

# **MODULE-5**

# Syllabus

- Application layer
- HTTP,
- FTP,
- SMTP,
- DNS.
- Network security:
- Common Threats- Firewalls (advantages and disadvantages),
- Cryptography.

# Application Layer

- 7th Layer.
- Responsible for providing services to the user.
- Most popular protocols are:
  - TELNET- Remote Logging
  - SMTP-For electronic mail
  - FTP- File Transfer

# REMOTE LOGGING

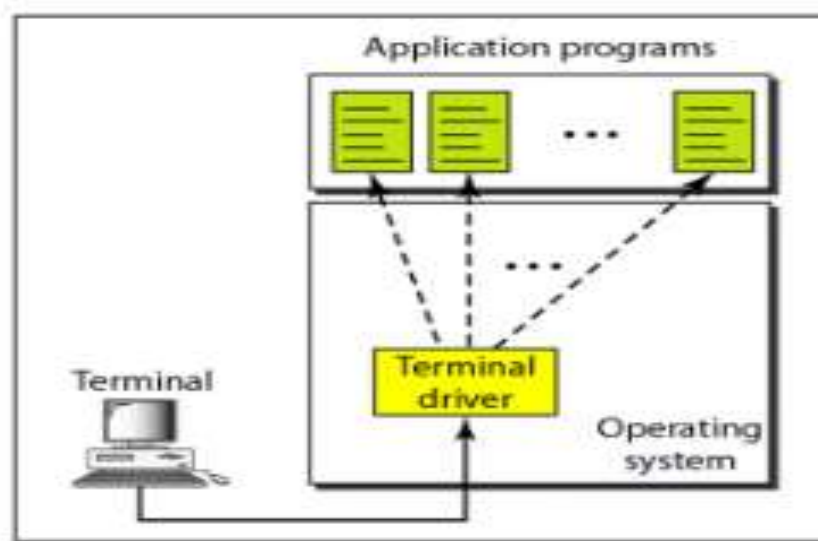
- On the Internet, users may want to run application programs at a remote site and create results that can be transferred to their local site.
- For example, students may want to connect to their university computer lab from their home to access application programs for doing homework assignments or projects

# TELNET

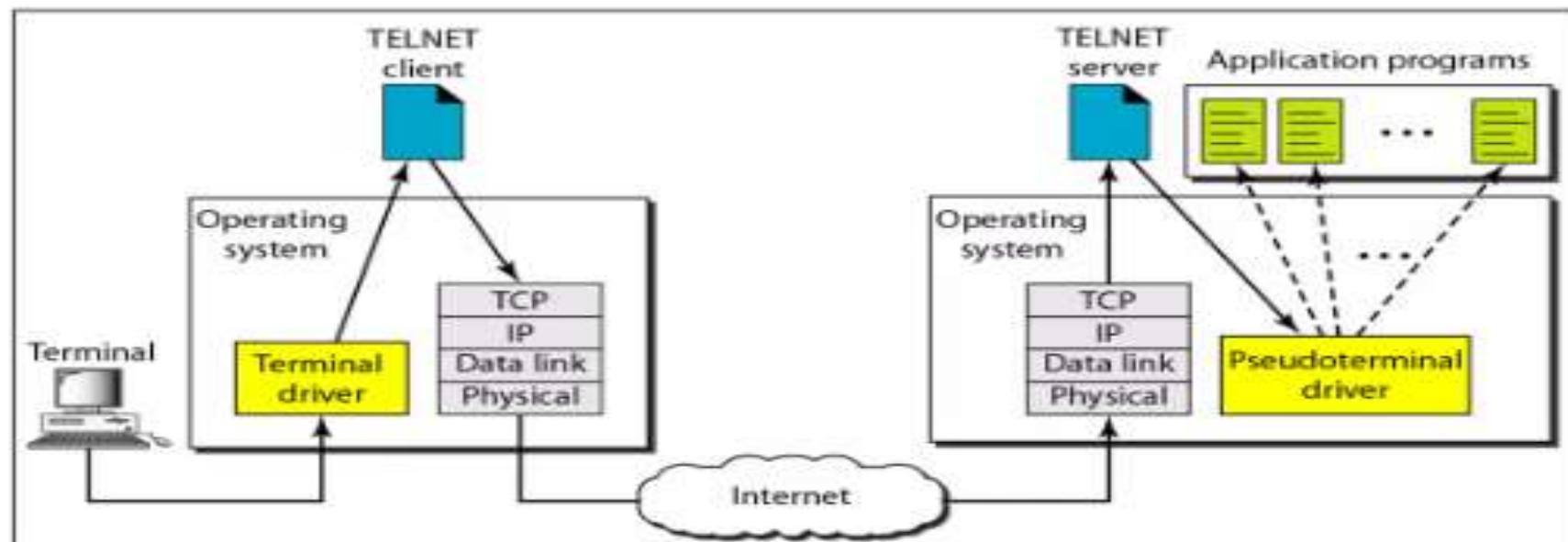
- TELNET is a general-purpose client/server application program
- TELNET is an abbreviation for TErminaL NETwork.
- TELNET enables the establishment of a connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system

# Logging Process

- Timesharing Environment-In such an environment, a large computer supports multiple users.
- In a timesharing environment, users are part of the system with some right to access resources
- To access the system, the user logs into the system with a user id or log-in name. The system also includes password checking to prevent an unauthorized user from accessing the resources.



a. Local log-in



b. Remote log-in

## **Local login**

- The user's terminal is directly connected to the computer. The terminal driver accepts the user's keystrokes and forwards them to the operating system, which then launches the required application.

## **Remote Login**

- The user's computer is connected to the computer they want to access through the internet. Remote login allows users to access a network from a remote location, without being physically present at the computer.



- Here(Remote Login) the TELNET client and server programs come into use.
- The user sends the keystrokes to the terminal driver, where the local operating system accepts the characters but does not interpret them.
- The characters are sent to the TELNET client, which transforms the characters to a universal character set called network virtual terminal (NVT) characters and delivers them to the local TCP/IP protocol stack.

- The commands or text, in NVT form, travel through the Internet and arrive at the TCP/IP stack at the remote machine.
- Here the characters are delivered to the operating system and passed to the TELNET server, which changes the characters to the corresponding characters understandable by the remote computer.

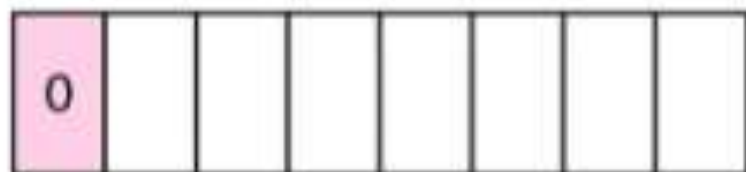
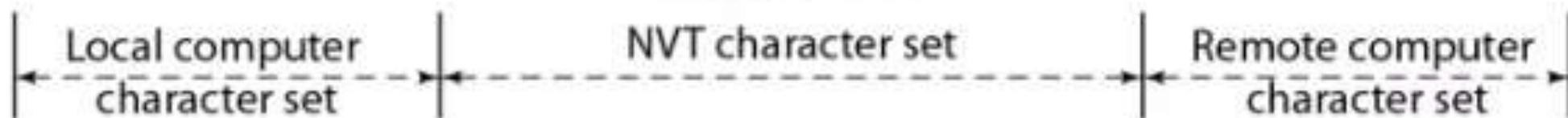
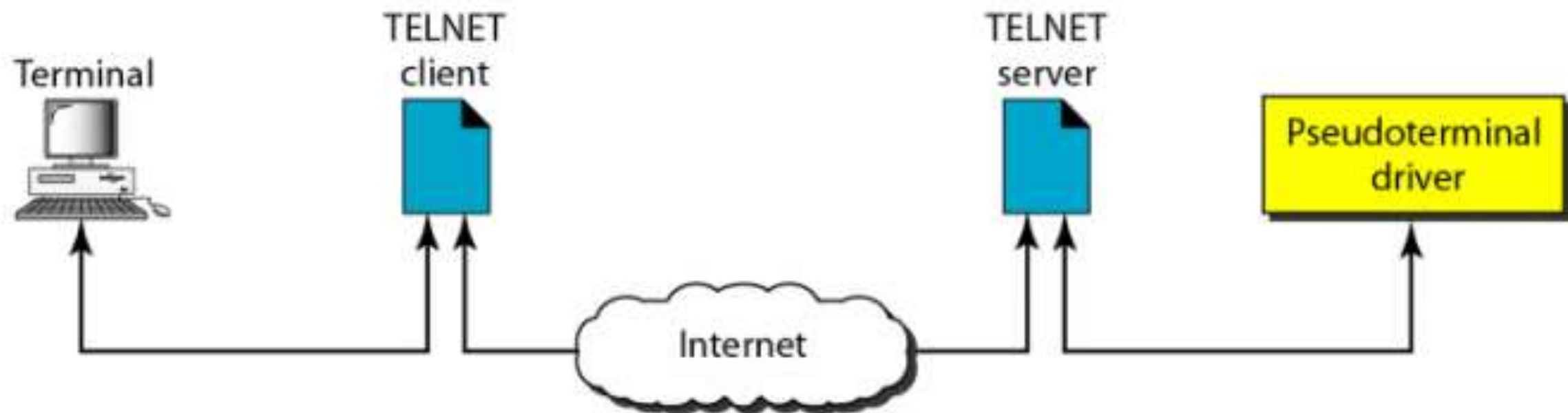
- However, the characters cannot be passed directly to the operating system because the remote operating system is not designed to receive characters from a TELNET server:
  - It is designed to receive characters from a terminal driver.
  - The solution is to add a piece of software called a *pseudoterminal driver* which pretends that the characters are coming from a terminal.
- The operating system then passes the characters to the appropriate application program

# Network Virtual Terminal

- The mechanism to access a remote computer is complex.
- Because every computer and its operating system accept a special combination of characters as tokens.
- For example, the end-of-file token in a computer running the DOS operating system is Ctrl+z, while the UNIX operating system recognizes Ctrl+d. We are dealing with **heterogeneous systems**.

# Solution

- TELNET solves this problem by defining a universal interface called the network virtual terminal (NVT) character set.
- Via this interface, the client TELNET translates characters (data or commands) that come from the local terminal into NVT form and delivers them to the network.
- The server TELNET, on the other hand, translates data and commands from NVT form into the form acceptable by the remote computer



Data character



Control character

# SMTP

- **Simple Mail Transfer Protocol**
- Protocol for transmitting and receiving email messages.
- connection-oriented scenario.
- The SMTP client, the initiating agent, sender, or transmitter, initiates the communication session.
- It issues the command strings and opens the session for corresponding responses from the SMTP server, which involves the listening agent or receiver.

# SMTP email transaction follows four command

## 1. HELO/EHLO command

- It tells the email server that the client wants to start the mail transaction. The client mentions its domain name after this command.

---

Sending server	HELO client.gmail.com	Identifies itself with the domain name or IP address to start the conversation
----------------	-----------------------	--------------------------------------------------------------------------------

---

Receiving server	250	OK
------------------	-----	----



## 2. MAIL command

- Specifies the sender of the email
- It lays down the bounce address/return address, defining the return or reverse paths.

Sending server	MAIL FROM <a href="mailto:mark@gmail.com">mark@gmail.com</a>	Specifies the sender of the email
Receiving server	250	OK

### 3.RCPT command

- It specifies the recipient of the message.

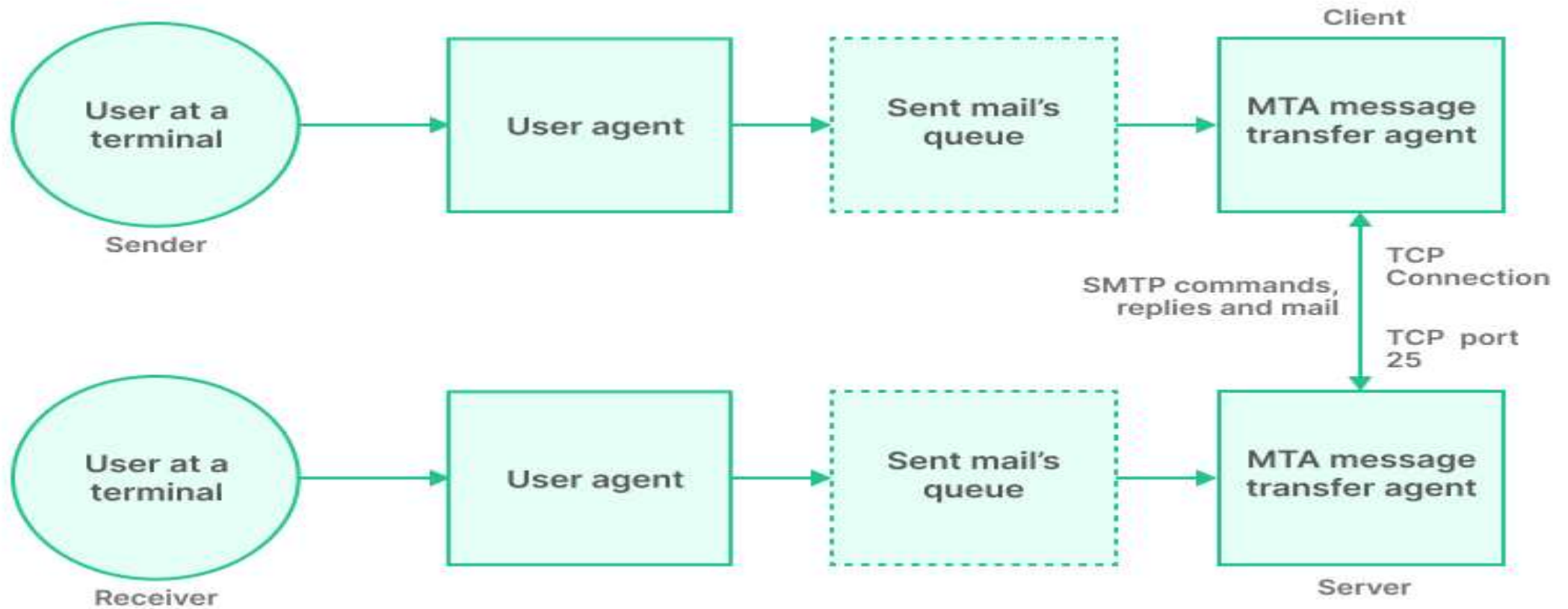
Sending server	RCPT TO <a href="mailto:clair@yahoo.com">clair@yahoo.com</a>	Specifies the recipient of the email
Receiving server	250	OK

## **4. DATA**

- It shows where the content of the message starts

Sending server	DATA	Request to start the transfer of email contents, date, subject line, etc.
Receiving server	354	Allows the sender to start the transfer of information
Sending server	Date: Sunday, 30 July 2023	Date of the email
Sending server	Subject: Welcome to the party!	Subject of the email
Sending server		Empty line
Sending server	We welcome you to the party. We hope to have a lot of fun together	Content of the email
Sending server	.	Terminates the transfer of email contents
Receiving server	250	Ok
Sending server	QUIT	Request to terminate the session
Receiving server	221	Closing session

# Components of SMTP



## **User-Agent (UA)**

- The **User-Agent (UA)** is responsible for preparing, creating, and putting the message in the form of an envelope for transmission.

## **Mail Transfer Agent (MTA)**

- The **Mail Transfer Agent (MTA)** then transfers this message to the recipient across the internet.

# How are emails sent using SMTP?

## 1. Composition

- With the help of a Mail User Agent (MUA) program, the user sends an email. The content of the email consists of two parts, the email header and the email body.

## 2. Submission

- The mail client (mail user agent, MUA) submits the email to a mail server (known as a mail submission agent, MSA). The MSA further delivers the mail to its mail transfer agent, MTA.

### 3. Mail delivery

- The two parts of an email address are the recipient's **username** and the **domain name**.
- For example, [mayank@gmail.com](mailto:mayank@gmail.com), 'Mayank' is the username, and '[gmail.com](mailto:mayank@gmail.com)' is the domain.
- If the **domain name** of the recipient's email address **does not match** the sender's domain name, then the MTA will search for the particular domain to **relay the mail**.
- This email transfer from one SMTP server to another is called an **SMTP relay**.



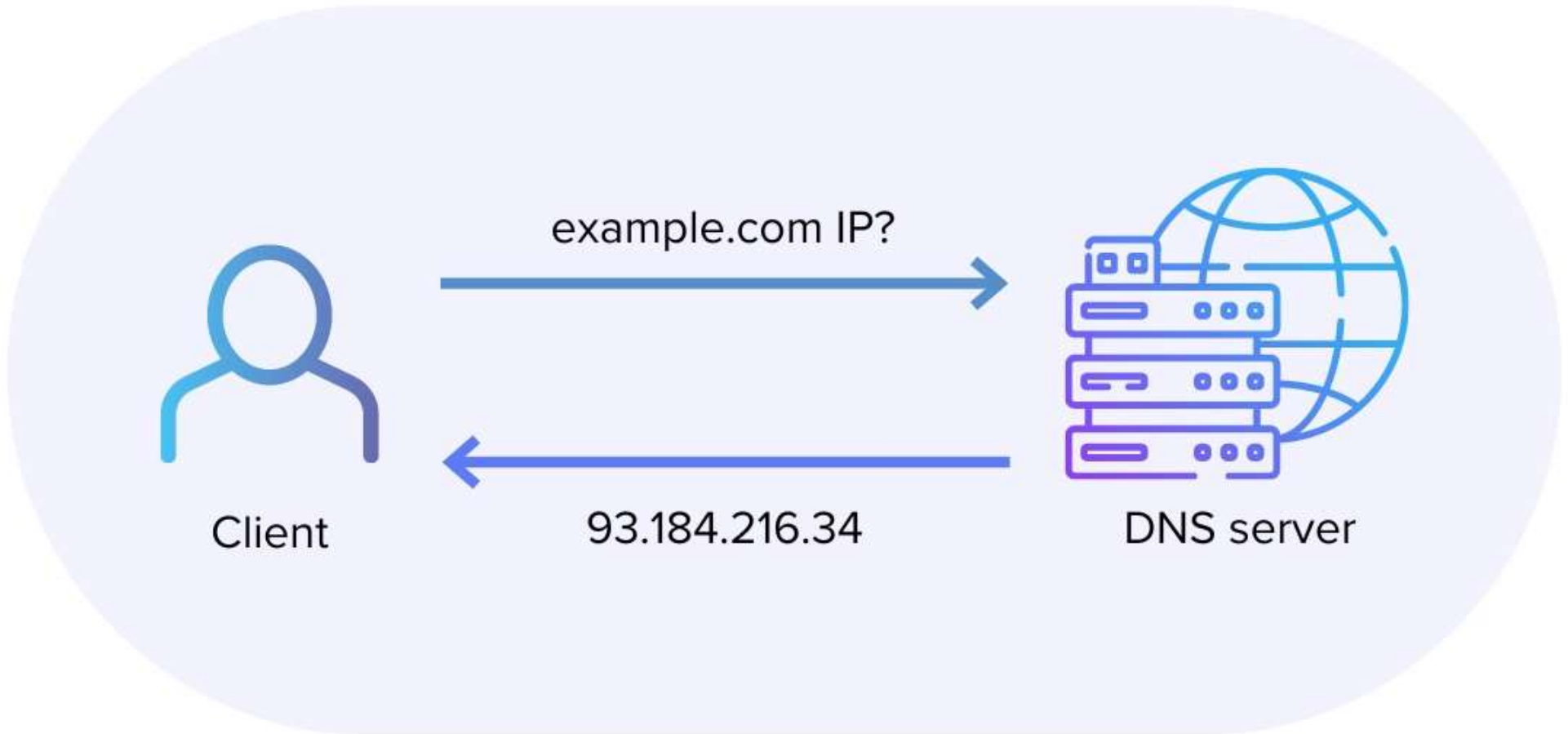
## 4. Receipt and processing

- After receiving the incoming message, the exchange server delivers it to the **Mail Delivery Agent**. Then, it stores the email and waits for the user to retrieve it.

## 5. Access and retrieval

- The user can access the MUA with the login and password. In addition, the MUA helps retrieve the email stored by the MDA.

# DNS



# DNS

- Every host is identified by the IP address but remembering numbers is very difficult for people also the IP addresses are not static therefore a mapping is required to change the domain name to the IP address.
- The DNS system serves as a kind of "phone book" that stores a database of domain names and their corresponding addresses.

- Without DNS, you would have to remember long strings of numbers to visit your favorite websites.
- DNS translates domain names to IP addresses

# Types of Domain

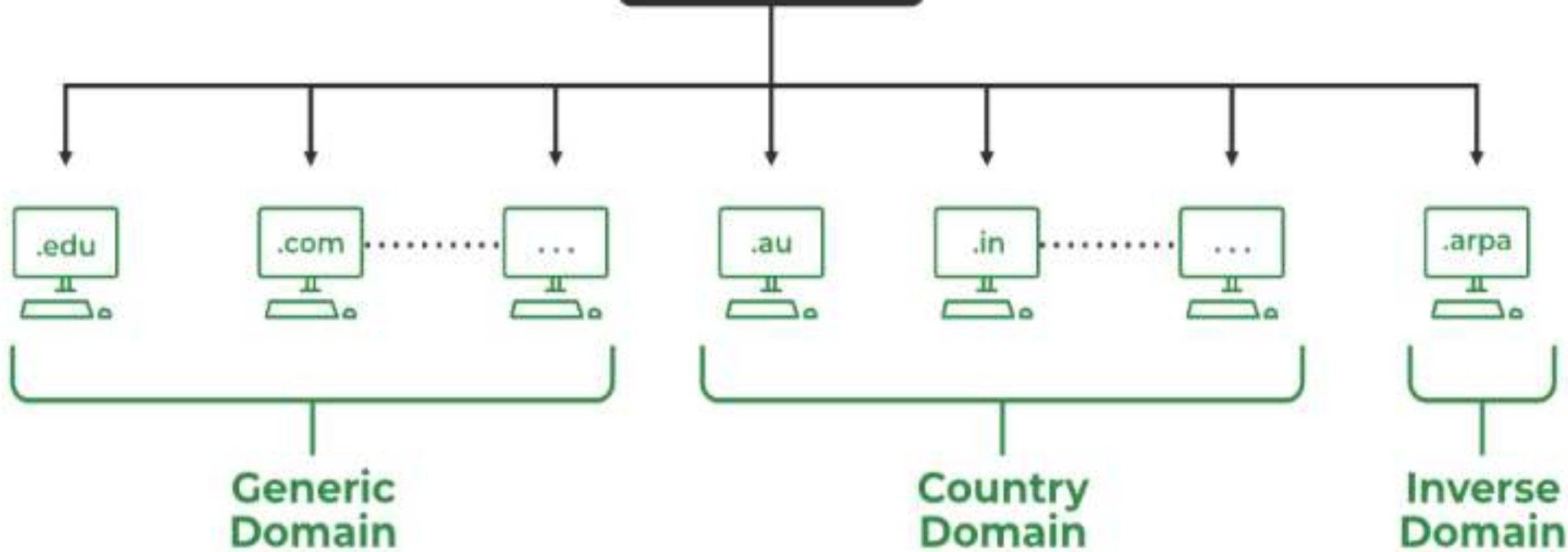
There are various kinds of domains:

- **Generic Domains: It is** a domain that is not associated with any specific country or geographical location, unlike country code TLDs (ccTLDs). Instead, gTLDs are generally used to represent different types of organizations, industries, or purposes.
- Examples-.com(commercial), .edu(educational), .mil(military), .org(nonprofit organization), .net(similar to commercial) all these are generic domains.

- **Country Domain:** (DNS) that is reserved for a specific country, sovereign state, or territory. Each ccTLD is two letters long, and it is assigned based on the country codes defined by the **ISO**
- Examples-.in (India) .us .uk

- **Inverse Domain:** if we want to know what is the domain name of the website. IP to domain name mapping.
- It is a type of DNS (Domain Name System) query that is used to map an **IP address** back to its associated domain name. This process is called **Reverse DNS Lookup** or simply **rDNS**.
- Forward DNS: Resolves a domain name (e.g., www.example.com) to an IP address (e.g., 192.0.2.1).
- Reverse DNS (Inverse Domain): Resolves an IP address (e.g., 192.0.2.1) to a domain name (e.g., www.example.com).

# Root





# Domain Name Space

- The **Domain Name Space** refers to the **hierarchical structure** used in the Domain Name System (DNS) to manage and organize domain names.
- This hierarchical naming system is essential for organizing the vast number of domains on the internet and **ensuring unique domain names.**
- The domain name space is organized in a **tree-like hierarchy** with different levels, each separated by dots **(.)**.
- The hierarchy flows from general (the root) to specific (host names or individual resources).



# Sections of Domain Name

## 1. Root Level

- Represented as a **dot (.)** and typically invisible in domain names.
- It is the starting point of the domain name space, managing the top-level domains.

## 2. Top-Level Domain (TLD)

- Directly below the root, TLDs are the **most general part** of a domain name (e.g., .com, .org, .net, country-specific like .uk or .in).

## Example:

`mail.sales.example.com`

## Complete Example Breakdown:

- **Root Level ( . ):** The invisible dot at the end of the domain.
- **TLD:** `.com`
- **Second-Level Domain:** `example`
- **Subdomain:** `sales`
- **Host Name:** `mail`

### 3. Subdomain

- A part of the domain that is a **subdivision** of the second-level domain.  
Organizations can create multiple subdomains to represent different services, departments, or functions.
- For instance, sales.example.com might represent the sales department of the company.

## Host/Resource Names:

- The **lowest level** in the domain name hierarchy, representing **specific devices or services**
- The host name is usually the first part of the full domain.
- In mail.sales.example.com, **"mail"** is the **hostname**, which might represent a specific mail server within the sales department.

## Example:

`mail.sales.example.com`

## Complete Example Breakdown:

- **Root Level ( . ):** The invisible dot at the end of the domain.
- **TLD:** `.com`
- **Second-Level Domain:** `example`
- **Subdomain:** `sales`
- **Host Name:** `mail`

# Fully Qualified Domain Name (FQDN):

- A **Fully Qualified Domain Name (FQDN)** is a domain name that specifies its **exact location** in the domain name hierarchy of the Domain Name System (DNS).
- It includes the **entire domain name** from the host to the top-level domain (TLD), and it ends with a dot representing the root of the DNS hierarchy.

# Characteristics of an FQDN:

- **Complete**
- **Ends with a dot**
- **Globally Unique**

## Example of an FQDN:

- `www.example.com.`
  - `www` → Hostname
  - `example` → Second-level domain (organization's name)
  - `.com` → Top-level domain (TLD)
  - `.` → Root (DNS hierarchy root)



# Partially Qualified Domain Name (PQDN)

- A **Partially Qualified Domain Name (PQDN)** is an incomplete domain name that does **not include all parts** necessary to specify its location in the DNS hierarchy.
- It may omit parts like the top-level domain (TLD) or the root dot.

## Characteristics of a PQDN

1. Incomplete
2. Relative Resolution
3. Local Scope

mail. sales → Could refer to mail.sales.example.com within an organization's internal DNS.

Aspect	Fully Qualified Domain Name (FQDN)	Partially Qualified Domain Name (PQDN)
Completeness	Complete domain name including TLD and root	Incomplete, may omit parts like TLD
DNS Hierarchy	Specifies the exact location in the DNS hierarchy	Relies on default context for resolution
Usage	Used when global, absolute resolution is required	Used for local or relative domain name resolution
Example	<code>www.example.com.</code>	<code>www</code> or <code>mail.sales</code>
Scope	Valid globally across the internet	Often valid only within a local network or domain

# Domain Name Resolution

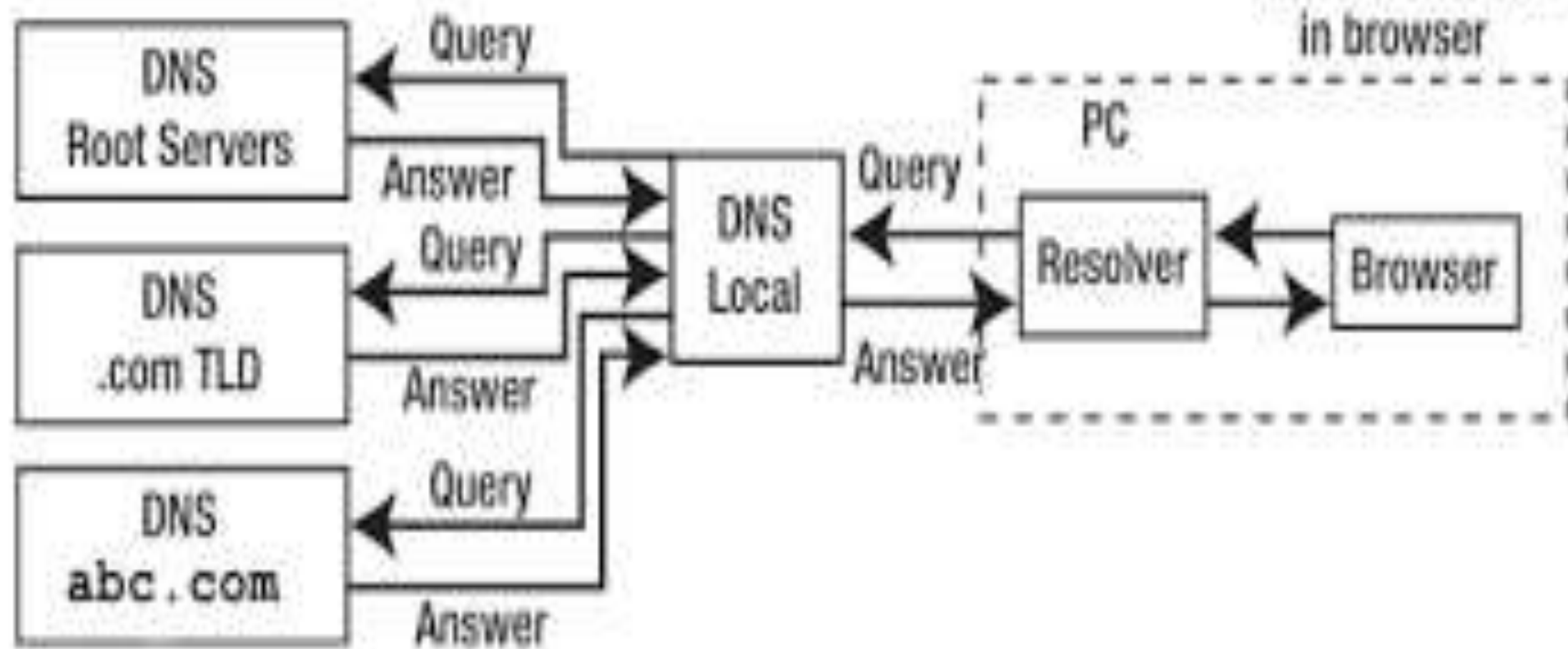
Mapping a domain name to an IP Address is known as Name-Address Resolution.

- ✓ The user enters a web address or domain name into a browser.
- ✓ The browser sends a message, called a *recursive DNS query*, to the network to find out which IP or network address the domain corresponds to.

- ✓ The query goes to a recursive DNS server, which is also called a recursive resolver, and is usually managed by the internet service provider ([ISP](#)).
- ✓ If the recursive resolver has the address, it will return the address to the user, and the webpage will load.
- ✓ If the recursive DNS server does not have an answer, it will query a series of other servers in the following order:
  - ✓ DNS root name servers,
  - ✓ top-level domain (TLD) name servers and
  - ✓ authoritative name servers.

- ✓ The three server types work together and continue redirecting until they retrieve a DNS record that contains the queried IP address. It sends this information to the recursive DNS server and the webpage the user is looking for loads.
- ✓ DNS root name servers and TLD servers primarily redirect queries and rarely provide the resolution themselves. The recursive server/resolver stores, or [caches](#), the a record for the domain name, which contains the IP address.
- ✓ The next time it receives a request for that domain name, it can respond directly to the user instead of querying other servers.
- ✓ If the query reaches the authoritative server and it cannot find the information, it returns an error message.

User types  
**www.abc.com**  
in browser



# Network Security

- **Confidentiality:** Ensuring that data is accessible only to authorized users or systems.
- **Integrity:** Protecting data from being altered, tampered with, or destroyed without authorization.
- **Availability:** Ensuring that network services and data are available to authorized users when needed.
- **Authentication:** Verifying the identity of users or systems to ensure that only legitimate entities are granted access.
- **Non-repudiation:** Preventing a sender or receiver from denying the transmission or reception of a message.



# Types of Attacks

- **Passive Attacks:** The attacker only monitors or intercepts the data being transmitted, such as eavesdropping or traffic analysis. These attacks aim to compromise confidentiality.
- **Active Attacks:** The attacker modifies or disrupts the transmission. Examples include man-in-the-middle attacks, replay attacks, and Denial-of-Service (DoS) attacks. These target the integrity, availability, and sometimes authentication.

# Passive Attacks

## Eavesdropping (or Sniffing):

- In this attack, the attacker listens in on network communications to capture and analyze data, such as passwords, private messages, or sensitive information.

## Traffic Analysis:

- Here, the attacker analyzes patterns of communication, such as the frequency, volume, and timing of messages between hosts, to infer sensitive information.

# Active Attacks

## **Man-in-the-Middle (MitM) Attack:**

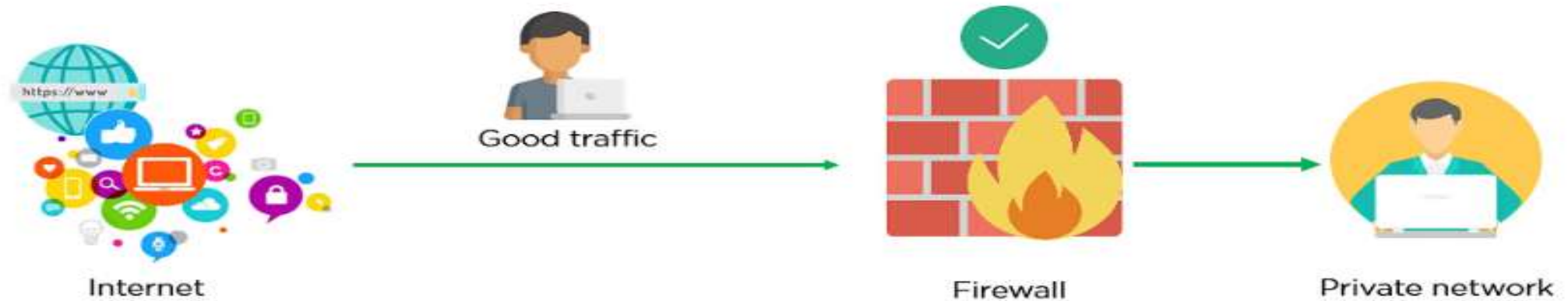
- The attacker secretly intercepts and alters communication between two parties without them knowing.

## **Denial-of-Service (DoS) Attack:**

- The attacker floods a target server, network, or service with excessive traffic, overwhelming its resources and rendering it unavailable to legitimate users.  
Overloading a website's server with excessive requests to make it crash.

# Firewalls


- Firewalls are essential components of network security, acting as barriers that control traffic between networks based on predefined security rules



# Key Functions of Firewalls:

- ✓ **Traffic Filtering:** Firewalls screen data packets (pieces of data) in the network's flow-in and flow-out directions, allowing or blocking them according to certain rules.
- ✓ **Access Control:** They decide which applications, services, and devices can access the network, thus protecting sensitive resources.
- ✓ **Threat Detection:** Some of them can detect and prevent other types of threats, such as viruses, malware, or even suspicious behavior.

# Advantages and Disadvantages of Firewalls

ADVANTAGES & DISADVANTAGES OF A FIREWALL	
Monitors Traffic	Server Resources Performance
Prevents Hacker Virus & Malware Injection	Hardware Firewall is Higher Cost
Access Control	User Restrictions
Software Firewall is Cost-Efficient	Complex Operations
Privacy	Internal Network Attacks
Easy Installation	
 Liquid Web™	

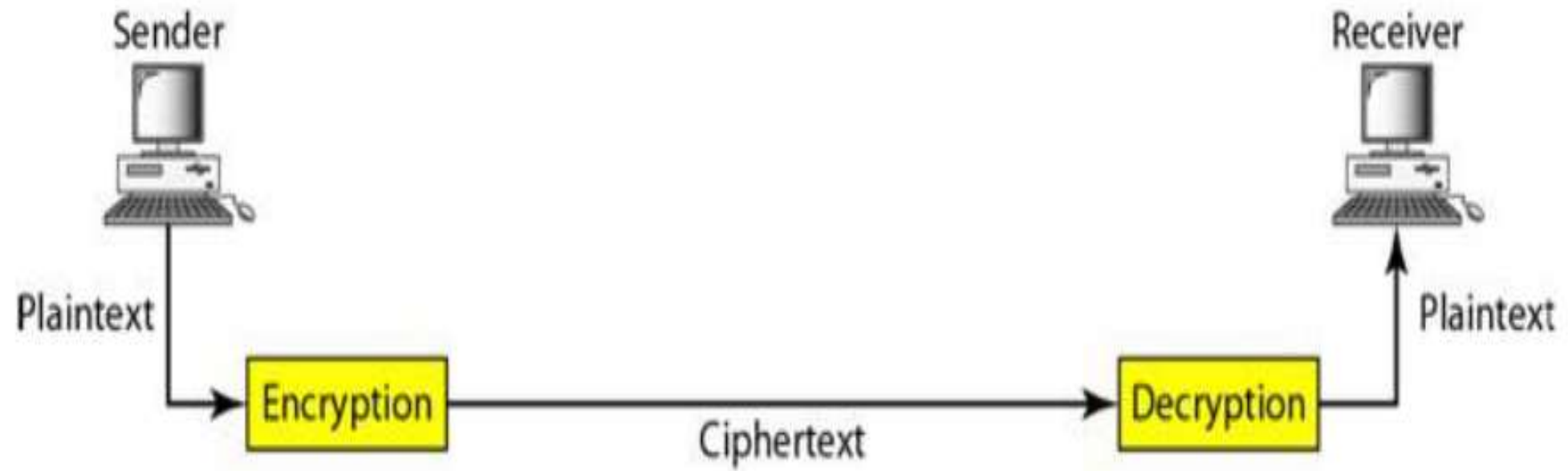
# Cryptography

- Cryptography, a word with Greek origins, means "secret writing."
- Cryptography is the science of securing data by transforming it into an unreadable format, called ciphertext, which can only be converted back to its original form, or plaintext, by authorized entities.

# Primary goals of cryptography

- ✓ **Confidentiality:** Ensuring that only authorized individuals can read the data.
- ✓ **Integrity:** Protecting the data from being altered or tampered with.
- ✓ **Authentication:** Verifying the identities of the entities involved in communication.
- ✓ **Non-repudiation:** Preventing either the sender or receiver from denying that they participated in the communication.





# Cryptography Components

## Plaintext and Ciphertext

- The **original message**, before being transformed, is called **plaintext**. After the message is transformed, it is called ciphertext.
- An encryption algorithm transforms the plaintext into ciphertext; a decryption algorithm transforms the ciphertext back into plaintext.
- The sender uses an encryption algorithm, and the receiver uses a decryption algorithm.

# Cipher

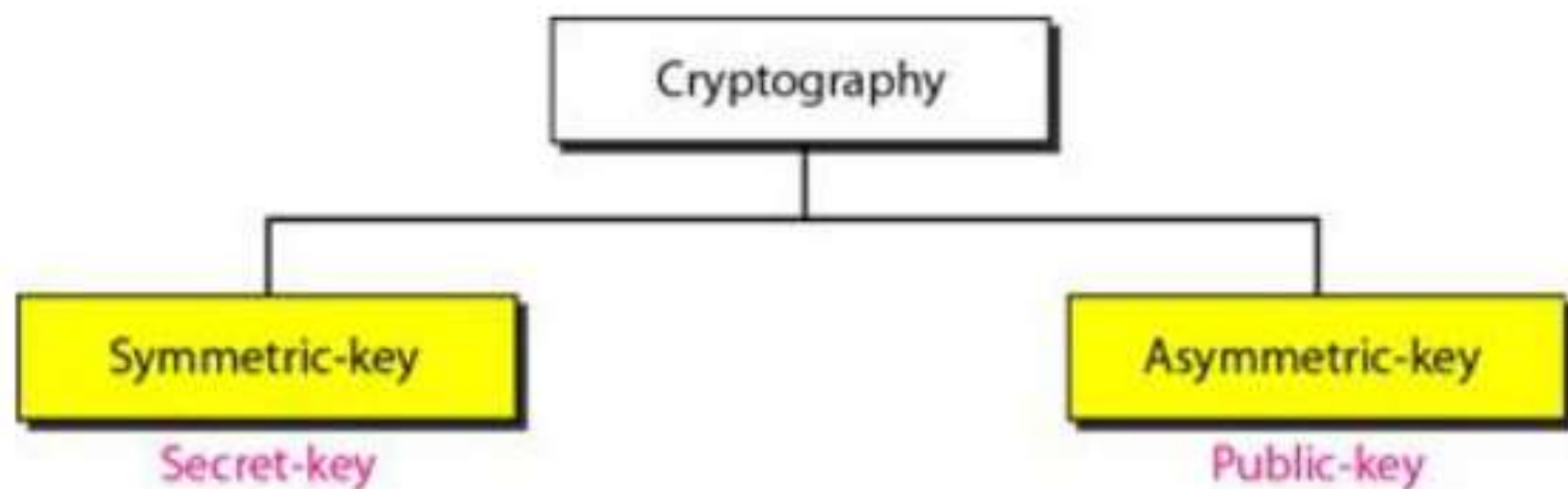
- Encryption and decryption algorithms as ciphers.
- The term cipher is also used to refer to different categories of algorithms in cryptography.

## Key

- A key is a number (or a set of numbers) that the cipher, as an algorithm, operates on.
- To encrypt a message, we need an encryption algorithm, an encryption key, and the plaintext. These create the ciphertext.
- To decrypt a message, we need a decryption algorithm, a decryption key, and the ciphertext. These reveal the original plaintext.

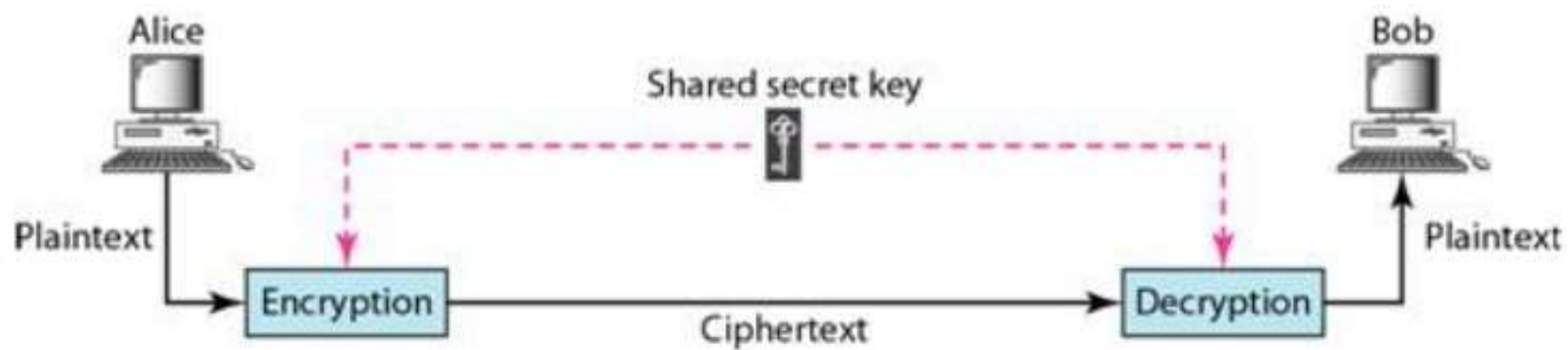
# Two Categories

- We can divide all the cryptography algorithms (ciphers) into two groups:
- Symmetric key (also called secret-key) cryptography algorithms and
- Asymmetric (also called public-key) cryptography algorithms



# Symmetric Key Cryptography

- In symmetric-key cryptography, the same key is used by both parties.
- The sender uses this key and an encryption algorithm to encrypt data;
- The receiver uses the same key and the corresponding decryption algorithm to decrypt the data





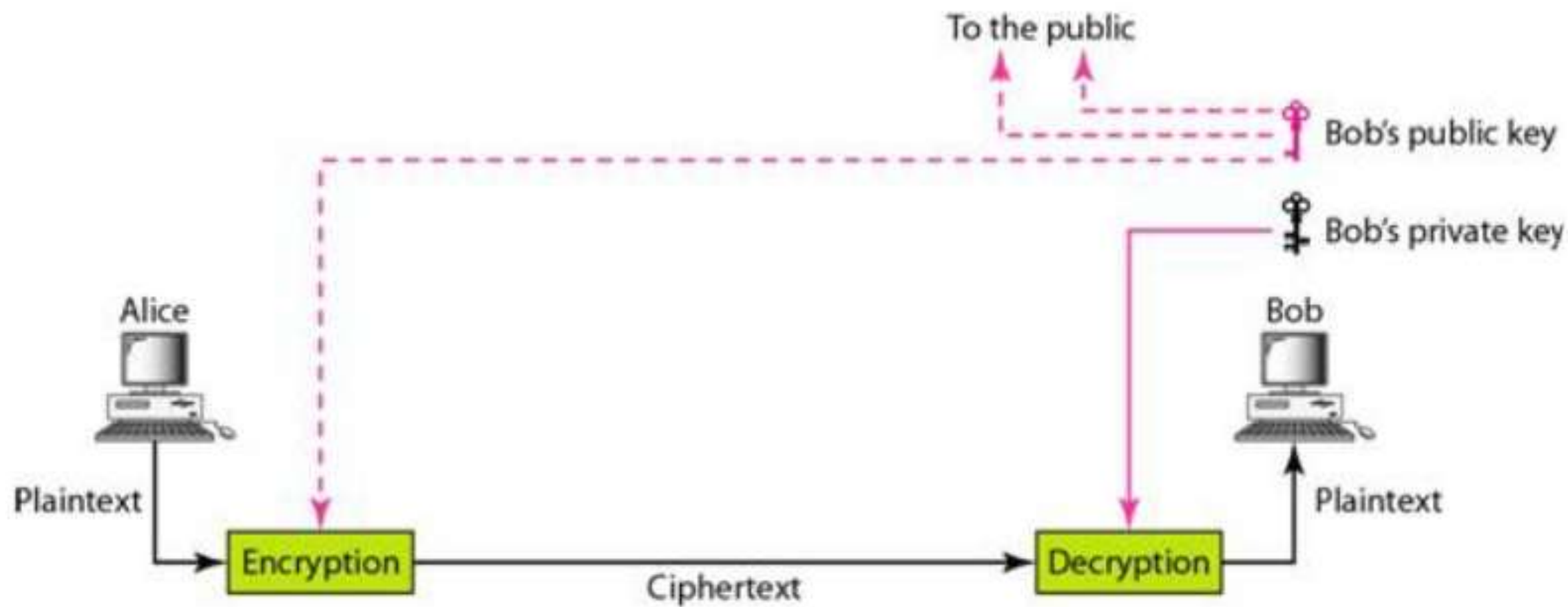
- The key is shared.
- The challenge lies in securely distributing and managing the shared

### Key Characteristics:

- Efficient and faster than asymmetric systems.
- Requires a secure method to exchange keys.
- If the key is compromised, all communications encrypted with that key are vulnerable.

# Asymmetric-Key Cryptography

- In asymmetric or public-key cryptography, there are two keys: a private key and a public key.
- The private key is kept by the receiver.
- The public key is announced to the public.
- In the Figure imagine Alice wants to send a message to Bob. Alice uses the public key to encrypt the message. When the message is received by Bob, the private key is used to decrypt the message.



# Three Types of Keys



Symmetric-key cryptography



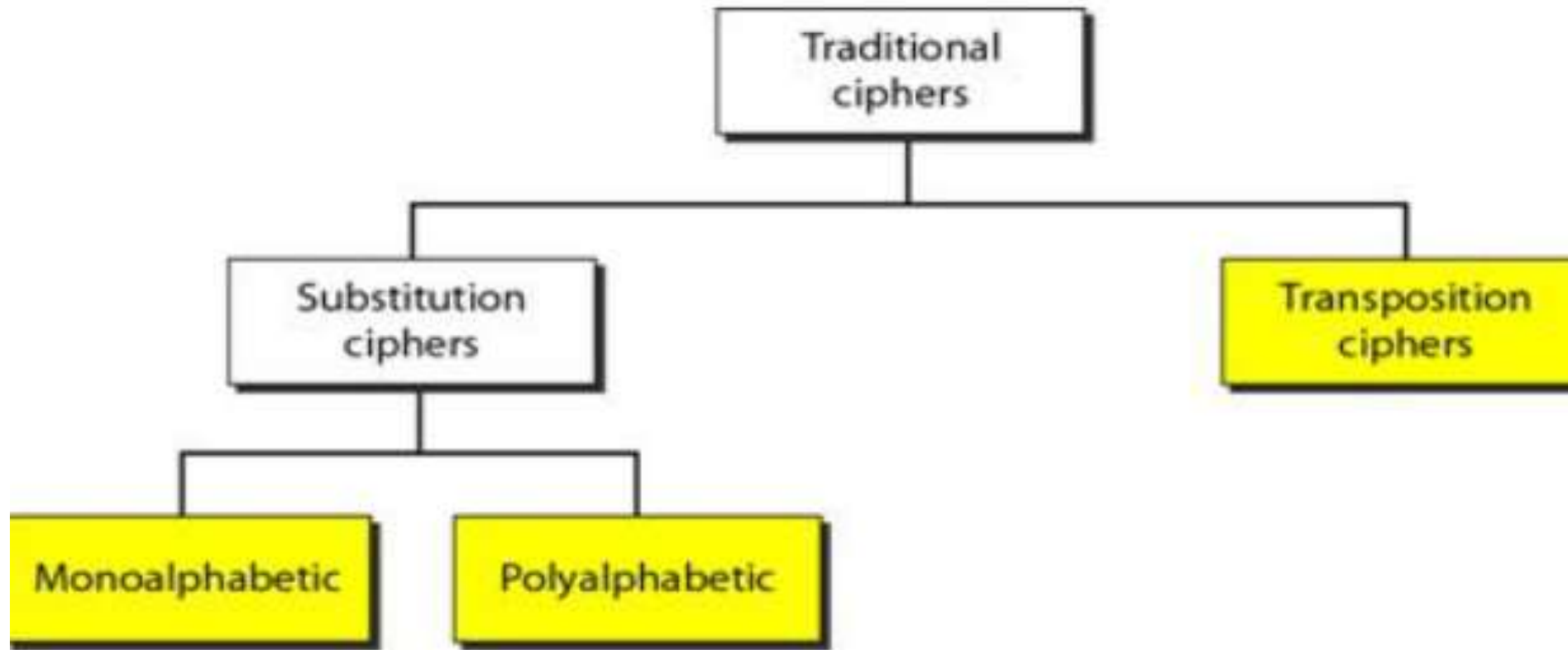
Asymmetric-key cryptography

# Comparison

- Encryption can be thought of as electronic locking; decryption as electronic unlocking.
- The sender puts the message in a box and locks the box by using a key; the receiver unlocks the box with a key and takes out the message.
- The difference lies in the mechanism of the locking and unlocking and the type of keys used.
- In symmetric-key cryptography, the same key locks and unlocks the box. In asymmetric-key cryptography, one key locks the box, but another key is needed to unlock it.

# SYMMETRIC-KEY CRYPTOGRAPHY-

## Traditional Ciphers



*Note*

**A substitution cipher replaces one symbol with another.**