NAME: CINDY OGHENETEGA ETUK

STUDENT ID: W21058580

SECURITY REPORT:

Website Security Measures:

Prepared SQL Queries explains how to correctly prepare SQL queries to stop SQL injection, which in turn stops your database from being exposed.

Validating form data that is provided to the server through GET and POST requests is what is meant by the term "basic validation" (username, password, email, etc.).

Session Administration – Create new sessions and save the data acquired from the database. A session's identifier is stored in the browser and synced with the session's data on the server.

Website Security Measures Implemented:

For this website I used the authenticate.php which allows users to be authenticated, establish a connection to the database, receive database results, verify form input, and start new sessions.

logout.php: will terminate any sessions in which the user was signed in and then take them to the login page.

home.php: is the default homepage shown to visitors that have successfully signed in.

profile.php: is responsible for retrieving the user's account information from our MySQL database and populating it with PHP and HTML.

Form — You'll see that I've used both the action and post properties there. The authentication file will have its action property changed to the appropriate value. When the form is submitted, the information that is included inside the form will be sent to the authentication file to be processed. In addition, the method will be specified as post since doing so will make it possible for us to handle the form data by utilising the POST request method.

Input (text/password) — Was used to give the form fields a name so that the server can identify them. The value of the attribute name may be declared as username, and then used as declaration to obtain the post variable in the authentication file that receive the data. For instance, $_POST['username'] would be an appropriate expression.

Input (submit) — Once the form is submitted, the data entered will be processed and transmitted to the authentication file.

After the conclusion of the initialization of the database; the authentication file, which oversees processing and verifying the form input, will be loaded.

The session is then started as soon as the code is executed since doing so allows us to save customers credentials on the server, which can then be used in the future to remember users who are signed in.

Establishing a connection to the database is necessary. If there’s no DB Connection, we won't be able to retrieve or store any information relating to our users. As a result, we need to make sure that the variables are updated to match the credentials that we use to access the MySQL database.

A straightforward error message from a list of SQL prepared statement will be generated if the user attempts to access the file without first submitting the form data, or perform the necessary criteria for logging in.

The SQL statement that will select the excursionID and password_hash columns from the customers database will be prepared. In addition, it will run the SQL statement, after which it will save the result. Finally, it will tie the username to the statement.

To begin, we must determine whether the query has produced any results. If the username is not present in the database, the search will provide no results.

If the username already exists, the search's results are bound to the $excursionID and $password_hash variables.

After that, the password verify method ensure that the password is correct. Passwords that were generated using the password hash function are the only ones that will be accepted.

The password hashing functions are highly crucial for the security of the website because if your database is compromised in any way, then all the passwords that are saved in the customers table will also be compromised. A feeling of privacy will also be afforded to the user because of the fact that their password will be encrypted.

After the user has successfully authenticated themselves, the session variables will be initialised and maintained until either the user logs out or the session time runs out, at which point they will be deleted. These session variables are saved on the server, and their location on the server is be linked to a session ID that is kept on file in the user's browser. These variables will be used by us to ascertain whether the user is logged in, as well as to correlate the session variables with the MySQL database results that have been fetched.

Keep in mind that the passwords are encrypted, which means that we will not be able to see the decrypted password until you first establish a new session variable and save the password in the authenticate.php file.

The only thing that needs to be done to complete the logout script is to terminate all of the sessions that were specified in the authenticate file.

After the sessions have been initialised and destroyed, the user will be sent to the login page. We utilise sessions to identify whether the user is logged in; hence, deleting them will result in the user not being logged in.

Important Steps to Take in Order to Strengthen the Security for future Website:

- **Make use of a secure server by implementing HTTPS and an SSL Certificate**:

  An encrypted URL is essential for the safety of your website. Your site must use HTTPS, not HTTP, if your users volunteer to provide you personally identifiable information.

- **web application firewall (WAF):** It is placed in between the web server and your network connection. To ensure site’s safety, it must analyse all incoming and outgoing information. Most WAFs available today are hosted in the cloud and need no special configuration to begin protecting your network. As an intermediary between the outside world and your data in the cloud, the service prevents all efforts at hacking. Furthermore, it blocks harmful bots and spammers from accessing the system.

Reference:

Drew Hendricks, Article on “10 Essential Steps To Improve Your Website Security” https://www.computer.org/publications/tech-news/trends/10-essential-steps-to-improve-your-website-security Accessed 03/01/2023.

https://www.lucidadvertising.com/blog/security-measures-protect-website-different-threats/ Accessed 03/01/2023.