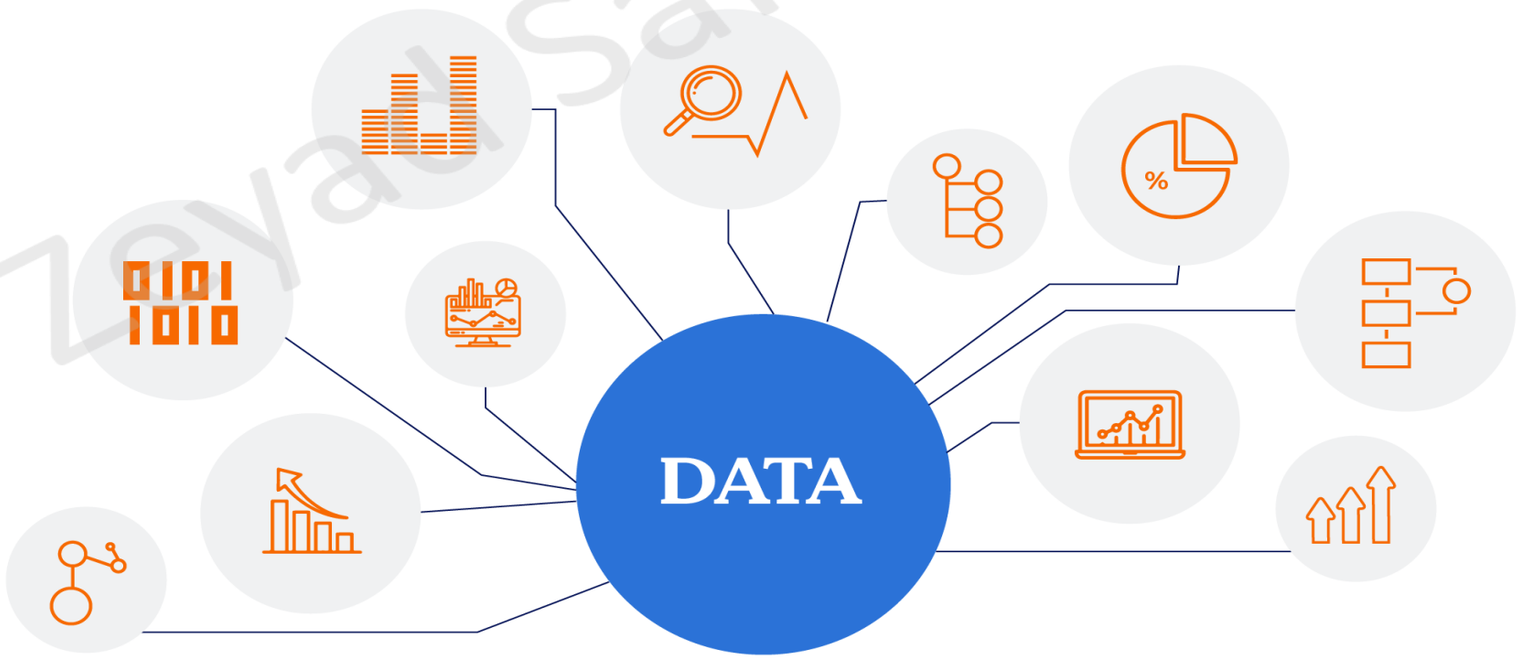
Software Security

Secure Data & Database Management

Dr. Zeyad Safaa Younus Saffawi

What is Data Security?

- **Data security** refers to the **process** of **protecting digital information** where a **company stores** its most **valuable assets** such as (**usernames, passwords, credit card info, financial records**, etc.) from **unauthorized access, corruption, modification, or theft**.
- It involves implementing **technical** and organizational **measures** to **ensure that sensitive data remains protected throughout its lifecycle.**



Why is Data Security Important?

- **Data security** is **important** for **organizations, individuals, and governments** for **several reasons**.
  - **Protects Sensitive Information: Keeps** personal data, financial records, medical information, etc., private to **prevent unauthorized users from accessing this information**.
  - **Preventing Cyber Attacks: Cyber attacks** such as **hacking, malware**, and **ransomware** target **sensitive data** by implementing **strong security controls helps protect systems from these threats**.
  - **Maintains Trust:** Organizations that **protect user data** effectively build **trust with customers and stakeholders**.
  - **Avoiding Financial and Legal Consequences: Data breaches** may lead to **financial losses, legal penalties, and damage** to an organization's **reputation**.

Common Data Security Threats

- **Unauthorized Access:** When an **attacker gains access** to a system **without permission**, for example by guessing passwords or exploiting a hidden backdoor.
- **SQL Injection:** A common attack **where attackers insert malicious SQL commands** into **input fields to access, modify, or steal data** from a database.
- **Data Corruption:** When **stored data becomes damaged or altered due to system failures**, **software errors**, or **malware**.
- **Insider Threats:** Security risks **caused** by **people inside the organization**, such as **employees who misuse their access to steal or leak sensitive data**.

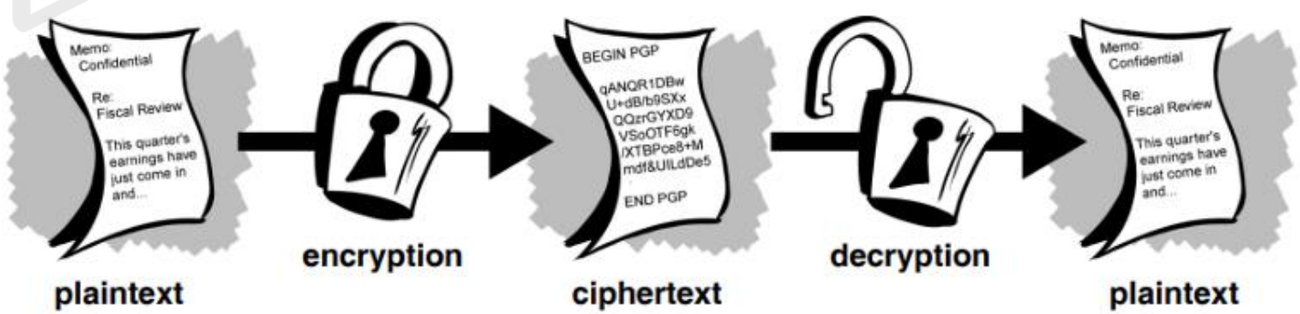
Data Security Measures

Common Data Security Measures

- Organizations use **multiple techniques** to **protect data** from security threats.
  - **Encryption: converts readable data** into an **unreadable format** so that **unauthorized users cannot understand it**.
  - **Access Control:** ensures that only **authorized users** can **access certain data** using **authentication mechanisms** such as **usernames, passwords, or multi-factor authentication**.
  - **Firewalls : monitor network traffic and block unauthorized access**,
  - **Antivirus and Anti-spyware Software: antivirus** software **detects and removes malicious programs**. while **Anti-spyware** software used to **detect, block, and remove malicious programs** that **secretly monitor user activity, steal data**, etc.
  - **Data Backup:** Backup systems create **copies of important data** so it can be **restored in case of data loss, corruption, or cyber attacks**.
  - **Security Policies and Training:** Organizations must **implement security policies** and **train employees on proper data protection practices**.

Protecting Data via Encryption

- **Encryption** is one of the most **important techniques** used to **secure data**.
- **The Basics components of Encryption**
  - **Encryption** is the process of **transforming readable data** (**plaintext**) into an **unreadable form** called **ciphertext**. Only **authorized users with the correct key** can **convert the encrypted data back into its original form**.
  - The process of **converting plaintext to ciphertext** is called **encryption**, while **converting ciphertext back to readable data** is called **decryption**.
  - Encryption relies on a **cryptographic key**, which is typically a **string of numbers or characters used by an encryption algorithm**.



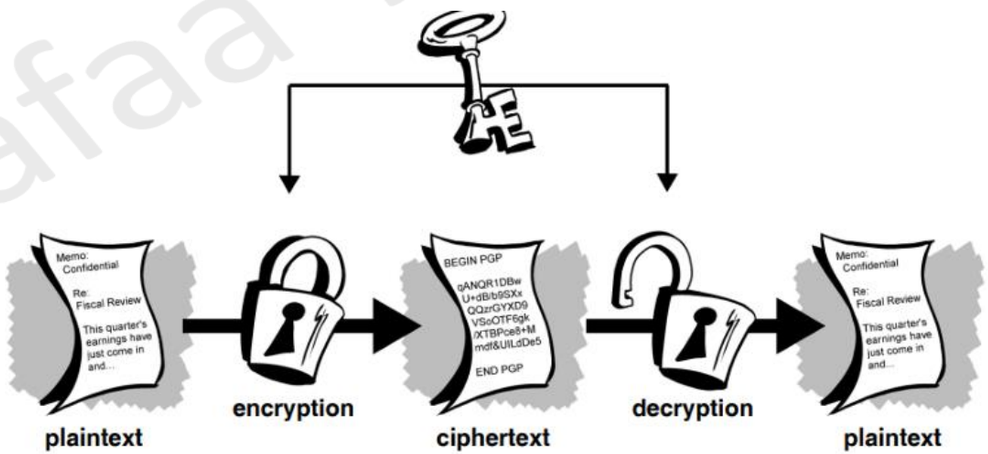
Types of Encryption

There are two main types of encryption used in modern systems:

- **Symmetric Encryption**
- **Asymmetric Encryption**

Symmetric Encryption

- **Symmetric encryption** uses the **same key** for **both encryption and decryption**.
- This means the **sender and receiver must both have access to the same secret key**.
- **Characteristics**
  - Fast and efficient
  - Suitable for encrypting large amounts of data
- **Examples:**
  - Caesar Cipher
  - DES (Data Encryption Standard)
- **Types of Symmetric Encryption**
  - **Block Ciphers**
    - Encrypt data in fixed-size blocks
    - Example: 64-bit or 128-bit blocks
  - **Stream Ciphers**
    - Encrypt data one bit or one byte at a time



- **Limitations of Symmetric Encryption**
  - If the **secret key is exposed**, the **encrypted data can be compromised**.
  - Secure key distribution between sender and receiver can be difficult.
  - The key may be **intercepted during transmission**.

Asymmetric Encryption

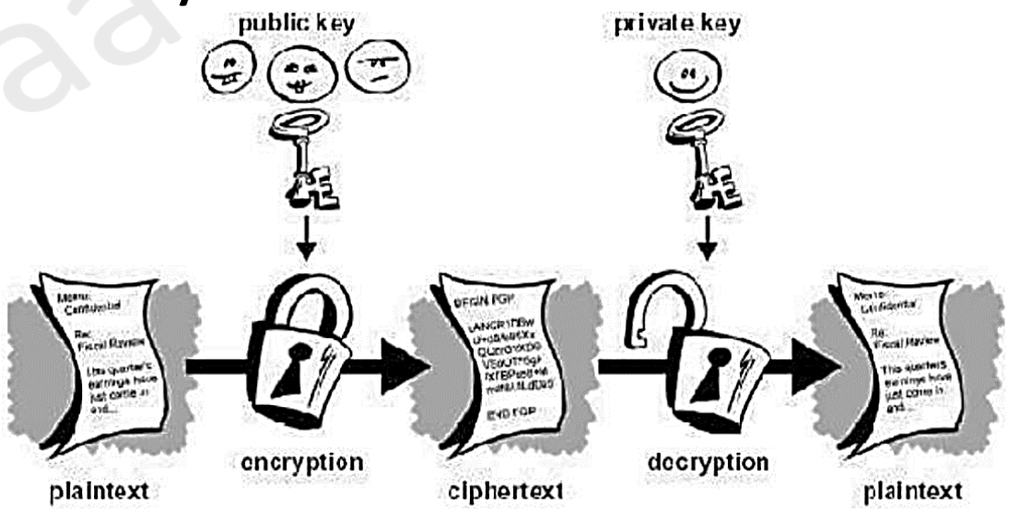
Asymmetric encryption, also known as **public-key encryption**, uses **two different keys**:

- **Public Key** – used for encryption
- **Private Key** – used for decryption

The **public key can be shared publicly**, while the **private key must remain secret**.

This **approach eliminates** the need to securely **send the secret key** between users.

- **Examples**
  - RSA
  - El Gamal



- **Limitation of Asymmetric Encryption**
  - It is **slower than symmetric encryption**
  - It may be **vulnerable** to **Man-in-the-Middle attacks**
  - **Generating and managing key pairs** can be **complex**
- For this reason, many real-world systems **combine** both **symmetric and asymmetric encryption** to achieve better security and performance.

Secure Database Management

- **Databases store large amounts of sensitive information, making them attractive targets for attackers**.
- Therefore, **securing databases** is a critical part of software security.
- **Secure Database Management** is the **set of rules and** practices **includes:**
  - Implementing **strong authentication mechanisms**
  - Using **role-based access control**
  - **Encrypting** sensitive data stored in databases
  - Regularly **updating and patching** database systems
  - **Monitoring database** activity to detect suspicious behavior
- These practices help reduce the **risk of data breaches and unauthorized data access**.

References

- Kohnfelder, L. (2021). *Designing secure software: A guide for developers*. O'Reilly Media.

- McGraw, G. (2006). *Software security: Building security in*. Addison-Wesley Professional.