Fingerprint Security and Access Tracking System

Samuel Tebbs

IT Innovation Centre, University of Birmingham

08/09/2017

Contents

Ref: Fingerprint Security and Access Tracking 08/09/17
Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Introduction

With the rise in the integration of smart devices in buildings and the prevalence of Internet of Things appliances in everyday life, biometric authentication is becoming a more widespread and affordable possibility for businesses to use when securing the workplace. This, paired with the fact that there are multitudes of libraries and frameworks available to developers when creating biometric systems, means that the prevalence of such systems is only going to rise throughout the coming years.

Requirements

With the growing trend outlined above in mind, the task was given to develop a fingerprint-based security and access tracking system with the following requirements:

Security

The security aspect of the project must be able to restrict and grant users' access to rooms that are equipped with the biometric system, and must have an easily-modifiable list of users. There must also be a system in place to give and revoke a user's access to particular rooms, which is reflected upon the next time the user tries to access that room.

Performance

The authentication system itself must also be fast enough so that it doesn't cause unnecessary delays to those attempting to access a room, although a reasonable delay is acceptable.

Access tracking

A persistent record of points in time when accessed has been granted to a room must be kept, composing the granted person's ID, the room ID along with the date and time at which access was granted. It must be possible to request a report for a certain room, listing the people that were granted access to the room given a time-window. It must also be possible to request a report for a certain person, listing the rooms to which they were granted access given a time-window.

Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Extensions

The known possible extensions to the project that are based on the security and/or access tracking systems outlined above are:

Work hour tracking

The access tracking system could be extended to record the number of hours worked by an employee in a given day, along with recording the duration and frequency of breaks taken. This could work as a basis for wage calculation and performance analysis, among other uses.

Theft deterrent

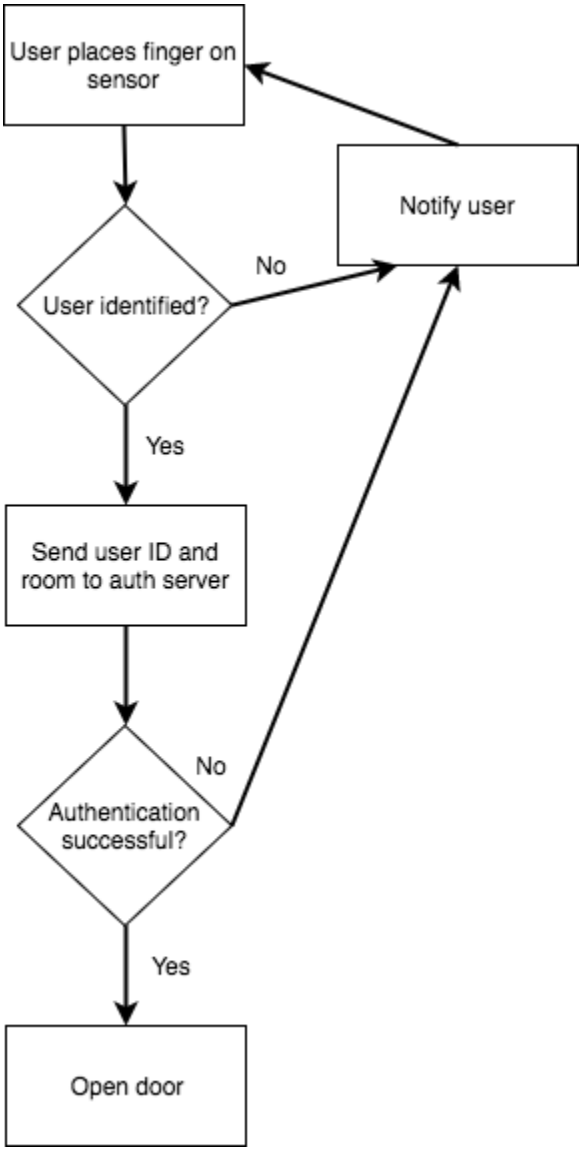
Access to a room is restricted to those with the correct permissions so in the eventuality that an item is stolen from the room, a list of possible suspects could be made with ease using the access report functionality outlined before. A flaw with this could be that someone with access could allow a restricted user in, who will then not appear in the access report, however the system could be paired with facial recognition to make the access list more accurate. In conclusion, the very knowledge that a potential thief's presence in a room is known could deter them from stealing from the room.

Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Design

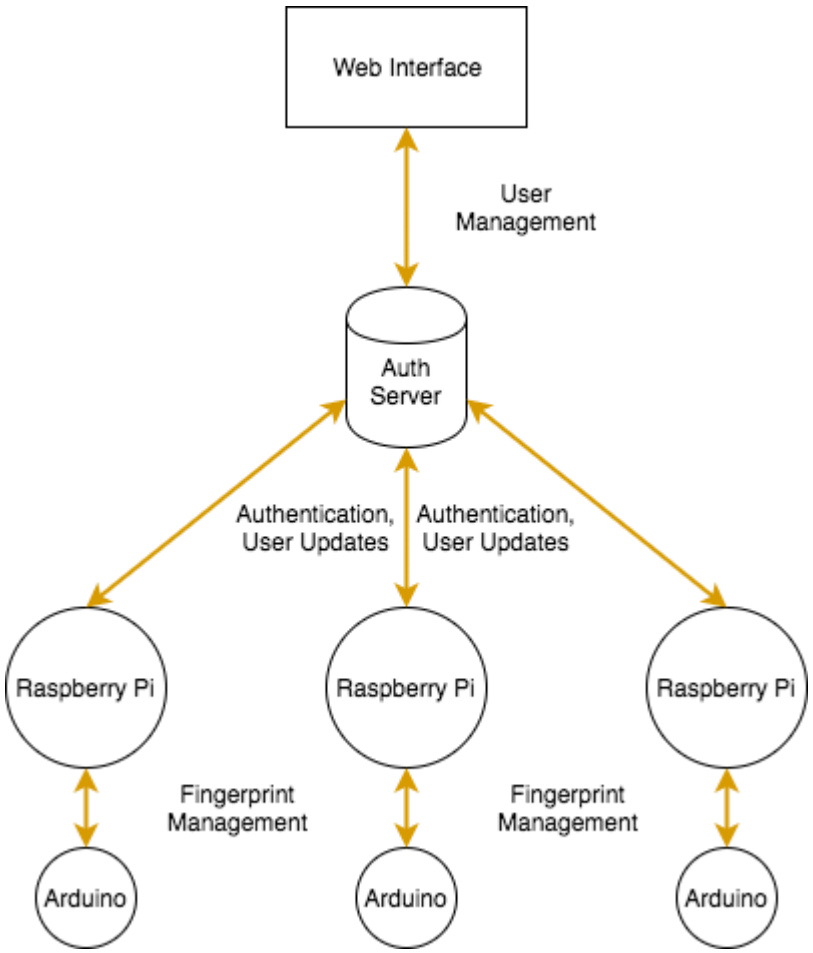
The system is split into two separate components; the authentication server and the fingerprint scanner. The authentication server stores all user data, their permissions, associated data and serves the website that handles user management. The fingerprint scanner consists of a Raspberry Pi board and an Arduino Uno, with the Arduino Uno serving as an interface to the fingerprint sensor and the Raspberry Pi serving as a processor of the information received by the authentication server and the Arduino Uno.

Below is a diagram detailing the process through which a user requests access to a room.



Ref: Fingerprint Security and Access Tracking 08/09/17
Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Below is a diagram outlining the communication structure.



The web interface communicates solely with the authentication server, the Raspberry Pi's communicate with both the authentication server (for authentication and user updates) and the Arduino (for fingerprint detection and management).

Ref: Fingerprint Security and Access Tracking 08/09/17
Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Implementation

Authentication server

The authentication server was implemented as a Java application which functions as both a file server (serving html, css, javascript files etc.) and an API handler. The Raspberry Pi uses this API to decide when to grant/deny access to a user, add new fingerprints and delete old fingerprints.

API Endpoints

Below are the various API endpoints that it provides.

- **Authentication**
  - **auth/check:** Check if a user has access to a given room
  - **auth/add:** Add a fingerprint to the database so it can be propagated to other sensors
- **Reports**
  - **reports/new:** Create a new report for a user/room within a given time frame, and send it to a given email address
- **Users**
  - **users/list:** List all users and the rooms they have access to
  - **users/add:** Add a user with a name and ID
  - **users/delete:** Delete a user with a given ID
  - **users/enroll:** Set a user to be enrolled at a certain room
- **Access**
  - **access/add:** Give a user access to a room
  - **access/delete:** Revoke a user's access to a room
- **Rooms**
  - **rooms/add:** Add a room with an ID and a name
  - **rooms/delete:** Delete a room with a given ID
  - **rooms/list:** List all rooms

Database

The authentication server also has a backing MySQL database, which is designed as follows.

Users

- **id:** primary key, unique, int
- **name**: not null, varchar (30)

Access

- **user_id**: foreign key (Users.id), not null, int
- **room_id**: foreign key (Rooms.id), not null, int

User Aliases

- **user_id**: foreign key (Users.id), not null, int
- **room_id**: foreign key (Rooms.id), not null, int
- **local_id**: not null, int

Rooms

- **id**: primary key, unique, int
- **name**: not null, varchar (30)

Enrollment

- **user_id**: foreign key (Users.id), not null, int
- **room_id**: foreign key (Rooms.id), not null, int

Record

- **user_id**: foreign key (Users.id), not null, int
- **room_id**: foreign key (Rooms.id), not null, int
- **date**: not null, datetime

Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

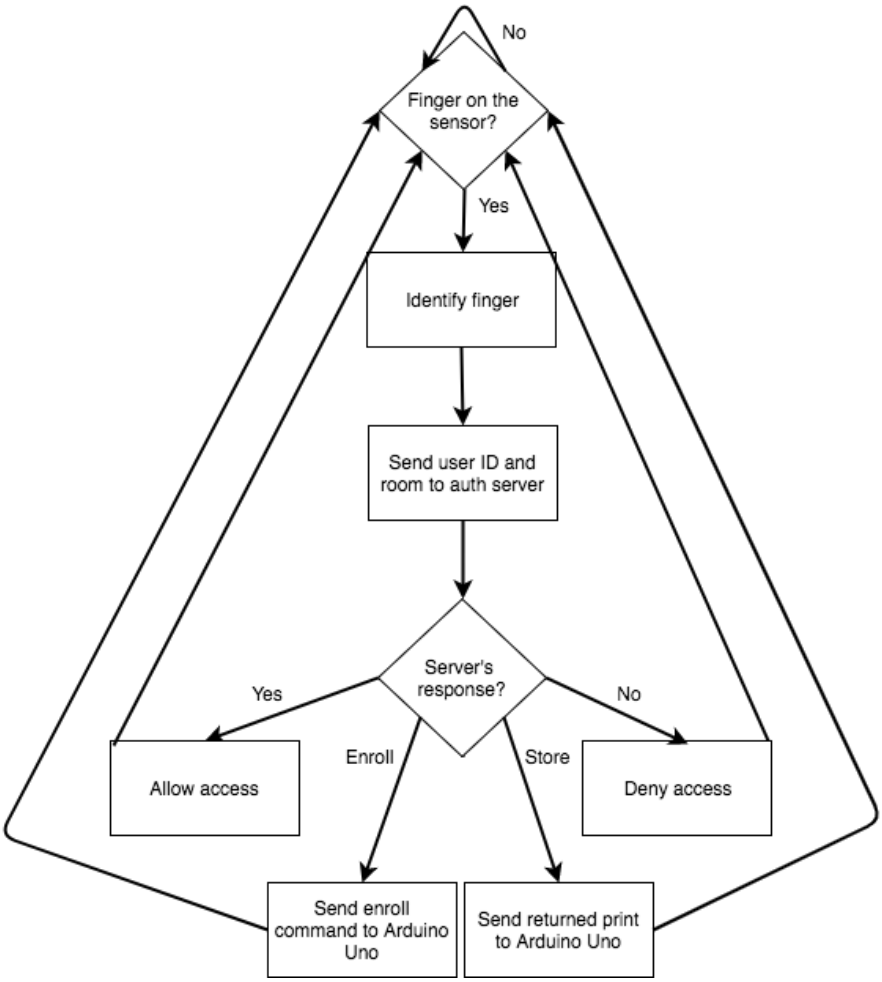
Arduino Uno

The Arduino Uno software is written in Arduino C, and is responsible for managing the fingerprint sensor and handling commands sent to it over serial by the Raspberry Pi. The commands (strings of characters) accepted are as follows:

- **?:** Check if there is a finger on the sensor and respond accordingly
- **E:** Enroll a user's fingerprint via this sensor using a given ID
- **I**: Identify the user whose finger is on the sensor
- **L:** Load the fingerprint associated with a given ID and send it over serial
- **S:** Store a fingerprint sent over serial, using a given ID

Raspberry Pi

The Raspberry Pi software is written in Python and acts as an intermediary between the Arduino Uno and the authentication server, processing the information received from both. Below is a diagram outlining its functionality:



Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Evaluation

Advantages

There are multiple advantages in using a fingerprint security system over a typical keycard or lock and key system. One being that there is no object that can be lost in the way that a key or keycard can, meaning that a user's access to a room is more reliable than it would be with a typical system. In addition, a person's possible lack of dexterity which could pose a problem when attempting to unlock a door with a key isn't an issue when using a fingerprint scanner, due to the less complex actions required. Finally, a lock and key system doesn't provide any way to identify the person unlocking the door, which a fingerprint security system does.

Disadvantages

One disadvantage that a fingerprint system has over a typical lock system, is that of power consumption as locks require none but each device attached to a door will require either a battery or mains electricity connection. In addition, fingerprint sensors are likely to be much more expensive than locks and keys. Finally, a disadvantage is that if the authentication server is taken offline, then all locks stop working, this issue could however be mitigated by introducing redundancy systems.

Existing products

There are multiple products already on the market that offer fingerprint sensors (see appendix), however, they are all more expensive than the project described in this document, and not all offer any security functionality, just access tracking.

Future plans

There are many improvements, extensions and changes that could be made to the project.

One such improvement would be the removal of the Raspberry Pi from the system, as its only role is to serve as an intermediary between the Arduino Uno and the authentication server and adds expense, power consumption and another potential point of failure. This would require improvements to the built-in WiFi capabilities of the Arduino Uno WiFi and an increase to the available program memory.

The sensor used can store **128** fingerprints, and so a possible improvement would be to find or develop a sensor that can store a greater number to allow more users to be added to one sensor.

Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

SWOT Analysis

Strengths	Weaknesses
• Not prone to key/keycard loss • Lack of dexterity isn't as much of a factor as it is in a key-based system • Enrolling a user is quick and easy	• Only 128 users can be stored on a fingerprint sensor • Regular cleaning of the sensors may be necessary to maintain reliability
Opportunities	**Threats**
• Could be integrated with facial recognition to improve access tracking • A company/university's most used rooms could be found using the access tracking functionality	• There are potential issues with data protection due to the storage of fingerprints in the sensors • Two distinct prints could erroneously match if the sensor is dirty or faulty

Ref: Fingerprint Security and Access Tracking 08/09/17
Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Appendix I. Evaluation Matrix Scores

Area	Scoring System	Score	Reason
Maturity	1 = Idea 5 = Mainstream Product	3	The fingerprint locker is in use, however, it is not widely used in university or big organizations.
Technology (Adoption timescales)	1 = > 3 years 5 = < 3 months	4	Adopt the technology is relatively easy and quick, as the commercial product is available.
Business Process (Adoption timescales)	1 = > 3 years 5 = < 3 months	3	To use it widely in the university and further support on it, it will require more resources and some process changes.
Adoption Overview	1 = v long time 5 = very short	3	People's acceptance of the product will affect the overall technology adoption.
Existing Technology (Impact)	1 = v large impact 5 = very little	4	Adopt fingerprint sensor will have some impact in the current technology, such as swipe card. However, it won't be a replacement at this stage.
Resources Required	1 = v large impact 5 = very little	2	Hardware and Software application will need to set up, together with resources to support.
Scope	1=very difficult 5=very easy	3	Fingerprint can be used in security, access tracking, potentially in other areas such as lecture attendance recording etc.
Usability	1=very difficult 5=very easy	3	Enroll the fingerprint is relatively easy, however, two distinct prints could erroneously match if the sensor is dirty or faulty.
Security	1 = very poor 5 = excellent	3	In the prototype, fingerprint are stored in the sensor and has an limitation of 128 fingerprints, which could lead to personal information leakage if the sensor is stolen.
Innovation Value	1 = low innov. 5 = high innov.	3	Biometric authentication is more convenient than using the keys, however, there is limitation as well.
Cost Effectiveness	1=very expensive 5=very cost effective	3	To use fingerprint sensor widely in the university, the cost for hardware and software setup is not cheap.
Adoption Readiness Score	*<20 - not ready* *20-29 - emerging* *30-39 - Adoptable* *>39 Fully Ready*	*27*	*Fingerprint security and access tracking system provides convenience for the user, however, it has some limitations, such as user acceptance, accuracy, and cost etc.*
***Note**: Rows that have no highlight colour indicate the score value is not added to the adoption readiness total. Instead, the overview score for that area is used as part of the total score.*			

Ref: Fingerprint Security and Access Tracking 08/09/17
Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk

Appendix II

Existing products

All prices given are for each sensor unit and include VAT

- Safescan TA-8020 Clocking in System (£458.59): https://www.safescan.com/en-gb/store/clocking-in-machines/safescan-ta-8020-clocking-in-system/vat?gclid=Cj0KCQjwub7NBRDJARIsAP7wlT-_LiKgaLkGNVUBA6t_WcE1zMZoRWX1lRHcNGizJG2D3VOACP_yUaMaAtjtEALw_wcB
- GeoTime 100 Fingerprint 2018 Version (£738): https://www.timesystemsuk.com/products/geotime100.html?gclid=Cj0KCQjwub7NBRDJARIsAP7wlT_ThzKGhJnqNHpzVj87cLpb410G8AsVAvy0IDXnFBz4ivvRbpYNMocaAmxVEALw_wcB
    - Note that this price is for the version with the same fingerprint capacity as the project described in this document (**128**), the cheapest version has a capacity of only 15 fingerprints and costs £258

Samuel Tebbs, IT Innovation Intern
sxt567@student.bham.ac.uk