Identity management

A way stop unauthorized access to the database is to require users to login with a username and password. If the system recognizes the username and the password supplied by the user, then the user is granted access to connect to the system. If the credentials entered are not recognized, then they cannot connect to the system. This ensures that only the users that you have authorized can access the system as they are the only ones that will have an account.

To stop users of the database from doing things that they are not supposed to be doing either intentionally or accidentally, privileges can be set. Privileges decide what a user can and cannot do such as only allowing a user to query a table or allowing a u different user to also insert data into the table. These privileges are set by an administrator and can be granted or revoked when needed.

Roles can be created so that users can be grouped together, and their privileges more easily managed. Distinct roles can be granted access to certain parts of the database, for instance basic users may only have access to certain tables while an admin has access to the whole system.

Password handling

Obscuring passwords from any type of user by using a method of encryption means that if there is a data leak, users are not compromised. Using a hashing algorithm before the password is stored in a table encrypts the password but this can be brute forced and the algorithm can be worked out.

To combat this a salt can be added to the hash to further encrypt it. A salt is pseudo-random data added during the hashing process to obscure the password.

Protection against SQL injections

An SQL injection is a method of attack that tries to pass an SQL statement as a string in the hopes of gaining unauthorized access to the system.

One way to try and limit stop is to use prepared statements to try and limit what the users can do. Instead of the users writing the whole statement, they could choose variables and have the system construct a statement.

Another way is to use the DBMS to validate input values and to filter input strings to try and stop users performing malicious acts using SQL injections.

Create backups

Creating backups allows the system to be recovered if anything goes wrong, such as hardware failure or natural disaster or sabotage.

A full backup can be created where all user information and any transaction logs up until that point is recorded in an offline file and placed into secure storage. This can take a long time as there is often a large amount of data requiring a large file to store it, which can be time consuming to transfer from one location to another.

An incremental backup only saves the changes that have been applied to the database since the last backup. These are significantly smaller and easier to manage.

These 2 types of backups can be used together depending on the requirements of the business. Full backups can be taken periodically such as once a week and incremental backups can be made at intervals such as every 24hrs between each full backup. These files can be deleted once a new full backup is taken to save storage space.