Development of an intelligent security control system

Cameron J. Perrie
Mentored by Brian Filizzi

Introduction

Widespread adoption of biometrics presents the opportunity for biometrically related technology to integrate with daily life. The purpose of this project was to develop software that operates as a security control system that could be applied to biometric sensors. It is already common to use biometrics, such as fingerprints, to unlock devices, because of the speed and ease which biometrics provide (Zimmerman, 2002). The software designed will be the bridge between biometric technology and human life by providing an interactive interface between the user and their technology. The control system was put into development to provide users with information regarding the status of their belongings and keep track of who has access to them. The software would allow for the users to be notified whenever their locks are accessed and, if the user is registered, by whom. This software could be used to provide updates to users via email or text message, allowing people to feel secure about their belongings and not have to worry about the safety of their property.

Materials and Method

The first step to develop this piece of software was to plan its overall features and functions. Then decisions were made regarding the inputs the software needs to recognize, what it needs to display, and how to manipulate the information the program is given. Completing the software followed the steps displayed in Figure 1. This specific software required an abundance of user information and an extensive graphical user interface (GUI) to work with optimal customizability for the user. To generate the sizable GUI, the coding language of C$^{\#}$ was used within Microsoft Visual Studio®.

Each piece of the software required its own form, and multiple forms require access to information entered by the user and data from the software and sensors. The data is stored in multiple text files. Each form may write to and read from these text files in order to transfer data around the software and keep information saved when the software is closed. Upon launch, it is necessary for the software to detect which files the program has and needs, generating the files it needs in a location the software knows how to reach. The first launch sequence of the software needs to be different than every other time it launches in order to

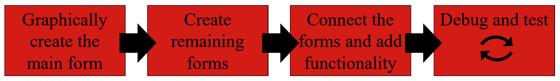

Graphically create the main form
Create remaining forms
Connect the forms and add functionality
Debug and test

Figure 1 (above): The main form is the most important as it requires the user to have access to all other forms and display the most important information. Once the main form was completed, the other forms were made to fulfill the initially laid out requirements. With each necessary component completed, the functionality was added and the GUI became usable. Finally, the cycle of debugging and testing took place.

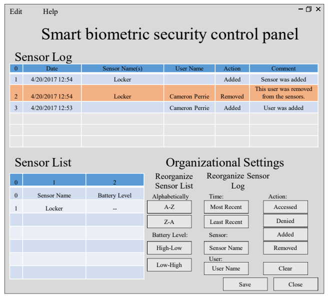
Materials and Method (cont.)

generate the secure files for storing data. The means of reading and writing required extensive functions to maintain data security while formatting and displaying information. Important information is hashed using the Secure Hash Algorithm 1 (SHA-1), which prevents an attacker from being able to access the software's sensors even if they look through the source files and find the stored password. The employment of the SHA-1 was done prior to the collision attack by Google® against the algorithm.

Results

The finished capabilities of the software include the notification function, user addition and removal, a secure login, a logging system, model sensor addition and removal organizational features, and a help file. Figure 2 shows the main form, the hub interface. All the other forms are accessed through this form with the dropdown "Edit" button in the top left corner.

The program sufficiently handles user input and displays appropriate information, regarding users and sensors added to the software. The notification system operates for several actions, represented in Figure 3. Notifications operate through e-mail and text messaging. The software accomplished is a solid foundation for the future fulfillment of the project.



Smart biometric security control panel

Sensor Log

0	Date	Sensor Name(s)	User Name	Action	Comment
1	4/20/2017 12:54	Locker		Added	Sensor was added
2	4/20/2017 12:54	Locker	Cameron Perrie	Removed	This user was removed from the sensors.
3	4/20/2017 12:53		Cameron Perrie	Added	User was added

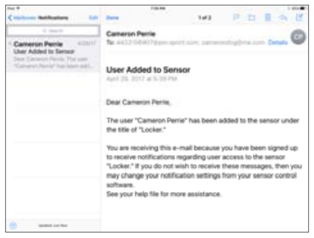
Sensor List

0	1	2
0	Sensor Name	Battery Level
1	Locker	--

Organizational Settings

Figure 2 (left): A mock up of the main form of the software that displays the log of information, a list of all sensors, and several organizational settings for user preference. This form grants access to the remainder of the software, except for an initial login screen, through the "Edit" and "Help" buttons in the top left corner. The "Help" button opens a help file for the software.

Figure 3 (right): An example message from the software sent to an iPad™. The message states that a user was just added to a specific sensor, a sensor that the recipient of the e-mail is set to be notified of changes regarding user access to the sensor. The same message was also sent via text message, as seen in the "to" section, but only the email is displayed.



Cameron Perrie 4/20/17
User Added to Sensor
Dear Cameron Perrie, The user "Cameron Perrie" has been add...

Cameron Perrie
To: [illegible]

User Added to Sensor
April 20, 2017 at 5:39 PM

Dear Cameron Perrie,

The user "Cameron Perrie" has been added to the sensor under the title of "Locker."

You are receiving this e-mail because you have been signed up to receive notifications regarding user access to the sensor "Locker." If you do not wish to receive these messages, then you may change your notification settings from your sensor control software.
See your help file for more assistance.

Conclusion

This software was successful in providing an application that logs and notifies users of changes made within the software and provides a foundation for future applications. There is educational value in the software, including learning about encryption, information storage, and basic electronic communications. One advantage of this software is that it is easy to customize by a novice C$^{\#}$ user.

The application's customizability allows it be worked on with the implementation of sensors and be expanded for the tracking of other security based sensors. It was intended that sensors would be linked to the software initially, but the time required to make the GUI was underestimated. It was decided that the software should be completed first to reduce error when sensors are implemented, however the completion of the software did not permit enough time for sensor addition. The time after software completion was used to develop an in-depth help file that was nearly completed, and will need to be updated as future improvements are made to the software.

There are a variety of potential future improvements to the software, such as the integration of sensors and the development of an online database for storing user accounts, which would allow for improved security by supporting better user identification. Currently the software works by a computer to computer basis, meaning that each account will only work from the computer it was initially made on. By having an online database a user would be able to log their information from multiple sources. Additionally, a verification function for e-mails and phone numbers would allow the prevention of users from spamming unsuspecting people with unwanted notifications. The verification would also guarantee users that they entered in the right e-mail or number.

Another point to be addressed is multiplatform expansion. The current software does not perform on UNIX® systems and the extension of the project to a cellular-phone application would improve mobility, security, and efficiency.

References

Zimmerman, M. (2002). Biometrics and user identification. *SANS institute*. Retrieved from https://www.sans.org/reading-room/whitepapers/authentication/biometrics-user-authentication-122

Acknowledgements

I would like to thank my faculty advisor Nico Werps and my mentor Brian Filizzi for supporting this project and for answering any questions I had.