Security Document

StagBar: Bar Inventory system

Table of Contents:

1.0 Handling of User Information

1.1 User Information on system end

No user information will be stored on the system side. This includes usernames and passwords. Before sending any user names or passwords to the database the information will be hashed. This will avoid sending user information in plain text over the Internet. This will also avoid saving plain text credentials in the database. When retrieving from the database the data will be un-hashed and used only to verify the users credentials, and once again will not store the information any where on the users local system.

1.2 User Information on database end

Unfortunately, we do not have a lot of control over the database server. The server is leased through Amazon and therefore we cannot modify it. Therefore the overhead for the security must be on the system end. As mentioned above no user information will be stored in the database in a readable form, it will all be hashed. It will be up to the system to then un-hash the data to verify the user. Also users will not actually be given any permissions for the database. They will not be a user at the database level.

2.0 Database Security

2.1 Handling of database credentials

As of now database credentials are stored within the source code. This is less then ideal, but upon research by various team members we have yet to come up with a better solution. Outside consulting by some one more familiar with cloud-based databases is needed to overcome this issue.

2.2 Avoiding Improper database manipulation

All data taken from the user will have to first go through the system. This should provide an intermediate between the user and the system to not allow the user to manipulate the database directly. Also the use of prepared statements will be used on all SQL statements that take information from the user. No concatenation of user provided data would be used. This should help to sanitize the data and avoid SQL injection. Within the server all instances of the software are kept in isolation from each other. This keeps user from accessing tables from an organization outside their own.

3.0 Known Issues

-Database credentials stored in source code.
-Not much control over the server verification of users.