Computer Science Tripos

Part II Project Proposal Coversheet

Please fill in Part 1 of this form and attach it to the front of your Project Proposal.

Part 1

Name: Sophie Walker

CRSID: smw98

College: St. Catharine's College

Project Checkers:(Initials) AB/SK

Title of Project: Bluetooth Tracking Privacy Bubble

Date of submission:

Will Human Participants be used? No

Project Originator: Dr Ramsey Faragher

Project Supervisor: Dr Markus Kuhn, Dr

Directors of Studies: Dr John Fawcett

Special Resource Sponsor:

Special Resource Sponsor:

Part 2

Project Checkers are to sign and comment in the students comments box on Moodle.

Part 3 For Teaching Admin use only

Date Received:

Admin Signature:

Bluetooth Tracking Privacy Bubble

Sophie Walker

5 October 2023

1 Introduction

Bluetooth Low Energy (BLE) tags such as AirTags or Tile Trackers are useful tools for locating personal items. However, stalkers can maliciously use these to track their victims. My project is to create a prototype of a similar tracking system and investigate the effectiveness of adding an antistalking 'privacy bubble' feature. This involves having the system set tags as inhibitors or trackers. Tracking tags will function as regular tags so their owner can locate them and their items. However, if they are in the range of an inhibitor tag, the owner will not receive the tracking tag's location. A privacy bubble means any tracking tags near in time or distance ranges to an inhibitor will not report their location back to their owner. This has the potential to stop stalking as inhibitors will prevent stalkers locating any planted trackers on their victim.

Additionally, this will give people peace of mind that stalking will be stopped even before other tools can detect it, as this will block any tracking tags nearby. Benign tags will not be permanently co-located with inhibitor tags as they move through the environment and will eventually move away from them. So the blocking should not be disruptive, whilst true stalking is prevented. This can also prevent annoyances from false positive stalker detection that picks up tags owned by friends, family or colleagues, as this mechanism works without requiring notifications.

Currently, there are other stalking prevention features in similar devices, but typically detecting stalkers takes at least 4 hours [1] and can repetitively alert users to false positives if we are near a person with their own tag, for extended periods of time. This method of indiscriminately blocking nearby tags should not pose issues for people finding their own tags but solves the issue other tools have where we cannot distinguish malicious from safe tracking.

2 Description of Project

My project does not aim to create new tags or hardware, instead, I want to prototype the system used to interact with tags, and then I will implement inhibitor tags and modes on the preexisting system. This will provide additional scope to change the tag's modes, find lost inhibiting tags and provide anti-stalking to users who already own these tags as they can modify tag type. The tracking system involves a user-facing app and a server to communicate the tag's location.

So, this project's key items are:

- **The App:** This will be modified from a current open source app that can scan for BLE tags and obtain their Received Signal Strength Indicator (RSSI). I will adapt it to provide data packets of information including RSSI, phone position estimate, timestamp, and any other required information to the server. The app can be further modified in extensions which can improve locating tags and providing security.
- **The Server:** I will build this using Python Flask and MySQL to create a database which stores the tag IDs, which mode they are registered as, their location, and if they are inhibited. It will also calculate the location of tags and which tags are in range of inhibitors to block these. Additionally, it will share the last known location of trackers back to the user.

I can also solve difficulties regarding the imperfect BLE system and investigate if I can create one that can cope with the interference of the BLE signal through people, multipath fading and other spurious results [2]. This will also have to solve if tags are in range of each other, but detected by separate devices, or if a tracking tag is detected before the nearby inhibitor is. These are a few of the issues my system should aim to be robust against.

I can evaluate the system based on the false positive rates, false negative rates, and any other stats of interest for various settings and scenarios. This will start with an empty environment without obstacles at close ranges which will test the basic system in an environment, such as an empty outside area, that should reduce the effect of inaccurate distance calculations and purely test the base system. I could then introduce obstacles and other environments to evaluate the effectiveness of the inhibitor tags in different situations and which situations this could work with the basic location calculations. I could potentially design an experiment based on the usability of this system, and whether introducing these inhibitor tags is an effective way to reduce the risk/impact/ease of stalking.

3 Success Criteria

- I will produce a modified app that will register BLE tags in one of two modes and scan for nearby tags. It also needs to communicate with the server to share at least the ID and RSSI of any detected tags and the phone's current location.
- I will produce a server that will receive reports about tags detected by the app. It will use this to identify the mode and locate all tags. The server will find and inhibit any tracking tags it can colocate within an inhibitor's time and distance bounds. The server will send the owner's app the location of their uninhibited tag.
- The time and distance bounds will be configurable on the server. I will investigate the appropriate ranges with experiments based on the accuracy of these bounds. I will explore the tradeoff between preventing stalking and disruption to other users with these bounds.

4 Possible Extensions

- Explore tag type being detected by phones. This can colocate and block nearby inhibited tags within the app, and not broadcast their location to the server at all. I can investigate if this is a feasible implementation likely to be used.
- Implementing the same security procedures used by Apple to cycle IDs regularly and encrypt data in the server.
- Improving methods for locating tags, this can be based on an epoch of measurements[3] or use one of the smoothing algorithms suggested [4].
- Identifying stalkers even if they have been blocked, so those that have been blocked for a long period can be found by those being stalked and how other stalking detection tools can interact with inhibiting tags.
- Implementing local finding, if the devices are in range so we can allow genuine tags to be found only if their owner is in range. use case For example, so a user does not stop neighbours from finding their items while at home. This can also prevent the malicious use of inhibitor tags that can block an area of tags to prevent owners from finding items.

5 Starting Point

This will be completed in Java/Kotlin (based on the most appropriate app chosen) which I have used in the OOP course/ which I have not used except for a basic tutorial this summer. I will also use Python Flask to build the webserver, which I have used briefly before in the Group Project last year. I have used SQL in the databases course, but not MySQL specifically. I have not previously used tools or libraries related to BLE tracking tags before.

6 Work Plan

Week	Start	Work Package	Deliverable
1-2	16/10/2023	Set up existing scanner app, explore and test unchanged. Set up and learn to use MySQL with Flask. Do some basic tests and evaluation of received tracking tags with a standard app.	Write up of completed experiments and limitations that could affect this project. Write up the understanding of modifications and requirements of the app.
3-4	30/10/2023	Begin Draft of Introduction Chapter of Dissertation. Create a document detailing my implementation plan and specification	Full Specification of the System. Start of Draft Introduction Chapter.
5-6	13/11/2023	Draft Preparation Chapter. Implement basic modifications to the app and set up the web server.	The app will be able to detect tags and share specified data with the server. Start of Draft Preparation Chapter.
7-8	27/11/2023	The Web server should have locating tags implemented and the app needs to be able to receive locations of owned tag(s) whether as basic coordinates or if there is time in a usable map. The UoA deadline is on the 1st. Mainly use the second week for this. End of Michaelmas.	The basic tracking system will be complete, so evidence basic testing that tags can be tracked in an ideal environment.
9-10	11/12/2023	Implement registering inhibitor tags on app. Implement colocating inhibitor tags with tracker tags, and blocking their location.	Completed Inhibitor system. Evidence of basic testing of Inhibitor system in an ideal environment.
11-12	25/12/2023	Part of this package will be spent on the Christmas Holiday. I will start the implementation draft. If core implementation has not been completed, finish here. Otherwise, improve the Introduction and Preparation chapter Drafts.	Any Previous Unfinished Deliverables. Start of Implementation Draft Chapter.
13-14	08/01/2024	Start of Lent term. Plan experiments and evaluation tasks to complete and record. Create a draft Progress Report and Presentation	An evaluation plan describing planned experiments and data to be collected. Draft Progress Report and Presentation.
15-16	22/01/2024	Completed Progress Report and Presentation. Time permitting, implement Security Extension to introduce ID cycling and encryption to the system.	Completed Progress Report and Presentation. Time permitting implemented Security Extension.
17-18	05/02/2024	Complete evaluation experiments. Begin the Evaluation Chapter with recorded results from experiments.	Draft of Evaluation Chapter with recorded results from experiments.
19-20	19/02/2024	Improve implementation and Evaluation Chapters. Time Permitting, finish incomplete extension or implement improved methods of locating tags based on an epoch of measurements.	Improved Draft of Implementation and Evaluation Chapters. Time Permitting, implemented Improved Locating extension.
21-22	04/03/2024	Plan and evaluate any completed extensions. End of Lent Term.	Include completed extensions in the evaluation draft. Complete any unfinished deliverables.
23-24	18/03/2024	Start conclusions chapter draft. Finish any evaluation and improve other chapters. Time Permitting Implement extension for local owners of tags to be able to find them within the inhibitor bubble, and evaluate this.	Draft of Conclusions chapter. Time Permitting Implemented local finding extension.
25-26	01/04/2024	Complete First Draft Dissertation. Complete Conclusions Chapter of Dissertations	First Draft Dissertation
27-28	15/04/2024 -	Improve on the first draft from feedback	Updated Draft Dissertation
29-30	29/04/2024	Final improvements and Submission of Dissertation.	Submit Dissertation

In the timeline, extensions can be switched based on which is most appropriate for the project at that stage, given the completed preparation and core implementation.

7 Resource Declaration

The resources that this project requires are BLE tracking tags to test and create the inhibitor and tracking system and Android phones are required to detect the tags. Both of these will be provided by Dr. Ramsey Faragher. Although these have already been ordered, a final contingency is to simulate BLE beacons on my laptop and use my personal phone. I will need to host the server for a longer time period than I can with my own devices. I will use the Azure for students to host my server and database as this will give me 750hr per month for free. If any costs are accidentally incurred, this will be covered by the $100 free credit for students. If there are issues with this, I will use the Student-Run Computing Facility's free hosting. A final contingency for longer experiments is to locally record the timestamps and data the app should have sent the server. I would later replay them to my personal laptop to test how the system would have responded.

I will use my personal laptop to build this, with another personal laptop as a contingency. I will backup all code to GitHub and use its version control. Any documents created I will back up to GitHub. I am using TeXworks and Obsidian to write these. I accept full responsibility for this machine and I have made contingency plans to protect myself against hardware and/or software failure.

References

[1] K.I. Turk, A. Hutchings, A. Beresford, "Can't Keep Them Away: The failures of anti-stalking protocols in personal item tracking devices" to appear in the proceedings of the Security Protocols Workshop, March 2023.

[2] Faragher, R., Harle, R., "An Analysis of the Accuracy of Bluetooth Low Energy for Indoor Positioning Applications," Proceedings of the 27th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2014), Tampa, Florida, September 2014, pp. 201-210.

[3] R. Faragher and R. Harle, "Location Fingerprinting With Bluetooth Low Energy Beacons," in IEEE Journal on Selected Areas in Communications, vol. 33, no. 11, pp. 2418-2428, Nov. 2015, doi: 10.1109/JSAC.2015.2430281.

[4] Mustafa, Abdul and Sykes, Edward. (2021). A high fidelity indoor navigation system for users in motion using BLE with beacons