SYSC4907 PROJECT:
SENSOR-BASED ACCESS CONTROL SYSTEM

By
Craig Shorrocks, Jessica Morris, Richard Perryman
March 2017

A Fourth Year Project Report
submitted to the Dept. of Systems & Computer Engineering
in partial fulfillment of the requirements
for the degree of
Bachelors of Engineering

Abstract

This report tells you all you need to know about something.

Acknowledgements

I would like to thank my supervisor, anyone who paid me money, gave me equipment, etc.

Contents

List of Figures

List of Tables

List of Abbreviations

APDU	Application Protocol Data Unit
API	Application Programming Interface
NFC	Near Field Communication
HMAC	Keyed-hash message authentication code

Chapter 1

Introduction

Give an introduction to your project. This might include:

- Motivation for your project
- Problem you are trying to solve
- Scope of your project
- Organization of your report

You should tune this appropriately for what best suits your project.

Chapter 2

The Engineering Project

2.1 Health and Safety

Using the Health and Safety Guide posted on the course webpage, students will use this section to explain how they addressed the issues of safety and health in the system that they built for their project.

2.2 Engineering Professionalism

Using their course experience of ECOR 4995 Professional Practice, students should demonstrate how their professional responsibilities were met by the goals of their project and/or during the performance of their project.

2.3 Project Management

One of the goals of the engineering project is real experience in working on a long-term team project. Students should explain what project management techniques or processes were used to coordinate, manage and perform their project.

2.4 Individual Contributions

This section should carefully itemize the individual contributions of each team member. Project contributions should identify which components of work were done by each individual. Report contributions should list the author of each major section of this report.

2.4.1 Project Contributions

Give the individual contributions of the each team member towards the project.

2.4.2 Report Contributions

Give the individual contributions of the each team member towards writing the final report.

Chapter 3

Technical Background

3.1 NFC

3.2 Cloud Computing

3.2.1 AWS

3.3 Security

3.4 Singe-board computers

3.4.1 Raspberry pi

3.4.2 Arduino

Chapter 4

Business Use Cases

4.1 Online Order Secure Pickup

4.2 Central Mail Package Pickup

4.3 Long Term Storage

4.4 Service Provider

Chapter 5

Problem Analysis and System Design

5.1 Overall System Analysis

5.2 NFC

Determining how NFC communication should take place required analysis of two hardware systems: NFC card readers, as well as mobile devices. The most desirable protocol would be able to handle the widest variety of available hardwares for the two devices. The performance considerations between modes was fairly minimal, so preference was placed on the portability of the solution.

5.2.1 Card Readers

Since NFC cards are primarily designed for NFC communications, there were few restrictions that stemmed from potential choices in card reader. Since NFC communications are specified by the ISO, most cards support enough protocols that any decision on our part would be very likely to be supported by any card that would be desirable for any other reason.

5.2.2 Mobile Devices

The two most popular operating systems for mobile devices are iOS and Android []. Since iOS devices have NFC disabled for everything except Apple Pay[], the only option that remained was Android. Apple devices would represent a large part of the potential market, so alternatives to NFC would have to be considered.

Among Android devices, there exist devices which have hardware support for NFC communications, and those which rely on host-based card emulation. Devices with hardware support have a component called a Secure Element which performs all of the communication with the external NFC terminal. Later, applications can query this element to determine the status of the transaction, as well as other data. Devices which use host-based card emulation use a software implementation of secure elements. Since host-based card emulation is done through software, it will run on all Android devices running version 4.4 or greater[], which represents over 99% of all devices currently in use.

Android offers an API called Beam which is the only way Anndroid devices can use NFC in active mode []. Beam, however, does not support sending more than one message between devices. Since the information we are sending can be fairly large in the interest of security, this was not feasible given the restrictions of the NFC protocols we used. Further, active communications are easier to eavesdrop on, as discussed in the background section. We decided that these costs outweighed the simplicity of the Beam API, so passive communications were chosen for the implementation.

5.2.3 Protocol

Since our NFC communications may require more data than can be fit within an Application Protocol Data Units (APDUs), we required a protocol which would handle segmenting and recombining the message. APDUs are defined in ISO 7816-4 [] and are the units used by ISO 14443-4 [], which describes the transmission protocol used by NFC devices. They are restricted to 256 bytes, including headers.

To work around this, the hardware device connected to the shield maintains a buffer. Under ISO 14443-4, messages can be reliably transferred, so managing this buffer is the main consideration of our protocol. The hardware determines the maximum amount of data that can be stored in one APDU, and fills in this value into the length field of the APDU that it sends to the Android device. Then, the Android application responds with the minimum of that much data, and all of the remaining data that it has to send. Once the hardware receives an amonut of data less than the potential maximum, it deactivates the connection. In the event that the data from the Android application fits exactly into the last message that would be sent, the protocol still works, as the application will then respond with zero data bytes.

5.3 Android

5.4 Hardware

5.5 Cloud

5.6 Lock Demonstration

Chapter 6

System Implementation

6.1 NFC

6.2 Android

6.3 Hardware

6.4 Cloud

Chapter 7

Testing and Bug Fixes

Chapter 8

Conclusions

References

[1] T. Me and R. You, "A great result," *Wonderful Journal*, vol. 5, no. 9, pp. 1–11, 1998.

[2] J. Him and K. Her, "An even better result that you won't believe," *Best Journal Ever*, vol. 4, no. 8, pp. 55–66, 2002.

Appendix A

Extra Simulation Results

Appendix B

Review of Linear Algebra