Encryption

Encryption in Databases refers to the process of converting data into a secure format that can only be read or accessed by someone with the correct decryption key. This helps protect sensitive information from unauthorized access

For example, an MNC might use encryption to protect records in their database. Even if a hacker gains access to the database, they wouldn't be able to read the data without the decryption key.

The encrypted data will be visible as a long string of random characters making it unable to read, if read without the decryption key.

Types of Encryption
1. At-Rest Encryption
2. In-Transit Encryption
3. Column-Level Encryption
4. Full Disk Encryption:

- **At-Rest Encryption:** At-rest encryption protects data stored in the database by encrypting it when it is saved to disk. This means that if someone gains access to the physical storage or the database files, they won't be able to read the data without the encryption key.

- Example: A company encrypts employee salary records stored in their database. If someone accesses the database files directly, they see encrypted data like xYz1234!@# instead of the actual salary figures.

- **In-Transit Encryption:** In-transit encryption secures data as it moves between the database and applications or users. This is typically done using secure communication protocols like SSL/TLS to prevent eavesdropping or tampering during data transmission.

- Example: When you send an email, in-transit encryption ensures that the email content is scrambled while traveling to the recipient's server, so even if intercepted, it appears as random characters like qW9rTy!@ and is unreadable.

- **Column-Level Encryption:** Column-level encryption encrypts specific columns within a database. This is used when only certain pieces of data, like credit card numbers or social security numbers, need to be protected, leaving the rest of the data unencrypted for easier access.

- Example: A database holds user information with a column for Social Security Numbers (SSNs). This column is encrypted, so an SSN like 123-45-6789 might be stored as aBcXyZ!@3 in the database.

- **Full Disk Encryption**: Full disk encryption encrypts the entire disk where the database is stored. This ensures that all data on the disk is protected, not just the database but also any files, logs, or backups stored on that disk.

- Example: A laptop used by an employee is encrypted. If the laptop is stolen, the entire disk is protected, and the thief would see only encrypted data like r9T!@2d3Xy instead of readable files.

Advantages

- It ensures that sensitive information remains secure, even if the database is breached.

- Helps organizations meet regulatory requirements for data security, such as GDPR, HIPAA

- Limits who can view or modify encrypted data to those with the appropriate decryption keys.

Challenges

- Encryption can slow down database operations because of the additional processing required.

- If decryption keys are lost, the data may become permanently inaccessible.