Ping Command

The **ping** utility may be applied in several situations, including:

- **ping 127.0.0.1** –

  This command tests the local loopback interface and is used to confirm that the TCP/IP configuration is functioning correctly.

- **ping <host IP address>** –

  When sent to a host on the network, this command verifies the host's TCP/IP address configuration and checks connectivity between the local system and the remote host.

- **ping <IP address of the default gateway>** –

  Sending a ping request to the default gateway allows verification of the accessibility of the router that connects the local network to external networks.

# CHAPTER IV: INTERCONNECTION DEVICES

Interconnection devices are used to connect hosts within a network. Depending on their function, these devices can be classified according to specific layers of the OSI model.

## I. REPEATERS

A repeater is a device used to regenerate an analog or digital signal that has been distorted due to transmission loss caused by attenuation. Repeaters provide a physical connection between two segments of the same logical network.

A repeater may provide either media extension or media adaptation between two networks. An example of such adaptation is the conversion from twisted-pair cabling to optical fiber.

By allowing additional hosts to be connected, repeaters increase overall network traffic, which may lead to a higher number of collisions within the network.

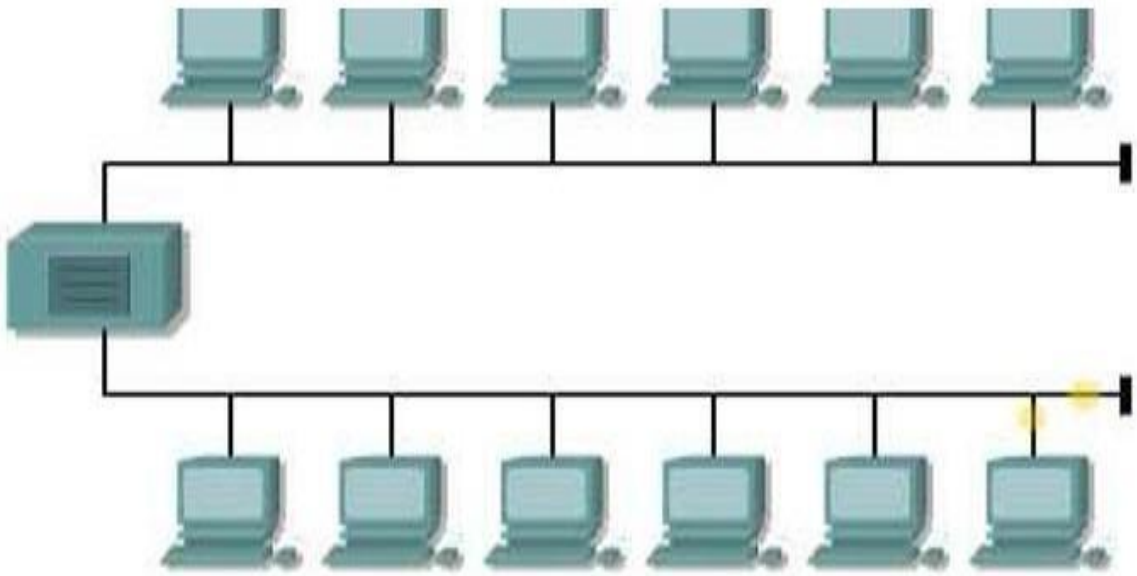


**Figure 4-1: Repeater Connecting Two Network Segments**

Repeaters operate within a single **collision domain**. A collision domain is a shared environment in which any disturbance or fault occurring in one segment affects the entire domain.

The so-called **"5-4-3-2-1" rule** specifies the conditions that must be respected to ensure proper network operation:

- **Five network media segments**
- **Four repeaters or hubs**
- **Three segments connected to hosts**
- **Two link segments without hosts**
- **One large collision domain**

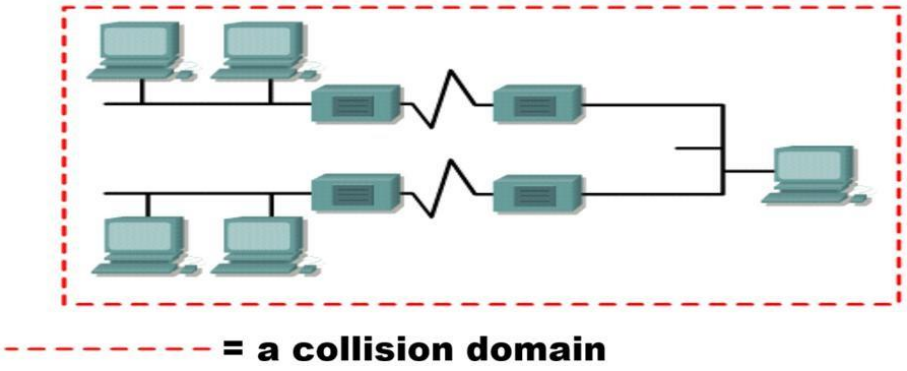



**Figure 4-2: Application of the 5-4-3-2-1 Rule to the Collision Domain**

# II. BRIDGES

A large Local Area Network (LAN) must be divided into several smaller segments in order to simplify network management and reduce the collision rate.

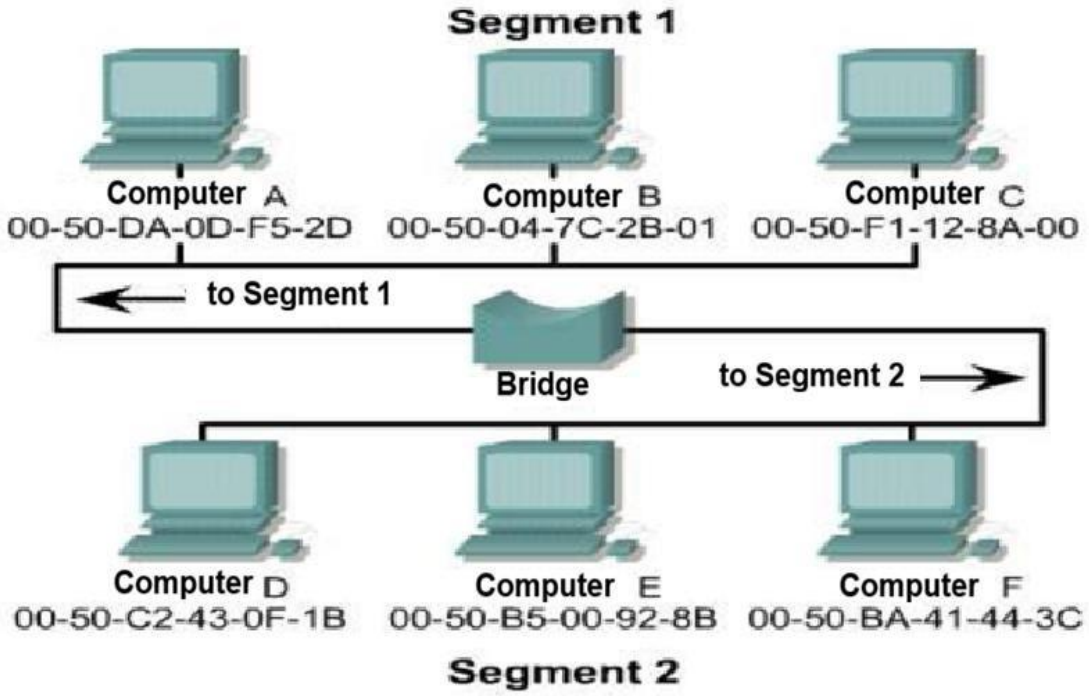



**Figure 4-3: Network Segmentation Using a Bridge**

Using a bridge makes it possible to reduce traffic within a LAN while increasing its geographical reach. Bridges operate at the **data link layer** of the OSI model. They provide data rate or media adaptation between similar networks (Ethernet/Ethernet or Token Ring/Token Ring) as well as between dissimilar networks (Ethernet/Token Ring).

Bridges and switches forward data across network segments using **MAC addresses**.

**Filtering:**

If the destination device is located on the same segment as the frame, the bridge does not forward the frame to other segments. This process is known as filtering.

**Forwarding:**

If the destination device is located on a different segment, the bridge forwards the frame to the appropriate segment. This process is referred to as forwarding.

**Flooding:**

If the bridge does not recognize the destination address, it forwards the frame to all segments except the one on which the frame was received. This process is called flooding.

A bridge divides a network into multiple collision domains, but it does not affect the broadcast domain. If a network does not include any device operating at layer 3, such as a router, the entire network shares the same logical broadcast address space.

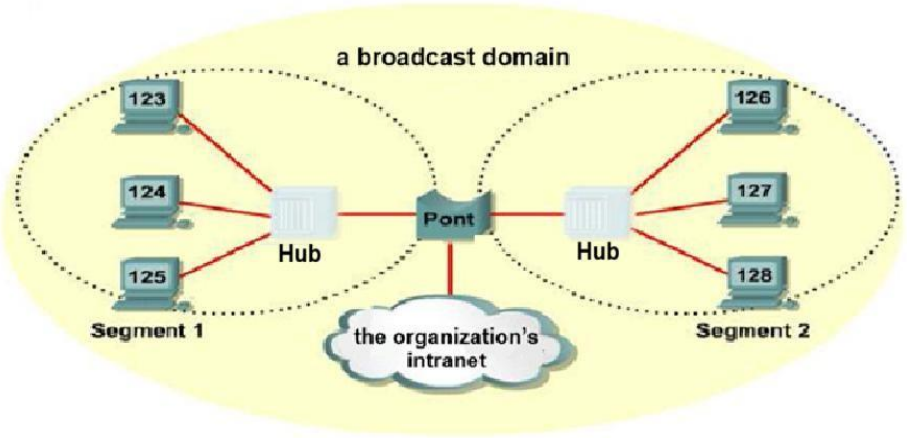



**Figure 4-4: Collision Domain and Broadcast Domain**

# III. SWITCHES

Switches are also referred to as **multiport bridges**. In contrast to traditional bridges, switches record the MAC addresses of received data frames in a **CAM (Content Addressable Memory)**.

Using these stored addresses, switches construct switching tables that enable them to identify the appropriate destination for each frame. These tables include both the port number and the MAC address of the device associated with that port.

| interface | MAC address |
|---|---|
| **E0** | **0260.8c01.1111** |
| **E1** | **0260.ec01.2222** |
| **E2** | **0260.ec01.3333** |
| **E3** | **0260.8c01.4444** |

**Figure 4-5: Switch Forwarding Table**

Switching is a networking technology designed to reduce congestion in Ethernet LANs. Switches can effectively replace hubs because they operate over the same existing cabling infrastructure.

## III-1 Switching Modes

### 1. Cut-Through Mode

In cut-through switching, a switch begins forwarding a frame immediately after identifying the destination MAC address in its switching table.

This mode is characterized by very low latency. In cut-through switching, the data rates of the source and destination ports must be identical in order to avoid frame corruption. This type of switching is referred to as **symmetric switching**.

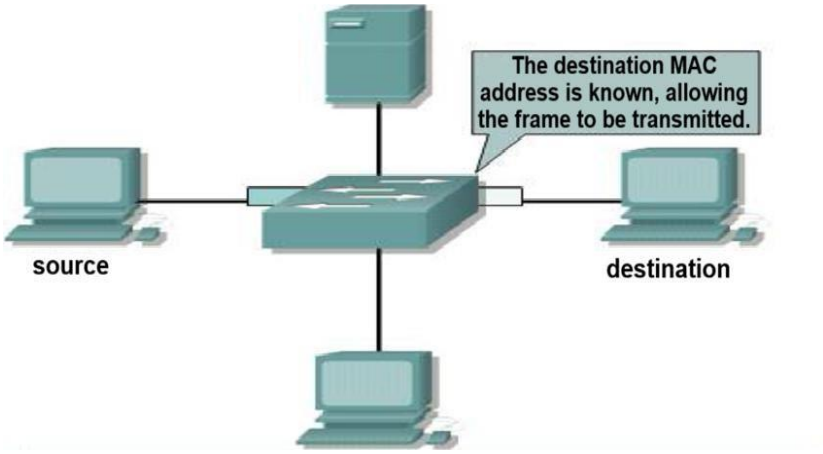



**Figure 4-6: Cut-Through Switching Mode**

### 2. Store-and-Forward Switching

In this switching mode, the switch must receive the entire frame, store it temporarily, and then forward it to the destination port. This process allows the switch to verify the **Frame Check Sequence (FCS)**. If the frame is found to be invalid, it is discarded. In 'Store-and-Forward' mode, the transmission and reception rates of the frame must be different.

This type of switching is called asymmetric switching.

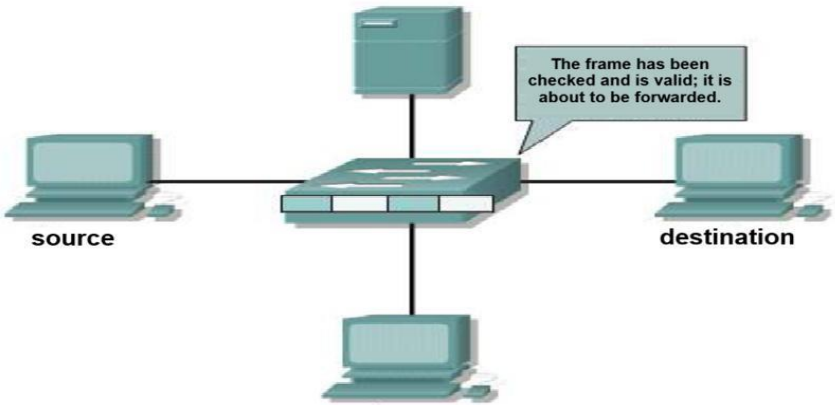



**Figure 4-7: Store-and-Forward Switching Mode**

### 3. Fragment-Free Switching

Fragment-free switching is a hybrid switching technique positioned between cut-through and store-and-forward modes. It allows the switch to read the first **64 bytes** of a frame, including the frame header, and to start forwarding the frame before the entire frame has been received. This approach checks the validity of address information and **LLC-related fields** to ensure that frames are properly processed and forwarded to the correct destination.

**Note:**

Switches operate at significantly higher speeds than bridges and can support advanced features such as **Virtual Local Area Networks (VLANs)**. As a result, switches are often considered to provide certain **Layer 3 capabilities**.

## III-2 Broadcast Domain

A **broadcast domain** consists of a set of collision domains interconnected by **Layer 2 devices**. Broadcast frames are propagated through all bridges and switches within the network. When a node needs to communicate with every host on the network, it sends a broadcast frame using the destination MAC address **0xFFFFFFFFFFF**, which is received by all hosts.

An excessive number of broadcast frames can significantly reduce the overall performance of a **LAN.**

**Note:**

Devices operating at Layers 1 and 2 cannot block broadcast traffic. Such traffic is controlled at Layer 3 by routers.

# IV. ROUTERS

Routers are networking devices responsible for making routing decisions necessary for communication between different networks. They can connect to wide area networks and interconnect local area networks separated by large geographical distances. Routers are also capable of regenerating signals, aggregating multiple connections, converting data transmission formats, and managing data forwarding

# CHAPTER V: TRANSMISSION MEDIUM

# I. TYPES OF CABLING

Copper transmission media represent the most widely used cabling solutions for data communication in computer networks. Among these media, the following types are commonly employed:

## I.1 Coaxial Cable

A coaxial cable is composed of a cylindrical structure that contains two conductive elements. The inner conductor is a copper core that carries the data signals and is surrounded by a flexible insulating layer. Around this insulation, a braided copper shield forms the second conductor, which helps reduce interference and external noise. The cable is protected by an outer rubber jacket.

Coaxial cables are low-cost, easy to install, and suitable for relatively long transmission distances, supporting data rates of up to **10 Mbps**. They typically use **BNC connectors** to interface with network interface cards.

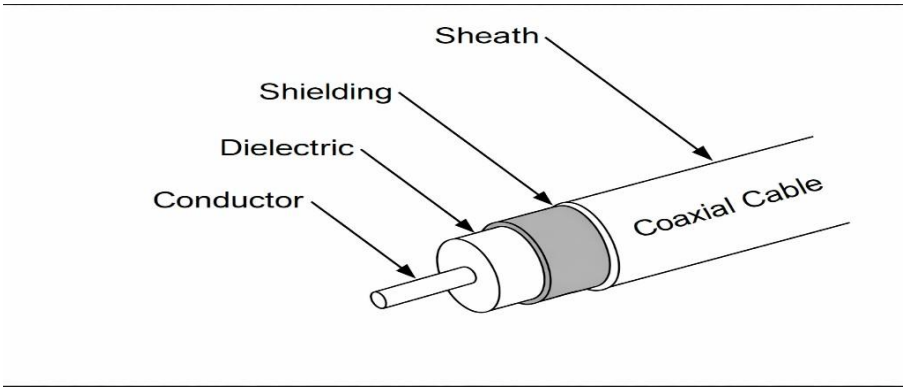



**Figure 5-1: Coaxial Cable**

Two main categories of coaxial cables can be identified:

- **10Base5 thick coaxial cable (Thicknet, Thick Ethernet, also known as Yellow Cable):**
  This cable has a diameter of about **12 mm**. Due to its rigidity, it is difficult to install and handle. It is mainly used to interconnect network segments or subnetworks. The maximum length of a single segment is **500 meters**.

- **Thin coaxial cable 10Base2 (Thin Ethernet):**

  This cable has a diameter of approximately **6 mm**. It is highly flexible and allows direct connection between computers, and it is used to connect computers to hubs, repeaters, or directly to other computers. The maximum segment length is about **185 meters**. It uses **BNC T-connectors**.

**Figure 5-2: BNC Connector**

## I.2 Twisted Pair Cable

Twisted pair cabling provides higher bandwidth and allows data rates beyond **10 Mbps**. It operates in **full-duplex** mode. **Unshielded Twisted Pair (UTP)** cabling, commonly used in Ethernet local area networks, consists of **four pairs of color-coded wires** that are twisted together and enclosed within a flexible plastic jacket.

The color coding allows easy identification of individual pairs and conductors, which facilitates cable installation and termination. The twisting of the wire pairs reduces electromagnetic interference between adjacent pairs, a phenomenon referred to as **crosstalk.**

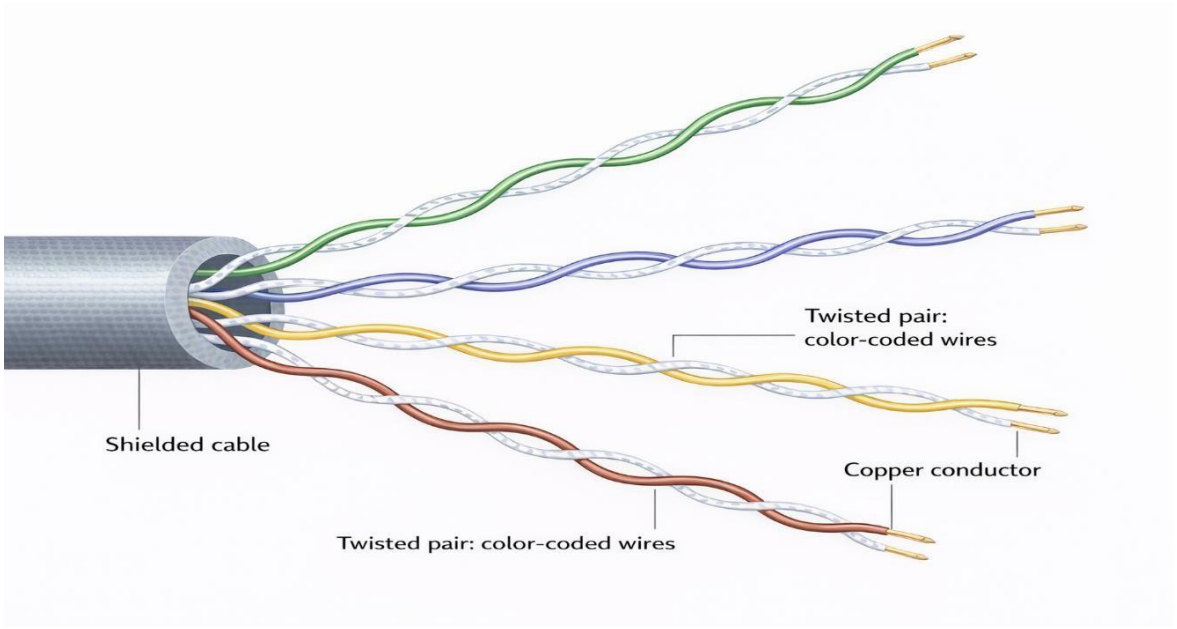



**Figure 5-3: Twisted Pair Cable**

- **UTP (Unshielded Twisted Pair) Standards:**

  This type of cabling follows the **10Base-T** specification and defines the number of wire twists per foot (33 cm) based on the intended application. It is very sensitive to electromagnetic interference but can still support data rates of up to **100 Mbps**.

- **STP (Shielded Twisted Pair):**

  This cable incorporates metallic shielding around each twisted pair. It enables higher data rates and longer transmission distances compared to UTP cabling.

- **FTP (Foiled Twisted Pair):**

  In this type of cable, shielding is provided by a thin aluminum foil that surrounds the twisted pairs.

- **S-FTP (Shielded Foiled Twisted Pair):**

  This cable offers very high resistance to interference but is more expensive and more difficult to install than unshielded twisted pair (UTP) cable. It consists of shielding for the entire cable as well as shielding for each cable pair.

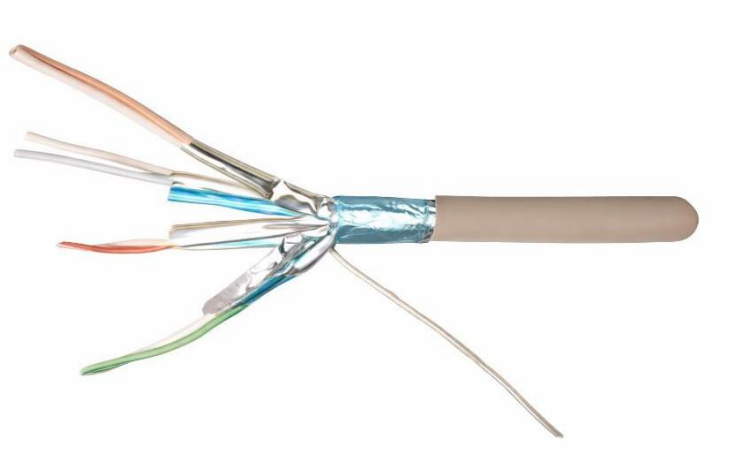


**Figure 5-4: Shielded Foiled Twisted Pair (S-FTP) Cable**

These standards have evolved to support data rates ranging from 1 to 10 gigabits per second over copper cabling.

- **UTP cabling**, extensively deployed in professional workplaces, educational facilities, and residential environments, complies with standards standards jointly defined by the **TIA (Telecommunications Industry Association)** and **the EIA (Electronics Industries Alliance)**. The **TIA/EIA-568A** and **TIA/EIA-568B** specifications are presented in the following figure.

| EIA/TIA-568A | | |
|---|---|---|
| 1 | T3 | White - Green |
| 2 | R3 | Green |
| 3 | T2 | White - Orange |
| 4 | R1 | Blue |
| 5 | T1 | White - Blue |
| 6 | R2 | Orange |
| 7 | T4 | White - Brown |
| 8 | R4 | Brown |

| EIA/TIA-568B | | |
|---|---|---|
| 1 | T2 | White-Orange |
| 2 | R2 | Orange |
| 3 | T3 | White - Green |
| 4 | R1 | Blue |
| 5 | T1 | White - Blue |
| 6 | R3 | Green |
| 7 | T4 | White - Brown |
| 8 | R4 | Brown |

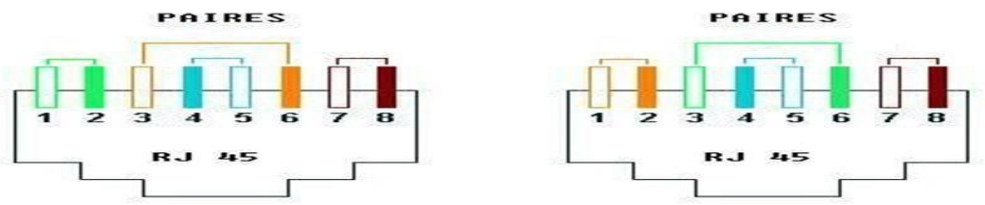



**Figure 5-5: TIA/EIA-568A and TIA/EIA-568B Standards**

The **TIA/EIA-568A and TIA/EIA-568B standards** define a set of fundamental specifications, including:

- types of cables**,**
- cable lengths**,**
- connectors**,**
- cable termination methods**,**
- cable testing procedures**.**

- **Straight-through, crossover, and rollover cabling**

The **TIA/EIA-568A and TIA/EIA-568B standards** specify three types of twisted pair cables:

- **Straight-Through Cable**

A **straight-through cable** is used to connect two different types of network devices, such as a **PC to a switch** or a **switch to a router**. Its wiring configuration is illustrated in the following figure.

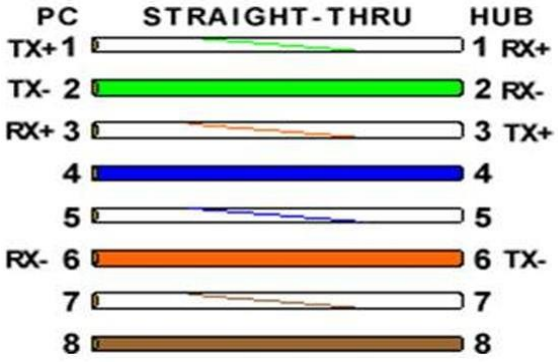



**Figure 5-6: TIA/EIA-568A and TIA/EIA-568B Straight-Through Cable**

- **Crossover Cable**

A **crossover cable** is used to connect two similar network devices, such as **PC-to-PC** or **switch-to-switch** connections. Its wiring configuration is illustrated in the following figure.

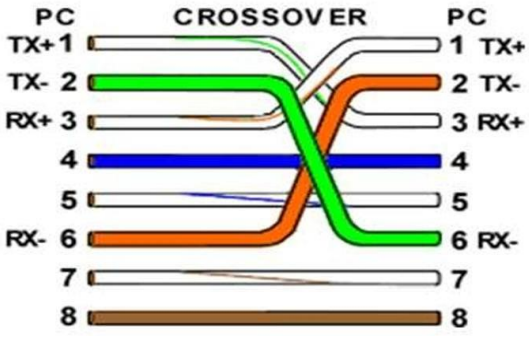



**Figure 5-7: TIA/EIA-568A and TIA/EIA-568B Standards – Crossover Cable**

- **Rollover (Console) Cable:**

  A rollover cable is used to **access and configure network devices**. It establishes a connection between a **computer terminal** and a **network device**, such as a **router or a switch**, through the **console port**, allowing initial setup and administrative management.

## I.3 Optical Fiber Cabling

Optical fiber cabling is a networking medium designed to transmit modulated light signals. Although it is more expensive than copper-based media, optical fiber it is resistant to electromagnetic interference and supports significantly higher data transmission rates. An optical fiber consists of a glass (silica) cylindrical structure whose physical properties allow light to be guided along its core.

In computer networks, fiber-optic cables typically contain two fibers enclosed within separate protective layers. An optical fiber is composed of three essential elements:

- **Core:** the central section through which light signals transmitted.
- **The optical cladding:** a layer with a lower refractive index than the core, which confines the light within the core.
- **Protective coating:** an external layer that provides mechanical protection for the fiber.

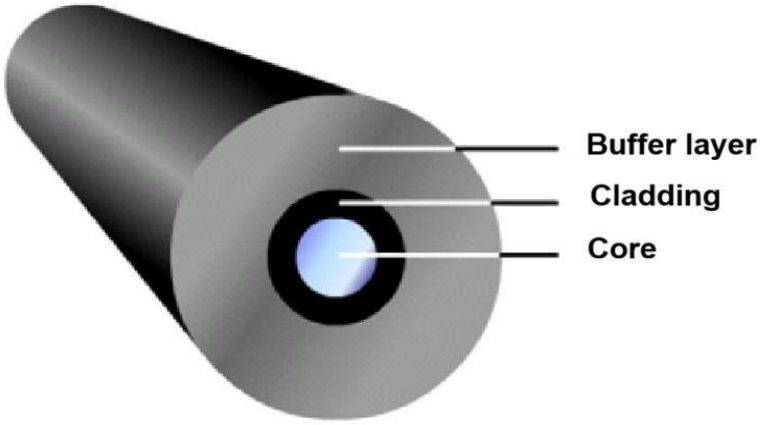



**Figure 5-8: Optical Fiber**

There are two main types of optical fibers:

- **Multimode fiber:**

  This type of fiber has a core diameter ranging from **50 to 62.5 micrometers**. It allows higher signal dispersion, which leads to greater attenuation. Multimode fiber typically uses **LEDs** as the light source.

- **Single-mode fiber:**

  This fiber has a very small core diameter, which significantly reduces dispersion. It is used for **long-distance transmissions** (up to approximately **3 km**) and employs a **laser** as the light source.

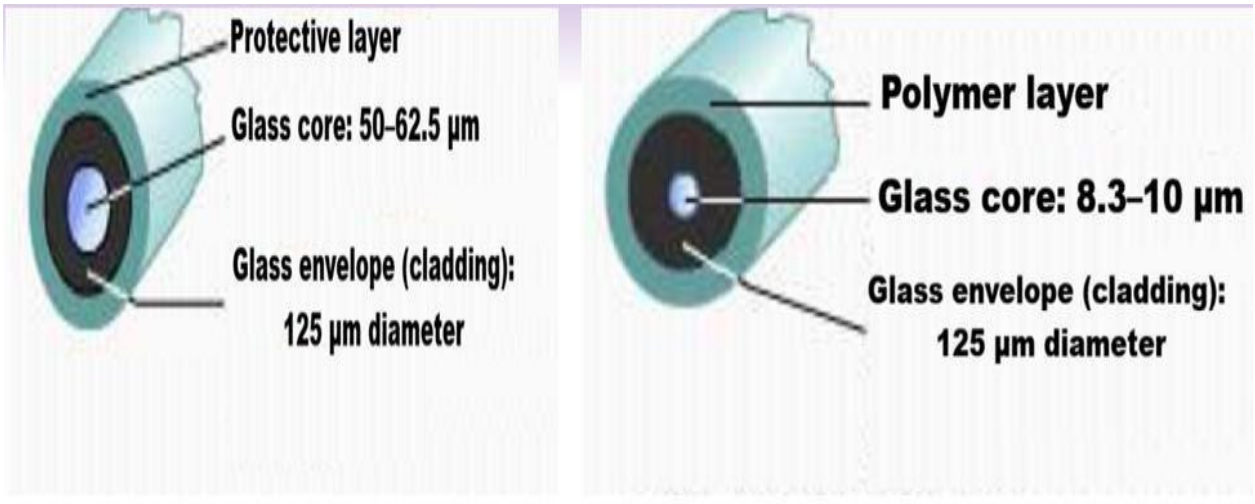



**Figure 5-8: Single-Mode and Multimode Optical Fibers**

Connectors installed at the ends of the optical fibers to allow them to be connected to the transmitter and receiver ports. The most commonly used connectors include:

- **SC connectors (Subscriber Connector):** typically used with **multimode fiber**,
- **ST connectors (Straight Tip):** commonly used with **single-mode fiber**.

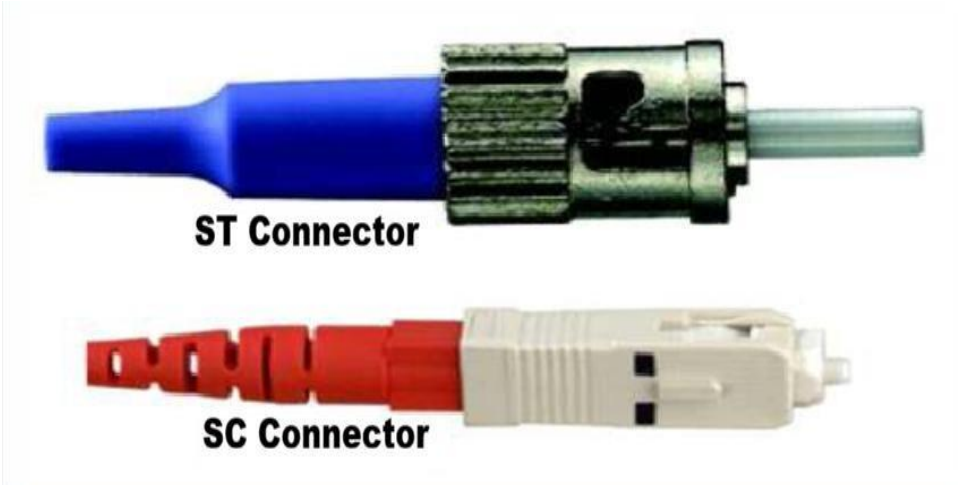



**Figure 5-9: Single-Mode and Multimode Optical Fiber Connectors**

# CHAPTER VI: DATA ENCODING

Data networks rely on the digitization of information, which makes it possible to represent data as sequences of binary digits (0s and 1s). To perform this digitization, **encoding schemes** are used. These schemes associate each character with a specific binary sequence. The number of bits assigned to a character determines the number of symbols supported by the code. Such encoding methods have been standardized. The most widely used codes include:

- **Telegraphic code:** a 5-bit code capable of representing **32 characters**.
- **ASCII code:** a 7-bit encoding scheme that represents **128 characters**.
- **EBCDIC code:** an 8-bit code that supports up to **256 characters**.

After the encoding phase, the data transmission stage takes place. Data may be transmitted either **serially** or **in parallel**. In serial transmission, bits are sent sequentially, one after another. The transmission of characters can occur in two different modes: **asynchronous transmission** or **synchronous transmission**.

The principal challenge in data transmission lies in determining how a transmitter can encode and transmit a signal in such a way that the receiver correctly identifies it as a binary **0** or **1**. The simplest approach consists of sending electrical signals along the transmission medium that represent the bits of a character in the form of square pulses. A zero-level signal corresponds to a **0**, while a positive signal corresponds to a **1**. This technique is known as **baseband transmission**.

## I. Baseband Transmission

Several encoding techniques are used for baseband transmission. Among these techniques is:

### I-1) NRZ (Non-Return to Zero) Encoding:

In this encoding method, the binary signal is represented by voltage levels in order to avoid a non-zero direct current (DC) component, which could lead to increased power consumption. This technique closely resembles basic binary encoding, where a **1** is represented by **+V** and a **0** by **–V**.

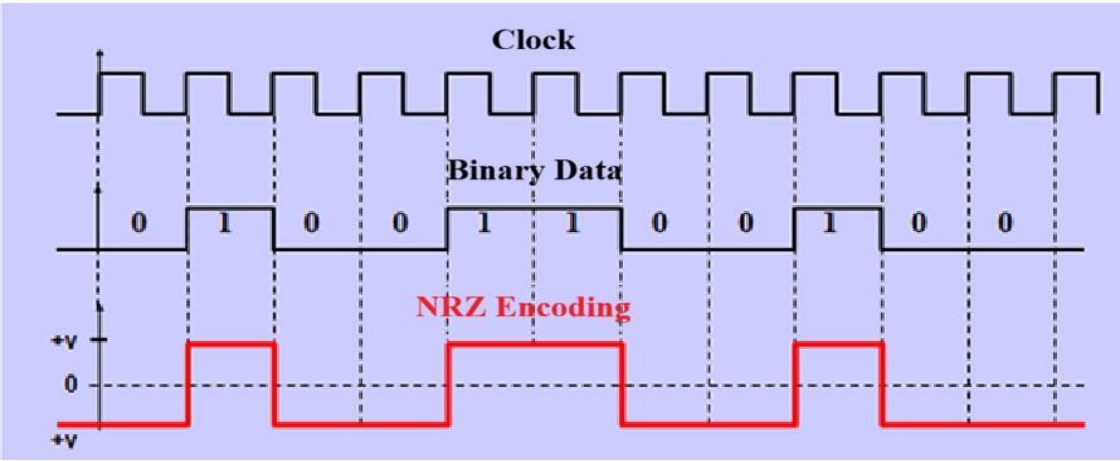



**Figure 6-1: NRZ Encoding**

## I-2) NRZI CODING (Non Return to Zero Inversed):

**NRZI** encoding is significantly different from **NRZ** coding: In this type of coding, the signal:

- Remains in the same state to encode a 0.
- Changes state to encode a 1.

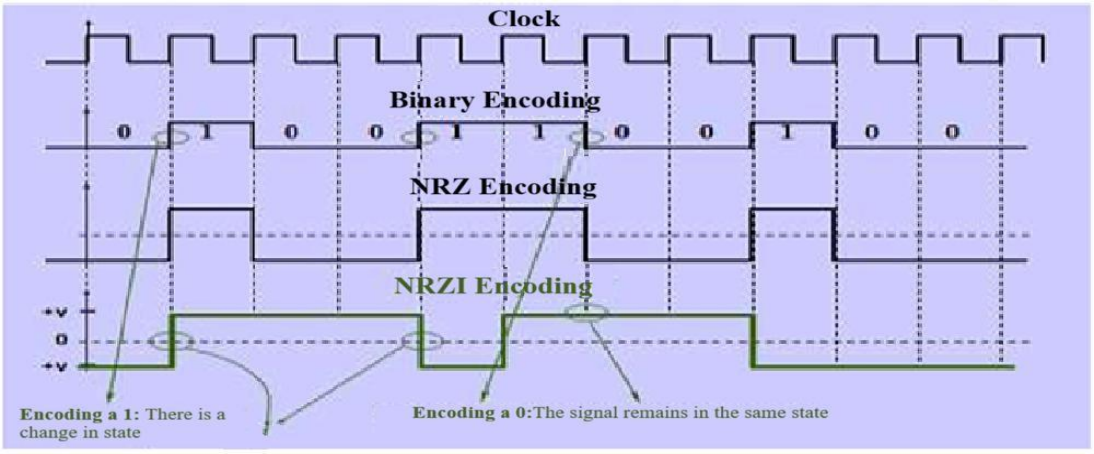



**Figure 6-2: NRZI Encoding**

In **NRZI encoding**, the transmission of long sequences of zeros results in a signal with no transitions over an extended period, which interferes with synchronisation between the transmitter and receiver.

## I-3) Manchester Encoding

One of the main limitations of baseband transmission is the rapid degradation of the signal as the transmission distance increases. If the signal is not periodically regenerated, without regular signal regeneration, the waveform becomes distorted and can no longer be correctly interpreted by the receiver. For this reason, baseband transmission is suitable only for short-distance communication.

**Manchester encoding** was adopted in **Ethernet networks.** It is employed in **10 Mbps Ethernet technologies**, including **10Base5, 10Base2, 10BaseT,** and **10BaseFL**, as illustrated in the following figure.

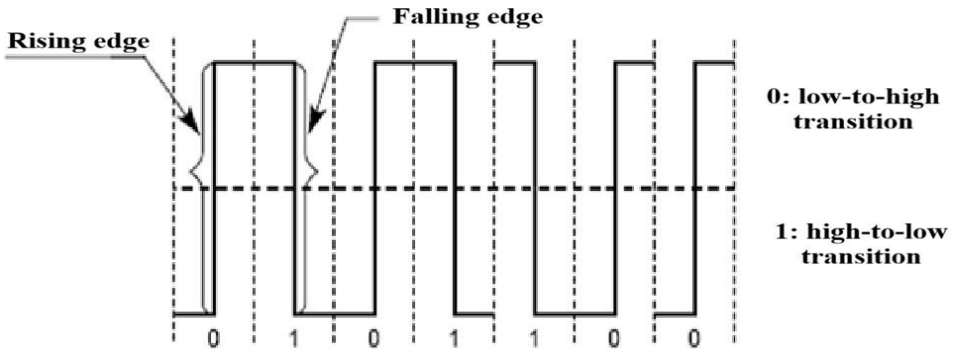



**Figure 6-3: Manchester Encoding**

**Advantages of Manchester Encoding**

Manchester encoding ensures reliable synchronization between the transmitter and the receiver, even when long sequences of binary **0s** or **1s** are transmitted. Moreover, since each binary value is represented by a **signal transition** rather than a constant signal level, this encoding method is highly resistant to transmission errors.

Manchester encoding imposes bandwidth requirements that are twice those of conventional binary encoding for the same data transfer rate.

**Disadvantages of Manchester Encoding:**

Manchester encoding requires a transmission channel capacity that is twice as high as that needed for conventional binary encoding:

- A transmission rate of 10 Mbps requires a signal frequency of 20 MHz.
- At 100 Mbps (100Base-T), the required frequency rises to 200 MHz.

## I-4) MLT-3 Encoding (Multi-Level Threshold 3)

MLT-3 encoding was developed as an alternative to Manchester encoding in order to reduce bandwidth requirements. This encoding method is specifically designed for use in **Fast Ethernet networks operating at 100 Mbps**. Its behavior is illustrated in the following figure.

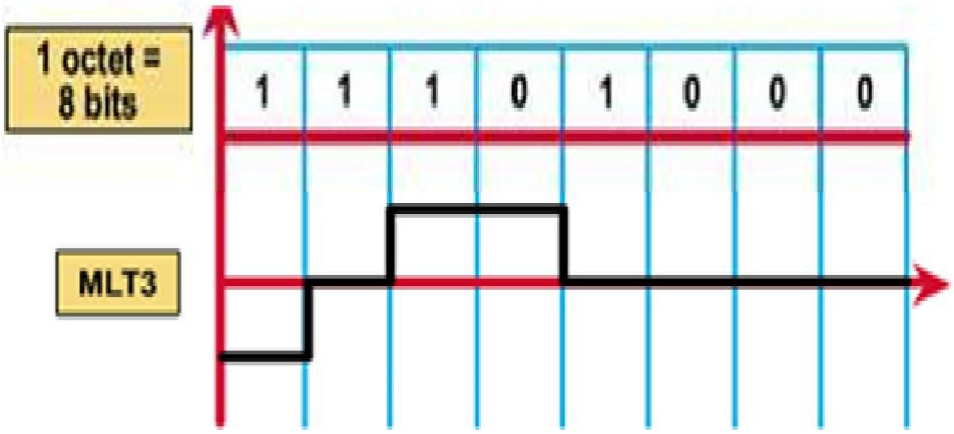



**Figure 6-4: MLT-3 Encoding**

## I-5) 4B/5B Encoding (4 Bits–5 Bits)

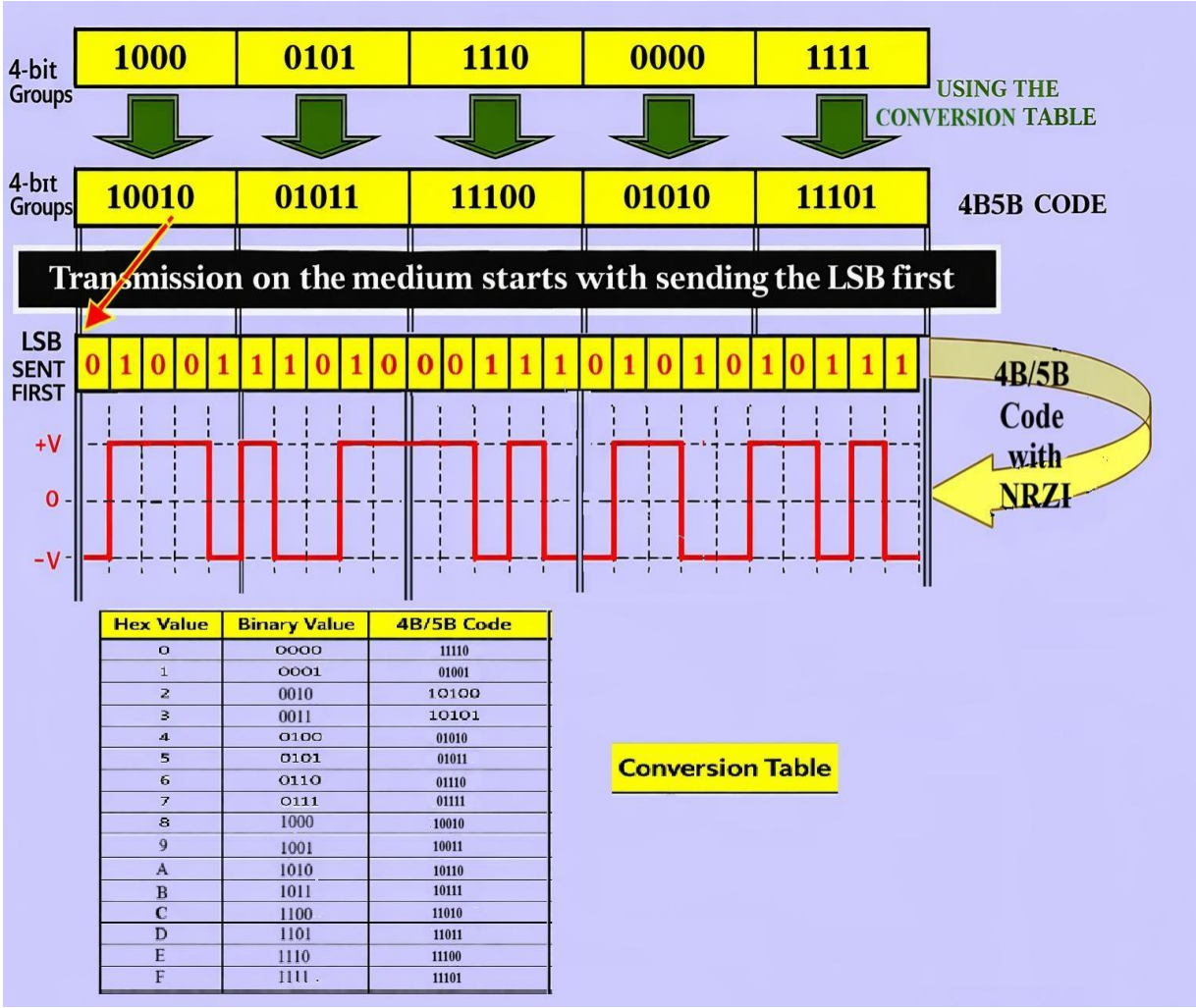
The **4B/5B encoding** technique is used in **high-speed data transmission**. It combines two encoding mechanisms:

- **4B/5B coding (4 data bits mapped to 5 transmitted bits):**
  This method reduces the risk of transmission errors by adding an extra bit to each group of four data bits. As a result, the signal frequency is increased by a factor of **5/4**.

1. **NRZI encoding:**
   After 4B/5B mapping, NRZI encoding is applied to significantly reduce the effective signal frequency by limiting unnecessary signal transitions.

**Example:**

Assume the following binary sequence is to be transmitted:
`1000010111110000011111`




| Hex Value | Binary Value | 4B/5B Code |
|---|---|---|
| 0 | 0000 | 11110 |
| 1 | 0001 | 01001 |
| 2 | 0010 | 10100 |
| 3 | 0011 | 10101 |
| 4 | 0100 | 01010 |
| 5 | 0101 | 01011 |
| 6 | 0110 | 01110 |
| 7 | 0111 | 01111 |
| 8 | 1000 | 10010 |
| 9 | 1001 | 10011 |
| A | 1010 | 10110 |
| B | 1011 | 10111 |
| C | 1100 | 11010 |
| D | 1101 | 11011 |
| E | 1110 | 11100 |
| F | 1111 | 11101 |

**Figure 6-5: 4B/5B Encoding and NRZI Encoding**

# II. ERROR DETECTION AND CORRECTION

Techniques used to detect transmission errors on communication links can be divided into two major categories:

1. **The first category** is based on adding redundant information to the transmitted data, allowing errors to be **detected and, in some cases, corrected**.
2. **The second approach** is based on calculated check values, enabling the detection of corrupted frames so that they can be retransmitted.

## II-1. Error Detection by Redundancy

This method relies on introducing redundancy into the transmitted data. Several techniques can be applied to detect and potentially correct errors:

- **Echo-based detection:**

  The receiver sends back a copy of the received message to the sender. If the returned message differs from the original, the sender retransmits the data. This technique is commonly used in **asynchronous communication systems** (such as Telnet and Minitel).

- **Repetition checking:**

  Each transmitted message is immediately followed by a duplicate copy. If the two messages differ upon reception, the receiver requests retransmission. This technique is used in **highly disturbed secure environments.** and in certain **real-time applications**.

## II-2. Error Detection Using a Computed Check Value

In systems based on computed check values, a **control sequence (CTL1)** is generated from a mathematical operation applied to the message before transmission and is sent together with the data. The receiver performs the same operation on the received message. If the resulting value **(CTL2)** matches the value calculated by the sender **(CTL1)**, the data block is considered valid; otherwise, the block is rejected.

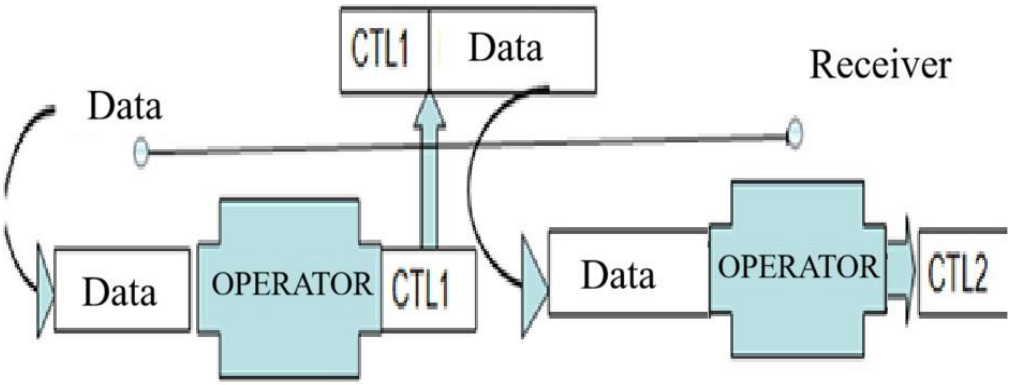



**Figure 6-6: Error Detection Using a Computed Check Value**

- **VRC Technique (Vertical Redundancy Check):**

  This technique is used in **asynchronous transmission** (character-by-character). It consists of adding a single bit to the binary sequence to be protected so that the total number of bits set to 1 is either **even** (even parity bit) or **odd** (odd parity bit).

  **VRC** is a simple method that introduces minimal redundancy; however, its error detection capability is limited to **individual characters**.

  The table shown in the following figure illustrates the application of the **VRC method** to the characters of the word **OSI**.

| Character | O | S | I |
|---|---|---|---|
| Bit 6 | 1 | 1 | 1 |
| Bit 5 | 0 | 0 | 0 |
| Bit 4 | 0 | 1 | 0 |
| Bit 3 | 1 | 0 | 1 |
| Bit 2 | 1 | 0 | 0 |
| Bit 1 | 1 | 1 | 0 |
| Bit 0 | 1 | 1 | 1 |
| Parity Bit | 1 | 0 | 1 |
| Odd Parity Bit | 0 | 1 | 0 |

**Figure 6-7: Error Detection Using the VRC Method**

- **LRC Technique (Longitudinal Redundancy Check):**

  This technique is used in **synchronous transmission**, where characters are transmitted **in blocks** rather than individually.

| Character to transmit | parity Bit | Character to transmit | Parity Bit | ..... | LRC Character | Parity Bit |
|---|---|---|---|---|---|---|

**Figure 6-8: Structure of a Character Block Protected by LRC**

When used separately, the VRC and LRC parity techniques do not provide sufficient error detection. Consequently, they are combined to achieve a more effective synchronous transmission. The following figure illustrates this combined approach.

| | H | E | L | L | O | LRC |
|---|---|---|---|---|---|---|
| Bit 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| Bit 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| Bit 2 | 0 | 1 | 1 | 1 | 1 | 0 |
| Bit 3 | 1 | 0 | 1 | 1 | 1 | 0 |
| Bit 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit 5 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit 6 | 1 | 1 | 1 | 1 | 1 | 1 |
| VRC | 0 | 1 | 1 | 1 | 1 | 0 |

**Figure 6-9: Structure of a Character Block Protected by LRC and VRC**

- **Cyclic Redundancy Codes (CRC)**

The **CRC (Cyclic Redundancy Check)** technique, also referred to as the **Frame Check Sequence (FCS)**, is an error detection method based on a computed check value. In this approach, the data block of **N bits** to be transmitted is interpreted as a polynomial of degree **N−1**, denoted **P(x)**. This polynomial is divided by a predefined **generator polynomial G(x)** using **binary (modulo-2) arithmetic**.

The remainder obtained from this division, denoted **R(x)**, represents the CRC value. This value is appended to to the original data block and transmitted together with it. Upon reception, the receiving system performs the same division on the received data. The CRC calculated by the receiver is then compared with the CRC that was transmitted; if the two values differ, a transmission error is detected.

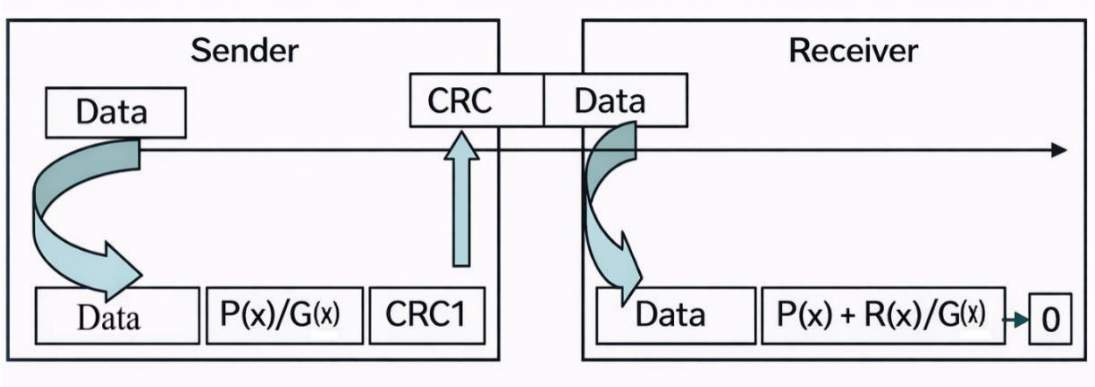



**Figure 6-10: Error Detection Using CRC Calculation**

## Example of a Generator Polynomial:

**Used by IEEE 802 standards**:

**CRC-32**:

$$G(X)=X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1$$

**Used by ATM cells:**

$$\mathbf{X8+X2+X+1.}$$

## Example:

The binary message **110111** is to be protected by a control key calculated using the generator polynomial X2+X+1

$$\mathbf{G(X) = X^2+X+1}.$$

The binary sequence **110111** is mapped to the polynomial

$$\mathbf{P(X)=1*X^5 + 1*X^4 + 0*X^3 + 1*X^2 + 1*X + 1}$$

$$\mathbf{P(X) = X^5+X^4+X^2+X+1}$$

The division process is implemented by hardware circuits based on **exclusive-OR (XOR)** logic. Multiplying the message polynomial by $X^N$ is equivalent to appending **N zero bits** to the original message, where **N** represents the degree of the generator polynomial.

Let the generator polynomial be

$$G(x) = x2+x+1,$$

Which can be written as $\mathbf{1(x^2) + 1(x^1) + 1(x^0)}$, corresponding to the binary sequence **111**.

As this polynomial has degree 2, multiplying the representative polynomial by x2x^2x2 is equivalent to appending **two zero bits (00)** to the end of the polynomial.

$$P(x) = x5+x4+x2+x+1$$

Therefore, the polynomial 110111 becomes 11011100 and is then divided by 111, which represents the generator polynomial.

# CHAPTER VII: IP ROUTING

The role of the network layer is to determine the most efficient path within a network. To achieve this, it relies on an addressing method known as IP addressing. There are two main addressing approaches: linear addressing and hierarchical addressing.

A linear addressing system assigns the next available address to a device. This type of addressing is used for MAC addresses. Ethernet technology uses MAC addresses that are 48 bits long and represented by twelve hexadecimal digits.

The first six hexadecimal digits, which are managed by IEEE, identify the manufacturer or vendor. This part of the MAC address is known as the Organizationally Unique Identifier (OUI). The remaining six hexadecimal digits represent the interface serial number or another value assigned by the manufacturer.

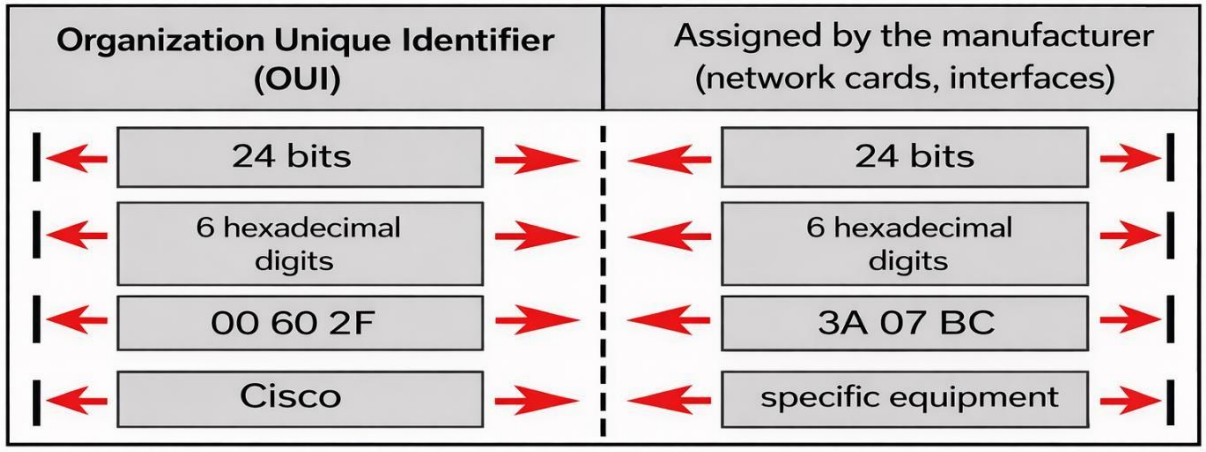



**Figure 7-1: Structure of a MAC Address**

A hierarchical addressing scheme, such as IP addressing, is employed by the network layer to determine the position of a host in a network. An IP address must be unique and conform to a standardized structure. Two versions of IP addresses are in use: IPv4 (Internet Protocol version 4) and IPv6 (Internet Protocol version 6). IPv4 is based on unique 32-bit addresses, while IPv6 uses unique 128-bit addresses.

## I. IPv4 Addressing

Each IPv4 address is composed of two parts: a network identifier (Network ID) and a host identifier (Host ID). An IP address has a total length of 32 bits, divided into four octets separated by dots. Each octet represents a decimal value ranging from 0 to 255.

### A. IPv4 Address Classes

IP addresses in IPv4 are classified to accommodate networks of various sizes, such as large-scale, medium-scale, and small-scale networks. Each class defines which part of the address identifies the network and which part identifies the host.

| Class A | Network | Host | | |
|---|---|---|---|---|
| Octet | 1 | 2 | 3 | 4 |

| Class B | Network | | Host | |
|---|---|---|---|---|
| Octet | 1 | 2 | 3 | 4 |

| Class C | Network | | | Host |
|---|---|---|---|---|
| Octet | 1 | 2 | 3 | 4 |

| Class D | Hote | | | |
|---|---|---|---|---|
| Host | 1 | 2 | 3 | 4 |

**Figure 7-2: IP Address Classes**

Class D addresses are reserved for multicast groups. The four most significant bits always have the binary value **1110**. This class enables multipoint communication between nodes that share the same multicast IP address.

Class E addresses are reserved for experimental and research purposes.

| Class | High-order bits | Range | Usage | Capacity |
|---|---|---|---|---|
| A | 0xxx xxxx | 0.0.0.0 – 127.255.255.255 | unicast classe A | $2^{31}$ addresses |
| B | 10xx xxxx | 128.0.0.0 – 191.255.255.255 | unicast classe B | $2^{30}$ addresses |
| C | 110x xxxx | 192.0.0.0 – 223.255.255.255 | unicast classe C | $2^{28}$ addresses |
| D | 1110 xxxx | 224.0.0.0 – 239.255.255.255 | multicast | $2^{28}$ addresses |
| E | 1111 xxxx | 240.0.0.0 – 255.255.255.255 | reserved | $2^{28}$ addresses |

**Table 7-1: IPv4 Address Classes and Their Address Ranges**

Each address class A, B, and C includes a range of private addresses. Private addresses are not routable on the Internet.

| Class | Range | Network | IP Address | Capacity |
|---|---|---|---|---|
| A | 10.0.0.0 – 10.255.255.255 | 10.0.0.0 /8 | 10.x.y.z – 10.x.y.z | $2^0$ networks, $2^{24}$ hosts |
| B | 172.16.0.0 – 172.31.255.255 | 172.16.0.0 /12 | 172.16.x.y – 172.31.xy | $2^4$ networks, $2^{16}$ hosts |
| C | 192.168.0.0 – 192.168.255.255 | 192.168.0.0 /16 | 192.168.0.x – 192.168 255.x | $2^8$ networks, $2^8$ hosts |

**Table 7-2: Private Address Ranges**

**IP address 0.0.0.0**: indicates that a node does not have an assigned IP address.

**IP address 127.0.0.0/8** (typically **127.0.0.1**) refers to the local host. Packets sent to this address remain within the device and are not forwarded onto the network (loopback).

**IP address 169.254.0.0/16**: used as a default address range for automatic configuration when a DHCP server is unavailable or cannot be reached.

**IP address 255.255.255.255** corresponds to the limited broadcast address and it is delivered to all hosts on the same local network. Routers do not forward this address.

### B. Network Address

A network address refers to an IP address where all bits in the host section are set to zero.

**Examples:**

- The address **192.168.10.100/24** corresponds to the network address **192.168.10.0**
- The address **172.16.10.100/16** corresponds to the network address **172.16.0.0**
- The address **10.10.10.100/8** corresponds to the network address **10.0.0.0**

### C. Broadcast Address

A broadcast address refers to an IP address in which all bits belonging to the host section are set to **1**.

**Examples:**

- The address **192.168.10.100/24** corresponds to the broadcast address **192.168.10.255**
- The address **172.16.10.100/16** corresponds to the broadcast address **172.16.255.255**
- The address **10.10.10.100/8** corresponds to the broadcast address **10.255.255.255**

### D. Network Mask

A network mask is an IP address in which all bits of the network portion are set to **1** and all bits of the host portion are set to **0**.

**Examples:**

- The IP address **192.168.10.100/24** has the subnet mask **255.255.255.0**, which is the default mask
- The IP address **172.16.10.100/16** has the subnet mask **255.255.0.0**, which is the default mask
- The IP address **10.10.10.100/8** has the subnet mask **255.0.0.0**, which is the default mask for Class A

## II. Subnetting

Subnetting makes it possible to divide a single network into multiple subnetworks of equal size by applying a new subnet mask instead of the default one. The reasons for creating subnetworks include:

- Reducing traffic within each network segment
- More efficient management of IP addresses per segment

- Improving the security of network segments
- Enabling the use of different technologies for each segment (Ethernet, Token Ring, etc.)

To perform subnetting, a portion of the bits originally assigned to the host part of the address is borrowed.

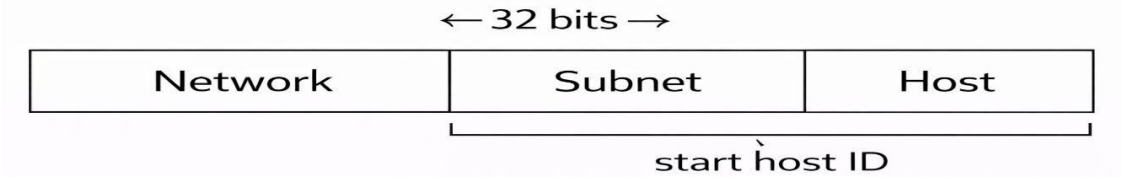



**Figure 7-3: Position of the Network, Subnet, and Host Identifiers in the IPv4 Address**

The process of creating subnetworks involves several key steps:

- Identifying how many subnetworks are required
- Defining how many IP addresses each subnet must support
- Calculating the subnet mask
- Creating the subnets by calculating the subnet address, determining the valid IP address range for each subnet, and identifying the broadcast address associated with each subnet

**Subnetting Calculation Example**

A company plans to use the network address **192.168.90.0** and requires a maximum of **25 hosts per subnet**.

**Tasks:**

- Determine the available subnets
- For the first two subnets, specify the subnet address, subnet mask, usable IP address range, and broadcast address

**Solution:**

- To accommodate **25 hosts**, at least **32 addresses** are required, which means **5 bits** must be allocated to the host portion. The remaining **3 bits** are therefore used for subnet identification.
- The resulting subnet mask is **255.255.255.224**.

| Subnet ID | ASubnet Address | Subnet Mask | IP Address Range | Broadcast Address |
|---|---|---|---|---|
| 1 | 192.168.90.0/27 | 255.255.255.224 | 192.168.90.0/27<br>192.168.90.30/27 | 192.168.90.31/27 |
| 2 | 192.168.90.32/27 | 255.255.255.224 | 192.168.90.33/27<br>192.168.90.62/27 | 192.168.90.63/27 |

While subnetting improves network performance and enhances security, it can also cause inefficient utilization of IP address space. This occurs because all subnets must share the same subnet mask, which forces them to have an equal number of addresses. (**Example,** if one subnet requires only 4 hosts and another needs 30 hosts, both subnets must be sized to support 32 hosts). As a result, many IP addresses remain unused.

To overcome this limitation, a more efficient addressing approach was introduced, known as **CIDR (Classless Inter-Domain Routing)**.

## III. CIDR (Classless Inter-Domain Routing)

CIDR emerged in **1993** to overcome the problem of IP address exhaustion. This approach allows subnetworks to use **different subnet masks** and to support **varying numbers of hosts**, depending on actual needs.

**Example:**

A company plans to use the network address **192.168.90.0** and divide it into **three subnetworks**:

- The first subnet requires **60 hosts**
- The second subnet requires **100 hosts**
- The third subnet requires **20 hosts**

**Tasks:**

- Identify the subnet addresses
- Determine the subnet mask for each subnet
- Define the usable IP address range for each subnet
- Identify the broadcast address of each subnet

| Subnet ID | ASubnet Address | Subnet Mask | IP Address Range | Broadcast Address |
|---|---|---|---|---|
| I (100 hôtes) | 192.168.90.0/25 | 255.255.255.128 | 192.168.90.1/25<br>192.168.90.126/25 | 192.168.90.127/25 |
| II( 60 hôtes) | 192.168.90.128/26 | 255.255.255.192 | 192.168.90.129/26<br>192.168.90.190/26 | 192.168.90.191/26 |
| III (20 hôtes) | 192.168.90.192/27 | 255.255.255.224 | 192.168.90.193/27<br>192.168.90.222/27 | 192.168.90.223/27 |

To simplify the administrator's work, a **DHCP (Dynamic Host Configuration Protocol)** server was developed to dynamically assign the calculated IP addresses to network hosts.

# IV. ARP and RARP Protocols

IP addressing is a logical addressing scheme that allows hosts to communicate within a network without having to consider physical addressing. However, during data transmission over a network, physical addresses are encapsulated to ensure that data frames reach the correct destination at the hardware level. This physical identification is achieved using the MAC address.

Therefore, in order to transmit data correctly, a mapping mechanism is required to associate the logical address (IP address at **Layer 3** of the **OSI** model) with the physical address (**MAC** address at **Layer 2** of the **OSI** model). This function is performed by the ARP protocol.

**ARP (Address Resolution Protocol)** is a data link layer protocol that allows a host to determine the physical **(MAC)** address of a destination when its logical (IP) address is known. Each host maintains an **ARP table**, also referred to as the **ARP cache**, which stores these IP-to-MAC associations. This table is dynamic, the table is dynamic, and its entries are retained for a limited duration. The contents of the ARP table can be displayed using the `arp -a` command.

**RARP (Reverse Address Resolution Protocol)** is also a data link layer protocol, but it performs the reverse operation: it determines the logical (IP) address of a host based on its physical **(MAC)** address. RARP is typically used by hosts that do not have an IP address at startup, (such as diskless workstations).

# CHAPTER VIII: DHCP SERVER (Dynamic Host Configuration Protocol)

## I. DHCP Overview

A DHCP (Dynamic Host Configuration Protocol) server is a network service designed to dynamically assign IP configuration parameters to clients for a limited period of time. These parameters typically include the IP address, subnet mask, default gateway, and the address of the name server.
DHCP assigns these parameters to all hosts that rely on dynamic addressing. Each allocation is valid only for a specific time interval known as the **Lease Duration**.

## II. Advantages of DHCP in Network Administration

The DHCP service offers several benefits, including:

- Centralized control over the use of IP addresses
- Elimination of manual IP address configuration
- Prevention of address conflicts by avoiding duplicate IP assignments
- Improved network performance and faster updates of IP configuration settings
- More efficient use of IP addresses, as the network administrator controls address assignment through the configuration of DHCP leases; IP addresses are automatically released once the lease expires

## III. DHCP Server Operation

When a host configured for automatic addressing starts up on a network, the process of locating a DHCP server is initiated and occurs in four main phases:

- **DHCP Discovery (DHCPDISCOVER):**

  The client broadcasts a message across the network to request an IP lease. This message is sent with the source IP address set to **0.0.0.0** and includes the client's physical (MAC) address, while the destination IP address is **255.255.255.255**.

- **DHCP Offer (DHCPOFFER):**

  Any DHCP server that receives the discovery message and has unassigned IP addresses responds by offering an IP address from its address pool, along with a specified lease duration. This response is referred to as a **DHCP offer**

- **DHCP Request (DHCPREQUEST):**
  The client selects the first IP address offered by the **DHCP** servers and broadcasts a request message over the network. This message includes the chosen IP address and identifies the chosen **DHCP** server.

The purpose of this request is twofold:

1. To confirm acceptance of all IP configuration parameters provided by the selected DHCP server (subnet mask, default gateway, DNS server address, etc.).
2. To notify the other DHCP servers that their offers have not been selected, allowing those IP addresses to be returned to the available pool.

- **DHCP Acknowledgment (DHCPACK):**
  Once the offer has been accepted, the selected DHCP server sends a confirmation message indicating that the IP address has been successfully assigned. This message includes the allocated IP address, the subnet mask, the default gateway, the DNS server information, and the lease duration associated with the assignment.

# IV. IP Lease Renewal Process

When a client system restarts, it continues to use its previously assigned IP address provided that the lease has not expired. The client then attempts to renew the lease for the same address with the original DHCP server by sending a **DHCPREQUEST** message.

- DHCP clients operating in a Windows DHCP environment (NT/2000) attempt to renew their lease once **50% of the lease duration** has elapsed. This renewal request is sent using a **DHCPREQUEST** message, and if approved, the server responds with a **DHCPACK** message that extends the lease period.
- If the lease renewal is unsuccessful at the 50% threshold, the client makes another renewal attempt when **87.5% of the lease duration** has passed. At this point, DHCP servers may respond with either a **DHCPACK** or a **DHCPNACK**.

If the lease expires or a **DHCPNACK** message is received, the client must immediately discontinue use of the IP address and initiate a new DHCP discovery phase by sending a **DHCPDISCOVER** message.

# CHAPTER IX: DNS SERVER (DOMAIN NAME SYSTEM)

## I. DNS Service Description

The DNS (Domain Name System) is a naming service that enables the identification of hosts by their **Fully Qualified Domain Names (FQDNs)** rather than by their IP addresses. Its primary function is to map each domain name to its corresponding IP address.

Hosts send queries to a DNS server in order to obtain the IP address of the requested host. The DNS service operates primarily over the **UDP protocol**, using **port number 53**.

- **Example of name resolution:**

  `www.google.com → 209.85.229.99`

- **Illustrative example:**

  In this example, a user attempts to establish a remote connection (for example, via **Telnet**) to the host `www.transfert.dz`.Tthe user first sends a DNS query to the DNS server to retrieve the IP address associated with the target host. Once the IP address is received, the remote connection can be initiated using that address.

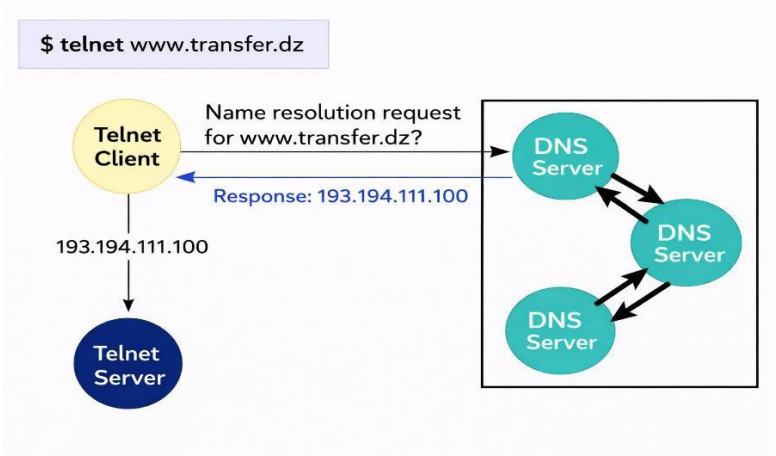



**Figure 9-1: DNS Resolution**

The DNS service cannot be centralized, as a single server is unable to store and manage all the name-to-address mappings required by Internet applications. This limitation is due to several factors:

- **Internet scale:** The number of queries sent to DNS servers is extremely large.
- **Fault tolerance:** A failure of a single DNS server would prevent hosts from accessing Internet resources.
- **Traffic load:** The volume of DNS traffic cannot be efficiently handled by a single server.
- **Response time:** DNS servers must provide fast responses, which is not achievable when relying on only one DNS server
- Issues Related to DNS Database Maintenance and Updates

# II. Domain Name Space

The DNS system is implemented through a **globally distributed database**. It is based on a **hierarchical tree structure** with **decentralized data management**, (where each DNS server is responsible for the information contained within its own zone). The DNS naming system is organized as an **inverted tree**, with the "**root"** at the top and a set of nodes representing domains identified by labels (such as *fr*, *education.fr*, *org*, *com*, etc.).A domain can be considered as a portion or sub-portion of the overall naming hierarchy.

Each name server manages a specific node in the hierarchy or a group of nodes over which it has authority. Such a server is said to manage a **zone of authority**.

Domain names are administered by a global organization known as **InterNIC,** as well as by delegated organizations such as RIPE, NIC France, NIC England, and others. The process of translating a domain name into an IP address by a DNS server is referred to as **name resolution**.

- Each data unit stored in the DNS database is indexed by a name.
- Domain names form a path within an inverted tree structure known as the **domain name space**.
- A domain label may contain up to **63 characters**, and letter case (uppercase or lowercase) is not significant.
- This hierarchical organization is similar to a file system structure.
- Each node in the hierarchy is identified by a name.
- The root node, known as the **root**, is identified by the symbol "."
- The hierarchy supports a maximum depth of **127 levels**.
- The structure of the domain name space is entirely independent of the physical or geographical organization of the network. Only registered **name servers** are visible within the naming system.

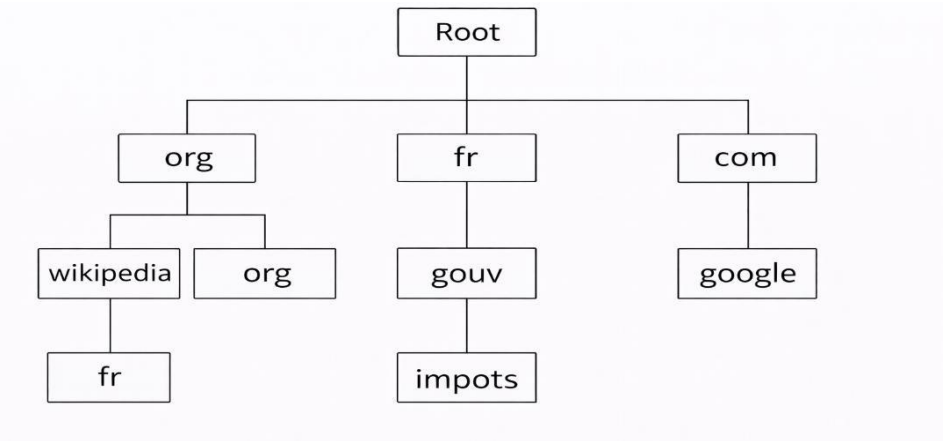



**Figure 9-2: DNS Namespace**

Each "part" of a domain name is called a **label**, and the collection of these labels forms a **Fully Qualified Domain Name (FQDN)**. An FQDN is unique. By convention, an FQDN ends with a dot (."). This dot is usually omitted when users type a domain name in a web browser; however, it is important to include it when configuring a DNS server.

The DNS system defines only a small number of naming rules:

- There are several predefined **top-level domains (TLDs)**, including:
  - **.com**: commercial organizations (e.g., *ibm.com*)
  - **.edu**: educational institutions (e.g., *mit.edu*)
  - **.gov**: government organizations (e.g., *nsf.gov*)
  - **.mil**: military organizations (e.g., *army.mil*)
  - **.net**: Internet network-related organizations (e.g., *worldnet.net*)
  - **.org**: non-profit organizations (e.g., *eff.org*)
  - **.int**: international organizations (e.g., *nato.int*)
- **Country-code top-level domains (ccTLDs)** are assigned to national organizations, such as **.dz**, **.fr**, **.uk**, **.de**, **.it**, **.us**, **.au**, **.ca**, **.se**, and others.

# III. Reverse Name Resolution

The purpose of reverse name resolution is to associate a domain name with a given IP address. This process is the opposite of forward name resolution, where an IP address is assigned to a domain name.

To perform reverse name resolution, a specific DNS zone is used, known as the **in-addr.arpa** zone.

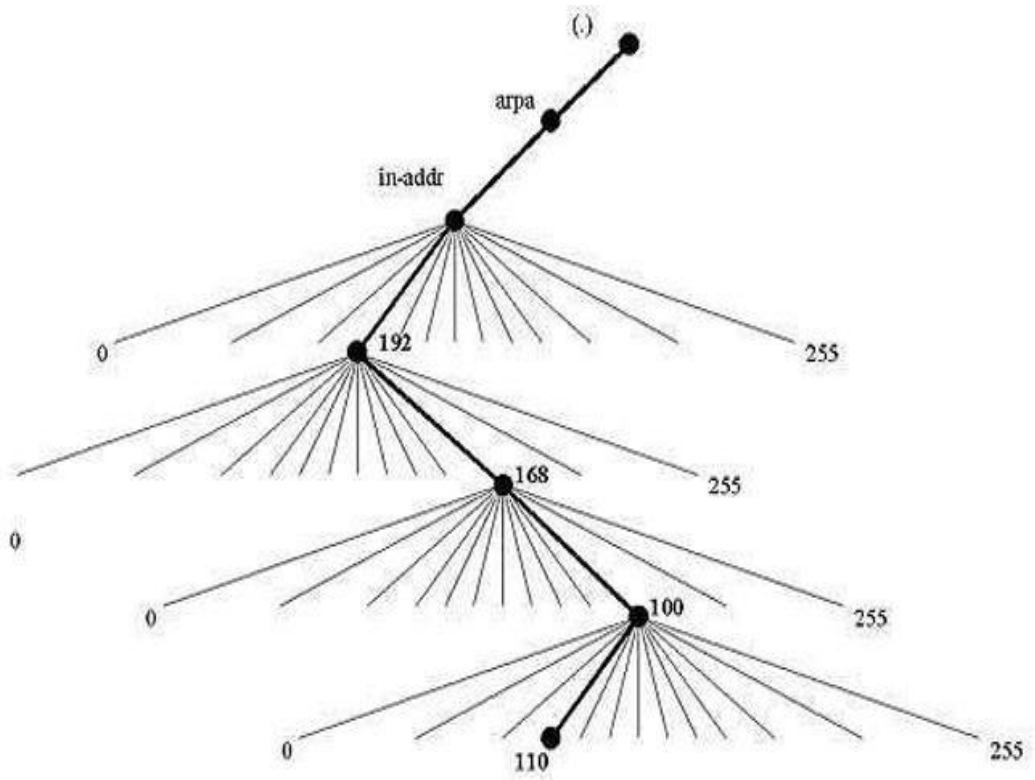



**Figure 9-3: Reverse Name Resolution**

# IV. DNS Configuration on Linux

## A. DNS Server Configuration

A DNS server manages name assignment using a database implemented through **zone files**, with a separate zone file for each node or zone. Each basic piece of information stored in the DNS database is represented by an object known as a **Resource Record (RR)**.

To configure a DNS server, the following steps must be performed:

- First, install the **BIND (Bind9)** package.
- Configure the **/etc/bind/named.conf** file, which is used to define DNS zones and specify the location of their corresponding zone files.
- Configure the **zone files**, which contain the resource records associated with each DNS zone.

The main types of DNS resource records include:

- **SOA (Start of Authority) record:**

Defines the authority over a DNS zone. This record contains essential domain information, including database update intervals between primary and secondary name servers, as well as the name and email address of the domain administrator.

- **NS (Name Server) records:**

These records provide the addresses of the domain name servers to be configured.

- **A (Address) records:**

Associate hostnames with fixed IP addresses. These records are commonly used for hosts with static IP addresses, such as servers and routers.

- **MX (Mail Exchanger) records:**

Specify the mail servers responsible for handling electronic mail for the domain.

- **CNAME (Canonical Name) records:**

Define alternative names (aliases) that reference existing canonical hosts.

- **PTR (Pointer) records:**

  Support reverse name resolution within the **in-addr.arpa** domain.

## 1. Structure of an SOA Record

Each DNS zone file begins with a **Start of Authority (SOA)** record. This record defines the authoritative information for the zone. An example of an SOA record is shown below:

```
ENPO.org. IN SOA ns1.ENPO.org.  admin.ENPO.org. (

          20001210011  ; Serial number
          10800        ; Refresh interval
          3600         ; Retry interval
          604800       ; Expiration time (one week)
          86400        ; Minimum TTL (one day)

      )
```

```
ENPO.org.   IN   SOA   ns1.ENPO.org.   admin.ENPO.org. (
    20001210011 ; Serial number
    10800       ; Refresh interval
    3600        ; Retry interval
    604800      ; Expiration time (one week)
    86400       ; Minimum TTL (one day)
)
```

## 2. NS Records for the Domain ENPO.org

**Name Server (NS)** records are used to identify the authoritative DNS servers for a given domain.
For the domain **ENPO.org**, the NS records are defined as follows:

**ENPO.org. IN NS ns1.ENPO.org.**

**ENPO.org. IN NS ns2.ENPO.org.**

```
ENPO.org.   IN   NS   ns1.ENPO.org.
ENPO.org.   IN   NS   ns2.ENPO.org.
```

### 3. A Records

**Address (A)** records define the mapping between a host name and its corresponding IP address.
For the domain **ENPO.org**, the following A records are configured:

ns1.ENPO.org. IN A 192.168.0.1

ns2.ENPO.org. IN A 192.168.0.2

```
ns1.ENPO.org.    IN    A    192.168.0.1
ns2.ENPO.org.    IN    A    192.168.0.2
```

### 4. CNAME Records

**CNAME (Canonical Name)** records are used to define aliases for existing domain names. When a request such as **`http://www.ENPO.org`** is issued, it is redirected to **ns1.ENPO.org**, since **www** is configured as an alias for **ns1**.

**www.ENPO.org. IN CNAME ns1.ENPO.org.**

**ftp.ENPO.org. IN CNAME ns2.ENPO.org.**

```
www.ENPO.org.    IN    CNAME    ns1.ENPO.org.
ftp.ENPO.org.    IN    CNAME    ns2.ENPO.org.
```

### 5. PTR Records

**PTR (Pointer)** records are used for **reverse name resolution**, which maps an IP address to a corresponding domain name.

1.0.168.192.in-addr.arpa. IN PTR ns1.ENPO.org.

2.0.168.192.in-addr.arpa. IN PTR ns2.ENPO.org.

```
1.0.168.192.in-addr.arpa.    IN    PTR    ns1.ENPO.org.
2.0.168.192.in-addr.arpa.    IN    PTR    ns2.ENPO.org.
```

## B. DNS Client Configuration

The DNS client configuration is defined in the **/etc/resolv.conf** file. This file specifies the information required for domain name resolution, including the domain search path and the address of the DNS server to be used.

# /etc/resolv.conf file

search ENPO.org # Domain name nameserver

192.168.1.1 # DNS server address

# Appendix A: Network Laboratory (Lab Work)

## Lab 1: Network Cabling

### Objectives

- Design and assemble a **straight-through Ethernet cable**
- Design and assemble a **crossover Ethernet cable**

### Required Equipment

- UTP cable
- Crimping tool
- RJ45 connectors

The **TIA/EIA 568A** and **568B** standards define the wiring schemes used to construct **twisted-pair network cables**. These standards are specified by **TIA/EIA** and are illustrated in the following table.

Standard T568A

| pin | color | pair |
|---|---|---|
| 8 | brown | 4 |
| 7 | brown-white | |
| 6 | orange | 2 |
| 5 | blue-white | 1 |
| 4 | blue | |
| 3 | orange-white | 2 |
| 2 | Green | 3 |
| 1 | Green-white | |

Standard T568B

| pair | color | pin |
|---|---|---|
| 4 | brown | 8 |
| | brown-white | 7 |
| 3 | Green | 6 |
| 1 | blue-white | 5 |
| | blue | 4 |
| 3 | Green-white | 3 |
| 2 | orange | 2 |
| | orange-white | 1 |

Define a Straight-Through Cable: .....

When is a straight-through cable used? .....

1. **Creating a Straight-Through Cable *(According to the wiring standard of your choice)*Follow the steps below:**
   - Strip the outer insulation of the cable
   - Untwist the wire pairs
   - Adjust and align the wires to the same length
   - Verify the required color-coding scheme
   - Insert the flattened wire pairs into the RJ45 connector
   - Crimp the connector using a crimping tool
   - Test the cable by connecting one end to an RJ45 wall outlet and the other end to a computer's network interface. On Windows 7, open the Control Panel from the Start menu, select Network and Internet, then Network and Sharing Center, and verify that the local area network connection is active. If a physical connectivity issue is detected,

a red cross will appear on the Local Area Connection icon. In this case, the cable must be reconstructed.

Define a Crossover Cable: .....

When is a crossover cable used? .....

Construct a **crossover cable** (according to the appropriate standard) by following the same steps described above.

# Lab 2: Cisco Packet Tracer Network Simulation

## Objectives

- Use **Cisco Packet Tracer** to design and implement a local area network
- Configure IP addresses on network devices
- Verify network connectivity

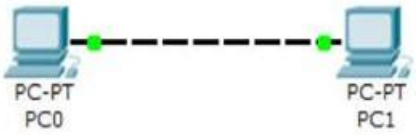
## Prerequisites

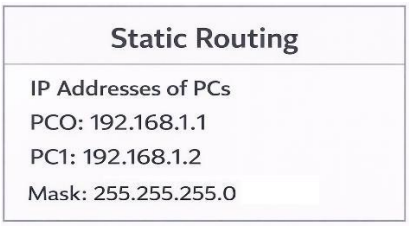
- Knowledge of **twisted-pair cabling**
- Basic understanding of **IP addressing**

## 1) Creation of a Peer-to-Peer Network

A **peer-to-peer network** is a simple network architecture that requires minimal hardware. To set it up, the following components are sufficient:

- Two computers, each equipped with an **Ethernet network interface card**
- One **crossover twisted-pair cable** to directly connect the two computers




- Create the network and configure IP parameters by assigning only the IP address and subnet mask to the two workstations, PC0 and PC1.
- Verify connectivity between the two PCs using the ping command.If the connectivity test fails, check the type of cable in use and review the IP addressing configuration, then perform the connectivity test again.

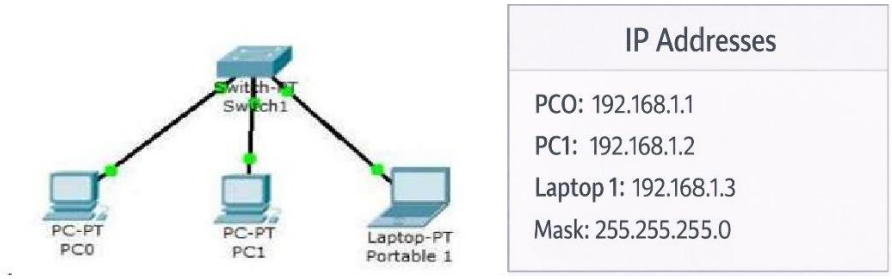
## 2) Network Creation Using a Switch

### 1. Basic Ethernet Network

In this task, a network composed of **three end devices** (two PCs and one printer) interconnected through a **switch** will be implemented.

### 2. Creating a Network Using a Switch

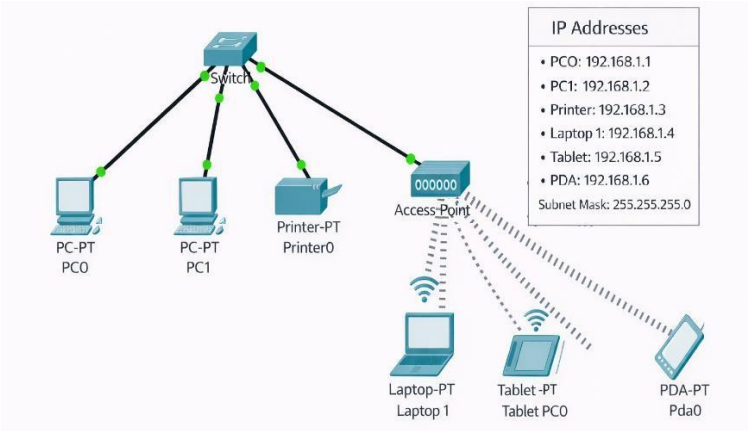
Create a network composed of **two desktop PCs and one laptop**, all connected through a **network switch**.




- Determine the **class of the network**.
- Identify the **network address** and the **broadcast address** of this network.
- Build the above network using the **IP parameters specified earlier**.
- Which **type of cable** is used to connect the PCs to the switch?
- Verify the IP configuration using the **`ipconfig`** command.
- Provide the **MAC addresses** of each host (**which command is used?**).
- Test connectivity between the hosts using the **`ping`** command with IP addresses.

### 3. Hybrid Wired and Wireless Network

Creation of a network that enables the interconnection of a **wired Ethernet network** with a **wireless network**.




- Build the above network using the **specified IP parameters**.
- Which **type of cable** is used to connect the **switch** to the **wireless access point**? **Justify your answer.**

**Note:**
To enable wireless connectivity on the laptop, a wireless network adapter named **Linksys WPC300** must first be installed and then configured accordingly.

- Test the **connectivity** between all hosts**.**

# Lab 3: TCP/IP LAN Network

## Design or implement a star-topology LAN using a network switch.

### Objectives

- Deploy a **real network** using a **star topology**, with workstations running **Windows or Linux operating systems**.
- Verify connectivity between hosts using common **network diagnostic commands**, such as: **`ipconfig /all`**, **`ping`** (Packet Internet Groper), **`arp`**, and **`ifconfig`**

### Required Resources

- Computers running **Windows**, each equipped with a **network interface card**
- **Twisted-pair cables** and **RJ45 connectors**
- **Network switches**

#### a. Complete the following table:

| | |
|---|---|
| Network Address | 192.168.1.0/24 |
| Subnet Mask | 255.255.255.0 |
| Broadcast Address | |
| Starting Host Address | |
| Ending Host Address | |
| Network Address Range End | |

#### b. IP Configuration

Assign an IP address and a subnet mask to each host.

- **Under Windows:**

  Open the **Control Panel**, then navigate to **Network and Internet → Network and Sharing Center → Change adapter settings**.


- **Under Linux:**

  Use the following command to configure the network interface: `ifconfig eth0 <IP_address> netmask <subnet_mask>`

- Test network connectivity in **both environments** using the **`ping`** command.

**Note 1:**

To properly test **TCP/IP connectivity** between computers, the **Windows firewall must be disabled** on both systems.

**Note 2:**

To access the **Windows Firewall**, open the **Start menu**, go to the **Control Panel**, select **System and Security**, and then disable the firewall.

Type the command **`ping 192.168.1.254`** and press **Enter**. A successful response indicates tha **IP connectivity is confirmed**. The output window should be similar to the following example:

```
Envoi d'une requête 'ping' sur 192.168.0.254 avec 32 octets de données :

Réponse de 192.168.0.254 : octets=32 temps<10 ms TTL=64
Réponse de 192.168.0.254 : octets=32 temps<10 ms TTL=64
Réponse de 192.168.0.254 : octets=32 temps<10 ms TTL=64
Réponse de 192.168.0.254 : octets=32 temps<10 ms TTL=64

Statistiques Ping pour 192.168.0.254:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    minimum = 0ms, maximum = 0ms, moyenne = 0ms
```

Execute the **`arp -a`** command on each workstation, then confirm that the listed IP addresses match the corresponding MAC addresses by checking the network configuration using **`ipconfig /all`** on Windows and **`ifconfig`** on Linux.

```
C:\>arp -a

Interface : 192.168.1.31 on Interface 0x1000003
  Adresse Internet      Adresse physique      Type
  192.168.1.2           00-d0-cf-03-60-42     dynamique
  192.168.1.33          00-10-75-03-05-f8     dynamique
  192.168.1.36          00-17-9a-c3-a6-c8     dynamique
```

Provide the results of the following commands:

**Ipconfig /release**

**ipconfig /renew**

**`tracert`** Traces the IP path taken by packets to reach a destination, displaying the **names of the intermediate hosts** traversed

**tracert -d <hostname>** Performs the same path tracing operation, but displays **only the IP addresses** of the intermediate hosts, without resolving their names.

**`Netstat`** Displays active network connections and indicates the **type of transport protocol used**, such as **TCP** or **UDP**.

# Lab 4: Peer-to-Peer Resource Sharing

## Objectives

- Share resources within a **peer-to-peer network**
- Examine the role and functionality of the **SAM (Security Account Manager)**

## Overview

A **peer-to-peer network** is a network model in which security management is handled locally by each workstation. Under **Windows XP Professional**, authentication and access control and security information are stored in a local database known as the **SAM (Security Account Manager)**.

The SAM database contains the list of user accounts defined on each individual workstation

## Configuration Steps

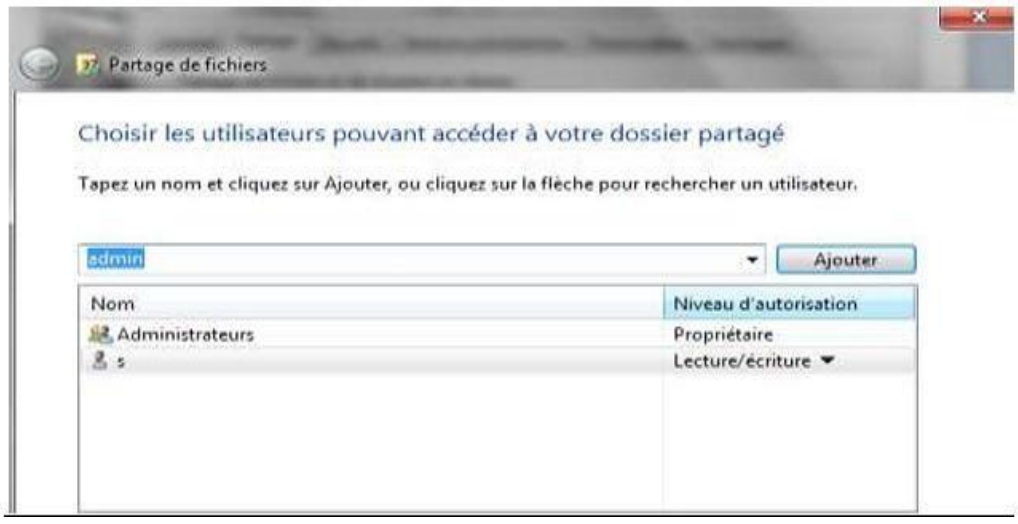
1. **Installation and configuration of the network interface**

   Assign the **IP address** and **subnet mask** to the network card.

2. **Connectivity testing**
   - Use the command **`ipconfig /all`** to verify the IP configuration.
   - Test **local connectivity** using: `ping 192.168.10.1`
   - Test **remote connectivity** using: `ping 192.168.10.2`

3. **Resource sharing**

   Create a folder, then right-click on it and select **Properties** → **Sharing**. The sharing configuration window will appear. Select the users who are allowed to access the shared resource and assign the appropriate **permission level**.



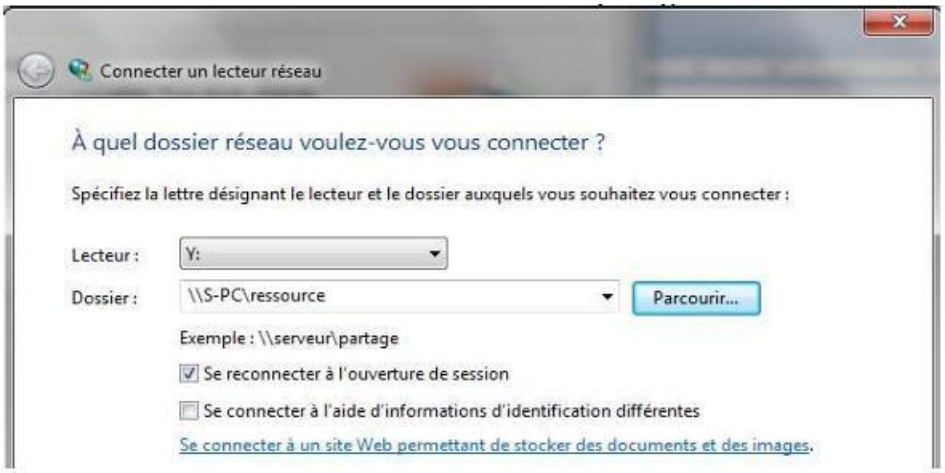

Once a resource has been shared, it becomes accessible over the network. To access a shared resource, follow these steps:

- Open the **Start Menu**, then select **Run**.
- Enter the path `\\<IP_address_of_the_remote_host>` and confirm. A window will then appear displaying the list of **shared folders** available on the remote machine.


**Note:**

Remote access is established between two hosts that have **identical user accounts defined in their respective SAM databases**.

- Access all machines on your network and provide a list of their **shared resources**.
- If a device, such as a **laptop**, does not have at least one user account in common with your local **SAM**, a dialog window will appear requesting:
  - a **username**
  - a **password**
- **Mapping a Network Drive:** A shared resource can be assigned as a network drive by following these steps: Open the Start Menu, right-click on Computer, select Properties, then open the Tools Menu and choose Map Network Drive. Next, specify the drive letter to be created and enter the path of the shared folder. The configuration process is illustrated as follows.




The shared resource will then appear in the list of available drives.

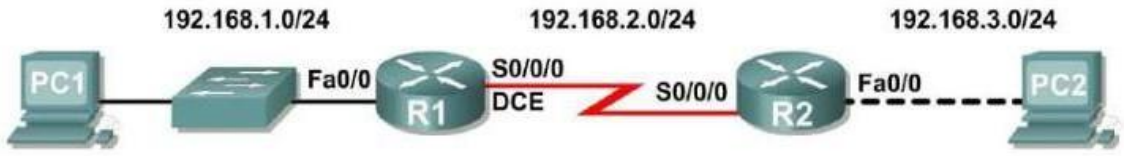
- Create a **network drive** using one of the shared resources available on the network.

# Laboratory 5: IP Routing

## Objectives

- Divide the network into **multiple subnets**
- Configure an **IP router**
- Carry out essential **router configuration operations**

Consider the following topology




The addressing table for this network is defined as follows.

| Device | Interface | IP Address | Subnet Mask | Default Gateway |
|---|---|---|---|---|
| R1 | Fa0/0 | 192.168.1.1 | 255.255.255.0 | S0/0 |
| | S0/0/0 | 192.168.2.1 | 255.255.255.0 | S0/0 |
| R2 | Fa0/0 | 192.168.3.1 | 255.255.255.0 | S0/0 |
| | S0/0/0 | 192.168.2.2 | 255.255.255.0 | S0/0 |
| PC1 | S0/0 | 192.168.1.10 | 255.255.255.0 | 192.168.1.1 |
| PC2 | S0/0 | 192.168.3.10 | 255.255.255.0 | 192.168.3.1 |

## Answer the following questions:

- Which **type of cable** is used to connect the **Ethernet interface of a host PC** to the **Ethernet interface of a switch**?
- Which **type of cable** is used to connect the **Ethernet interface of a switch** to the **Ethernet interface of a router**?
- Which **type of cable** is used to connect the **Ethernet interface of a router** to the **Ethernet interface of a host PC**?

### Router Configuration

- **Enable privileged EXEC mode:**

Router> enable

Router#

```
Router> enable
Router#
```

- **Enter global configuration mode:**

Router# configure terminal
Router(config)#

```
Router# configure terminal
Router(config)#
```

- **Assign the router name as R1:** At the command prompt, type:

Router(config)# hostname R1
R1(config)#

```
Router(config)# hostname R1
R1(config)#
```

- **Configuring the privileged EXEC mode password:**

Enter the command `enable secret` followed by the password. Use **ING_2018** as the secret password.

R1(config)# enable secret ING_2018
R1(config)#

```
R1(config)# enable secret ING_2018
R1(config)#
```

- **Configuring the virtual terminal (VTY) line password:**

Set the password to **cisco**, enable login authentication, then exit line configuration mode.

R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)#

```
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)#
```

- **Configuring the FastEthernet0/0 interface:**

Assign the IP address **192.168.1.1/24** to the FastEthernet0/0 interface and enable the interface.

R1(config)# interface fastethernet 0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown

```
R1(config)# interface fastethernet 0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown
```

- **Configuring the Serial0/0/0 interface:**

  Assign the IP address **192.168.2.1/24** to the Serial0/0/0 interface and set the clock rate to **64,000 bps**.

  R1(config)# interface serial 0/0/0

  R1(config-if)# ip address 192.168.2.1 255.255.255.0

  R1(config-if)# clock rate 64000

  R1(config-if)# no shutdown

```
R1(config)# interface serial 0/0/0
R1(config-if)# ip address 192.168.2.1 255.255.255.0
R1(config-if)# clock rate 64000
R1(config-if)# no shutdown
```

- **Saving the configuration of R1:**

  Use the following command to save the current configuration to non-volatile memory:

  R1# copy running-config startup-config

```
R1# copy running-config startup-config
```

- **Configuring router R2:**

  Apply the same configuration steps to **router R2**, with the exception that the **serial interface must not include the `clock rate` command**.

- **Verification of routing tasks:**

  Verify correct routing operation using appropriate verification commands.

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, Serial0/0/0
R1#
```

- **Interface Verification**

```
R1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.1.1     YES manual up                    up
FastEthernet0/1        unassigned      YES unset  administratively down down
Serial0/0/0              192.168.2.1     YES manual up                    up
Serial0/0/1              unassigned      YES unset  administratively down down
Vlan1                    unassigned      YES manual administratively down down
```

- **Connectivity Testing**

  Test network connectivity by sending **ping packets** from each host to its configured **default gateway**.

  From the host connected to **R1**, is it possible to successfully send a ping packet to the default gateway? ……..

  From the host connected to **R2**, is it possible to successfully send a ping packet to the default gateway? …….

# Lab 6: DHCP Server

## Objective

- Configure a **DHCP server under Linux**

---

## 1) Configuration Steps

- Install the **dhcp3-server** package.

```
apt-get install dhcp3-server
```

- Modify the appropriate **DHCP configuration file**.

**The DHCP configuration file is `/etc/dhcp3/dhcpd.conf`.**

- Configure the DHCP parameters by assigning the following IP information:
  - **Domain name:** ENPO.dz
  - **DNS servers:** 192.168.10.34, 194.2.10.50
  - **Default gateway:** 192.168.10.1
  - **Lease duration:** 3600 seconds
  - **Network address:** 192.168.10.0
  - **Subnet mask:** 255.255.255.0
  - **IP address range:** 192.168.10.150 to 192.168.10.254

```pgsql
option domain-name "ENPO.dz";
option domain-name-servers 192.168.10.34, 194.2.10.50;
option routers 192.168.10.1;
default-lease-time 3600;

subnet 192.168.10.0 netmask 255.255.255.0 {
    range 192.168.10.150 192.168.10.254;
}
```

## Configuration Example (Solution)

The DHCP configuration file includes the following directives:

- option domain-name "ENPO.dz";
- option domain-name-servers 192.168.10.34, 194.2.10.50;

- option routers 192.168.10.1;
- default-lease-time 3600;
- subnet 192.168.10.0 netmask 255.255.255.0 {

  range 192.168.10.150  192.168.10.254;

  }

## Explanation of the Configuration

- The first directive specifies the **domain name** provided to DHCP clients.
- The second directive defines the **DNS servers** assigned to clients.
- The third directive indicates the **default gateway address**.
- The fourth directive sets the **lease duration** in seconds.
- The last block specifies the **network address**, **subnet mask**, and the **IP address range** managed by the DHCP server.

- ## Restarting the DHCP Service

➢ After completing the configuration, the DHCP service must be started or restarted to apply the changes:

**/etc/init.d/dhcp3-server restart**

```
/etc/init.d/dhcp3-server restart
```

- ## Configuring Static IP Addresses with DHCP

➢ To assign a **fixed IP address** to a host using DHCP, additional entries must be added to the configuration file:

/etc/dhcp3/dhcpd.conf

➢ The following example demonstrates how to assign a static IP address based on the **MAC address** of a host.

host ING {

hardware ethernet 00:A5:5D:F5:08:02;

fixed-address 192.168.10.15;

}

- ## Restarting the DHCP Service

➢ To apply the configuration changes, restart the DHCP daemon:

/etc/init.d/dhcp3-server restart

- ## Client Configuration

➢ To configure client machines, modify the following file:

**/etc/network/interfaces**

- This file should include the following entries to enable DHCP:

auto lo eth0
iface lo inet loopback
iface eth0 inet dhcp

```
auto lo eth0
iface lo inet loopback
iface eth0 inet dhcp
```

- **Restarting the Network Service**
  - After modifying the configuration, restart the networking service to apply the changes:

**/etc/init.d/networking restart**

  - Client Configuration Verification

    Verify that the client has been correctly configured using the appropriate command:

    - **On Linux:** `ifconfig`
    - **On Windows:** `ipconfig /all`

- **Important Note**

The DHCP server itself must be configured with a **static IP address** on interface **eth0**. The network configuration file should include the following parameters:

auto eth0
iface eth0 inet static
address 192.168.10.100
netmask 255.255.255.0
broadcast 192.168.10.255
gateway 192.168.10.1

```
auto eth0
iface eth0 inet static
address 192.168.10.100
netmask 255.255.255.0
broadcast 192.168.10.255
gateway 192.168.10.1
```

# Lab No. 7: DNS Server Installation on Ubuntu Linux

## Objective

- Install and configure a **DNS server** under **Ubuntu Linux**

## Problem Statement

- This practical work describes the **installation and configuration of a DNS server** for a **primary zone** named **ENPO.dz**.
- The DNS zone includes **two name servers**:
  - **dns1.ENPO.dz** with IP address **192.168.10.1**
  - **dns2.ENPO.dz** with IP address **192.168.10.2**
  - Additional services within the domain:
  - A **mail server** named **mail.ENPO.dz** with IP address **192.168.10.3**
  - An **FTP server** named **ftp.ENPO.dz** with IP address **192.168.10.1**

## Important Note

Before deploying the DNS service, ensure that all machines are correctly configured with their respective **IP addresses** using the command **`ifconfig`**.

Assign hostnames to the machines as follows:

- Host **192.168.10.1** → `dns1`
- Host **192.168.10.2** → `dns2`

Use the **`hostname`** command to configure these names.

## DNS Configuration Procedure

To set up name resolution on a Linux server, follow the steps below:

### 1) Install the BIND9 Package

```
sudo apt-get install bind9
```

### 2) Configure DNS Zone Files

The DNS configuration relies on the following files and directories:

- **`/etc/named.conf`**

  Used to declare the **forward zone** and the **reverse zone**.

- **`/var/named/ING/ENPO.dz`**

  Contains the **forward lookup zone file**, which maps hostnames to IP addresses for all machines in the network.

- **`/var/named/ING/1.168.192`**

  Contains the **reverse lookup zone file**, used for reverse name resolution (`in-addr.arpa`).

### 3) DNS Configuration Files

- *named.conf file for the ENPO.dz domain*

  This file specifies the location of the DNS database files and defines the DNS zones.

  options {

  directory "/var/named";

  };

- Root hints configuration

  zone "." in {

  type hint;

  file "named.ca";

  };

- Creating the Primary Zone

  Define the DNS zone as follows:

  zone "ENPO.dz" in {

  type master; # This server acts as the primary DNS server for the domain

  file "path_and_filename_of_forward_zone";

  };

## Creating the Reverse DNS Zone

- Create the reverse lookup zone as follows:

```
zone "1.168.192.in-addr.arpa" in {
    type master;   # Indicates that this is a primary zone
    file "path_and_filename_of_reverse_zone";
};
```

### Note:

The directive **`type master`** specifies that the server is the **primary (authoritative) DNS server** for this zone.

The naming of resource files is not fixed and follows a convention only. A name server can manage multiple domains, and such conventions help organize DNS resource files efficiently.

## Forward Zone Configuration

Configure the forward zone file located at:

```
/var/named/ING/ENPO.dz
```

This zone file must include the declaration of **SOA, NS, MX, and A records**.

## Example Forward Zone File

```
$TTL 86400
@   IN  SOA dns1.ENPO.dz. admin.ENPO.dz. (
        86400
        3600
        3600000
        604800
)
;
```

**Name Server record: declaration of the authoritative DNS server**

```
IN  NS  dns1.ENPO.dz.
```

**Declaration of hosts for name resolution**

**Note:** The absence of a trailing dot allows automatic domain name extension

```
dns1    IN  A   192.168.1.1
```

CNAME alias declarations

The dns1 machine provides mail, FTP, news, and DNS services

```
mail    IN  CNAME  dns1
```

## Reverse Name Resolution for the ENPO.dz Zone

Configure the reverse lookup zone file located at:

```
/var/named/ING/1.168.192
```

## Example Reverse Zone File

```
$TTL 3h
@  IN  SOA dns1.ENPO.dz. admin.ENPO.dz. (
        16
        86400
        3600
        3600000
        604800
)
;
```

**Name Server record**

```
IN  NS  dns1.ENPO.dz.
```

Declaration of nodes in the 1.168.192.in-addr.arpa domain

Fully qualified domain names (ending with a dot) are required

Example extension: 1.1.168.192.in-addr.arpa.

```
1   IN  PTR dns1.ENPO.dz.
```

## Finalizing the Configuration

Starting and Stopping the DNS Service

Use the following commands to stop or start the DNS service and enable name resolution:

```
/etc/rc.d/init.d/named stop
/etc/rc.d/init.d/named start
```

After starting the service, test **name resolution** using the `dig` or `nslookup` commands.

## Client Configuration Procedure

## 1. Windows Client Configuration

This section applies to a **Windows client**. Each client must have the **TCP/IP protocol** enabled and a valid **IP address**.

The client must be configured to specify the **DNS server** it should query.

Proceed as follows:

- Open **Network and Sharing Center**
- Select **Local Area Connection → Properties**

- Choose **Internet Protocol (TCP/IP) → DNS settings**
- Enter the **IP address of the configured DNS server**
- Validate the configuration and **restart the machine**

## 2. Unix/Linux Client Configuration

On Unix/Linux systems, configure the DNS resolver by editing the following file:

```
/etc/resolv.conf
```

Add the following lines:

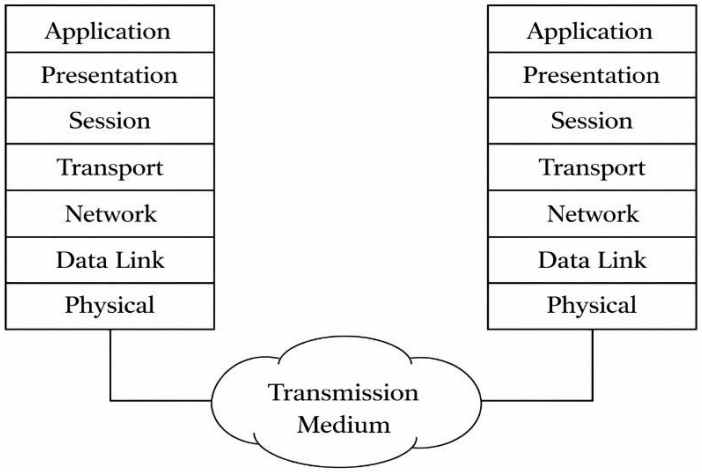
```
search ENPO.dz
nameserver 192.168.1.1    # IP address of your DNS server
```

These settings specify the **default search domain** and the **DNS server** used for name resolution.

# APPENDEIX B

# Tutorial No. 1: OSI Model

## Exercise I:




- In the **OSI** model, what is a **PDU (Protocol Data Unit)?**
- What are the names of the **PDUs** associated with the different **OSI** layers?
- What is meant by a protocol machine?
- Provide some examples of protocol stacks.

## Exercise II: Physical Layer

To design a local area network architecture for a computer laboratory containing **20 workstations**, a **100 Mbps Ethernet network** has been selected.

You are provided with an excerpt from a **technical documentation** to assist you in this task:

| Standards | Connectors | Cables | Maximum Length | Topology | Network Interface Card (NIC) |
|---|---|---|---|---|---|
| 10Base-T | RJ45 | Twisted Pair (UTP Cat 5) | 100 m | Star | TX NIC |
| 100Base-T | RJ45 | Twisted Pair (UTP Cat 5) | 100 m | Star | TX NIC |
| 100Base-TX | RJ45 | Shielded Twisted Pair (STP Cat 5) | 100 m | Star | TX NIC |
| 100Base-FX | ST | Optical Fiber | 1000 m | Star | FX NIC |
| 10Base2 | BNC | Thin Coaxial Cable | 185m | Bus | BNC NIC |
| 10Base5 | Vampire Tap | Thick Coaxial Cable | 500m | Bus | AUI NIC |

- ❖ Which type of cabling would you choose to implement this type of network?
- ❖ Calculate the number of cable segments required.

## Exercise III: Ethernet Frame

The format of the information transmitted over the communication medium is shown below. The fields highlighted in **bold** represent the **Ethernet frame**.

| Preamble | Start Frame Delimiter | Destination Address | Source Address | Type | Informations | FCS |
|---|---|---|---|---|---|---|
| 7 bytes | 1 byte | 6 bytes | 6 bytes | 2 bytes | 46 to 1500 octets | 4 bytes |

- What is the minimum length of an Ethernet frame?
- What is the minimum size of the payload (data) that can be carried?
- Why does the physical layer add a preamble?
- Define an Ethernet address.
- What is the difference between an Ethernet address and an IP address?
- Can two machines have the same Ethernet address? Why?

Below is a trace of a point-to-point communication captured by a line monitoring tool (SNOOP):

ETHER: ----- Ether Header -----

ETHER: Packet 1 arrived at 18:29:10.10

ETHER: Packet size = 64 bytes

ETHER: Destination = 8:0:20:18:ba:40, Sun

ETHER: Source = aa:0:4:0:1f:c8, DEC (DECNET)

ETHER: Ethertype = 0800 (IP)

```
ETHER: ----- Ether Header -----
ETHER: Packet 1 arrived at 18:29:10.10
ETHER: Packet size = 64 bytes
ETHER: Destination = 8:0:20:18:ba:40, Sun
ETHER: Source = aa:0:4:0:1f:c8, DEC (DECNET)
ETHER: Ethertype = 0800 (IP)
```

This trace is to be compared with a **group communication**:

ETHER: ----- Ether Header -----

ETHER: Packet 1 arrived at 11:40:57.78

ETHER: Packet size = 60 bytes

ETHER: Destination = ff:ff:ff:ff:ff:ff, (broadcast)

ETHER: Source = 8:0:20:18:ba:40, Sun

ETHER: Ethertype = 0806 (ARP)

```
ETHER: ----- Ether Header -----
ETHER: Packet 1 arrived at 11:40:57.78
ETHER: Packet size = 60 bytes
ETHER: Destination = ff:ff:ff:ff:ff:ff, (broadcast)
ETHER: Source = 8:0:20:18:ba:40, Sun
ETHER: Ethertype = 0806 (ARP)
```

 Which field indicates whether the communication is intended for a single destination or multiple destinations?

 How can a single message be delivered to multiple recipients simultaneously?

## Exercise IV

1. Compare Ethernet 10Base-T and Ethernet 100Base-TX by identifying their common features and key differences.

   Recall that the **minimum Ethernet frame size is 64 bytes**.
   For the purpose of this exercise, assume that the **signal propagation speed** in the Ethernet transmission medium is **200,000 km/s**.

2. What is the theoretical maximum distance that can separate two stations in a shared Ethernet **100Base-T** network (using repeaters)?

   In real deployments, the recommended distances are **90 meters** between a station and a repeater over **UTP Category 5 cabling**, with the possibility of cascading **two repeaters**, separated by **5 meters of cable**.

3. What explains the difference between the theoretical result obtained in the previous question and the practical values recommended?

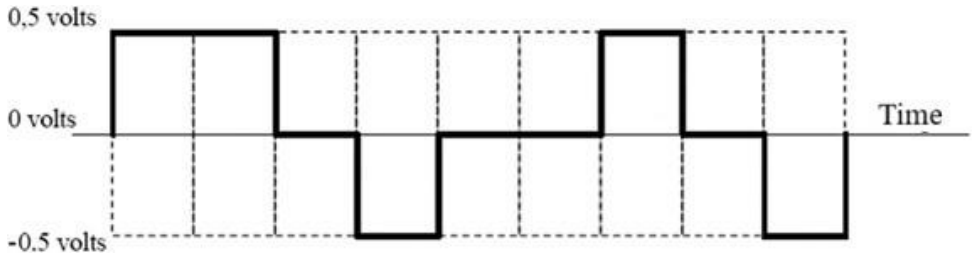
   Some Ethernet standards allow communication distances of up to 2 kilometers when **optical fiber** is used. Which type of Ethernet corresponds to such standards?

# Tutorial No. 2: Data Encoding

## Exercise I

An **oscilloscope** connected to a **local area network cable** displays the signal shown in the following figure.

This signal represents the **bit modulation at the physical layer**.




Does this signal correspond to baseband encoding or to carrier-wave modulation encoding? Justify your answer.

1. What is the binary sequence encoded by this signal?
2. What is the name of this signal encoding technique?
3. Specify the characteristics of the local area network that uses this encoding, including:
    - transmission medium,
    - connector type,
    - network topology,
    - Data rate.

## Exercise II

Consider a polynomial code generated by the polynomial

$$G(X)=X^4+X+1.$$

1. Explain the principle of polynomial codes. How many redundant bits are introduced by the polynomial **G(X)?**
2. **What is the length of the original data words**Which **error types** can this code detect?

We now aim to compute the redundancy bits for the message **M=1101011011**

3. Determine the polynomial **D(X)** associated with the binary sequence **M**.
4. Calculate the remainder **R(X)** obtained by dividing $\mathbf{D(X)\cdot X^4}$ by **G(X).**
5. Deduce the final encoded message that will be transmitted.

# Tutorial No. 3: IPv4 Addressing

## Exercise I

An **IPv4 address** assigned to a host is composed of two parts: a **network identifier** and a **host identifier**.

For example, a machine with the IPv4 address **192.33.159.6** has:

- **Network identifier:** 192.33.159 **(3 octets)**
- **Host identifier:** 6 **(1 octet)**

1. On the Internet, is it possible for two hosts located in different geographical areas to share the same IPv4 address?
2. If so, under which conditions can this situation occur?
3. Within the same IPv4 network, can two distinct hosts be assigned the same IPv4 address at different times? Provide a justification for your answer.

The following output corresponds to the execution of the UNIX ifconfig command on a machine:

**le0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu 1500**

**inet 192.33.159.212 netmask ffffff00 broadcast 192.33.159.255**

**ether 8:0:20:18:ba:40**

```
le0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu 1500
inet 192.33.159.212 netmask ffffff00 broadcast 192.33.159.255
ether 8:0:20:18:ba:40
```

4. Identify and explain the informations that can be obtained from this output.

## Exercise II: Address Classes

Consider the following IPv4 addresses:

- **204.150.200.95**
- **138.100.65.5**
- **18.181.0.31**
- **226.192.60.40**

## Questions

1. Determine the address class of each IPv4 address listed above.
2. Specify the number of usable host addresses available for each class.

## Answers

- **204.150.200.95**
  - The first octet is **204**, which in binary is **11001100**.
  - This address belongs to **Class C**.
  - A Class C network provides $2^8-2=254$ usable host addresses.
  - The two excluded addresses correspond to the **network address** and the **broadcast address** (e.g., broadcast: **204.150.200.255,** network: **204.160.241.0**).

- **138.100.65.5**
  - The first octet is **138**, which in binary is **10001010**.
  - This address belongs to **Class B**.
  - A Class B network offers $2^{16}-2=65,534$ usable host addresses.

- **18.181.0.31**
  - The first octet is **18**, which in binary is **00010010**.
  - This address belongs to **Class A**.
  - A Class A network allows $2^{24}-2=16,777,214$ usable host identifiers.

- **226.192.60.40**
  - The first octet is **226**, which in binary is **11100010**.
  - This address belongs to **Class D**, which is reserved for **multicast communication**.
  - Addresses in this class are reserved for multicast transmission and are not assigned to individual hosts.

## Exercise III

Consider the following IPv4 addresses:

1. **145.200.40.225**
2. **202.2.50.100**
3. **97.124.30.140**

For each address, determine:

1. The address class
2. The default network mask
3. The network address and the broadcast address
4. The first and last usable host addresses

## Answers

### 1. Address: 145.200.40.225

- The first octet is **145**, which falls within the range **[128–191] → Class B**
- Default network mask: **255.255.0.0**
- Network address: **145.200.0.0**
- Broadcast address: **145.200.255.255**
- First usable host address: **145.200.0.1**
- Last usable host address: **145.200.255.254**

### 2. Address: 202.2.50.100

- The first octet is **202**, which belongs to the range **[192–223] → Class C**
- Default network mask: **255.255.255.0**
- Network address: **202.2.50.0**
- Broadcast address: **202.2.50.255**
- First usable host address: **202.2.50.1**
- Last usable host address: **202.2.50.254**

### 3. Address: 97.124.30.140

- The first octet is **97**, which lies in the range **[0–127] → Class A**
- Default network mask: **255.0.0.0**
- Network address: **97.0.0.0**
- Broadcast address: **97.255.255.255**
- First usable host address: **97.0.0.1**
- Last usable host address: **97.255.255.254**

# Tutorial No.4: IPv4 Subnetting and CIDR

## Exercise I

In a **Class B network**, it is required to create several subnets using the following subnet mask:

**255.255.255.240**

1. How many hosts can be supported per subnet?
2. How many subnets can be created using this mask?
3. 

## Answer

The subnet mask **255.255.255.240** allocates **4 bits** from the host portion to identify subnet IDs. As a result, **12 bits** remain for host addressing.

- **Number of hosts per subnet:**

  $$2^{12}-2$$

  *(Excluding the network address and the broadcast address)*

- **Number of available subnets:**

  $$2^{4}=16$$

## Exercise II

Consider the network **192.168.1.0/24**, which must be divided into **three subnets** with the following host requirements:

- **Subnet 1:** 60 hosts
- **Subnet 2:** 100 hosts
- **Subnet 3:** 20 hosts

For **each subnet**, determine:

- Subnet address
- Subnet mask
- Usable host address range
- Broadcast address

## Answer :

| Range | Subnet mask | Subnet address | Broadcast address | Host address range |
|---|---|---|---|---|
| 192.168.1.0 _ 192.168.1.127 | 255.255.255.128(/25) | 192.168.1.0/25 | 192.168.1.127 | 192.168.1.1 _ 192.168.1.126 |
| 192.168.1.128 _ 192.168.1.191 | 255.255.255.192(/26) | 192.168.1.128/26 | 192.168.1.191 | 192.168.1.129 _ 192.168.1.190 |
| 192.168.1.192 _ 192.168.1.223 | 255.255.255.224(/27) | 192.168.1.192/27 | 192.168.1.223 | 192.168.1.193 _ 192.168.1.222 |

## Exercise III

A company owns **73 computers** that must be distributed across **three subnets**.
The company uses the IPv4 network **192.168.0.0/24**.

The required subnet sizes are as follows:

- **Subnet 1:** 21 hosts
- **Subnet 2:** 29 hosts
- **Subnet 3:** 23 hosts

Answer the following questions:

1. Identify the IP address class of the given network.
2. Determine the number of bits required for subnetting.
3. Compute the appropriate subnet mask.
4. Calculate the maximum number of hosts that can be configured in each subnet.
5. Determine the first and last usable IP addresses assigned to the hosts in each subnet.

# REFERENCES

1. **Postel, J.** *Internet Protocol (IPv4)*, RFC 791.Internet Engineering Task Force (IETF), **September 1981**.
2. **Droms, R.** *Dynamic Host Configuration Protocol*, RFC 2131. IETF, **March 1997**.
3. **Mockapetris, P.** *Domain Names – Concepts and Facilities*, RFC 1034. IETF, **November 1987**.
4. **Mockapetris, P.** Domain Names – Implementation and Specification, RFC 1035.IETF, **November 1987.**
5. **Telecommunications Industry Association (TIA)** *ANSI/TIA-568-C.2: Balanced Twisted-Pair Telecommunications Cabling and Components Standard*.TIA, **2009**.
6. **Forouzan, B. A.** *Data Communications and Networking*, 5th Edition. McGraw-Hill Education, New York, **2013**.
7. **Tanenbaum, A. S., & Wetherall, D. J.** *Computer Networks*, 5th Edition. Pearson Education, Upper Saddle River, NJ, **2011**.
8. **Cisco Networking Academy** *Introduction to Networks (ITN)*. Cisco Systems, **2020**.