Erik Marrero

Final Project Report

5/2/2022

Video recording #12

Video recording #11 no video

Project: Security Database to store files at different security levels. As well as set users security credentials in order to secure file access.

Basically what I have implemented is a file that has entries of user names, hashed passwords with salts and authentication level such as UNCLASSIFIED, CLASSIFIED,SECRET, and TOPSECRET with a couple of compartment A,B and ab which allows for both compartments. When the user tries to login implement, I do authentication using a salt and the password that was input and the original salt for the user then I compare the 2 hashed passwords to see if they are the same.  After the user logs in the user’s privileges corresponds with their authentication level, which is their security clearance, compartment, and model type. When the user logs in they chose which Biba or BLP model type they wish to implement. The program Also allows for file creation and user creation along with reading and writing to files depending on the users privileges.

I feel the design I chose for the security database was well suited for the application functionality because it displays the integrity and confidentiality of the BIBA and BLP model. It Also displays this using multiple security levels and compartments. The project allows readers to read and write based off of the model they chose. So if the person wants to make sure they don’t write information to a lower security clearance they can set the program to blp star and it will guarantee that they don’t write some classified information at a lower security level therefor compromising the information. Or if someone didn’t want a user to read or write on different levels they could set the program to biba/blp strong star guaranteeing that the person will only read and write to its own security levels.

Class Concepts:

Biba Demonstrated: This is used to maintain the **Integrity** of Security.

* SIMPLE INTEGRITY RULE: NO READ DOWN
* STAR INTEGRITY RULE: NO WRITE-UP
* STRONG STAR INTEGRITY RULE: NO READ WRITE UP DOWN

BLP Demonstration : This is used to maintain the **Integrity** of Security.

* SIMPLE CONFIDENTIALITY RULE: NO READ-UP
* STAR CONFIDENTIALITY RULE: NO WRITE-DOWN
* STRONG STAR CONFIDENTIALITY RULE: NO READ WRITE UP DOWN

How does ethics apply to my project:   
Ethics apply to my project because the entire project is about keeping integrity and confidentiality. As well as I tried Salting and hashing the user’s password that way if someone was to try and read the user information they cant access the users password. The user’s password is never stored anywhere in plaintext. So even if someone tries to access a user’s login password it is secure. The only thing that is stored is the random salt along with a hash. And this was so I could keep the user’s information secure from anyone trying to take their password. I tried to keep ethics in mind when creating my project and how ethically I would want to secure a user’s password. Along with the BIBA and BLP model Which ethically try and keep a user from accessing information out of there security clearance and compartment type to keep the files integrity and confidentiality at the most importance of the project.