RDBMS Security

Database security is essential for a database to protect the data stored in it, to preserve the integrity or confidentiality and to maintain the user privileges across the database. Thus, the database security measures must include protecting the underlying infrastructure of the database system, enforce backup strategies and access to the data itself [15]. Some of the key considerations for addressing these potential security issues and to keep the any database secure are

PostgreSQL

PostgreSQL comes in with some good built-in security features and is addressed in various levels of the database management software [17]. Some of the important security features provided by Postgres are

* Database File Protection – All the files/data stored in the database are restricted from access, to read and modify, except for by the Postgres superuser.
* User and Group Privileges – The superuser has the options to issue database or file access to other users directly or by creating groups and assigning the users to the groups. These groups can have limited or definite permissions to access a table, database, or file and can also define type of access to the respective.
* Control over Remote connections – By default, Postgres only allows local connections over Local Unix Socket [17]. The backend server must be started manually to allow non-local connections. The superuser can also configure/restrict these non-local clients to a particular IP address or to a set of IP domains.

PostgreSQL has a flexible yet powerful built-in security methods to protect the database from exploits. But these methods must be properly implemented and configured based on the application it is used for. No configuration is ideal to every application and thus is not secure from every vulnerability [18].

[15]. <https://www.imperva.com/learn/data-security/database-security/#:~:text=The%20most%20common%20ways%20that,for%20compromises%20are%20as%20follows>.

[16] <https://looker.com/definitions/database-security>

[17] <https://www.postgresql.org/docs/7.0/security.htm>

[18] <https://www.upguard.com/blog/10-ways-to-bolster-postgresql-security>