# Master Thesis

*Mapping smart environments: analysing and attributing wireless communication*

Nicolai Rønning

Thesis submitted for the degree of
Master in Informatics: Programming and System Architecture

60 credits

Department of Informatics

Faculty of mathematics and natural sciences

UNIVERSITY OF OSLO

Spring 2021

# Master Thesis

*Mapping smart environments: analysing and attributing wireless communication*

Nicolai Rønning

Master Thesis

https://www.mn.uio.no/ifi/?vrtx=person-view&uid=nicolr

Printed: Reprosentralen, University of Oslo

## Abstract

Our environment has become filled with wireless devices communicating with each other constantly. This provides convenient solutions for many, but are people aware of the privacy risks it entails? This thesis explores various ways the internet of things (IoT) can prove problematic for privacy. It does so by proposing a prototype that demonstrates what kind of information someone can obtain simply by observing signals coming from mobile devices. In addition, the prototype is used to conduct experiments that shed light on specific methods for exploiting these privacy risks.

# Contents

**List of Figures** **v**

**List of Tables** **viii**

**Listings** **ix**

**1 Introduction** **1**
- 1.1 Motivation . . . 1
- 1.2 Contribution . . . 2
- 1.3 Research questions . . . 3

**2 Background** **5**
- 2.1 Internet of things . . . 5
- 2.2 MAC address . . . 5
- 2.3 Wi-Fi . . . 6
- 2.4 Bluetooth . . . 7
- 2.5 Wardriving . . . 8
- 2.6 Prototype technologies . . . 8
- 2.7 Privacy . . . 9
- 2.8 Visualization tools . . . 9
- 2.9 Related work . . . 9

**3 Data collection** **11**
- 3.1 Collecting data . . . 11
- 3.2 Data format . . . 11
- 3.3 Data exporting . . . 12

**4 Design** **17**
- 4.1 Purpose . . . 17
- 4.2 Specification . . . 17
  - 4.2.1 Functionality . . . 17
- 4.3 Use-cases . . . 20
  - 4.3.1 Detecting residents in homes . . . 20
  - 4.3.2 Detecting visitors in homes . . . 20
  - 4.3.3 Adding context from location . . . 21
  - 4.3.4 Detecting frequent locations and frequently taken paths . . . 21
- 4.4 Technology . . . 21

4.5 Concepts . . . 22
4.5.1 Trilateration . . . 22
4.6 Architecture . . . 24
4.6.1 Database design . . . 24

**5 Implementation 27**
5.1 Project . . . 27
5.2 Upload . . . 27
5.3 Compare . . . 29
5.4 Map . . . 30
5.5 Datatable . . . 30
5.6 Dashboard . . . 30
5.6.1 Device distribution . . . 30
5.6.2 Device distribution graph . . . 30
5.6.3 Get vendor info . . . 31
5.6.4 TVs scanned . . . 31
5.6.5 Filter selection . . . 31
5.6.6 Find location of device . . . 31
5.6.7 Find devices at address . . . 31
5.6.8 Find time device was spotted . . . 32
5.6.9 Filter distant duplicates . . . 32
5.6.10 Remove duplicates . . . 33
5.6.11 Remove access points . . . 33

**6 Experiments 35**
6.1 Experiment 1 - Detecting residents in homes . . . 35
6.1.1 About . . . 35
6.1.2 Description . . . 35
6.1.3 Method . . . 37
6.2 Experiment 2 - Detecting visitors in homes . . . 38
6.2.1 About . . . 38
6.2.2 Description . . . 38
6.2.3 Method . . . 40
6.3 Experiment 3 - Adding context from location . . . 42
6.3.1 About . . . 42
6.3.2 Description . . . 42
6.3.3 Method . . . 43
6.4 Experiment 4 - Detecting frequent locations and frequently taken paths . . . 46
6.4.1 About . . . 46
6.4.2 Description . . . 46
6.4.3 Method . . . 48

**7 Results 53**
7.1 Experiment 1 - Detecting residents in homes . . . 53
7.1.1 Data . . . 53
7.1.2 Discussion . . . 54
7.2 Experiment 2 - Detecting visitors in homes . . . 55
7.2.1 Data . . . 55

7.2.2 Discussion . . . 56
7.3 Experiment 3 - Adding context from location . . . 56
7.3.1 Data . . . 56
7.3.2 Discussion . . . 57
7.4 Experiment 4 - Detecting frequent locations and frequently taken paths . . . 58
7.4.1 Data . . . 58
7.4.2 Discussion . . . 59

**8 Conclusion** **61**
8.1 Summary . . . 61
8.2 Research questions revisited . . . 61
8.2.1 RQ1 . . . 61
8.2.2 RQ2 . . . 62
8.2.3 RQ3 . . . 62
8.2.4 RQ4 . . . 62
8.3 Future work . . . 62
8.3.1 Probe requests . . . 62
8.3.2 Larger datasets . . . 62

**Bibliography** **63**

# List of Figures

2.1 Wi-Fi infrastructure mode . . . . . . . . . . 6
2.2 Bluetooth piconet . . . . . . . . . . 7

3.1 WiGLE Wi-Fi Wardriving Android app . . . . . . . . . . 12
3.2 Example CSV entries . . . . . . . . . . 14
3.3 Exporting CSV file . . . . . . . . . . 16

4.1 Sum . . . . . . . . . . 18
4.2 Difference . . . . . . . . . . 18
4.3 Intersection . . . . . . . . . . 19

5.1 Upload section . . . . . . . . . . 28
5.2 Compare section . . . . . . . . . . 29
5.3 Map section . . . . . . . . . . 30
5.4 Datatable section . . . . . . . . . . 31
5.5 Dashboard section . . . . . . . . . . 34

6.1 Experiment 6.1 . . . . . . . . . . 36
6.2 Uploading $S_2$ . . . . . . . . . . 38
6.3 Comparing $S_1$ . . . . . . . . . . 39
6.4 Comparing $S_1$, pt. 2 . . . . . . . . . . 40
6.5 Experiment 1 results . . . . . . . . . . 40
6.6 Experiment 6.2 . . . . . . . . . . 41
6.7 Uploading $S_2$ . . . . . . . . . . 42
6.8 Comparing $S_1$ . . . . . . . . . . 43
6.9 Comparing $S_1$, pt. 2 . . . . . . . . . . 44
6.10 Experiment 2 results . . . . . . . . . . 44
6.11 Experiment 6.3 . . . . . . . . . . 45
6.12 Uploading $S_1$ . . . . . . . . . . 46
6.13 Comparing $S_2$ . . . . . . . . . . 47
6.14 Comparing $S_2$, pt. 2 . . . . . . . . . . 48
6.15 Experiment 3 results . . . . . . . . . . 48
6.16 Experiment 6.4 . . . . . . . . . . 49
6.17 Experiment 4 map view . . . . . . . . . . 50
6.18 Experiment 4 filter distant duplicates . . . . . . . . . . 50
6.19 Experiment 4 after filtering with a distance of 100m . . . . . . . . . . 51
6.20 Experiment 4 after filtering with a distance of 200m . . . . . . . . . . 51
6.21 Experiment 4 after filtering with a distance of 400m . . . . . . . . . . 52

6.22 Experiment 4 after filtering with a distance of 800m . . . . . . . . . . . . . . . . . . . . 52

7.1 Experiment 1 results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 53
7.2 Experiment 1 results with no duplicates . . . . . . . . . . . . . . . . . . . . . . . . . . . 54
7.3 Experiment 1 results with no duplicates and no access points . . . . . . . . . . . . . . . 54
7.4 Experiment 2 results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55
7.5 Experiment 2 results with no access points . . . . . . . . . . . . . . . . . . . . . . . . . . 55
7.6 Experiment 3 results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57
7.7 Experiment 3 results with no duplicates . . . . . . . . . . . . . . . . . . . . . . . . . . . 57
7.8 Experiment 4 iPhone . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 58
7.9 Experiment 4 Peugeot . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 59
7.10 Experiment 4 Highway controller . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 60

# List of Tables

2.1 Bluetooth classes [18] . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

3.1 Wigle Wi-Fi CSV format for Wi-Fi devices [36] . . . . . . . . . . . . . . . . . . . . . . 13
3.2 Wigle Wi-Fi CSV format for Bluetooth devices. [36] . . . . . . . . . . . . . . . . . . . 13
3.3 Wigle Wi-Fi CSV format for cell networks pt. 1 [36] . . . . . . . . . . . . . . . . . . . 14
3.4 Wigle Wi-Fi CSV format for cell networks pt. 2 [36] . . . . . . . . . . . . . . . . . . . 15

4.1 Database design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25

6.1 Known devices for experiment 1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37
6.2 Devices turned on for both scans . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 41
6.3 Devices turned on only for second scan . . . . . . . . . . . . . . . . . . . . . . . . . 42
6.4 Device present in both scans . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43

# Listings

4.1 Trilateration in JavaScript . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
5.1 Parsing CSV file in JavaScript . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
5.2 Functions for comparing sets in JavaScript . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29
5.3 Finding devices within radius distance from point(lat1 lon1) . . . . . . . . . . . . . . . . 31
5.4 Finding distant duplicates in JavaScript . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32

# Chapter 1

# Introduction

## 1.1 Motivation

IoT is a term used to describe electronics that are capable of being controlled remotely over the internet [1]. It is growing rapidly and has become a billion-dollar industry [2]. More and more consumer electronics are shipping equipped with IoT functionality. Many people appreciate the convenience, but they might not be aware of the potential privacy issues associated with having electronics in their home or on their person connected to the internet. These devices might broadcast their presence to their surroundings. IoT also encompasses connected devices in public spaces. These devices could also pose a threat to privacy. It has become common knowledge that companies like Facebook and Google are collecting data about anyone who uses their services, yet concerns regarding IoT devices have yet to reach the mainstream.

As IoT grows, people find that more and more of their home appliances and consumer electronics come equipped with Bluetooth and Wi-Fi. However, they might not be aware of what this means in terms of privacy and security. These devices are constantly broadcasting information about themselves to the world, and these signals can travel past the walls of the home. This could potentially give unauthorized parties information about what type of devices are in the house. There is also the potential of IoT devices sensing other devices in the neighborhood or apartment complex. Furthermore, software vulnerabilities in appliances and other vital electronics in a home could allow hackers to control essential functions like heating or alarm systems.

Besides the concerns of IoT devices in private homes, there is also the issue of phones, smartwatches, cars, and other connected devices in public spaces tied to a person. These devices could potentially be giving out information about where someone is or what paths they are taking. This information could be used to discover people's private information, for example, the route someone takes to get to work or in which café someone usually has their lunch.

This thesis will discuss possible threats to privacy through IoT-based networking and provide a working prototype that lets users map out wireless devices around them, to give users an understanding of what information they are giving out to the world.

## 1.2 Contribution

This thesis will provide the theory regarding risks and concepts related to IoT-based networking, the development of a software tool for mapping and systematizing IoT devices, and propose a prototype for this system. The prototype will take data from a device that scans a location for wireless devices and store these devices with various metadata. It will include a systematic way of reviewing these devices to give the user a broad understanding of the wireless devices in an area and how IoT-based surveillance could be taking place around them.

The theory part of the thesis will focus on Wi-Fi and Bluetooth-enabled devices both in homes and in public. It will assess privacy and confidentiality by looking at what personally identifiable information from people's devices is accessible to unknown third parties and discuss how this could potentially present a privacy risk. The thesis will also look at how information not necessarily personally identifiable can cause privacy risks. This entails discussing Wi-Fi and Bluetooth in detail, focusing on what information is accessible over these wireless technologies without establishing a connection.

The prototype for the system is intended to be fully functioning, enabling the user to add data from their own device and review the data in both a database lookup with a search function and an interactive map, where the user is able to see all scanned devices in their respective location on a map. Every device should be stored with metadata like location, connection types, device capabilities, and more. The thesis will include working examples and a description of how the system works.

## 1.3 Research questions

This thesis will attempt to answer the following research questions:

- **RQ1:** What information are personal devices giving out to third parties through wireless communication?
- **RQ2:** How can information from personal devices be exploited?
- **RQ3:** How can users best understand how their privacy is affected by the internet of things?
- **RQ4:** What other risks besides privacy risks are present with the internet of things?

# Chapter 2

# Background

## 2.1 Internet of things

The internet of things was defined by the International Telecommunication Union as: "a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies." [3]. More concisely, it can be seen as the network of all manufactured objects that can be assigned an IP address, so-called *things* [4]. This includes everything from smart fridges to security cameras to heart monitor implants. Both devices communicating over Wi-Fi and devices communicating over Bluetooth are generally considered part of the internet of things.

The internet of things is commonly divided into three layers: Application layer, network layer, and perception layer. The application layer encompasses software and tools used to manipulate IoT devices. The network layer encompasses the communication between the application layer and the perception layer. The perception layer encompasses the sensors, cameras and other aspects of IoT devices used to perceive the physical world. [5]

## 2.2 MAC address

A network interface card, or NIC for short, is a computer component that provides networking capabilities for the computer. All computers that use some kind of networking need to have a NIC [6]. All NICs have a designated unique 48-bit address, called a MAC address [7]. Because this address is virtually unique, it can identify one specific device in a network. The MAC address is part of the link layer of the OSI model and therefore is only visible within a local area network. This makes it hard for someone to trace a specific device because a server cannot see the device-specific MAC address for a request, only the MAC address of the router. However, if someone is in physical range of the network, they could use a device with a NIC and packet sniffing software to obtain the MAC addresses of devices near the router.

## 2.3 Wi-Fi

Wi-Fi is a wireless communication technology trademarked by the Wi-Fi alliance. It uses the IEEE 802.11 set of standards for wireless communication [8]. There are two main modes of operation for Wi-Fi, ad-hoc mode and infrastructure mode. Ad-hoc mode lets devices communicate with each other independently, while infrastructure mode allows devices to communicate through a central access point [9]. When connecting a smartphone or laptop to the internet over Wi-Fi, this uses infrastructure mode, which lets the device communicate with an access point, which in turn communicates with the internet.

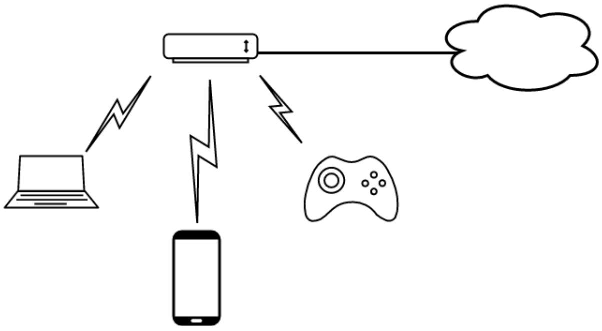


**Figure 2.1:** Wi-Fi infrastructure mode

Wi-Fi wireless signals can be transmitted in various frequency bands depending on the specific 802.11 protocol used. The most common bands for consumer electronics are 2400MHz-2500MHz, commonly referred to as the 2.4GHz band, and 5725MHz-5875MHz commonly referred to as the 5GHz band. Within these bands, there are smaller frequency ranges often referred to as channels [10]. Channels exist to avoid interference between different signals. By spreading different signals over different channels, interference between the signals becomes less likely [11]. Most devices, both access points and personal devices will only transmit and receive signals on a single channel per session [12].

802.11 works by sending information in packets. By default, network interface cards will filter packets received by SSID, channel, and MAC address. To limit or disable this filtering, some NICs have special modes that allows it to bypass some or all of this filtering, called promiscuous mode and monitor mode. Promiscuous mode disables the MAC-address filtering on the NIC, meaning it can receive all packets transmitted in the network it is connected to. Monitor mode disables the SSID filtering as well, meaning the NIC can capture all packets from all networks on the selected channel. Standard NICs are not able to capture packets on multiple channels simultaneously [12]. Some Wi-Fi access points can broadcast multiple SSIDs on a single radio. Each SSID on the access point then gets its own BSSID, which is a 48-bit identifier, similar to a MAC-address, which is derived from the MAC-address of the access point's NIC [13]. BSSID is therefore not the same as a MAC address. However, it is functionally equivalent for this thesis, so the terms will be used interchangeably.

When a Wi-Fi client is not connected to anything, it will typically use two scanning methods for finding suitable access points to connect to, passive scanning and active scanning. Passive scanning means that the client scans for signals coming from access points called beacon frames. Since access points can be on different channels, the client needs to cycle through channels to be able to find all possible access points nearby. This can take time. To speed up the process of finding access points, there is also active scanning. Active scanning means the client sends out signals called probe requests on all channels. If an access point receives a probe request, it can reply with a probe response, letting the client know that the access point is available and in range [14]. Probe requests include MAC-address of sender [15] and the SSIDs of networks that the sender is configured to join [16].

## 2.4 Bluetooth

Bluetooth is a low-powered, low-cost, and short-range wireless communications technology specified by The Bluetooth Special Interest Group (SIG). Similarly to Wi-Fi, Bluetooth works in the 2.4GHz band with channels spaced out by 1MHz. Bluetooth devices can establish connections with each other where one device is the so-called master, and the other connected devices are so-called slaves. The master device can transmit data between all its slave devices, but slave devices can not transmit data between each other. This small network of devices is called a piconet. [17]

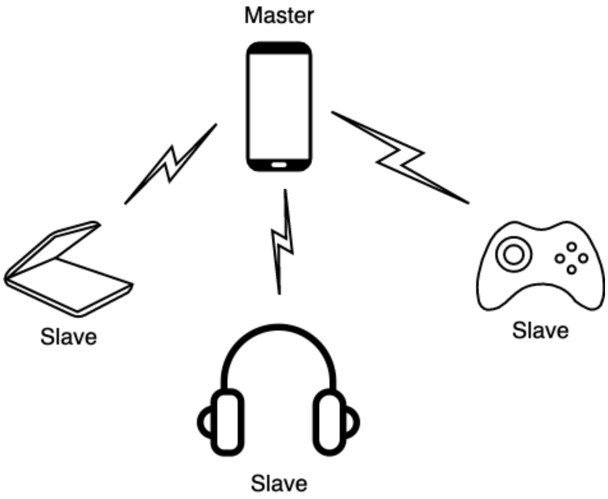


**Figure 2.2:** Bluetooth piconet

| Class | Range | Power |
|---|---|---|
| Class 1 | 100m | 100mW |
| Class 2 | 10m | 2.5mW |
| Class 3 | <10m | 1mW |

**Table 2.1:** Bluetooth classes [18]

Bluetooth devices use frequency hopping to minimize interference with other signals in the 2.4GHz band. The signals transmitted are sent one packet at a time, with each packet being sent on the next channel from the previous packet. For all the devices in the piconet to be on the same channel simultaneously, the master device sets a hopping sequence, and the slave devices follow the sequence. [17].

The connection process for Bluetooth devices is similar to that of Wi-Fi devices. One device sends out an inquiry message similar to a probe request, and other devices in range will respond with an inquiry response, similar to a probe response. The probe response contains the "BD_ADDRESS" of the device responding, which is not the same as the device's MAC address. However, it can still be used to identify a specific device. This means it would be possible to identify specific Bluetooth-enabled devices by sending inquiry messages out and collecting inquiry responses. BD_ADDRESS and MAC-address will be used interchangeably in this thesis, as the difference will not be relevant. [19]

## 2.5 Wardriving

Wardriving is the act of driving or traveling around with a portable device with GPS capabilities scanning and logging wireless networks and their location. The aim of wardriving is usually to find poorly secured or completely unsecured networks, potentially to return later to compromise those networks. Wardriving became popular in the early 2000s when computer security consultant Peter Shipley developed software that interacted with a GPS to enable mapping of unsecured networks. [20] [21]

## 2.6 Prototype technologies

**WiGLE.net** is a website acting as a worldwide central database for networks discovered through wardriving. Anyone can contribute by uploading scanned data to the central database. WiGLE uses triangulation of all the submissions of the same network to pinpoint a more accurate location. The website provides a worldwide map with all mapped devices going back to 2001. Users can also submit queries to the database. In addition to the website, WiGLE provides an Android app that lets anyone scan for devices using their android smartphone. [22]

**Vue.js** is an open-source, progressive framework for web development [23] [24]. Vue.js is used for creating Single-Page Applications (SPAs) and can create Progressive Web Apps (PWAs).

**OpenStreetMap** is a free and open-source mapping service operated by the OpenStreetMap Foundation. The maps are built using aerial imagery, GPS devices, and field maps from volunteers, and the data is free and open for anyone to use. [25]

**Leaflet** is an open-source JavaScript library for interactive maps. It is made to be lightweight and mobile-friendly. Leaflet allows developers to programmatically add markers, lines, popups, etc., to maps. [26]

## 2.7 Privacy

The Cambridge dictionary defines privacy as "someone's right to keep their personal matters and relationships secret" [27]. Article 7 of the European Convention on Human Rights states that everyone should have the right to respect for his or her private and family life, home, and communications [28]. The concept of privacy has roots going back to ancient Greece, but it has become increasingly important in modern times with the invention of various technologies making private information increasingly available to others.

Privacy in IoT can be seen as the right of someone not to have their personal matters and relationships made available to others without consent through their connected devices. One thing that separates privacy in IoT from other platforms is how all-encompassing IoT is becoming. Many people always carry at least one connected device on them, making them extra susceptible to having their personal matters disclosed. Furthermore, whereas privacy in generic systems is usually dependent on the proprietors of the system handling users' data securely, in IoT, merely the presence of a connected device could pose threats to the user's privacy. [29]

## 2.8 Visualization tools

Few visualization tools are focusing strictly on privacy. Any visualization tool will require a set of data to visualize. Visualizing privacy in IoT will then need a set of data regarding IoT devices, for example, from a wardriving session. Because today's cellphones come equipped with GPS capabilities, they can be used for scanning for wireless devices, and the data can be used to show locations and paths taken by nearby devices. The visualization can be done through graphs, lists, or maps. [30]

Visualization tools can consist of a web-based client to make it easy to use across different platforms. They might include a map with some overlay. The overlay can be used to demonstrate various information related to privacy. This could include mapping out access points, heat maps, or showing paths taken by a given device. It could also include charts relaying information related to the scanned data. Lastly, scanned devices can be viewed in a list. It is possible to use trilateration to pinpoint the location of scanned devices on a map more accurately. Using multiple scans and comparing them can make it possible to find what devices are stationary and what devices are mobile. [31] [32]

## 2.9 Related work

A 2018 article from Princeton University, USA, found that most consumers of IoT products are not concerned with potential privacy and security issues related to their IoT devices as long as the perceived convenience of having the device is great enough. It also found that the consumers relied heavily on the brand reputation of the company producing the device for judging the security of the product. [33]

A 2019 article by IEEE found that it is possible to monitor the devices in a given area with high accuracy by capturing probe request frames from Wi-Fi-enabled mobile devices. The monitoring entailed measuring the number of people in a given area in real-time with a Pearson's correlation coefficient between the actual number of people and the estimated number of people of 0.896. [34]

A 2010 research paper by Miranda Blogg, Conor Semler, Manu Hingorani, and Rod Troutbeck found that it was possible to monitor vehicle traffic using Bluetooth scanners along the roads. The paper study showed that at the time, the accuracy of the technique was not that good, but as more and more vehicles come equipped with Bluetooth, it is possible that it would be possible to monitor vehicles using Bluetooth technology accurately. [35]

A 2016 report by Almar Tillekens, Nhien-An Le-Khac, and Thanh Thoa Pham Thi titled "A Bespoke Forensics GIS Tool" discusses a web-based Geographic Information Systems tool used to extract, analyze and visualize geospatial data for law enforcement agencies. The tool could be used to visualize scanned devices in an area on a map, as well as compare different scans to each other to distinguish stationary devices from mobile devices. [32]

A 2013 conference paper by Valeros Verónica and Sebastián Garcia titled "How Bluetooth may jeopardize your privacy. An analysis of people behavioral patterns in the street." discusses how Bluetooth devices can be tied to a specific person, and proposes a tool that can visualize and aid in following Bluetooth devices in public spaces. [30]

# Chapter 3

# Data collection

## 3.1 Collecting data

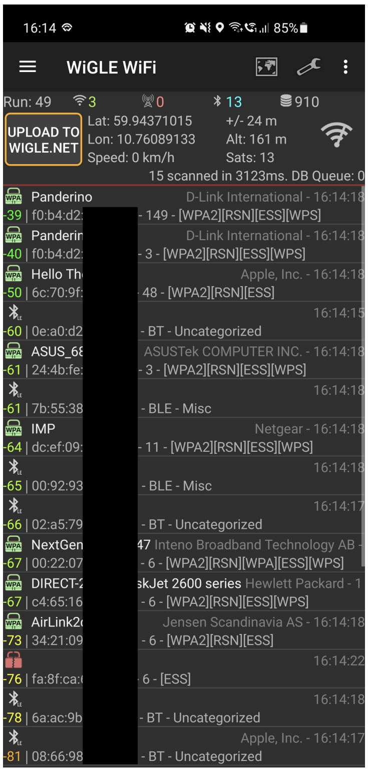
In order to demonstrate what information can be gathered through wireless scanning, some sample data is required. Collecting data of wireless devices can be done in various ways, but the easiest way is probably to use a phone app. There are various apps available that let the user scan for wireless devices, but the most functional one seems to be the "WiGLE Wi-Fi Wardriving" app. The Wigle app is made to be used with "wigle.net", a wardriving database website. However, it can also be used stand-alone and export scanned devices to a file.

The Wigle app lets the user scan for Wi-Fi and Bluetooth devices while using other apps or while the phone is locked as long as the app is running in the background. This makes it easy to scan for devices while driving or walking. The app has a local database where scanned devices are stored. The database entries can be uploaded to wigle.net directly or to a CSV file which can be saved locally on the device or sent via e-mail or other sharing services.

The app lets the user set a time interval for how often it will scan for devices. The same device can be scanned multiple times. However, the interval between each time a device is added as an entry in the database appears to be quite arbitrary, even if the device is in range for an extended period. The app is also reliant on a stable GPS signal. If the phone running the app loses the GPS signal while scanning, scanned devices' locations are estimated, and the timestamp is not recorded. Wi-Fi and Bluetooth-enabled smartphones do generally not appear on the Wigle app. The exception is when they are actively scanning for Bluetooth devices.

## 3.2 Data format

The CSV file has devices in rows with the format as seen in Table 3.1, 3.2, 3.3, and 3.4. An example of how typical entries might look is shown in Figure 3.2. There are three types of entries, Wi-Fi devices, Bluetooth devices, and Cell networks. All entries have 11 fields, and only the first two fields and the fifth field are different between Wi-Fi, Bluetooth and cell network entries. For Wi-Fi entries, the first two fields are: BSSID, SSID and the fifth field is Channel. For Bluetooth entries, the first two fields are: BD_ADDR, Device Name and the fifth field is Channel. For cell networks the first two fields are: Cell Key, Network Name and the fifth field is Frequency. All other fields are the same across all types of entries.



```
16:14                                   85%
WiGLE WiFi
Run: 49   3   0   13   910
UPLOAD TO   Lat: 59.94371015    +/- 24 m
WIGLE.NET   Lon: 10.76089133    Alt: 161 m
            Speed: 0 km/h       Sats: 13
                 15 scanned in 3123ms. DB Queue: 0
Panderino                D-Link International - 16:14:18
-39 | f0:b4:d2:[redacted] - 149 - [WPA2][RSN][ESS][WPS]
Panderin                 D-Link International - 16:14:18
-40 | f0:b4:d2:      - 3 - [WPA2][RSN][ESS][WPS]
Hello Th                       Apple, Inc. - 16:14:18
-50 | 6c:70:9f:      48 - [WPA2][RSN][ESS]
                                            16:14:15
-60 | 0e:a0:d2       - BT - Uncategorized
ASUS_68              ASUSTek COMPUTER INC. - 16:14:18
-61 | 24:4b:fe:      - 3 - [WPA2][RSN][ESS][WPS]
                                            16:14:18
-61 | 7b:55:38       - BLE - Misc
IMP                                Netgear - 16:14:18
-64 | dc:ef:09:      - 11 - [WPA2][RSN][ESS][WPS]
                                            16:14:18
-65 | 00:92:93       - BLE - Misc
                                            16:14:17
-66 | 02:a5:79       - BT - Uncategorized
NextGen        47 Inteno Broadband Technology AB -
-67 | 00:22:07       - 6 - [WPA2][RSN][WPA][ESS][WPS]
DIRECT-2       skJet 2600 series Hewlett Packard - 1
-67 | c4:65:16       - 6 - [WPA2][RSN][ESS][WPS]
AirLink2             Jensen Scandinavia AS - 16:14:18
-73 | 34:21:09       - 6 - [WPA2][RSN][ESS]
                                            16:14:22
-76 | fa:8f:ca:      6 - [ESS]
                                            16:14:18
-78 | 6a:ac:9b       - BT - Uncategorized
                               Apple, Inc. - 16:14:17
-81 | 08:66:98       - BT - Uncategorized
```

**Figure 3.1:** WiGLE Wi-Fi Wardriving Android app

- *Type specific*
- *Type specific*
- Capabilities
- First timestamp seen
- *Type specific*
- RSSI
- Latitude
- Longitude
- Altitude
- Accuracy
- Type

Both BSSID and BD_ADDR will hereby be referred to as MAC-address, as they are both functionally equivalent to the device's hardware address for this thesis.

## 3.3 Data exporting

Exporting data from the Wigle Wi-Fi wardriving app can be done by entering the database section of the app and pressing either "CSV EXPORT RUN" or "CSV EXPORT DB" depending on whether the

| Field Name | Description | Example |
|---|---|---|
| BSSID | Basic Service Set Identifier - hardware address | 1a:9f:ee:5c:71:c6 |
| SSID | Service Set Identifier - the access point name | Scampoodle |
| Capabilities | Capabilities array as specified by package. Based on Android capabilities sets. | [WPA2-EAP-CCMP][ESS] |
| First timestamp seen | First timestamp seen in SQL seconds-precision time format ( YYYY-MM-DD hh:mm:ss), assumes UTC | 2018-08-01 13:08:27 |
| Channel | Integer channel vaule for the observed signal | 161 |
| RSSI | Received Signal Strength Indicator (RSSI) as reported by the radio | -43 |
| Latitude | Observed latitude in decimal ( degrees.decimal degrees) format | 37.76578028 |
| Longitude | Observed longitude in decimal ( degrees.decimal degrees) format | -123.45919439 |
| Altitude | Estimated position altitude in integer meters | 67 |
| Accuracy | Estimated position accuracy in decimal meters | 3.2160000801086426 |
| Type | The device type - always Wi-Fi for Wi-Fi networks currently | Wi-Fi |

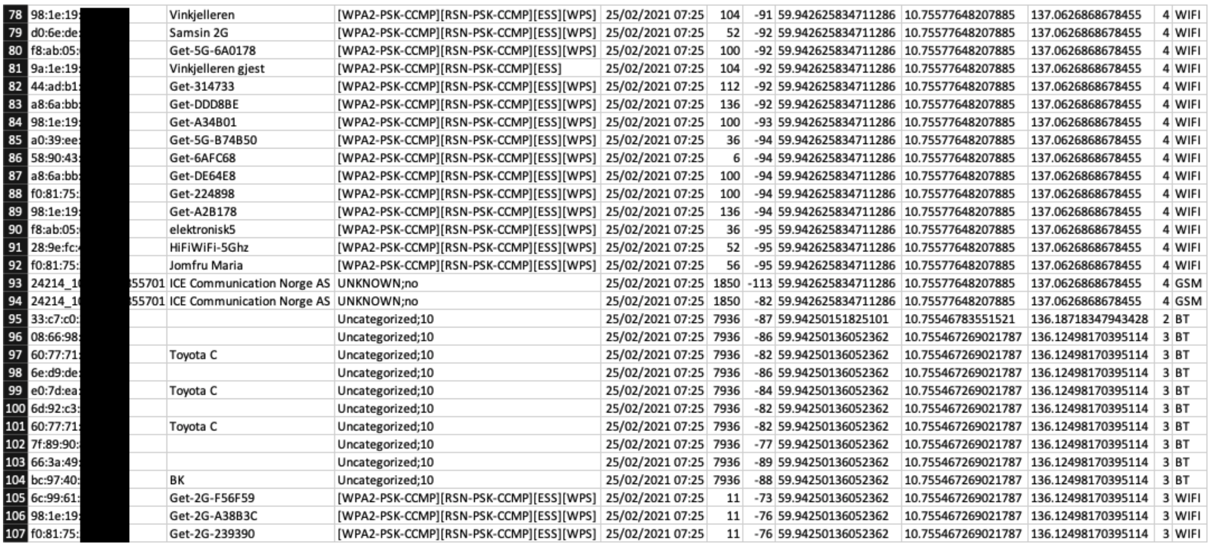
**Table 3.1:** Wigle Wi-Fi CSV format for Wi-Fi devices [36]

| Field Name | Description | Example |
|---|---|---|
| BD_ADDR | Bluetooth Device Address - hardware address | 1a:9f:ee:5c:71:c6 |
| Device Name | The published name of the blueooth device if available | Jabra Headset |
| Capabilities | List; includes the device type list if provided, and optionally [BT] or [BLE] enclosed in square brackets to describe the first scan type detecting the device. | Misc [LE] |
| First timestamp seen | First timestamp seen in SQL seconds-precision time format ( YYYY-MM-DD hh:mm:ss), assumes UTC | 2018-08-01 13:08:27 |
| Channel | 0 for Bluetooth devices | 0 |
| RSSI | Received Signal Strength Indicator (RSSI) as reported by the radio | -67 |
| Latitude | Observed latitude in decimal ( degrees.decimal degrees) format | 37.73090571 |
| Longitude | Observed longitude in decimal ( degrees.decimal degrees) format | -122.42877987 |
| Altitude | Estimated position altitude in integer meters | 104 |
| Accuracy | Estimated position accuracy in decimal meters | 49.3120002746582 |
| Type | The device type - always<br>BT<br>or<br>BLE for Bluetooth or BTLE devices respectively | BLE |

**Table 3.2:** Wigle Wi-Fi CSV format for Bluetooth devices. [36]

| Field Name | Description | Example |
|---|---|---|
| Cell Key | Composite Identifier - depends on network type:<br>For CDMA: [System]_[Network]_[Base Station]<br>For GSM: [Operator*]_[LAC]_[Cell ID]<br>For WCDMA: [Operator*]_[LAC]_[Cell ID]<br>For LTE: [Operator*]_[LAC]_[Cell ID]<br>*Operator:<br>for GSM-derived networks this is MCCMNC concatenated to a 6 digit format. MCC and MNC are searchable through the Wigle API via the https://api.wigle.net/api/v2/cell/mccMnc endpoint. | 310410_56967_4118917 |
| Network Name | For GSM/WCDMA/LTE:<br>Reported MCCMNC lookup value if available, otherwise reported network name from radio/SIM.<br><br>For CDMA:<br>reported network name from radio. | AT&T Mobility |
| Capabilities | For GSM/WCDMA/LTE:<br>Network type + ; + MCC+MNC (operator value)<br><br>For CDMA:<br>Network type + ; + System | WCDMA;310410 |
| First timestamp seen | First timestamp seen in SQL seconds-precision time format (YYYY-MM-DD hh:mm:ss), assumes UTC | 2018-08-01 13:08:27 |

**Table 3.3:** Wigle Wi-Fi CSV format for cell networks pt. 1 [36]



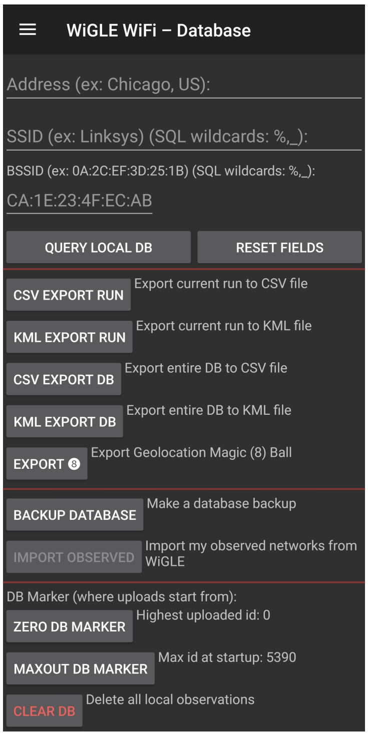
| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 78 | 98:1e:19:[redacted] | Vinkjelleren | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 104 | -91 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 79 | d0:6e:de:[redacted] | Samsin 2G | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 52 | -92 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 80 | f8:ab:05:[redacted] | Get-5G-6A0178 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 100 | -92 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 81 | 9a:1e:19:[redacted] | Vinkjelleren gjest | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 25/02/2021 07:25 | 104 | -92 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 82 | 44:ad:b1:[redacted] | Get-314733 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 112 | -92 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 83 | a8:6a:bb:[redacted] | Get-DDD8BE | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 136 | -92 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 84 | 98:1e:19:[redacted] | Get-A34B01 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 100 | -93 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 85 | a0:39:ee:[redacted] | Get-5G-B74B50 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 36 | -94 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 86 | 58:90:43:[redacted] | Get-6AFC68 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 6 | -94 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 87 | a8:6a:bb:[redacted] | Get-DE64E8 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 100 | -94 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 88 | f0:81:75:[redacted] | Get-224898 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 100 | -94 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 89 | 98:1e:19:[redacted] | Get-A2B178 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 136 | -94 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 90 | f8:ab:05:[redacted] | elektronisk5 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 36 | -95 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 91 | 28:9e:fc:[redacted] | HiFiWiFi-5Ghz | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 52 | -95 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 92 | f0:81:75:[redacted] | Jomfru Maria | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 56 | -95 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | WIFI |
| 93 | 24214_1[redacted]55701 | ICE Communication Norge AS | UNKNOWN;no | 25/02/2021 07:25 | 1850 | -113 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | GSM |
| 94 | 24214_1[redacted]55701 | ICE Communication Norge AS | UNKNOWN;no | 25/02/2021 07:25 | 1850 | -82 | 59.942625834711286 | 10.75577648207885 | 137.0626868678455 | 4 | GSM |
| 95 | 33:c7:c0:[redacted] | | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -87 | 59.94250151825101 | 10.75546783551521 | 136.18718347943428 | 2 | BT |
| 96 | 08:66:98:[redacted] | | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -86 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 97 | 60:77:71:[redacted] | Toyota C | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -82 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 98 | 6e:d9:de:[redacted] | | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -86 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 99 | e0:7d:ea:[redacted] | Toyota C | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -84 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 100 | 6d:92:c3:[redacted] | | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -82 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 101 | 60:77:71:[redacted] | Toyota C | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -82 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 102 | 7f:89:90:[redacted] | | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -77 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 103 | 66:3a:49:[redacted] | | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -89 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 104 | bc:97:40:[redacted] | BK | Uncategorized;10 | 25/02/2021 07:25 | 7936 | -88 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | BT |
| 105 | 6c:99:61:[redacted] | Get-2G-F56F59 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 11 | -73 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | WIFI |
| 106 | 98:1e:19:[redacted] | Get-2G-A38B3C | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 11 | -76 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | WIFI |
| 107 | f0:81:75:[redacted] | Get-2G-239390 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 25/02/2021 07:25 | 11 | -76 | 59.94250136052362 | 10.755467269021787 | 136.12498170395114 | 3 | WIFI |

**Figure 3.2:** Example CSV entries

| Field Name | Description | Example |
|---|---|---|
| Frequency | Integer center frequency value depending on network type:<br><br>For CDMA: 0<br>For GSM: Observed ARFCN<br>For WCDMA: Observed UARFCN<br>For LTE: Observed EARFCN | 4385 |
| RSSI | Received Signal Strength Indicator (RSSI) as reported by the radio | -81 |
| Latitude | Observed latitude in decimal ( degrees.decimal degrees) format | 37.72090053 |
| Longitude | Observed longitude in decimal ( degrees.decimal degrees) format | -122.44579219 |
| Altitude | Estimated position altitude in integer meters | 104 |
| Accuracy | Estimated position accuracy in decimal meters | 34.30400085449219 |
| Type | The device type - always one of CDMA, WCDMA, GSM or LTE. | WCDMA |

**Table 3.4:** Wigle Wi-Fi CSV format for cell networks pt. 2 [36]

user wants to export only the devices scanned since starting the app or since the database was last reset respectively, see Figure 3.3.



Address (ex: Chicago, US):

SSID (ex: Linksys) (SQL wildcards: %,_):

BSSID (ex: 0A:2C:EF:3D:25:1B) (SQL wildcards: %,_):

CA:1E:23:4F:EC:AB

Export current run to CSV file

Export current run to KML file

Export entire DB to CSV file

Export entire DB to KML file

Export Geolocation Magic (8) Ball

Make a database backup

Import my observed networks from WiGLE

DB Marker (where uploads start from):

Highest uploaded id: 0

Max id at startup: 5390

Delete all local observations

**Figure 3.3:** Exporting CSV file

Chapter 4

# Design

## 4.1 Purpose

To demonstrate privacy-related risks related to Wi-Fi and Bluetooth enabled devices, a prototype system for visualizing the risk will be developed. The prototype is meant to take the scanned data from the Wigle Wardriving Android app and demonstrate privacy-related risks related to the data to the user. It will be used both to demonstrate what information an adversary could obtain from scanning Wi-Fi and Bluetooth devices in people's homes and in public, as well as being used for experiments in this thesis.

## 4.2 Specification

The prototype should satisfy the following specifications:

1. The prototype shall be web-based to make it easily accessible on various devices.
2. It shall be responsive to also work on tablets and phones.
3. It shall be usable offline.
4. It shall be usable for users with no prior understanding of how wireless devices work.
5. It shall enable the user to perform the six use-cases described in section 4.3.

### 4.2.1 Functionality

The prototype shall support the following functionality:

**Uploading CSV file**

The prototype should let the user upload a CSV file in the Wigle format, as seen in Table 3.1, 3.2, 3.3, and 3.4. The entries in the CSV file should then be added to an internal data structure in the prototype application. The entries should then be available for the user to interact with in other parts of the application. Uploading a new file should erase the old entries from the internal data structure.

### Comparing multiple CSV files

Once a selection of entries has been added to the internal data structure, the prototype should let the user compare another file. The user should be able to find the sum, the difference, and the intersection of the current selection and an additional file and make this the current selection. Given the current selection, $B$, and the new file, $A$, the operations will work as shown in Figures 4.1, 4.2, and 4.3.

#### *Sum*

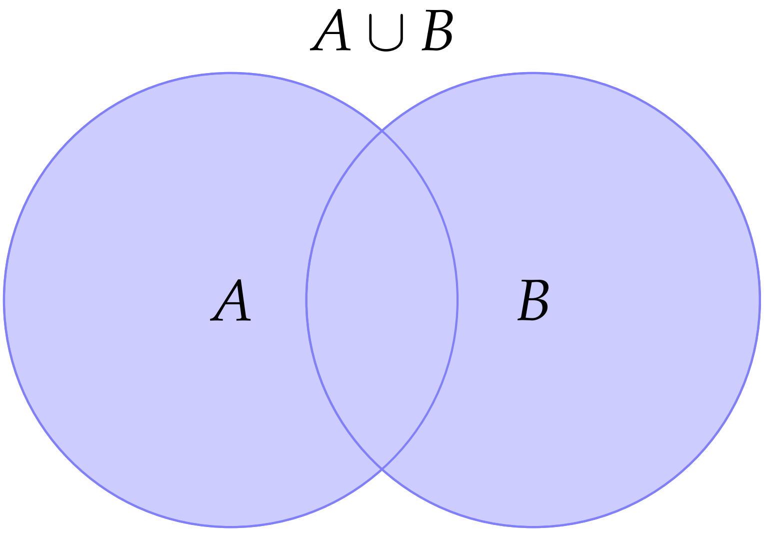


**Figure 4.1:** Sum

#### *Difference*

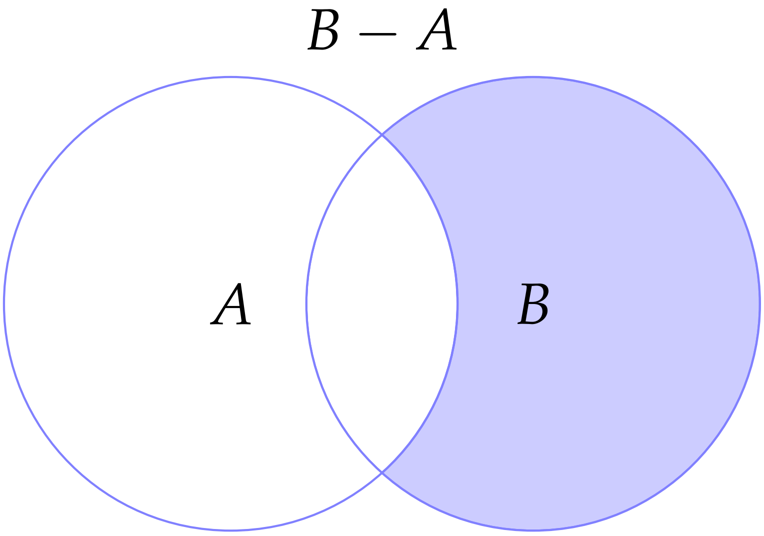


**Figure 4.2:** Difference

#### *Intersection*

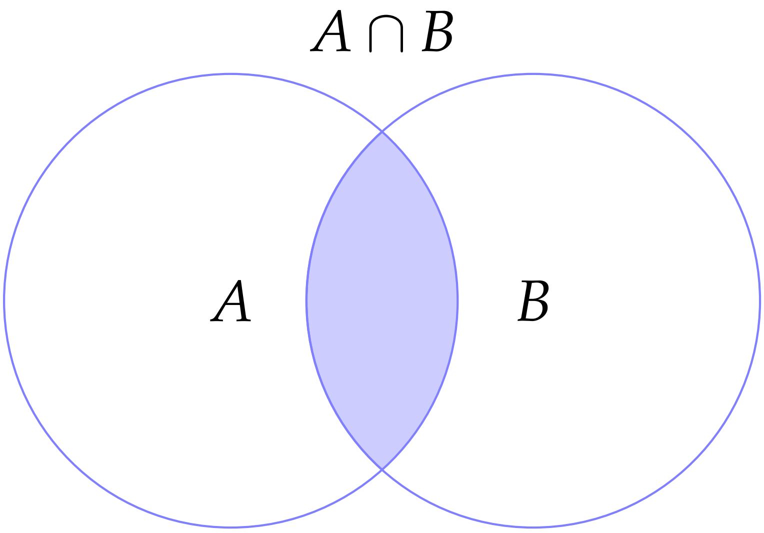


**Figure 4.3:** Intersection

### Interactive map

The prototype should have an interactive map. The map should show the user all the devices in the current selection in their respective location. The map should allow for standard map functions like moving around and zooming. The map should let the user press any of the devices to see information about that device. There should also be options for filtering the devices shown on the map based on SSID/name, MAC address, and type.

### Datatable view

The prototype should let the user see all the entries in the current selection in a table view. The table should follow the format as specified in section 3.2. The user should be able to search the entries in the table.

### Dashboard

The prototype should have a page where various information about the current selection is easily visible. In addition, there should be widgets that let the user manipulate the current selection. The widgets should include:

- **Get vendor info** - Lets the user see the distribution of network card vendors in the current selection and the countries of origin for those vendors.
- **Filter selection** - Lets the user filter the selection based on MAC-address, SSID/name or type.
- **Find location of device** - Lets the user find a more accurate location for a device based on its MAC address.
- **Find devices at address** - Lets the user see all devices in the current selection that are at a given address.
- **Find time device was spotted** - Lets the user enter the MAC address of a device in the current selection and returns a list of all the timestamps of times that device has been scanned.

- **Filter distant duplicates** - Lets the user enter a distance in meters, then filters the current selection to only include devices that have been scanned more than once and have at least the given distance between the scanned locations.
- **Remove duplicates** - Removes all duplicate entries in the current selection

## 4.3 Use-cases

The web app is designed to visualize privacy risks surrounding IoT devices. The prototype and the thesis will focus on four main use-cases:

### 4.3.1 Detecting residents in homes

***Method:*** Scanning for devices near a house or apartment, then analyzing the scanned data to determine whether someone is home at a given time and what type of consumer electronics are present in the home.

***Privacy breach:*** Privacy in the home

***Consequences:*** An adversary could figure out whether someone is home at a given time and what type of consumer electronics are present in the home. This could assist the person in deciding if a home is worth breaking into, when to break into the home, and what equipment they need to bring for the break-in.

***Data processing:*** Determining what devices are present in a house entails taking scanned data from when many devices are on in the house and also from when few devices are on, then finding the set difference between the scans. Finding the difference should eliminate devices not in the house since they will likely be present in both scans and therefore filtered out.

***Analyzing the data:*** The data processing returns a list of devices. The SSID/name of the devices can often tell what type of device it is. Smart-TVs will, for example, often have "TV" at the beginning of their SSID/name. This gives the user an overview of what kind of consumer electronics are present in the home. If the list contains one or more phones, that could be an index that someone was home when the phones were scanned.

### 4.3.2 Detecting visitors in homes

***Method:*** Scanning for devices near a house or apartment, then analyzing the scanned data to determine whether the people living in the home have visitors over at a given time.

***Privacy breach:*** Privacy in the home, privacy of relationships

***Consequences:*** An adversary could figure out if there are visitors in a given home based on what consumer electronics are present.

***Data processing:*** Determining if there is an unusually high amount of devices in a house entails first establishing a baseline for what constitutes a normal amount of devices, then scanning and comparing with the baseline to see if there are more devices present than normal.

***Analyzing the data:*** The data processing returns a list of devices. The SSID/name of the devices can often tell what type of device it is. This gives the user an overview of what kind of consumer electronics are present in the home. If the list contains more personal electronics than normal, this can indicate that there are visitors visiting when the devices were scanned.

### 4.3.3 Adding context from location

***Method:*** Scanning for devices near a house or apartment, then scanning near a location that can add context to what the person is doing. Examples would be specialized medical facilities, an office, a brothel, or another location that can add context to what the person is doing.

***Privacy breach:*** Privacy of medical data, context added by location

***Consequences:*** An adversary could figure out private information regarding medical data, work hours, or other sensitive information related to the context of a location.

***Data processing:*** Determining if a person from a given house or apartment has visited a given location can be done by cross-referencing devices scanned at the house or apartment with devices scanned at the given location.

***Analyzing the data:*** If one or more matches are found, that is a strong indicator that one or more persons from the given household have visited the given location. If the given location is of a certain facility/establishment, it could give context to what the person was doing there.

### 4.3.4 Detecting frequent locations and frequently taken paths

***Method:*** Scanning for devices in multiple locations over time in order to find patterns and paths taken by individuals.

***Privacy breach:*** Stalking, profiling, predicting locations

***Consequences:*** An adversary could use the information to ascertain where certain individuals usually are and move throughout a given time period to learn patterns in the individuals' daily lives and potentially predict where individuals will be at a given time.

***Data processing:*** Finding patterns of locations and paths taken by individuals can be done by scanning for devices in multiple locations over a longer period of time, then detecting if certain devices appear in more than one location and detecting patterns in this behavior.

***Analyzing the data:*** If a device appearing in more than one location is found, that is an indicator of the device being portable. It can then be singled out to let the user see patterns of how it is moving to potentially discover patterns and periodically visited locations for the given device. This could allow an adversary to stalk, profile or predict where someone will be at a given time.

## 4.4 Technology

To satisfy specification points 1, 2 and 3, the prototype will be developed as a web-based application. More specifically, it will be developed as a responsive Single-Page Application (SPA), Progressive Web

App (PWA), using the Vue.js framework for web development. In order to use maps in the application, it will use Leaflet, a mapping library for javascript. For quick and streamlined styling, the application will use Material Design for Bootstrap.

## 4.5 Concepts

### 4.5.1 Trilateration

***Theory:***
The scan entries from the Wigle app give the Latitude and Longitude from the GPS of the phone the app is running on, meaning it does not accurately represent the location of the scanned device. In order to more accurately represent the location of the access point, trilateration/multilateration can be used [31]. Trilateration/multilateration is the method of determining the precise location of a point based on the distance from that point to three other points. Therefore, in order to perform trilateration on a scanned device, at least three separate scans of the device are required.

The first thing needed for trilateration will be a way of determining the distance from the location of the scanning device from the scanned device. As seen in Table 3.1, 3.2, and 3.3, the Wigle android app gives the RSSI of scanned devices as an entry in the CSV file. RSSI is affected by much more than just the distance between the devices. However, it can be used as an indicator of an approximate distance. The paper "An RSSI-based Wireless Sensor Node Localisation using Trilateration and Multilateration Methods for Outdoor Environment" gives the formulas needed for finding the exact location of a device based on RSSI [31]. The equations are as follows:

The distance from a scanned device to the scanning device can be calculated as:

The distance, $d$, between the two devices is given as 10 to the power of the RSSI, $r$, divided by 10 times a constant, $n$, as follows:

$$d = 10^{\left(\frac{r}{10n}\right)} \tag{4.1}$$

This gives an approximated distance between a scanned device and the scanning device.

The distance can also be expressed as follows:

The distance squared, $d^2$, is given as the x coordinate of the scanned device, $x$, minus the x coordinate of the scanning device, $x'$, squared, plus the y coordinate of the scanned device, $y$, minus the y coordinate of the scanning device, $y'$, squared as follows:

$$d^2 = (x - x')^2 + (y - y')^2 \tag{4.2}$$

Which can be linearized as presented here:

$$\begin{gathered} -2(x_{n-1} - x_n)x - 2(y_{n-1} - y_n) \\ = \\ (d_{n-1}^2 - d_n^2) - (x_{n-1}^2 - x_n^2) + (y_{n-1}^2 - y_n^2) \end{gathered} \tag{4.3}$$

Which can be rewritten in matrices as:

$$AX = B$$

Where

$$A = \begin{bmatrix} -2(x_1 - x_n) & -2(y_1 - y_n) \\ -2(x_2 - x_n) & -2(y_2 - y_n) \\ \vdots & \vdots \\ -2(x_{n-1} - x_n) & -2(y_{n-1} - y_n) \end{bmatrix}, X = \begin{bmatrix} x \\ y \end{bmatrix} \tag{4.4}$$

$$B = \begin{bmatrix} (d_1^2 - d_n^2) - (x_1^2 - x_n^2) + (y_1^2 - y_n^2) \\ (d_2^2 - d_n^2) - (x_2^2 - x_n^2) + (y_2^2 - y_n^2) \\ \vdots \\ (d_{n-1}^2 - d_n^2) - (x_{n-1}^2 - x_n^2) - (y_{n-1}^2 - y_n^2) \end{bmatrix}$$

The approximated location of the scanned device can be solved using:

$$X = A^{-1} \cdot B \tag{4.5}$$

For A to have an inverse, it must be a square matrix, meaning this can only be solved when $n = 3$, meaning exactly three entries from the Wigle android app must be used to calculate the approximate position of the scanned device.

***Implementation:***

A javascript implementation (using the math.js library for matrix operations) would look something like this:

```
trilateration(p1, p2, p3) {
  const n = 2.185; // From An RSSI-based Wireless Sensor Node Localisation using Trilateration and Multilateration Methods for Outdoor Environment

  const r1 = p1.rssi; // RSSI of point 1
  const r2 = p2.rssi; // RSSI of point 2
  const r3 = p3.rssi; // RSSI of point 3

  const x1 = p1.x;
  const y1 = p1.y;
  const x2 = p2.x;
  const y2 = p2.y;
  const x3 = p3.x;
  const y3 = p3.y;

  const d1 = Math.pow(10, (r1) / (10 * n)); // Distance between point 1 and source
  const d2 = Math.pow(10, (r2) / (10 * n)); // Distance between point 2 and source
  const d3 = Math.pow(10, (r3) / (10 * n)); // Distance between point 3 and source

  const A = math.matrix([
    [-2 * (x1 - x3), -2 * (y1 - y3)],
    [-2 * (x2 - x3), -2 * (y2 - y3)]
  ]);

  const B = math.matrix([
    (Math.pow(d1, 2) - Math.pow(d3, 2)) -
    (Math.pow(x1, 2) - Math.pow(x3, 2)) -
    (Math.pow(y1, 2) - Math.pow(y3, 2)),
    (Math.pow(d2, 2) - Math.pow(d3, 2)) -
    (Math.pow(x2, 2) - Math.pow(x3, 2)) -
    (Math.pow(y2, 2) - Math.pow(y3, 2))
  ]);

  const X = math.multiply(math.inv(A), B);

  return {x: X._data[0], y: X._data[1]};
}
```

**Listing 4.1:** Trilateration in JavaScript

## 4.6 Architecture

### 4.6.1 Database design

The database will consist of a single table of key-value pairs to work as an IndexedDB database [37]. See Table 4.1. The first pair will have the key "devices" and the value will be a JavaScript array of all the devices in the current selection. The second pair will have the key "tvs" and the value will be a javascript array of all the devices with the substring "TV" in its SSID/name. The third pair will have the key "types" and the value will be a javascript array of length 2, containing the number of devices of type Wi-Fi as its first element and the number of devices of type Bluetooth as its second element.

| # | Key | Value |
|---|---|---|
| 0 | "devices" | *javascript array* |
| 1 | "tvs" | *javascript array* |
| 2 | "types" | *javascript array* |

**Table 4.1:** Database design

# Chapter 5

# Implementation

Project and CSV files are available at:

https://github.com/nicolaironning/Master

## 5.1 Project

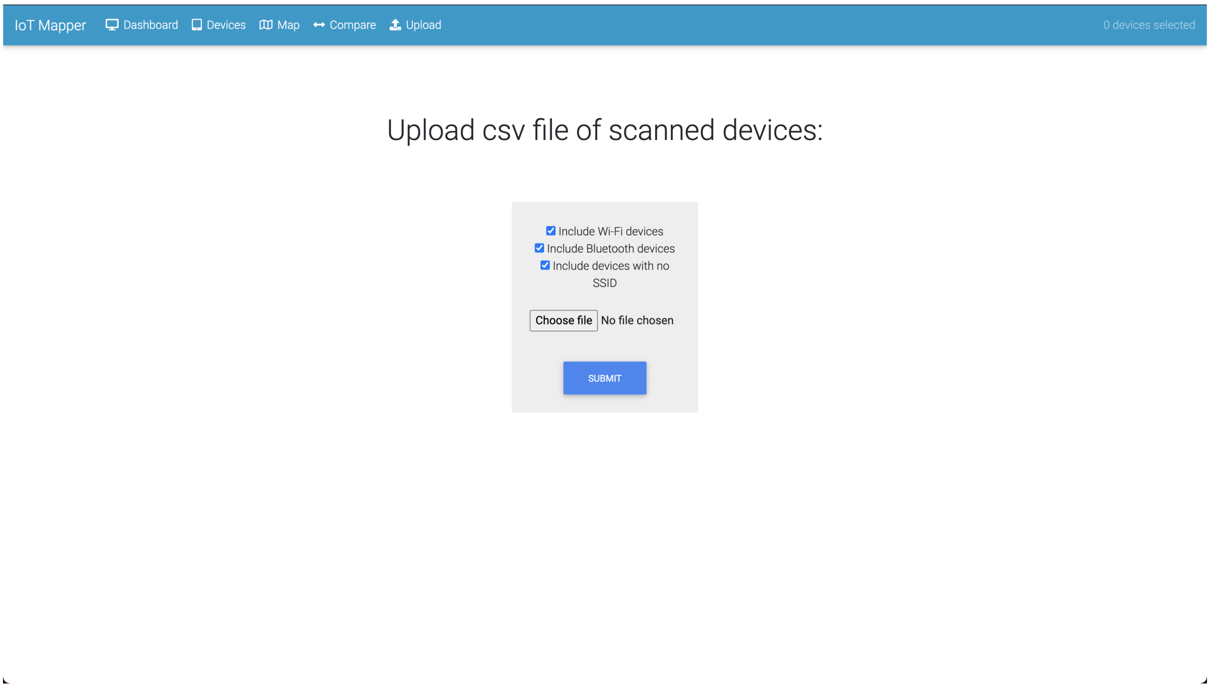
The project was set up using the Vue CLI tool and run using Node.js. Dependencies were added using Node Package Manager. The live version of the web app was packaged with Node Package Manager and hosted on Github. The web app was set up with tabs for each section of the application placed on a navigation bar at the top of the page. The navigation bar responds to mobile screen sizes by collapsing to a burger menu. The navigation bar is static and available from all sections of the page. The web app uses IndexedDB for storing the entries that users upload.

## 5.2 Upload

The upload section of the web app consists of a single upload module. The module allows the user to upload a CSV file. The user can select any CSV file on their device. When the user clicks the submit button, the upload module parses the devices in the CSV file into an array of objects. This is shown in Listing 5.1. The text from the CSV file is split into an array on the newline character. The first two rows of the file are skipped, as these are metadata and field names. Entries that have the type cell network or have no timestamp are skipped. A small amount of jitter is added to the coordinates to avoid the entries being right on top of each other on the map and not clickable. The IndexedDB of the web app is then initialized to remove any previous data that might be present. The array is then inserted into the IndexedDB database of the web app.

```
// Get text from CSV file and split into array on newline
var arr = e.target.result.split("\n");

// Skip first two rows, this is metadata and field names
for (var i = 2; i < arr.length - 1; i++) {
  var tmp = arr[i].split(",");

  // Skip cell devices
  if(tmp[10] == 'GSM' || tmp[10] == 'LTE') continue;

  // Skip devices with no timestamp
  if(tmp[3].includes(’1970’)) continue;

  // Add jitter to make all devices clickable on map
  const jitterLat = Math.random() ∗ 0.00001;
  const jitterLon = Math.random() ∗ 0.00001;
  const sign = Math.random() < 0.5 ? -1 : 1;

  var obj = {
    id: i - 2,
    mac: tmp[0],
    ssid: tmp[1],
    authMode: tmp[2],
    firstSeen: tmp[3],
    channel: tmp[4],
    rssi: tmp[5],
    currentLatitude: parseFloat(tmp[6]) + jitterLat ∗ sign,
    currentLongitude: parseFloat(tmp[7]) + jitterLon ∗ sign,
    altitudeMeters: tmp[8],
    accuracyMeters: tmp[9],
    type: tmp[10] == ’BLE’ ? ’BT’ : tmp[10],
  };

  this.devices.push(obj);
}
```

**Listing 5.1:** Parsing CSV file in JavaScript



Upload csv file of scanned devices:

☑ Include Wi-Fi devices
☑ Include Bluetooth devices
☑ Include devices with no SSID

**Figure 5.1:** Upload section

## 5.3 Compare

The compare section of the web app consists of the same upload module as the upload section. However, in this section, the CSV file is not immediately inserted into the IndexedDB database. The user is first presented with three options: intersection, difference, or sum. Based on the button the user presses, the array is then compared to the existing array in the IndexedDB database, and a new array is created. The code for these three operations is shown in Listing 5.2.

```javascript
intersectTask: (arr1, arr2) => {
  return arr1.filter(c => arr2.findIndex(x => x.mac == c.mac) > -1);
},
differenceTask: (arr1, arr2) => {
  const getDifference = (a, b, fn) => {
    const setB = new Set(b.map(item => fn(item)));
    return [...a.filter(item => !setB.has(fn(item)))]
  };
  return getDifference(arr1, arr2, (x => x.mac));
},
sumTask: (arr1, arr2) => {
  return arr1.concat(arr2);
}
```

**Listing 5.2:** Functions for comparing sets in JavaScript

The IndexedDB of the web app is then initialized to remove the existing array. The new array is then inserted into the IndexedDB database of the web app.

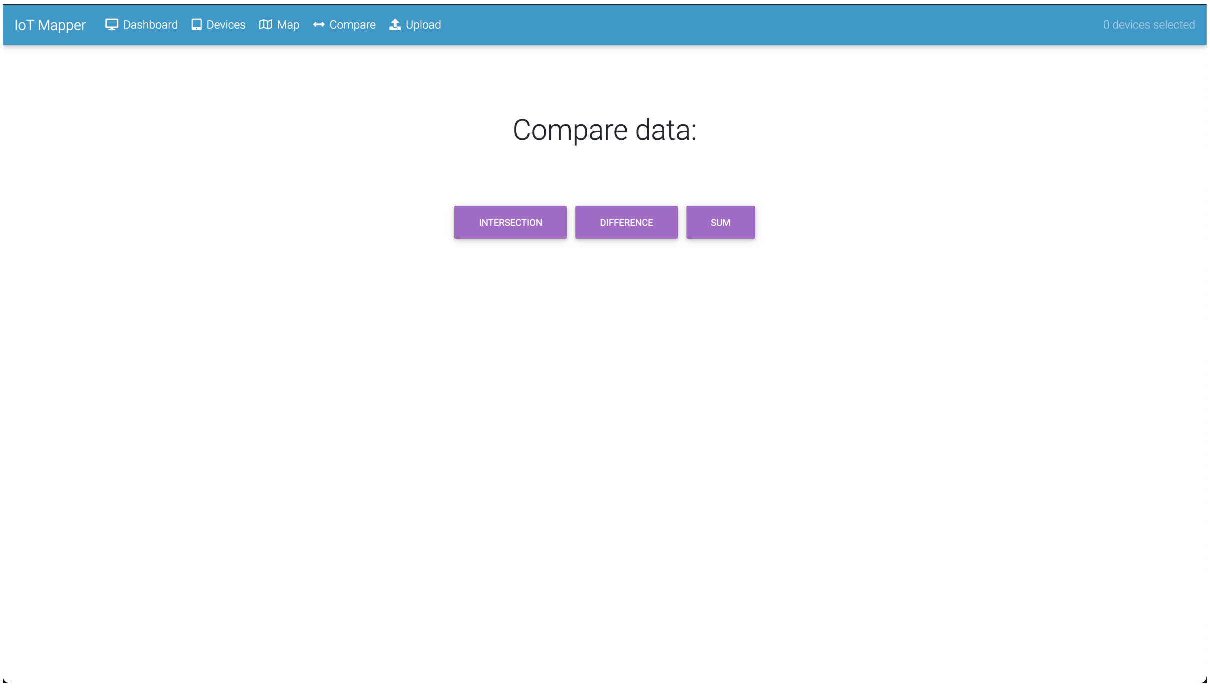


**Figure 5.2:** Compare section

## 5.4 Map

The map section of the web app consists of a single interactive map built with leaflet. The map displays all the devices in the current selection as circles. Pressing a circle opens a pop-up window that lets the user see details about that specific device. The pop-up window also contains a button letting the user filter the map to only show instances of that specific device. There is also a button to let the user open a side panel with options to filter devices based on SSID/name, MAC address, or type of device.

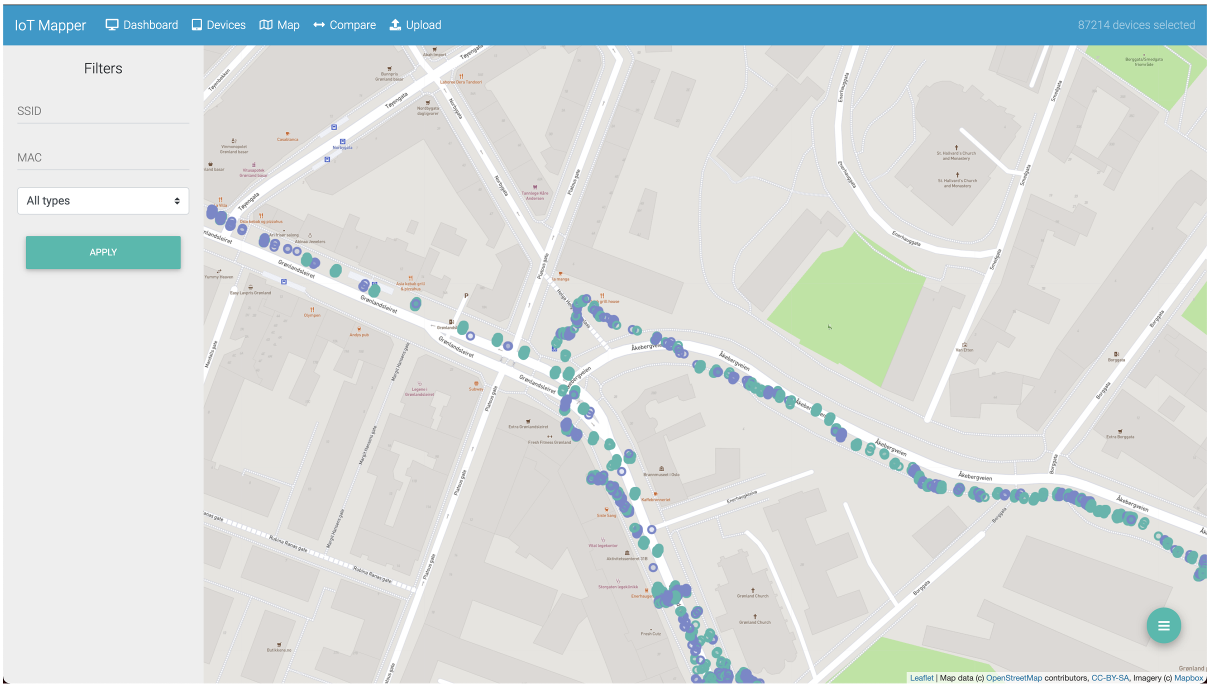


**Figure 5.3:** Map section

## 5.5 Datatable

The data table section of the web app consists of a single data table containing all the devices in the current selection. The table columns are based on the javascript objects in the "devices" key-value pair in the database.

## 5.6 Dashboard

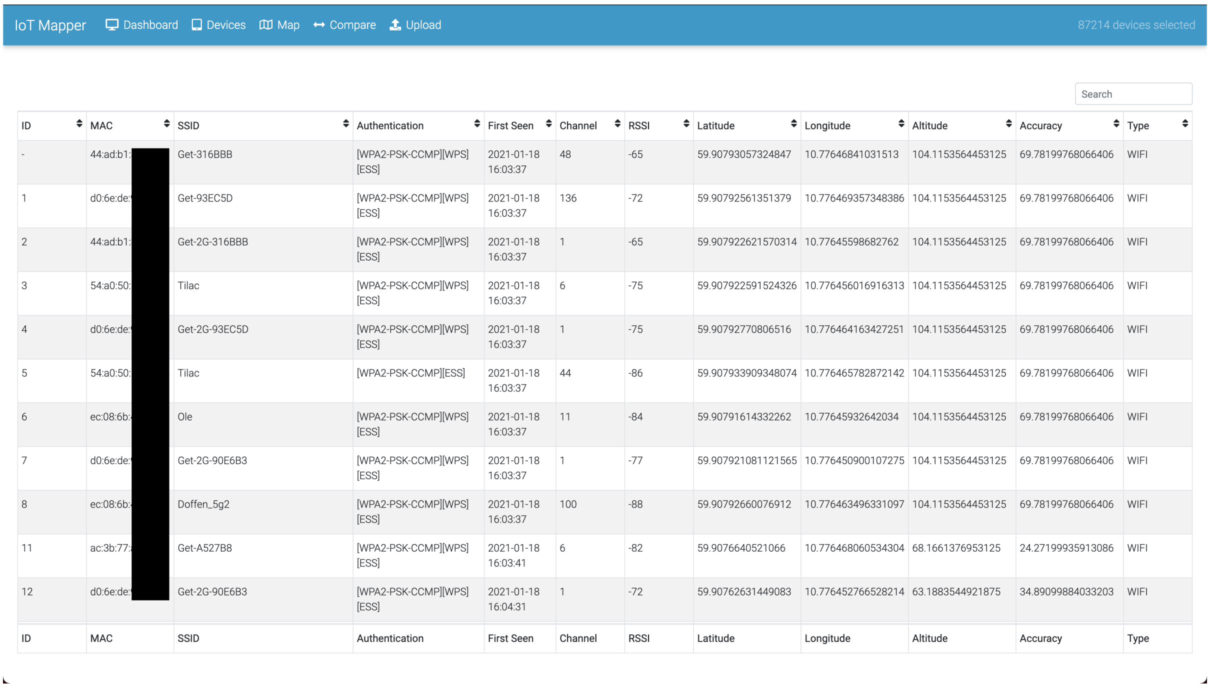
The dashboard section of the web app consists of several widgets aimed to give the user a general overview of the current selection and some options for manipulating it.

### 5.6.1 Device distribution

This widget shows the distribution of types of devices in the current selection as text.

### 5.6.2 Device distribution graph

This widget shows the distribution of types of devices in the current selection as a graph.



| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|---|
| - | 44:ad:b1:[redacted] | Get-316BBB | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 48 | -65 | 59.90793057324847 | 10.77646841031513 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 1 | d0:6e:de:[redacted] | Get-93EC5D | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 136 | -72 | 59.90792561351379 | 10.776469357348386 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 2 | 44:ad:b1:[redacted] | Get-2G-316BBB | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 1 | -65 | 59.907922621570314 | 10.77645598682762 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 3 | 54:a0:50:[redacted] | Tilac | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 6 | -75 | 59.907922591524326 | 10.776456016916313 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 4 | d0:6e:de:[redacted] | Get-2G-93EC5D | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 1 | -75 | 59.90792770806516 | 10.776464163427251 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 5 | 54:a0:50:[redacted] | Tilac | [WPA2-PSK-CCMP][ESS] | 2021-01-18 16:03:37 | 44 | -86 | 59.907933909348074 | 10.776465782872142 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 6 | ec:08:6b:[redacted] | Ole | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 11 | -84 | 59.90791614332262 | 10.77645932642034 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 7 | d0:6e:de:[redacted] | Get-2G-90E6B3 | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 1 | -77 | 59.907921081121565 | 10.776450900107275 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 8 | ec:08:6b:[redacted] | Doffen_5g2 | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:37 | 100 | -88 | 59.90792660076912 | 10.776463496331097 | 104.1153564453125 | 69.78199768066406 | WIFI |
| 11 | ac:3b:77:[redacted] | Get-A527B8 | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:03:41 | 6 | -82 | 59.9076640521066 | 10.776468060534304 | 68.1661376953125 | 24.27199935913086 | WIFI |
| 12 | d0:6e:de:[redacted] | Get-2G-90E6B3 | [WPA2-PSK-CCMP][WPS][ESS] | 2021-01-18 16:04:31 | 1 | -72 | 59.90762631449083 | 10.776452766528214 | 63.1883544921875 | 34.89099884033203 | WIFI |
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 5.4:** Datatable section

### 5.6.3 Get vendor info

This widget lets the user press a button to load the distribution of vendors and vendor countries of origin which is then displayed as bar charts.

### 5.6.4 TVs scanned

This widget shows a map of all the devices in the current selection with "TV" in its SSID/name on a map.

### 5.6.5 Filter selection

This widget lets the user filter the current selection based on SSID/name, MAC address, and type of device.

### 5.6.6 Find location of device

This widget lets the user find a more accurate location for a device in the current selection, based on trilateration, by entering its MAC address.

### 5.6.7 Find devices at address

This widget lets the user enter an address and returns a list of the devices in the current selection in close vicinity to the given address. Nominatim geocoder is used to find latitude and longitude from an address. The devices are then filtered based on the code found in Listing 5.3.

```
var tmpArr = [];
for(var i = 0; i < devices.length; i++) {
  const tmp = devices[i];
  const lat2 = parseFloat(tmp.currentLatitude);
  const lon2 = parseFloat(tmp.currentLongitude);
  const R = 6371e3;
  const phi1 = lat1 * Math.PI/180;
  const phi2 = lat2 * Math.PI/180;
  const deltaphi = (lat2-lat1) * Math.PI/180;
  const deltalambda = (lon2-lon1) * Math.PI/180;
  const a = Math.sin(deltaphi/2) * Math.sin(deltaphi/2) +
            Math.cos(phi1) * Math.cos(phi2) *
            Math.sin(deltalambda/2) * Math.sin(deltalambda/2);
  const c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
  const d = R * c;

  if(d < distance) {
    tmpArr.push(tmp);
  }
}
return tmpArr;
```

**Listing 5.3:** Finding devices within radius distance from point(lat1 lon1)

### 5.6.8 Find time device was spotted

This widget lets the user enter the device's MAC address in the current selection and returns the times it was spotted.

### 5.6.9 Filter distant duplicates

This widget lets the user enter a distance in meters and then filters the current selection to only include devices that appear at least twice and have at least the provided distance between themselves and other scans of the same device. This is shown in Listing 5.4.

```
filterDistantDuplicates(distance, devices) {
  function getNotUnique(array) {
    var map = new Map();
    array.forEach(a => map.set(a.mac, (map.get(a.mac) || 0) + 1));
    return array.filter(a => map.get(a.mac) > 1);
  }

  // Get all not unique devices and sort them by mac
  var tmp = getNotUnique(devices).sort((a,b) => (a.mac > b.mac) ? 1 : ((b.mac > a.mac) ? -1 :
      0));

  // Get only devices not close to each other
  let i = 0, j = 1;
  while(i < tmp.length) {
    let lat1 = parseFloat(tmp[i].currentLatitude);
    let lon1 = parseFloat(tmp[i].currentLongitude);
    let lat2 = parseFloat(tmp[j].currentLatitude);
    let lon2 = parseFloat(tmp[j].currentLongitude);

    // Distance
    const R = 6371e3; // metres
    const phi1 = lat1 * Math.PI/180; // phi, lambda in radians
    const phi2 = lat2 * Math.PI/180;
    const deltaphi = (lat2-lat1) * Math.PI/180;
    const deltalambda = (lon2-lon1) * Math.PI/180;
    const a = Math.sin(deltaphi/2) * Math.sin(deltaphi/2) +
              Math.cos(phi1) * Math.cos(phi2) *
              Math.sin(deltalambda/2) * Math.sin(deltalambda/2);
    const c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
    const d = R * c; // in metres


    // If distance less than 50 meters, remove from array, if not inc. j
    if(d < distance) {
      tmp.splice(j,1);
    } else {
      i += 2;
      j = i + 1;
    }

    // If done
    if(typeof tmp[j+1] === ’undefined’ && j - 1 === i) {
      tmp.splice(i, 1);
      break;
    }
    // If at last set of duplicates, but not done
    else if(typeof tmp[j+1] === ’undefined’) {
      i++;
      j = i+1;
    }
    // If comparing different MACs, go to next set of duplicates
    else if(tmp[i].mac !== tmp[j].mac && j - 1 === i){
      tmp.splice(i, 1);
    }
    // If last of duplicates and i = j - 1, skip to next set of duplicates
    else if(tmp[j+1].mac !== tmp[j].mac && j - 1 === i) {
      continue;
    }
    // If last of current duplicates, skip to next value of i
    else if(tmp[j+1].mac !== tmp[j].mac) {
      i++;
      j = i+1;
    }
    // If not, keep going
  }
  return tmp;
}
```

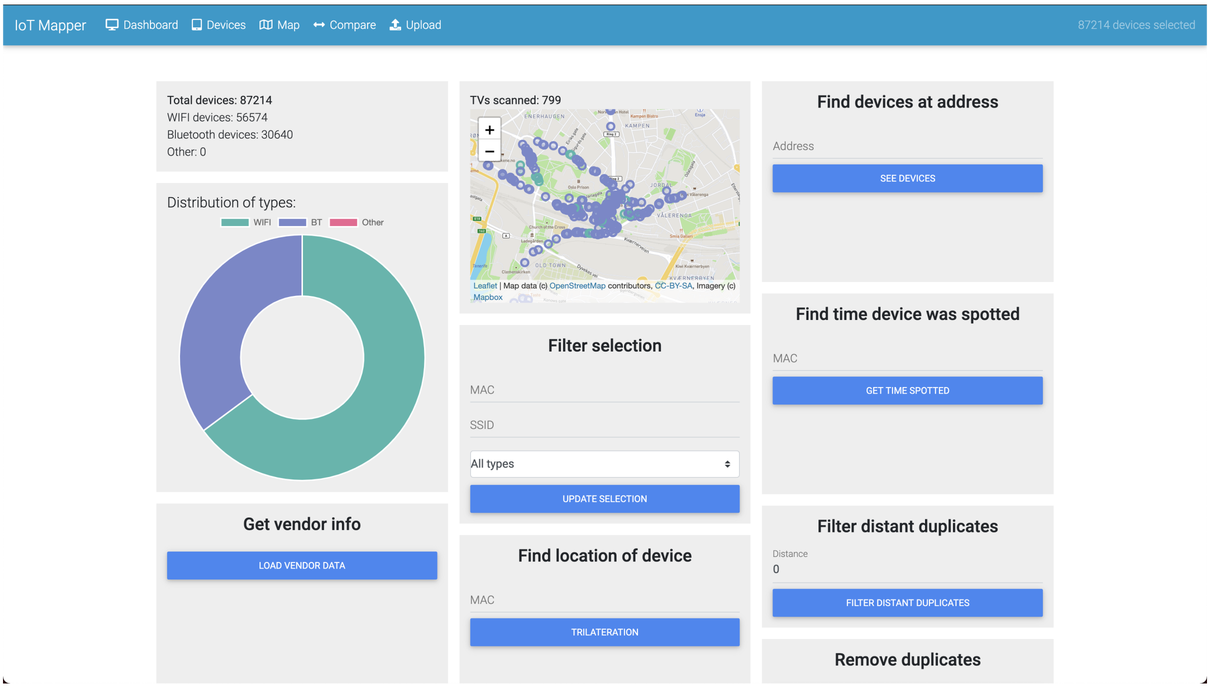
**Listing 5.4:** Finding distant duplicates in JavaScript

### 5.6.10 Remove duplicates

This widget lets the user remove all but one entry of the same MAC address for all devices in the current selection.

### 5.6.11 Remove access points

This widget lets the user remove all entries with the string "[WPA" in its AuthMode field. This removes most entries of access points.



Total devices: 87214
WIFI devices: 56574
Bluetooth devices: 30640
Other: 0

Distribution of types:

TVs scanned: 799

+
−
Leaflet | Map data (c) OpenStreetMap contributors, CC-BY-SA, Imagery (c) Mapbox

Find devices at address

Find time device was spotted

Filter selection

Get vendor info

Find location of device

Filter distant duplicates

Distance
0

Remove duplicates

**Figure 5.5:** Dashboard section

# Chapter 6

# Experiments

Project and CSV files are available at:

https://github.com/nicolaironning/Master

Files e1s2.csv and e2s2.csv are the same files. Files e3s1.csv and e2s1.csv are also the same files.

## 6.1 Experiment 1 - Detecting residents in homes

### 6.1.1 About

This experiment is based on use-case 4.3.1 from section 4.3. The aim of the experiment is to determine if it is possible for an adversary to detect if anyone is home in a given house at a given time by scanning for devices from right outside the house. In addition, the experiment is meant to demonstrate the extent to which the person scanning is able to detect what electronic devices are in the home.

If an adversary is able to accurately detect the presence of user electronics from outside houses, this information could be used to help the adversary efficiently determine which houses are potential targets for breaking into, as well as helping them decide what tools are required for breaking in. A house where mobile phones, mobile Bluetooth speakers, or other user electronics are present will likely have people present and will be a poor target for breaking into. A house where such devices are not present, but expensive TVs or other valuable electronics are will be a good candidate for breaking into. This would be not just a privacy risk but a risk for people's safety and valuables.

### 6.1.2 Description

**Step 1: Scanning**

The scanning process is to be performed twice. Once with known electronic devices in the house turned off, and once with all devices turned on. The devices should be spread out within the house. This is to determine how accurately it is possible to use wireless scanning to determine certain devices' presence. It should be done relatively close to the house in question. This is to emulate the best possible scenario for an adversary. Both scans should be done in a similar manner to limit other factors from influencing the results.

### Step 2: Using prototype

The CSV file from the second scan should first be uploaded to the prototype. The CSV file from the first scan should then be uploaded in the compare-section of the app. It will then be used to find the set difference between the second scan and the first scan. Which file is uploaded first matters, as this is the file that will be subtracted from. This is why the second scan is uploaded first. The list of devices after filtering based on the difference will then be analyzed manually.

### Step 3: Analysis

Scan 1, $S_1$, will contain all devices scanned while the known devices are switched off. Scan 2, $S_2$, will contain all devices scanned while the known devices are switched on. The devices present in $S_2$, but not in $S_1$, $(S_2 - S_1)$, marked in gray in Figure 6.1, will be the starting point for analysis. This is because in perfect conditions, with no other factors influencing the scanning, the only devices present in $S_2$ and not in $S_1$ will be the known devices. The degree to which this is the case in practice will indicate how accurate of a method this is.

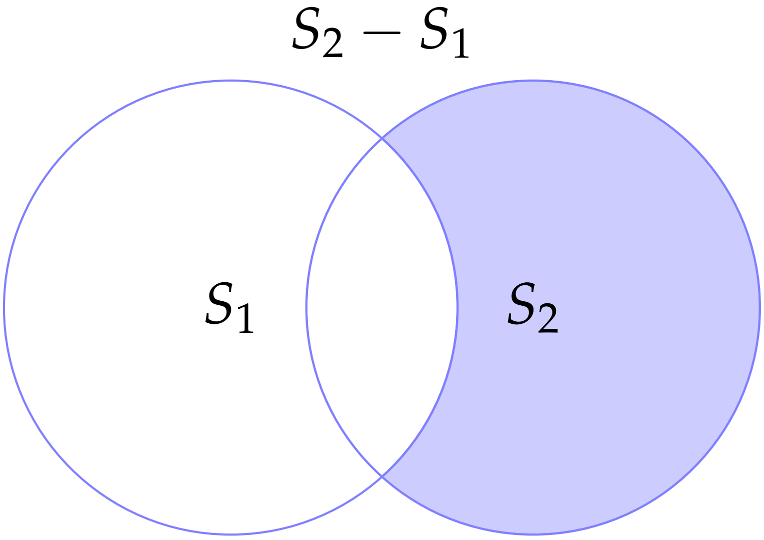


**Figure 6.1:** Experiment 6.1

The list should be compared to the list of known electronic devices in the house. The degree to which a potential intruder can detect devices in the house will be indicated by how similar the lists are. If a known device is not present in the list of scanned devices, this indicates that not all the devices in the house can be detected by a scanning phone from outside the house. If there are devices in the list of scanned devices that are not a known device, this indicates that there is noise in the scan, making it harder to determine the presence of people accurately. The analysis part of the experiment will be presented in Chapter 7.

### 6.1.3 Method

The known devices for this experiment are listed in Table 6.1.

| SSID/Name | MAC-address | Type of device |
|---|---|---|
| Galaxy S9 | f4:7d:ef:f1:ff:7e | Android smartphone |
| JBL Flip 5 | b8:f6:53:ac:09:a8 | Bluetooth speaker |
| G7-9F2CBB | fc:db:b3:9f:2c:bb | Wi-Fi enabled camera |
| LE-Bose NC 700 HP | 4c:87:5d:9e:45:be | Bluetooth headphones |

**Table 6.1:** Known devices for experiment 1

#### Scanning

The scanning was performed on a house in a suburban area. The devices were placed in different rooms, all on the ground floor of the house. The Android smartphone was put into Bluetooth pairing mode for the duration of the second scan, as this was the only way it could be picked up by Wigle.

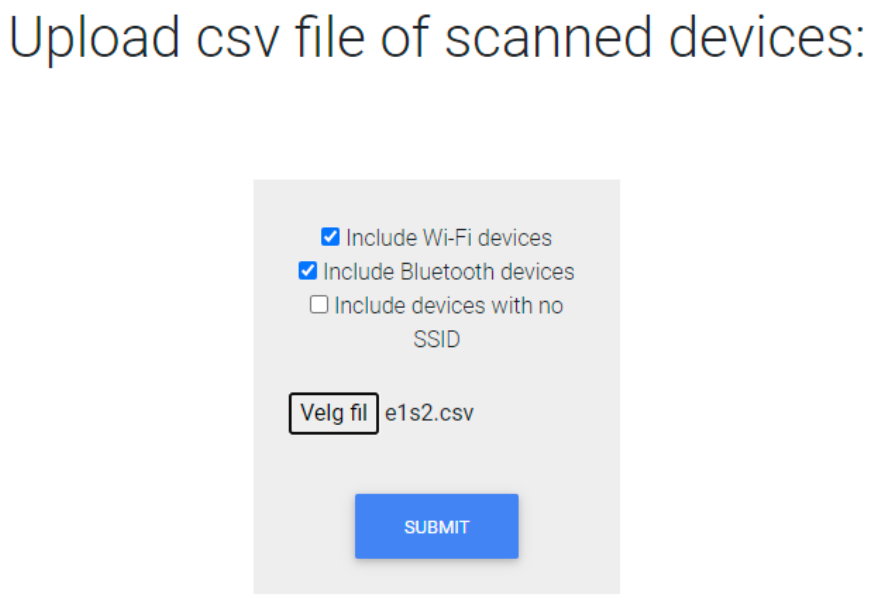
***Scan 1:*** The known devices were all switched off. The person scanning was standing 2 meters from the house wall when beginning to scan. The person then walked around the house twice in a row, at a normal walking pace, keeping approximately two meters from the outer wall at all times. The scanning was stopped once the person scanning reached the starting point after completing two laps.

***Scan 2:*** The known devices were all switched on. The person scanning then conducted a scan in the same way as in scan 1, making sure to follow a similar pace and distance from the outer wall.

Both scans were saved as individual CSV files, containing only the devices from the respective scan.

#### Using prototype

The CSV file of $S_2$ was uploaded to the prototype to make this the current selection. Only devices with an SSID/name will be included. See Figure 6.2. This is because all the known devices have SSIDs/names, and devices without SSIDs/names will therefore not be useful. The compare section of the prototype will then be used to upload the $S_1$, only including devices with an SSID/name, then find the difference between the two scans. See Figure 6.3 and Figure 6.4. The devices found in one scan, but not the other, will then be available in the data table section of the prototype, see Figure 6.5. This list will be analyzed in Chapter 7.



Upload csv file of scanned devices:

☑ Include Wi-Fi devices
☑ Include Bluetooth devices
☐ Include devices with no SSID

Velg fil e1s2.csv

**Figure 6.2:** Uploading $S_2$

## 6.2 Experiment 2 - Detecting visitors in homes

### 6.2.1 About

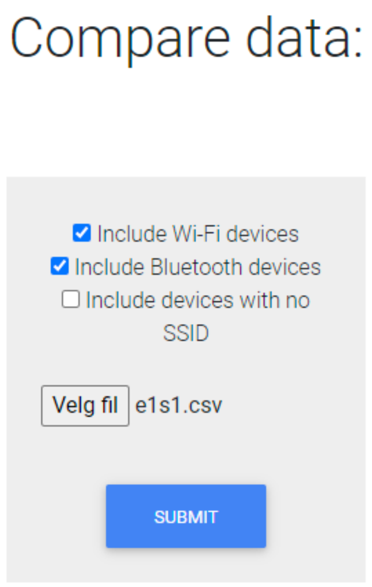
The experiment is based on use-case 4.3.2 from section 4.3. The aim of the experiment is to determine if it is possible for an adversary to detect if there are visitors present in a given house at a given time by scanning for devices from right outside the house. The experiment is also meant to demonstrate the extent to which the person scanning is able to detect the difference in devices in the house compared to a baseline.

If an adversary is able to accurately detect user electronics that are not usually present in houses from outside those houses, this information could be used to help the adversary determine if there are visitors present in houses. This requires that the adversary has scanned the houses before and knows what devices are usually present in the houses. This information could be used for someone to detect relationships between specific individuals. If the adversary is able to cross-reference the visiting devices with other houses, they could potentially find relationships between people. This would be a breach of privacy.

### 6.2.2 Description

#### Step 1: Scanning

The scanning process is to be performed twice. Once with only a set of known electronic devices turned on inside the house, and once with a larger superset of those devices turned on. The devices should be spread out within the house. This is to determine how accurately it is possible to use wireless scanning for determining the presence of devices not usually present in the house. It should be done relatively



Compare data:

☑ Include Wi-Fi devices

☑ Include Bluetooth devices

☐ Include devices with no SSID

Velg fil e1s1.csv

**Figure 6.3:** Comparing $S_1$

close to the house in question. This is to emulate the best possible scenario for someone scanning.

### Step 2: Using prototype

The CSV file from the second scan should first be uploaded to the prototype. The CSV file from the first scan should then be uploaded in the compare-section of the app. It will then be used to find the set difference between the second scan and the first scan. Which file is uploaded first matters, as this is the file that will be subtracted from. This is why the second scan is uploaded first. The list of devices after filtering based on the symmetric difference will then be analyzed manually.

### Step 3: Analysis

Scan 2, $S_2$, will contain all devices scanned while the superset of known devices was switched on. Scan 1, $S_1$, will contain the devices scanned while only the baseline known devices were switched on. The devices present in $S_2$, but not in $S_1$, ($S_2 - S_1$), marked in gray in Figure 6.6, will be the starting point for analysis. This is because in perfect conditions, with no other factors influencing the scanning, the only devices present in $S_2$ and not in $S_1$ will be the devices present in the superset of devices, but not in the baseline set of devices. The degree to which this is the case in practice will indicate how accurate of a method this is.

The list of devices in $S_2$, but not in $S_1$, should be compared to the list of devices present in the superset of known devices but not in the baseline set of devices. The degree to which an adversary is able to detect visitors in a house will be indicated by how similar the lists are.

If a device that was turned on for the second scan is not present in the list of scanned devices, this indicates that not all the devices in the house can be detected by a scanning phone from outside the

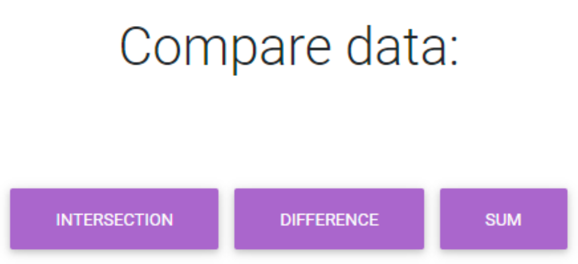
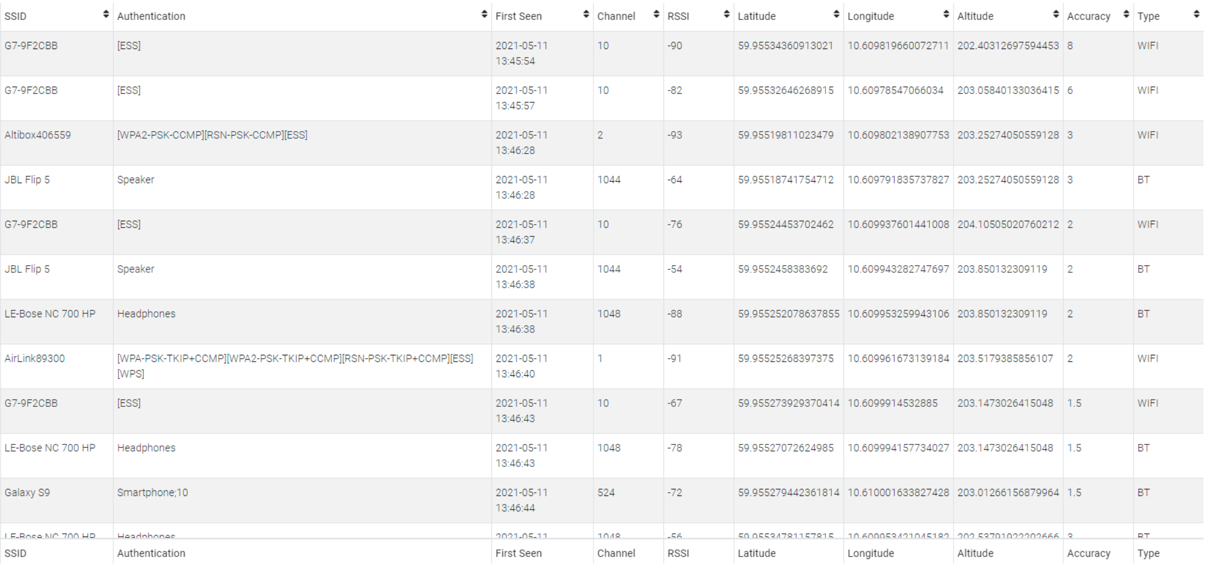


**Figure 6.4:** Comparing $S_1$, pt. 2



| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|
| G7-9F2CBB | [ESS] | 2021-05-11 13:45:54 | 10 | -90 | 59.95534360913021 | 10.609819660072711 | 202.40312697594453 | 8 | WIFI |
| G7-9F2CBB | [ESS] | 2021-05-11 13:45:57 | 10 | -82 | 59.95532646268915 | 10.60978547066034 | 203.05840133036415 | 6 | WIFI |
| Altibox406559 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:46:28 | 2 | -93 | 59.95519811023479 | 10.609802138907753 | 203.25274050559128 | 3 | WIFI |
| JBL Flip 5 | Speaker | 2021-05-11 13:46:28 | 1044 | -64 | 59.95518741754712 | 10.609791835737827 | 203.25274050559128 | 3 | BT |
| G7-9F2CBB | [ESS] | 2021-05-11 13:46:37 | 10 | -76 | 59.95524453702462 | 10.609937601441008 | 204.10505020760212 | 2 | WIFI |
| JBL Flip 5 | Speaker | 2021-05-11 13:46:38 | 1044 | -54 | 59.9552458383692 | 10.609943282747697 | 203.850132309119 | 2 | BT |
| LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:38 | 1048 | -88 | 59.955252078637855 | 10.609953259943106 | 203.850132309119 | 2 | BT |
| AirLink89300 | [WPA-PSK-TKIP+CCMP][WPA2-PSK-TKIP+CCMP][RSN-PSK-TKIP+CCMP][ESS] [WPS] | 2021-05-11 13:46:40 | 1 | -91 | 59.95525268397375 | 10.609961673139184 | 203.5179385856107 | 2 | WIFI |
| G7-9F2CBB | [ESS] | 2021-05-11 13:46:43 | 10 | -67 | 59.955273929370414 | 10.6099914532885 | 203.1473026415048 | 1.5 | WIFI |
| LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:43 | 1048 | -78 | 59.95527072624985 | 10.609994157734027 | 203.1473026415048 | 1.5 | BT |
| Galaxy S9 | Smartphone;10 | 2021-05-11 13:46:44 | 524 | -72 | 59.955279442361814 | 10.610001633827428 | 203.01266156879964 | 1.5 | BT |
| LE-Bose NC 700 HP | Headphones | 2021-05-11 | 1048 | -56 | 59.95534781157815 | 10.609953421045182 | 202.53791022202666 | 3 | BT |
| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.5:** Experiment 1 results

house. If there are devices in the list of scanned devices that are not a device that was turned on for the second scan, this indicates that there is noise in the scan, which could make it harder to determine the presence of people accurately.

### 6.2.3 Method

The devices turned on for the first scan are listed in Table 6.2.

The devices turned on for the second scan are listed in Table 6.3.

#### Scanning

The scanning was performed on a house in a suburban area. The devices were placed in different rooms, all on the ground floor of the house.

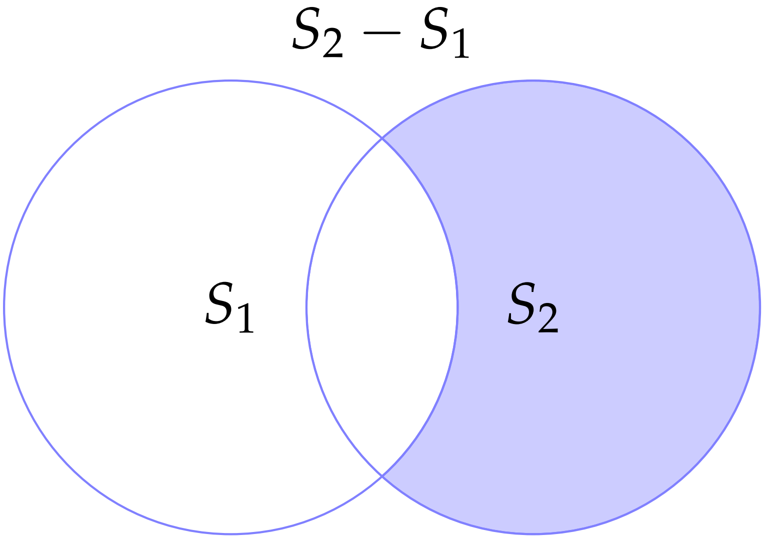


**Figure 6.6:** Experiment 6.2

| SSID/Name | MAC-address | Type of device |
|---|---|---|
| G7-9F2CBB | fc:db:b3:9f:2c:bb | Wi-Fi enabled camera |
| LE-Bose NC 700 HP | 4c:87:5d:9e:45:be | Bluetooth headphones |
| JBL Flip 5 | b8:f6:53:ac:09:a8 | Bluetooth speaker |

**Table 6.2:** Devices turned on for both scans

***Scan 1:*** The known devices in Table 6.3 were switched off. The person scanning was standing 2 meters from the house wall when beginning to scan. The person then walked around the house twice in a row, at a normal walking pace, keeping approximately two meters from the outer wall at all times. The scanning was stopped once the person scanning reached the starting point after completing two laps.

***Scan 2:*** The known devices from both Table 6.2 and 6.3 were all switched on. The person scanning then conducted a scan in the same way as in scan 1, making sure to follow a similar pace and distance from the outer wall.

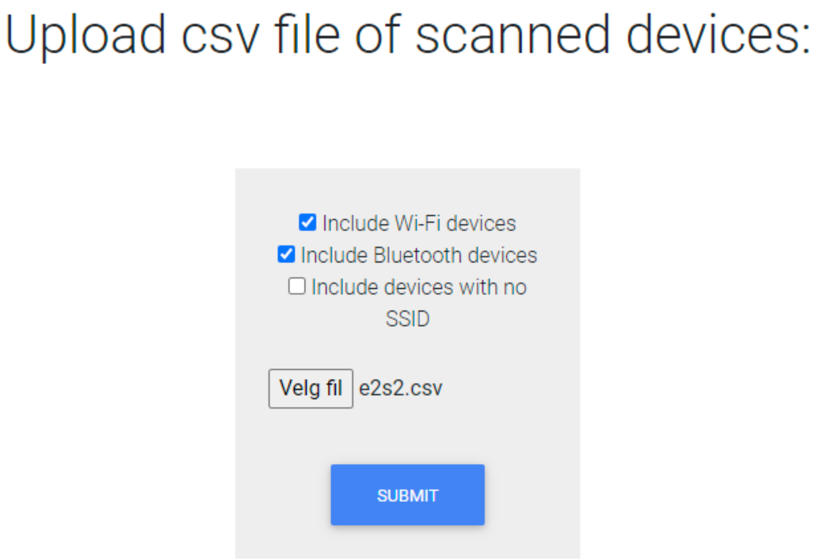
Both scans were saved as individual CSV files, containing only the devices from the respective scan.

### Using prototype

The CSV file from the second scan should first be uploaded to the prototype. The CSV file from the first scan should then be uploaded in the compare-section of the app. It will then be used to find the set difference between the second scan and the first scan. The list of devices after filtering based on the symmetric difference will then be analyzed manually. The devices found in one scan, but not the other, will then be available in the data table section of the prototype, see Figure 6.10. This list will be analyzed in chapter 7.

| SSID/Name | MAC-address | Type of device |
|---|---|---|
| iZound X-50 | a0:e9:db:51:16:f9 | Bluetooth speaker |

**Table 6.3:** Devices turned on only for second scan



Upload csv file of scanned devices:

☑ Include Wi-Fi devices

☑ Include Bluetooth devices

☐ Include devices with no SSID

Velg fil e2s2.csv

**Figure 6.7:** Uploading $S_2$

## 6.3 Experiment 3 - Adding context from location

### 6.3.1 About

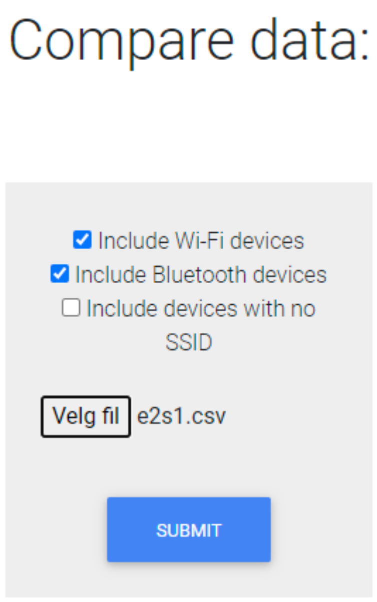
This experiment is based on use-case 4.3.3 from section 4.3. The aim of the experiment is to determine if it is possible for an adversary to detect the presence of a person at a location that can give context as to what the person is doing there. This information could reveal private or sensitive data about the person.

If an adversary is able to detect the presence of user electronics at a house and then detect the same device or devices at a location that can give context as to what the person is doing there, this could give the adversary sensitive information about the person from the context. This requires that the adversary has a large dataset of devices in an area and the ability to scan in a given location for a long duration. Any device that appears both in the scan at the location and the large dataset is potentially a privacy risk.

### 6.3.2 Description

**Step 1: Scanning**

The scanning process is to be performed twice. Once at a house with a known set of devices turned on, and once at a location that gives context to what someone might be doing there while one of the known



Compare data:

☑ Include Wi-Fi devices

☑ Include Bluetooth devices

☐ Include devices with no SSID

e2s1.csv

**Figure 6.8:** Comparing $S_1$

devices is present.

### Step 2: Using prototype

The CSV file from one of the scans should be uploaded to the prototype. The CSV file from the other scan should then be uploaded in the compare-section of the app. It will then be used to find the intersection between the scans. Which file is uploaded first does not matter as the operation is symmetric. The list of devices after filtering based on the intersection will then be analyzed manually.

### Step 3: Analysis

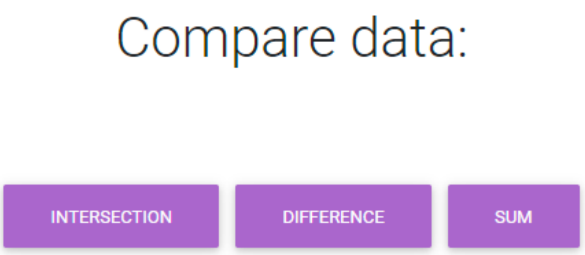
Scan 1, $S_1$, will contain all devices scanned at the house with the known devices switched on. Scan 2, $S_2$, will contain all devices scanned while scanning at the context-giving location. The devices present in both $S_1$ and $S_2$ ($S_1 \cap S_2$), marked in blue in Figure 6.11, will be the starting point for analysis.

## 6.3.3 Method

The known device for this experiment is listed in Table 6.4:

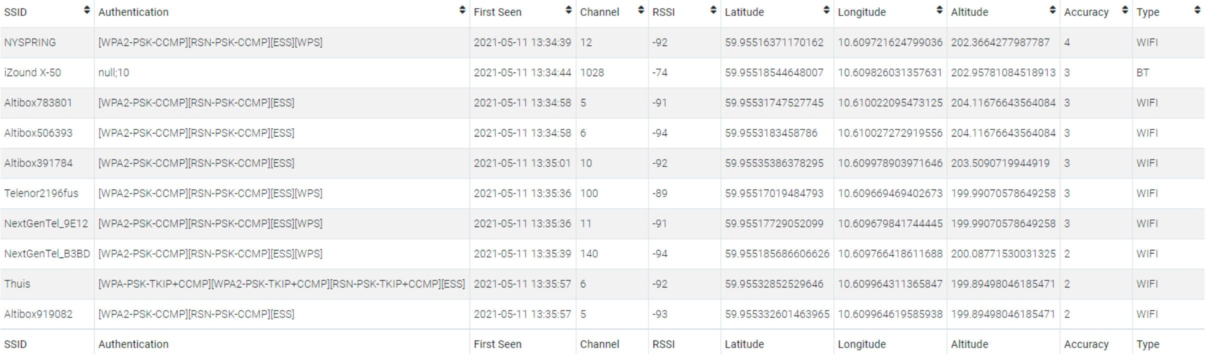
| SSID/Name | MAC-address | Type of device |
|---|---|---|
| LE-Bose NC 700 HP | 4c:87:5d:9e:45:be | Bluetooth headphones |

**Table 6.4:** Device present in both scans



Compare data:

**Figure 6.9:** Comparing $S_1$, pt. 2



| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|
| NYSPRING | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:34:39 | 12 | -92 | 59.95516371170162 | 10.609721624799036 | 202.3664277987787 | 4 | WIFI |
| iZound X-50 | null;10 | 2021-05-11 13:34:44 | 1028 | -74 | 59.95518544648007 | 10.609826031357631 | 202.95781084518913 | 3 | BT |
| Altibox783801 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:34:58 | 5 | -91 | 59.95531747527745 | 10.610022095473125 | 204.11676643564084 | 3 | WIFI |
| Altibox506393 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:34:58 | 6 | -94 | 59.9553183458786 | 10.610027272919556 | 204.11676643564084 | 3 | WIFI |
| Altibox391784 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:01 | 10 | -92 | 59.95535386378295 | 10.609978903971646 | 203.5090719944919 | 3 | WIFI |
| Telenor2196fus | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:36 | 100 | -89 | 59.95517019484793 | 10.609669469402673 | 199.99070578649258 | 3 | WIFI |
| NextGenTel_9E12 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:36 | 11 | -91 | 59.95517729052099 | 10.609679841744445 | 199.99070578649258 | 3 | WIFI |
| NextGenTel_B3BD | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:39 | 140 | -94 | 59.955185686606626 | 10.609766418611688 | 200.08771530031325 | 2 | WIFI |
| Thuis | [WPA-PSK-TKIP+CCMP][WPA2-PSK-TKIP+CCMP][RSN-PSK-TKIP+CCMP][ESS] | 2021-05-11 13:35:57 | 6 | -92 | 59.95532852529646 | 10.609964311365847 | 199.89498046185471 | 2 | WIFI |
| Altibox919082 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:57 | 5 | -93 | 59.955332601463965 | 10.609964619585938 | 199.89498046185471 | 2 | WIFI |
| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.10:** Experiment 2 results

### Scanning

The scanning was performed on a house in a suburban area and at "Sex & Samfunn", a medical clinic specializing in sexually related medical problems for people under 26 years old. At the house, the devices were placed in different rooms, all on the ground floor of the house.

***Scan 1:*** The known devices were all switched on. The person scanning was standing 2 meters from the house wall when beginning to scan. The person then walked around the house twice in a row, at a normal walking pace, keeping approximately two meters from the outer wall at all times. The scanning was stopped once the person scanning reached the starting point after completing two laps.

***Scan 2:*** The person scanning was standing next to the entrance of the clinic. Another person was wearing the Bluetooth headphones. The person wearing the headphones walked past the person scanning to the entrance of the clinic. The person wearing the headphones then walked away.

Both scans were saved as individual CSV files, containing only the devices from the respective scan.

### Using prototype

The CSV file of $S_1$ was uploaded to the prototype to make this the current selection. Only devices with an SSID/name were included. See Figure 6.12. This is because all the known devices have SSIDs/names,

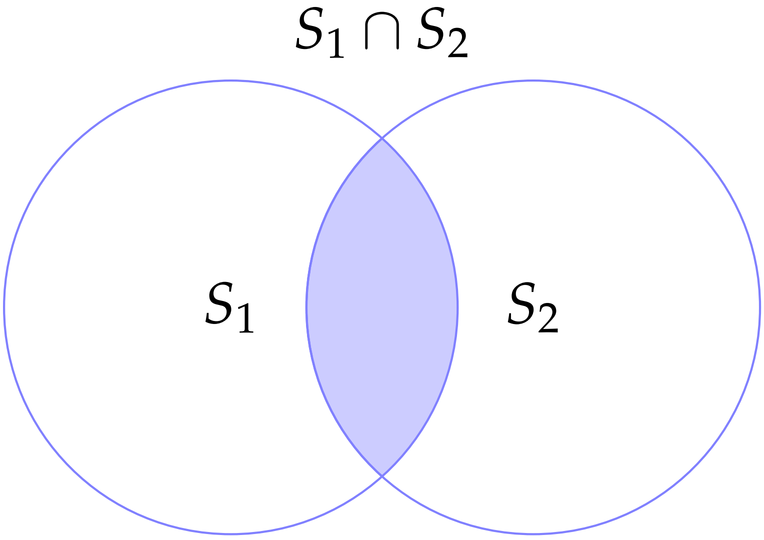
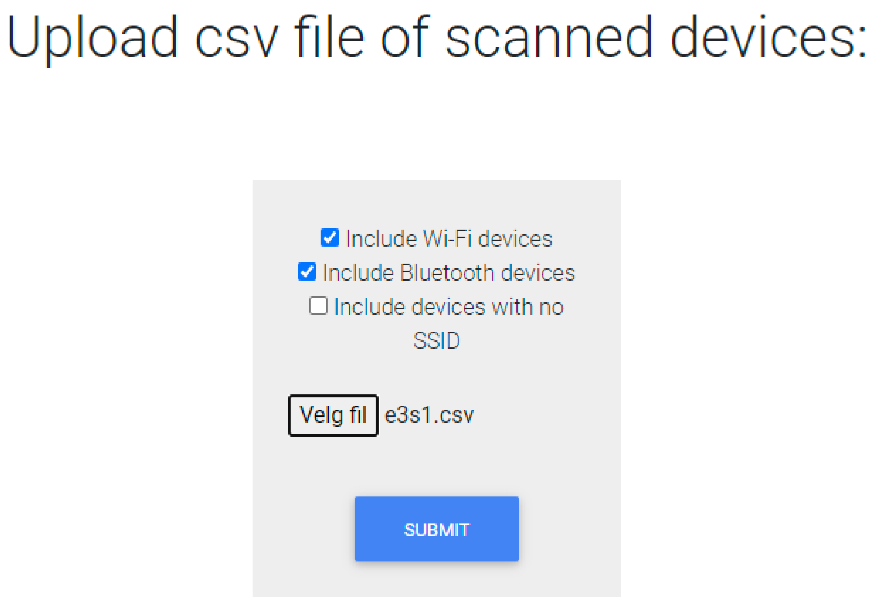


**Figure 6.11:** Experiment 6.3

and devices without SSIDs/names will therefore not be useful. The compare section of the prototype will then be used to upload $S_2$, only including devices with an SSID/name, then find the intersection of the two scans, see Figure 6.13 and Figure 6.14. Any devices present in both scans will then be available in the data table section of the prototype, see Figure 6.15. This list will be analyzed in chapter 7.



Upload csv file of scanned devices:

☑ Include Wi-Fi devices
☑ Include Bluetooth devices
☐ Include devices with no SSID

Velg fil e3s1.csv

**Figure 6.12:** Uploading $S_1$

## 6.4 Experiment 4 - Detecting frequent locations and frequently taken paths

### 6.4.1 About

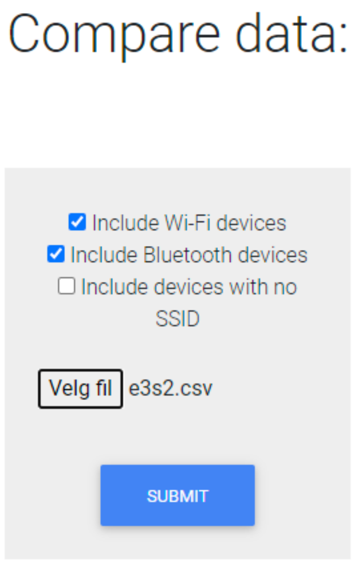
This experiment is based on use-case 4.3.4 from section 4.3. The aim of the experiment is to look at a more long-term scenario and determine what kind of data is possible to extract from scanning in multiple locations over a longer period of time. This experiment will differ from the other experiments in that it will not be controlled. The experiment will entail looking at a large set of data and using the prototype to attempt to extract potentially privacy-sensitive information.

One thing to look for in the dataset is if specific devices appear more than once. Especially if they appear in different locations. This is interesting because it confirms if devices are mobile and can filter out uninteresting devices. The devices that appear in multiple locations might say something about paths taken by individuals or patterns in their day-to-day life. Should an adversary obtain this information, it would be a significant privacy risk.

### 6.4.2 Description

**Scanning**

The scanning process for this experiment should cover a large area over a long period of time. This is to obtain a dataset that allows for picking up the same devices at different times in different locations. This will be required for detecting paths taken and patterns in people's daily lives. The scan should then be exported to a CSV file.



Compare data:

☑ Include Wi-Fi devices
☑ Include Bluetooth devices
□ Include devices with no SSID

e3s2.csv

**Figure 6.13:** Comparing $S_2$

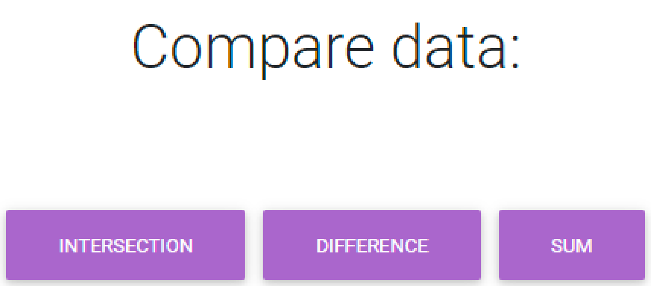
### Using Prototype

The CSV file of the large long-time scan should be uploaded to the prototype. The Wigle app captures the same device periodically, so there is likely to be a large number of devices that appear several times in the dataset. A significant portion of these are likely scanned within a short time frame and in the same area, meaning they do not give any information about how a person moves or patterns in their lives. To exclude these, the prototype will filter out all unique entries, calculate the distance between duplicates and only keep them in the selection if the distance between them is greater than a given value. The remaining entries should then be devices that were scanned at different locations at different times.

Different distances should be used for filtering out devices to see which distances give the most interesting results. The data table and map sections of the prototype will be used to find devices of interest and paths taken or patterns. The distant duplicates, $C$, will be a subset of all duplicates, $B$, which again is a subset of all the entries in the CSV file, $A$, see Figure 6.16. The number of elements in set $A$ and set $B$ will be constant, while the number of elements in set $C$ will be dependent on the distance given to the prototype when finding distant duplicates. A lower distance will mean more devices that are potentially interesting for finding paths taken and patterns while including more instances of a device being scanned multiple times on the same occasion and location. Finding a good middle ground for this will therefore be important before analyzing the data.

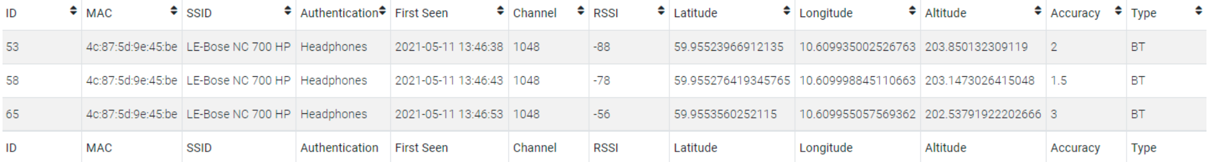
### Analysis

The analysis for this experiment will entail manually looking through the devices in set $C$ and see what information can be gained.



Compare data:

**Figure 6.14:** Comparing $S_2$, pt. 2



| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 53 | 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:38 | 1048 | -88 | 59.95523966912135 | 10.609935002526763 | 203.850132309119 | 2 | BT |
| 58 | 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:43 | 1048 | -78 | 59.955276419345765 | 10.609998845110663 | 203.1473026415048 | 1.5 | BT |
| 65 | 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:53 | 1048 | -56 | 59.9553560252115 | 10.609955057569362 | 202.53791922202666 | 3 | BT |
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.15:** Experiment 3 results

### 6.4.3 Method

#### Step 1: Scanning

The data to be used for this experiment was gathered over a longer period of time in Oslo by Lothar Fritsch, supervisor for this thesis. The scanning was done between the dates January 17th and February 27th. The set contains 87214 entries.

#### Step 2: Using prototype

The CSV file is added to the selection in the upload section of the web app. The number of entries is 87214. Opening the map section of the app reveals all the entries on a map, shown in Figure 6.17. Using the filter distant duplicates widget in the dashboard section requires entering a distance. The distances: 100, 200, 400, and 800 meters are used. For each of the distances, the resulting selection is inspected in both the data table section and the map section of the prototype.

***100m:***
Filtering with a distance of 100m reduced the size of the current selection to 15904 entries. Inspecting the devices in the data table section shows that many of the devices were scanned within a short span of time, see Figure 6.19, meaning they were probably scanned multiple times at the same occasion and location. Inspecting further in the map section confirms this.

***200m:***
Filtering with a distance of 100m reduced the size of the current selection to 3998. Inspecting the devices in the data table section shows that several entries are still within a short period of time, although

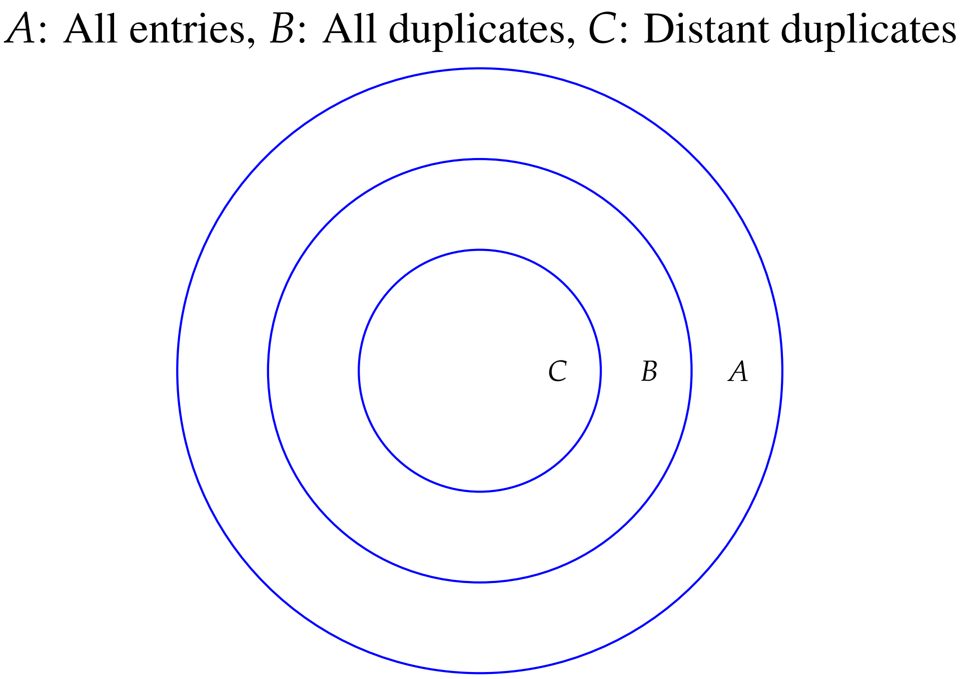


**Figure 6.16:** Experiment 6.4

significantly fewer. See Figure 6.20.

***400m:***
Filtering with a distance of 100m reduced the size of the current selection to 896. Inspecting the devices in the data table section shows that the number of duplicates scanned within a short period was significantly reduced, however still present, see Figure 6.21.

***800m:***
Filtering with a distance of 100m reduced the size of the current selection to 276. Inspecting the devices in the data table section shows that nearly all duplicates were captured on separate occasions. See Figure 6.22. Inspecting the map section of the prototype confirms this.

From looking at the current selection after filtering based on the different distances, it is decided to use 800m as the distance for analysis.

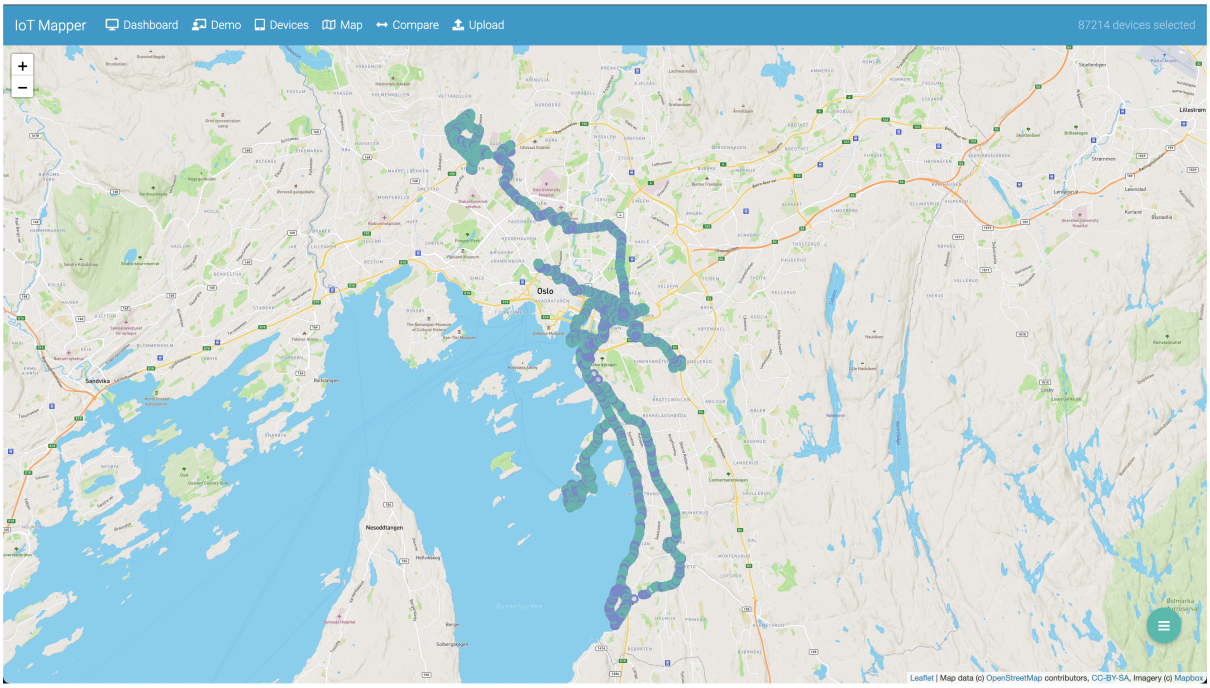
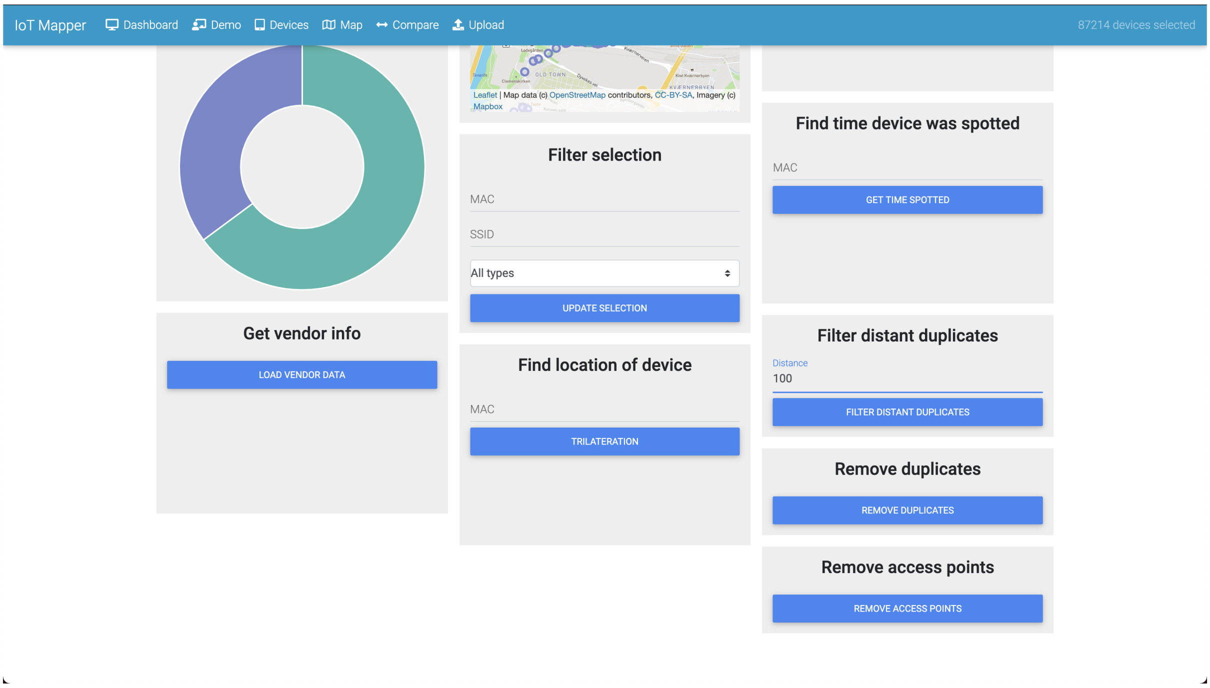


**Figure 6.17:** Experiment 4 map view


IoT Mapper
Dashboard
Demo
Devices
Map
Compare
Upload
87214 devices selected
Leaflet | Map data (c) OpenStreetMap contributors, CC-BY-SA, Imagery (c) Mapbox

Get vendor info

LOAD VENDOR DATA

Filter selection

MAC

SSID

All types

UPDATE SELECTION

Find location of device

MAC

TRILATERATION

Find time device was spotted

MAC

GET TIME SPOTTED

Filter distant duplicates

Distance
100

FILTER DISTANT DUPLICATES

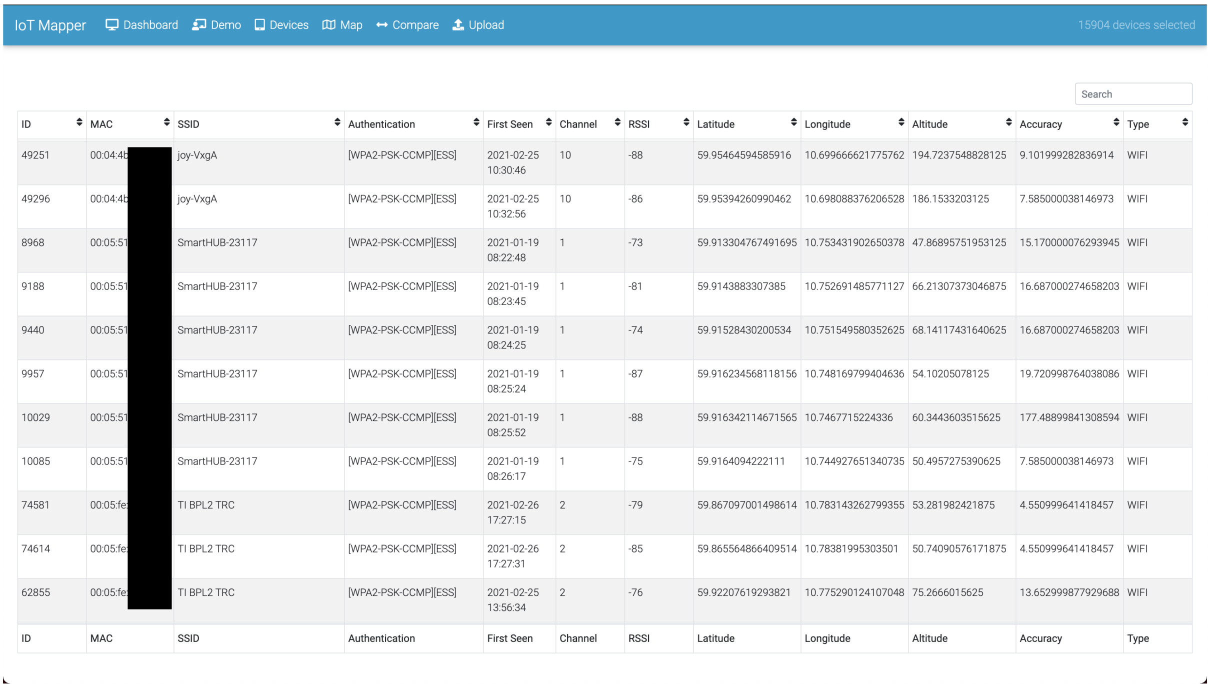
Remove duplicates

REMOVE DUPLICATES

Remove access points

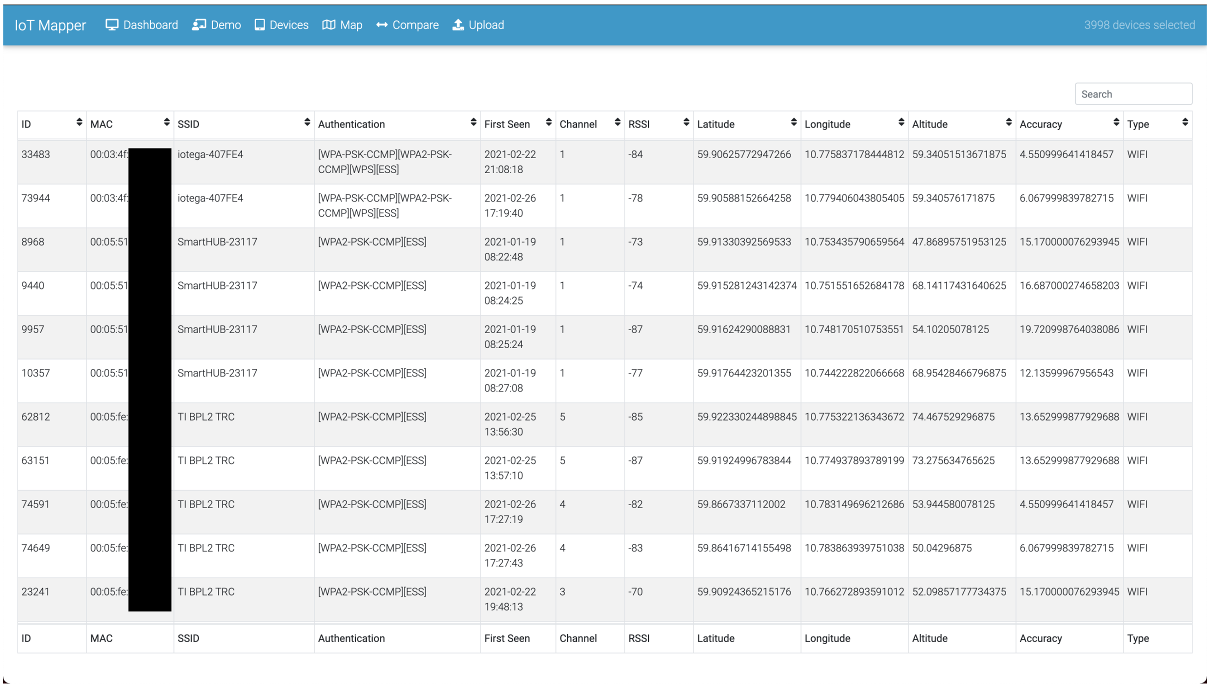
REMOVE ACCESS POINTS

**Figure 6.18:** Experiment 4 filter distant duplicates



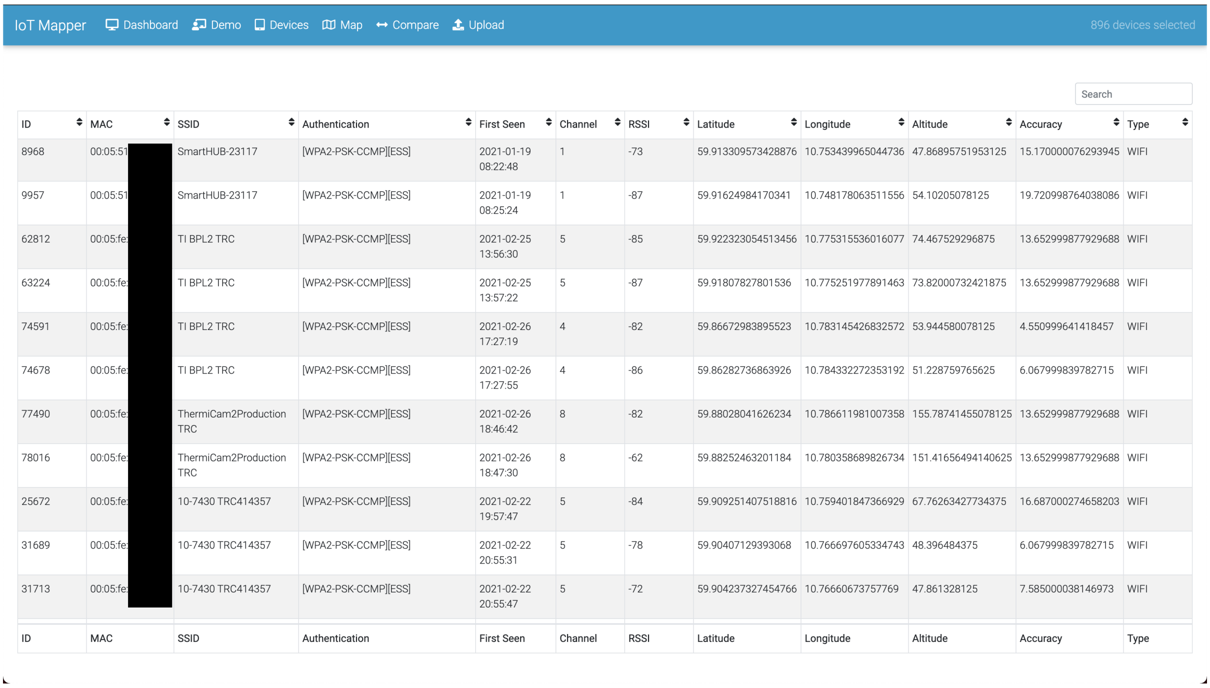
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 49251 | 00:04:4b[redacted] | joy-VxgA | [WPA2-PSK-CCMP][ESS] | 2021-02-25 10:30:46 | 10 | -88 | 59.95464594585916 | 10.699666621775762 | 194.7237548828125 | 9.101999282836914 | WIFI |
| 49296 | 00:04:4b[redacted] | joy-VxgA | [WPA2-PSK-CCMP][ESS] | 2021-02-25 10:32:56 | 10 | -86 | 59.95394260990462 | 10.698088376206528 | 186.1533203125 | 7.585000038146973 | WIFI |
| 8968 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:22:48 | 1 | -73 | 59.913304767491695 | 10.753431902650378 | 47.86895751953125 | 15.170000076293945 | WIFI |
| 9188 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:23:45 | 1 | -81 | 59.9143883307385 | 10.752691485771127 | 66.21307373046875 | 16.687000274658203 | WIFI |
| 9440 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:24:25 | 1 | -74 | 59.91528430200534 | 10.751549580352625 | 68.14117431640625 | 16.687000274658203 | WIFI |
| 9957 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:25:24 | 1 | -87 | 59.916234568118156 | 10.748169799404636 | 54.10205078125 | 19.720998764038086 | WIFI |
| 10029 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:25:52 | 1 | -88 | 59.916342114671565 | 10.7467715224336 | 60.3443603515625 | 177.48899841308594 | WIFI |
| 10085 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:26:17 | 1 | -75 | 59.9164094222111 | 10.744927651340735 | 50.4957275390625 | 7.585000038146973 | WIFI |
| 74581 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:27:15 | 2 | -79 | 59.867097001498614 | 10.783143262799355 | 53.281982421875 | 4.550999641418457 | WIFI |
| 74614 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:27:31 | 2 | -85 | 59.865564866409514 | 10.78381995303501 | 50.74090576171875 | 4.550999641418457 | WIFI |
| 62855 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-25 13:56:34 | 2 | -76 | 59.92207619293821 | 10.775290124107048 | 75.2666015625 | 13.652999877929688 | WIFI |
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.19:** Experiment 4 after filtering with a distance of 100m



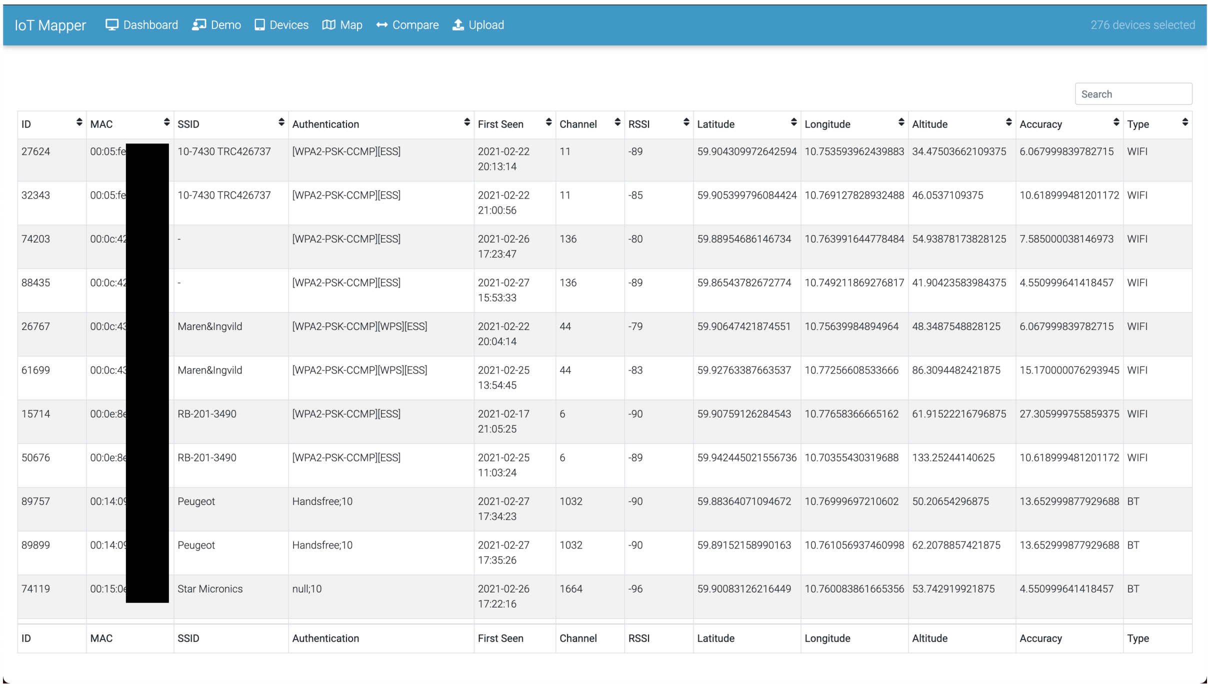
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 33483 | 00:03:4f:[redacted] | iotega-407FE4 | [WPA-PSK-CCMP][WPA2-PSK-CCMP][WPS][ESS] | 2021-02-22 21:08:18 | 1 | -84 | 59.90625772947266 | 10.775837178444812 | 59.34051513671875 | 4.550999641418457 | WIFI |
| 73944 | 00:03:4f:[redacted] | iotega-407FE4 | [WPA-PSK-CCMP][WPA2-PSK-CCMP][WPS][ESS] | 2021-02-26 17:19:40 | 1 | -78 | 59.90588152664258 | 10.779406043805405 | 59.340576171875 | 6.067999839782715 | WIFI |
| 8968 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:22:48 | 1 | -73 | 59.91330392569533 | 10.753435790659564 | 47.86895751953125 | 15.170000076293945 | WIFI |
| 9440 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:24:25 | 1 | -74 | 59.915281243142374 | 10.751551652684178 | 68.14117431640625 | 16.687000274658203 | WIFI |
| 9957 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:25:24 | 1 | -87 | 59.91624290088831 | 10.748170510753551 | 54.10205078125 | 19.720998764038086 | WIFI |
| 10357 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:27:08 | 1 | -77 | 59.91764423201355 | 10.744222822066668 | 68.95428466796875 | 12.13599967956543 | WIFI |
| 62812 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-25 13:56:30 | 5 | -85 | 59.922330244898845 | 10.775322136343672 | 74.467529296875 | 13.652999877929688 | WIFI |
| 63151 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-25 13:57:10 | 5 | -87 | 59.91924996783844 | 10.774937893789199 | 73.275634765625 | 13.652999877929688 | WIFI |
| 74591 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:27:19 | 4 | -82 | 59.8667337112002 | 10.783149696212686 | 53.944580078125 | 4.550999641418457 | WIFI |
| 74649 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:27:43 | 4 | -83 | 59.86416714155498 | 10.783863939751038 | 50.04296875 | 6.067999839782715 | WIFI |
| 23241 | 00:05:fe:[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-22 19:48:13 | 3 | -70 | 59.90924365215176 | 10.766272893591012 | 52.09857177734375 | 15.170000076293945 | WIFI |
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.20:** Experiment 4 after filtering with a distance of 200m



| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 8968 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:22:48 | 1 | -73 | 59.913309573428876 | 10.753439965044736 | 47.86895751953125 | 15.170000076293945 | WIFI |
| 9957 | 00:05:51[redacted] | SmartHUB-23117 | [WPA2-PSK-CCMP][ESS] | 2021-01-19 08:25:24 | 1 | -87 | 59.91624984170341 | 10.748178063511556 | 54.10205078125 | 19.720998764038086 | WIFI |
| 62812 | 00:05:fe[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-25 13:56:30 | 5 | -85 | 59.922323054513456 | 10.775315536016077 | 74.467529296875 | 13.652999877929688 | WIFI |
| 63224 | 00:05:fe[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-25 13:57:22 | 5 | -87 | 59.91807827801536 | 10.775251977891463 | 73.82000732421875 | 13.652999877929688 | WIFI |
| 74591 | 00:05:fe[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:27:19 | 4 | -82 | 59.86672983895523 | 10.783145426832572 | 53.944580078125 | 4.550999641418457 | WIFI |
| 74678 | 00:05:fe[redacted] | TI BPL2 TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:27:55 | 4 | -86 | 59.86282736863926 | 10.784332272353192 | 51.228759765625 | 6.067999839782715 | WIFI |
| 77490 | 00:05:fe[redacted] | ThermiCam2Production TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 18:46:42 | 8 | -82 | 59.88028041626234 | 10.786611981007358 | 155.78741455078125 | 13.652999877929688 | WIFI |
| 78016 | 00:05:fe[redacted] | ThermiCam2Production TRC | [WPA2-PSK-CCMP][ESS] | 2021-02-26 18:47:30 | 8 | -62 | 59.88252463201184 | 10.780358689826734 | 151.41656494140625 | 13.652999877929688 | WIFI |
| 25672 | 00:05:fe[redacted] | 10-7430 TRC414357 | [WPA2-PSK-CCMP][ESS] | 2021-02-22 19:57:47 | 5 | -84 | 59.909251407518816 | 10.759401847366929 | 67.76263427734375 | 16.687000274658203 | WIFI |
| 31689 | 00:05:fe[redacted] | 10-7430 TRC414357 | [WPA2-PSK-CCMP][ESS] | 2021-02-22 20:55:31 | 5 | -78 | 59.90407129393068 | 10.766697605334743 | 48.396484375 | 6.067999839782715 | WIFI |
| 31713 | 00:05:fe[redacted] | 10-7430 TRC414357 | [WPA2-PSK-CCMP][ESS] | 2021-02-22 20:55:47 | 5 | -72 | 59.904237327454766 | 10.76660673757769 | 47.861328125 | 7.585000038146973 | WIFI |
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.21:** Experiment 4 after filtering with a distance of 400m



| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 27624 | 00:05:fe[redacted] | 10-7430 TRC426737 | [WPA2-PSK-CCMP][ESS] | 2021-02-22 20:13:14 | 11 | -89 | 59.904309972642594 | 10.753593962439883 | 34.47503662109375 | 6.067999839782715 | WIFI |
| 32343 | 00:05:fe[redacted] | 10-7430 TRC426737 | [WPA2-PSK-CCMP][ESS] | 2021-02-22 21:00:56 | 11 | -85 | 59.905399796084424 | 10.769127828932488 | 46.0537109375 | 10.618999481201172 | WIFI |
| 74203 | 00:0c:42[redacted] | - | [WPA2-PSK-CCMP][ESS] | 2021-02-26 17:23:47 | 136 | -80 | 59.88954686146734 | 10.763991644778484 | 54.93878173828125 | 7.585000038146973 | WIFI |
| 88435 | 00:0c:42[redacted] | - | [WPA2-PSK-CCMP][ESS] | 2021-02-27 15:53:33 | 136 | -89 | 59.86543782672774 | 10.749211869276817 | 41.90423583984375 | 4.550999641418457 | WIFI |
| 26767 | 00:0c:43[redacted] | Maren&Ingvild | [WPA2-PSK-CCMP][WPS][ESS] | 2021-02-22 20:04:14 | 44 | -79 | 59.90647421874551 | 10.75639984894964 | 48.3487548828125 | 6.067999839782715 | WIFI |
| 61699 | 00:0c:43[redacted] | Maren&Ingvild | [WPA2-PSK-CCMP][WPS][ESS] | 2021-02-25 13:54:45 | 44 | -83 | 59.92763387663537 | 10.77256608533666 | 86.3094482421875 | 15.170000076293945 | WIFI |
| 15714 | 00:0e:8[redacted] | RB-201-3490 | [WPA2-PSK-CCMP][ESS] | 2021-02-17 21:05:25 | 6 | -90 | 59.90759126284543 | 10.77658366665162 | 61.9152221679875 | 27.305999755859375 | WIFI |
| 50676 | 00:0e:8[redacted] | RB-201-3490 | [WPA2-PSK-CCMP][ESS] | 2021-02-25 11:03:24 | 6 | -89 | 59.942445021556736 | 10.70355430319688 | 133.25244140625 | 10.618999481201172 | WIFI |
| 89757 | 00:14:0[redacted] | Peugeot | Handsfree;10 | 2021-02-27 17:34:23 | 1032 | -90 | 59.88364071094672 | 10.76999697210602 | 50.20654296875 | 13.652999877929688 | BT |
| 89899 | 00:14:0[redacted] | Peugeot | Handsfree;10 | 2021-02-27 17:35:26 | 1032 | -90 | 59.89152158990163 | 10.761056937460998 | 62.2078857421875 | 13.652999877929688 | BT |
| 74119 | 00:15:0[redacted] | Star Micronics | null;10 | 2021-02-26 17:22:16 | 1664 | -96 | 59.90083126216449 | 10.760083861665356 | 53.742919921875 | 4.550999641418457 | BT |
| ID | MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |

**Figure 6.22:** Experiment 4 after filtering with a distance of 800m

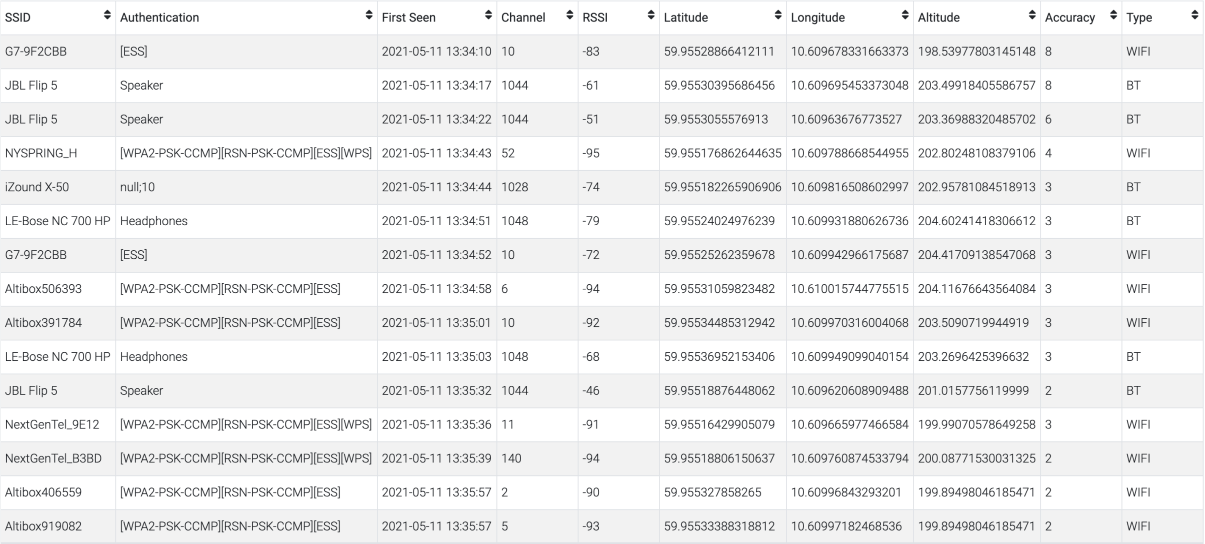
# Chapter 7

# Results

## 7.1 Experiment 1 - Detecting residents in homes

### 7.1.1 Data

The list of scanned devices in the current selection after conducting experiment 1 is shown in Figure 7.1.



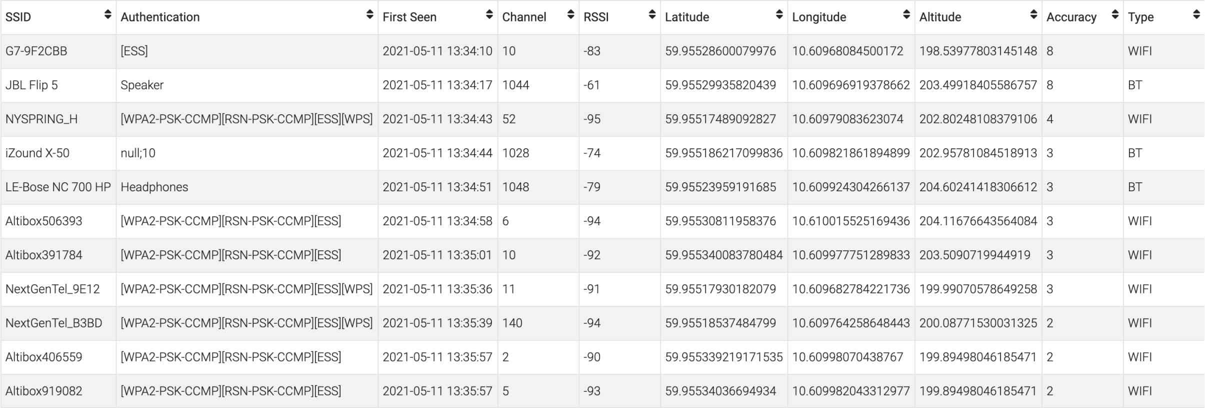
| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|
| G7-9F2CBB | [ESS] | 2021-05-11 13:34:10 | 10 | -83 | 59.95528866412111 | 10.609678331663373 | 198.53977803145148 | 8 | WIFI |
| JBL Flip 5 | Speaker | 2021-05-11 13:34:17 | 1044 | -61 | 59.95530395686456 | 10.609695453373048 | 203.49918405586757 | 8 | BT |
| JBL Flip 5 | Speaker | 2021-05-11 13:34:22 | 1044 | -51 | 59.9553055576913 | 10.60963676773527 | 203.36988320485702 | 6 | BT |
| NYSPRING_H | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:34:43 | 52 | -95 | 59.955176862644635 | 10.609788668544955 | 202.80248108379106 | 4 | WIFI |
| iZound X-50 | null;10 | 2021-05-11 13:34:44 | 1028 | -74 | 59.955182265906906 | 10.609816508602997 | 202.95781084518913 | 3 | BT |
| LE-Bose NC 700 HP | Headphones | 2021-05-11 13:34:51 | 1048 | -79 | 59.95524024976239 | 10.609931880626736 | 204.60241418306612 | 3 | BT |
| G7-9F2CBB | [ESS] | 2021-05-11 13:34:52 | 10 | -72 | 59.95525262359678 | 10.609942966175687 | 204.41709138547068 | 3 | WIFI |
| Altibox506393 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:34:58 | 6 | -94 | 59.95531059823482 | 10.610015744775515 | 204.11676643564084 | 3 | WIFI |
| Altibox391784 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:01 | 10 | -92 | 59.95534485312942 | 10.609970316004068 | 203.5090719944919 | 3 | WIFI |
| LE-Bose NC 700 HP | Headphones | 2021-05-11 13:35:03 | 1048 | -68 | 59.95536952153406 | 10.609949099040154 | 203.2696425396632 | 3 | BT |
| JBL Flip 5 | Speaker | 2021-05-11 13:35:32 | 1044 | -46 | 59.95518876448062 | 10.609620608909488 | 201.0157756119999 | 2 | BT |
| NextGenTel_9E12 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:36 | 11 | -91 | 59.95516429905079 | 10.609665977466584 | 199.99070578649258 | 3 | WIFI |
| NextGenTel_B3BD | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:39 | 140 | -94 | 59.95518806150637 | 10.609760874533794 | 200.08771530031325 | 2 | WIFI |
| Altibox406559 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:57 | 2 | -90 | 59.955327858265 | 10.60996843293201 | 199.89498046185471 | 2 | WIFI |
| Altibox919082 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:57 | 5 | -93 | 59.95533388318812 | 10.60997182468536 | 199.89498046185471 | 2 | WIFI |

**Figure 7.1:** Experiment 1 results

The list contains some entries that are duplicate devices that were scanned more than once. Removing duplicates gives the list shown in Figure 7.2.

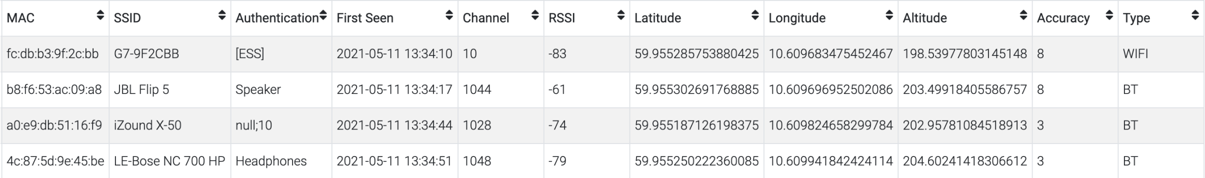
Removing access points by filtering out any entry with authentication that starts with the string "[WPA" is shown in Figure 7.3.

This list is a perfect match with the list of known devices.



| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|
| G7-9F2CBB | [ESS] | 2021-05-11 13:34:10 | 10 | -83 | 59.95528600079976 | 10.60968084500172 | 198.53977803145148 | 8 | WIFI |
| JBL Flip 5 | Speaker | 2021-05-11 13:34:17 | 1044 | -61 | 59.95529935820439 | 10.609696919378662 | 203.49918405586757 | 8 | BT |
| NYSPRING_H | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:34:43 | 52 | -95 | 59.95517489092827 | 10.60979083623074 | 202.80248108379106 | 4 | WIFI |
| iZound X-50 | null;10 | 2021-05-11 13:34:44 | 1028 | -74 | 59.955186217099836 | 10.609821861894899 | 202.95781084518913 | 3 | BT |
| LE-Bose NC 700 HP | Headphones | 2021-05-11 13:34:51 | 1048 | -79 | 59.95523959191685 | 10.609924304266137 | 204.60241418306612 | 3 | BT |
| Altibox506393 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:34:58 | 6 | -94 | 59.95530811958376 | 10.610015525169436 | 204.11676643564084 | 3 | WIFI |
| Altibox391784 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:01 | 10 | -92 | 59.955340083780484 | 10.609977751289833 | 203.5090719944919 | 3 | WIFI |
| NextGenTel_9E12 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:36 | 11 | -91 | 59.95517930182079 | 10.609682784221736 | 199.99070578649258 | 3 | WIFI |
| NextGenTel_B3BD | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:39 | 140 | -94 | 59.95518537484799 | 10.609764258648443 | 200.08771530031325 | 2 | WIFI |
| Altibox406559 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:57 | 2 | -90 | 59.955339219171535 | 10.60998070438767 | 199.89498046185471 | 2 | WIFI |
| Altibox919082 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:57 | 5 | -93 | 59.95534036694934 | 10.609982043312977 | 199.89498046185471 | 2 | WIFI |

**Figure 7.2:** Experiment 1 results with no duplicates



| MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|
| fc:db:b3:9f:2c:bb | G7-9F2CBB | [ESS] | 2021-05-11 13:34:10 | 10 | -83 | 59.955285753880425 | 10.609683475452467 | 198.53977803145148 | 8 | WIFI |
| b8:f6:53:ac:09:a8 | JBL Flip 5 | Speaker | 2021-05-11 13:34:17 | 1044 | -61 | 59.955302691768885 | 10.609696952502086 | 203.49918405586757 | 8 | BT |
| a0:e9:db:51:16:f9 | iZound X-50 | null;10 | 2021-05-11 13:34:44 | 1028 | -74 | 59.955187126198375 | 10.609824658299784 | 202.95781084518913 | 3 | BT |
| 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:34:51 | 1048 | -79 | 59.955250222360085 | 10.609941842424114 | 204.60241418306612 | 3 | BT |

**Figure 7.3:** Experiment 1 results with no duplicates and no access points

### 7.1.2 Discussion

The fact that the list found by using the prototype is a perfect match with the list of known devices shows that this method is quite accurate for detecting user electronics in houses, and thereby quite accurate for detecting people in houses. However, the experiment was conducted with nearly perfect conditions, so this only shows the best possible scenario for an adversary. In a real-life scenario, the adversary would likely face various problems that make the method less reliable.

Possible factors that could make the process less accurate in a real-life situation include:

- ***Not knowing what user electronics to look for in a house.*** This experiment was done with a list of known devices in the house. An adversary would not have this and would therefore not be sure if the devices they scan are coming from inside the house.
- ***Not being able to get as close to the house.*** This experiment was done by walking around the house twice with a distance of approximately two meters from the outer wall. This might not be something an adversary is able to do without causing suspicion from neighbors or passers-by. Scanning for devices at night might alleviate some of this. However, scanning for devices at night will likely not be as effective, as the house residents would likely be sleeping and not have their personal electronic devices turned on.
- ***Residents sleeping or not having any devices turned on.*** This experiment was done with two Bluetooth speakers, a Wi-Fi-enabled camera, and a pair of Bluetooth headphones turned on simultaneously. This might not be a likely scenario. A typical household might not have that many devices turned on simultaneously if any at all. Furthermore, as mentioned before, if the house residents are sleeping, chances are they have switched off all their devices.

The prototype worked well for this use case. The two CSV files alone would have been quite hard to work with without a way of finding the difference between the two files, removing duplicates, and removing access points. This becomes especially important once the use-case is applied to multiple houses.

A question to ask is if this method is any better than inspecting the house visually. It is fair to assume that in most cases when someone is at home and has their personal wireless devices turned on, it would be pretty easy to spot this from outside the house anyway. That being said, some people will let their lights stay on while away to dissuade intruders. In this case, scanning and seeing that there are no personal wireless devices turned on could be a useful clue for an adversary.

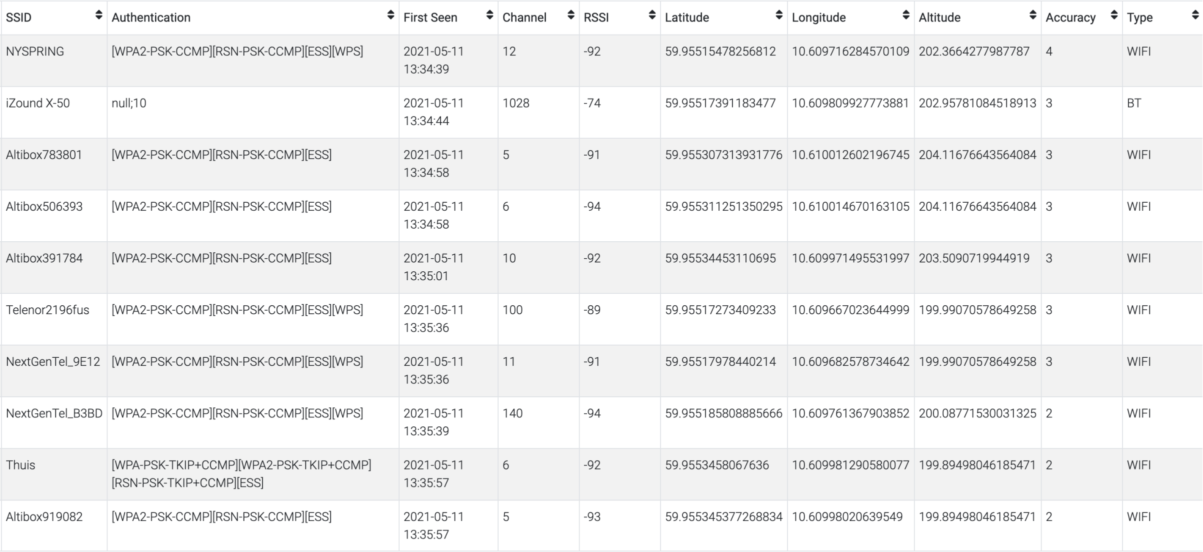
The use-case for this experiment was also about detecting expensive devices that could be potential targets for theft. A problem the adversary might face is that any house they find to be a potential target for breaking into will be a house where they did not find any wireless devices. This means the adversary will only be able to find out if nobody is home or if the house contains expensive electronic devices, never both.

All in all, this method of checking if someone is home is quite accurate, although its actual practical usefulness is debatable. It might serve as a tool to use in combination with other methods. Therefore, the privacy risk related to this method is moderate in theory, but probably relatively low in practice.

## 7.2 Experiment 2 - Detecting visitors in homes

### 7.2.1 Data

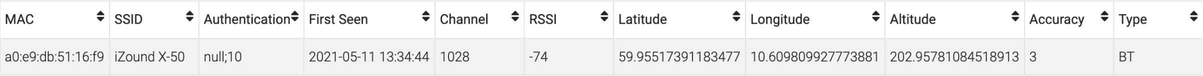
The list of scanned devices in the current selection after conducting experiment 2 is shown in Figure 7.4.



| SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|
| NYSPRING | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:34:39 | 12 | -92 | 59.95515478256812 | 10.609716284570109 | 202.3664277987787 | 4 | WIFI |
| iZound X-50 | null;10 | 2021-05-11 13:34:44 | 1028 | -74 | 59.95517391183477 | 10.609809927773881 | 202.95781084518913 | 3 | BT |
| Altibox783801 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:34:58 | 5 | -91 | 59.955307313931776 | 10.610012602196745 | 204.11676643564084 | 3 | WIFI |
| Altibox506393 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:34:58 | 6 | -94 | 59.955311251350295 | 10.610014670163105 | 204.11676643564084 | 3 | WIFI |
| Altibox391784 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:01 | 10 | -92 | 59.95534453110695 | 10.609971495531997 | 203.5090719944919 | 3 | WIFI |
| Telenor2196fus | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:36 | 100 | -89 | 59.95517273409233 | 10.609667023644999 | 199.99070578649258 | 3 | WIFI |
| NextGenTel_9E12 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:36 | 11 | -91 | 59.95517978440214 | 10.609682578734642 | 199.99070578649258 | 3 | WIFI |
| NextGenTel_B3BD | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS][WPS] | 2021-05-11 13:35:39 | 140 | -94 | 59.955185808885666 | 10.609761367903852 | 200.08771530031325 | 2 | WIFI |
| Thuis | [WPA-PSK-TKIP+CCMP][WPA2-PSK-TKIP+CCMP][RSN-PSK-TKIP+CCMP][ESS] | 2021-05-11 13:35:57 | 6 | -92 | 59.9553458067636 | 10.609981290580077 | 199.89498046185471 | 2 | WIFI |
| Altibox919082 | [WPA2-PSK-CCMP][RSN-PSK-CCMP][ESS] | 2021-05-11 13:35:57 | 5 | -93 | 59.955345377268834 | 10.60998020639549 | 199.89498046185471 | 2 | WIFI |

**Figure 7.4:** Experiment 2 results

Removing access points by filtering out any entry with authentication that starts with the string "[WPA" is shown in Figure 7.5.



| MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|
| a0:e9:db:51:16:f9 | iZound X-50 | null;10 | 2021-05-11 13:34:44 | 1028 | -74 | 59.95517391183477 | 10.609809927773881 | 202.95781084518913 | 3 | BT |

**Figure 7.5:** Experiment 2 results with no access points

This is a perfect match with the device only turned on for the second scan.

### 7.2.2 Discussion

The fact that the list found by using the prototype is a perfect match with the list of devices switched on for the second scan, but not the first, shows that this method is quite accurate for detecting devices that are not usually present, and thereby quite accurate for detecting visitors in houses. However, as with experiment 1, the experiment was conducted under nearly perfect conditions, thus only shows the best possible scenario for an adversary. In a real-life scenario, the adversary would likely face various problems making the method less reliable.

Possible factors that could make the process less accurate in a real-life situation include:

- ***Not knowing what user electronics to look for in a house.*** This experiment was done with both a baseline of devices usually present in the house as well as a known device in addition to this. An adversary would have to establish a baseline of devices that are usually present in the house, which would be very time-consuming. Also, the adversary would have to understand if a device not in the baseline set of devices is coming from the house.

- ***Not being able to get as close to the house.*** This experiment was done by walking around the house twice with a distance of approximately two meters from the outer wall. Establishing a baseline of devices for a house would require the adversary to do this several times over an extended time period. This might not be something an adversary is able to do without causing suspicion from neighbors or passers-by. Scanning for devices at night might alleviate some of this. However, scanning for devices at night will likely not be as effective, as the house residents would likely be sleeping and not have their personal electronic devices turned on anyway.

- ***Residents sleeping or not having any devices turned on.*** This experiment was done with two Bluetooth speakers, a Wi-Fi-enabled camera, and a pair of Bluetooth headphones turned on simultaneously. This might not be a likely scenario. A typical household might not have that many devices turned on simultaneously if any at all. Furthermore, as mentioned before, if the residents of the house are sleeping, chances are they have switched off all their devices.

The prototype worked well for this use case. The two CSV files alone would have been quite hard to work with without a way of finding the difference between the two files and removing access points. This becomes especially important once the use-case is applied to multiple houses.

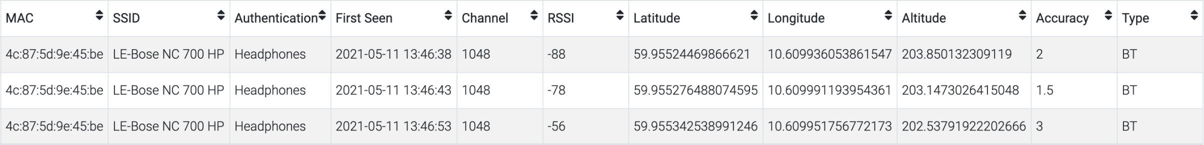
A question to ask is if this method is any better than inspecting the house visually. It seems it would be easier for an adversary just to watch the entrance of the house and see if anyone comes to visit.

All in all, this method of checking if someone is home is quite accurate, although its actual practical usefulness is seemingly low. It might serve as a tool to use in combination with other methods. Therefore, the privacy risk related to this method is moderate in theory but is likely quite very low in practice.

## 7.3 Experiment 3 - Adding context from location

### 7.3.1 Data

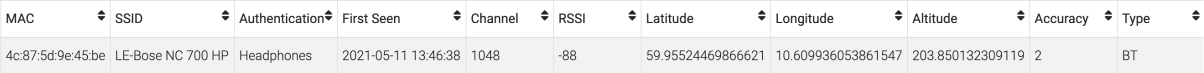
The list of scanned devices in the current selection after conducting experiment 2 is shown in Figure 7.6.



| MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|
| 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:38 | 1048 | -88 | 59.95524469866621 | 10.609936053861547 | 203.850132309119 | 2 | BT |
| 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:43 | 1048 | -78 | 59.955276488074595 | 10.609991193954361 | 203.1473026415048 | 1.5 | BT |
| 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:53 | 1048 | -56 | 59.955342538991246 | 10.609951756772173 | 202.53791922202666 | 3 | BT |

**Figure 7.6:** Experiment 3 results

Removing duplicates gives the list shown in Figure 7.7.



| MAC | SSID | Authentication | First Seen | Channel | RSSI | Latitude | Longitude | Altitude | Accuracy | Type |
|---|---|---|---|---|---|---|---|---|---|---|
| 4c:87:5d:9e:45:be | LE-Bose NC 700 HP | Headphones | 2021-05-11 13:46:38 | 1048 | -88 | 59.95524469866621 | 10.609936053861547 | 203.850132309119 | 2 | BT |

**Figure 7.7:** Experiment 3 results with no duplicates

This list is a perfect match with the single known device.

### 7.3.2 Discussion

The list found by using the prototype is a perfect match with the known device. This shows that this method is quite accurate for detecting a device at a location that can give context as to what someone could be doing there and comparing it with a dataset of devices to determine who it was. As with the former two experiments, this was done with nearly perfect conditions, so this only shows the best possible scenario for an adversary. In a real-life scenario, the adversary would likely face various problems that make the method less reliable.

Possible factors that could make the process less accurate in a real-life situation include:

- ***Requiring a large and specific dataset.*** Obtaining a large enough dataset with the right devices in it is likely very hard. The adversary would have to scan for devices in a large area for there to be enough devices in the dataset for this method to be viable. In addition, the types of devices that people carry with them are also not easy to scan. Many smartphones do not send out probe requests when they are not actively looking for devices to pair with. Also, devices like wireless headphones are likely to only be on while they are currently on a person, meaning scanning outside houses or from a car is likely not efficient for finding these devices.

- ***Scanning constantly at a specific location.*** scanning at a specific location will only be useful if it can be done for an extended period of time, letting enough devices pass by for there to be any statistical chance of picking up a device also found in the large dataset. This can be hard for an adversary if the adversary is not in control of the location. For locations like medical clinics or similar, this is likely the case. The adversary would have to either hide a scanning device at the location or be present at the location. At that point, the adversary could just as well have spotted people visually. Hiding a scanning device at a place like a medical clinic is likely hard without causing suspicion and involves the risk that someone finds the device. This is also dependent on the adversary being able to retrieve the device again at a later time.

The prototype worked well for this use case. The two CSV files alone would have been quite hard to work with without having a way of finding the intersection of the two files.

This method seems to be better than visual inspection, though it depends on the adversary hiding and retrieving a scanning device in the given location. In addition, even if the adversary is able to find a match between the two datasets, finding the identity of the person the device belongs to might be very hard.

All in all, the method is accurate but is likely hard to actually accomplish in a real-life scenario. It might be useful as a tool in combination with other methods. The privacy risk related to this method is therefore high in theory but somewhat low in practice.

## 7.4 Experiment 4 - Detecting frequent locations and frequently taken paths

### 7.4.1 Data

The current selection after conducting experiment 4 contains 276 devices. Some notable devices are:

- "iPhone"
- "Peugeot"
- "Highway controller"

#### "iPhone"

The iPhone was scanned in two completely different locations on two different days.

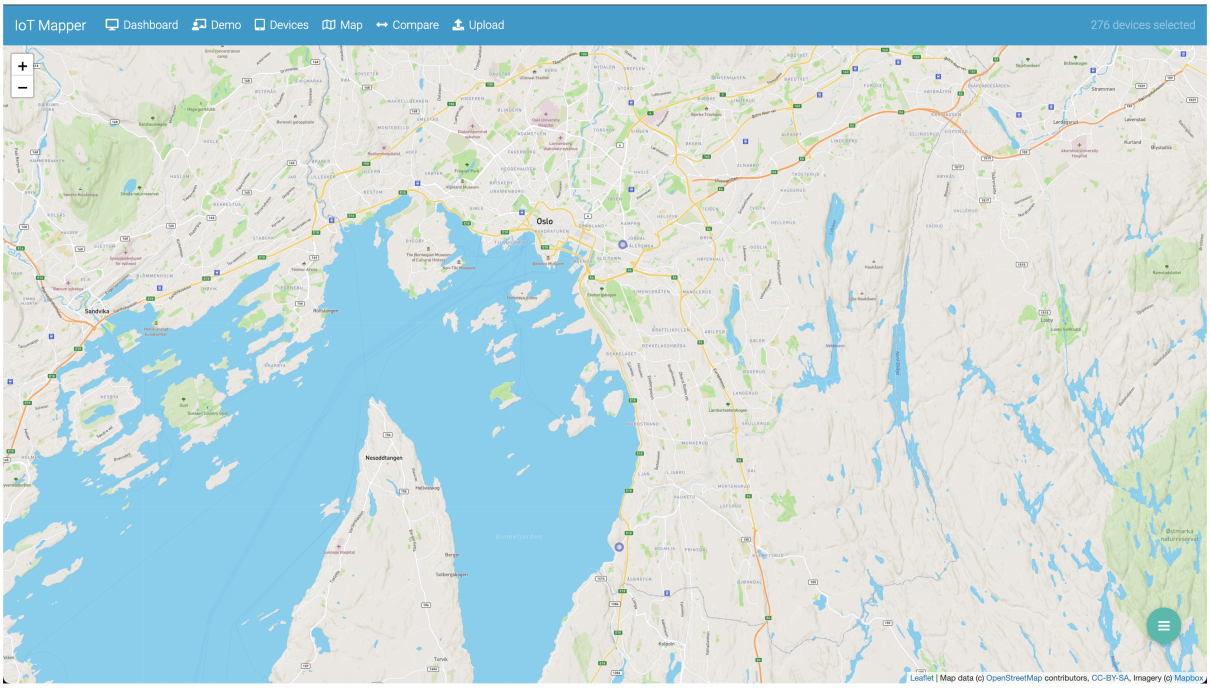


**Figure 7.8:** Experiment 4 iPhone

The iPhone is especially interesting because smartphones are only registered by Wigle when in Bluetooth pairing mode. This experiment shows that given a large dataset, it is possible to detect phones. In addition, with an even larger dataset, it might even be possible to find paths taken or patterns in a person's daily life with this method.

### "Peugeot"

The device with the name "Peugeot" was scanned along the same road within a short period of time. This, in combination with the name "Peugeot", suggests that it was scanned while driving.

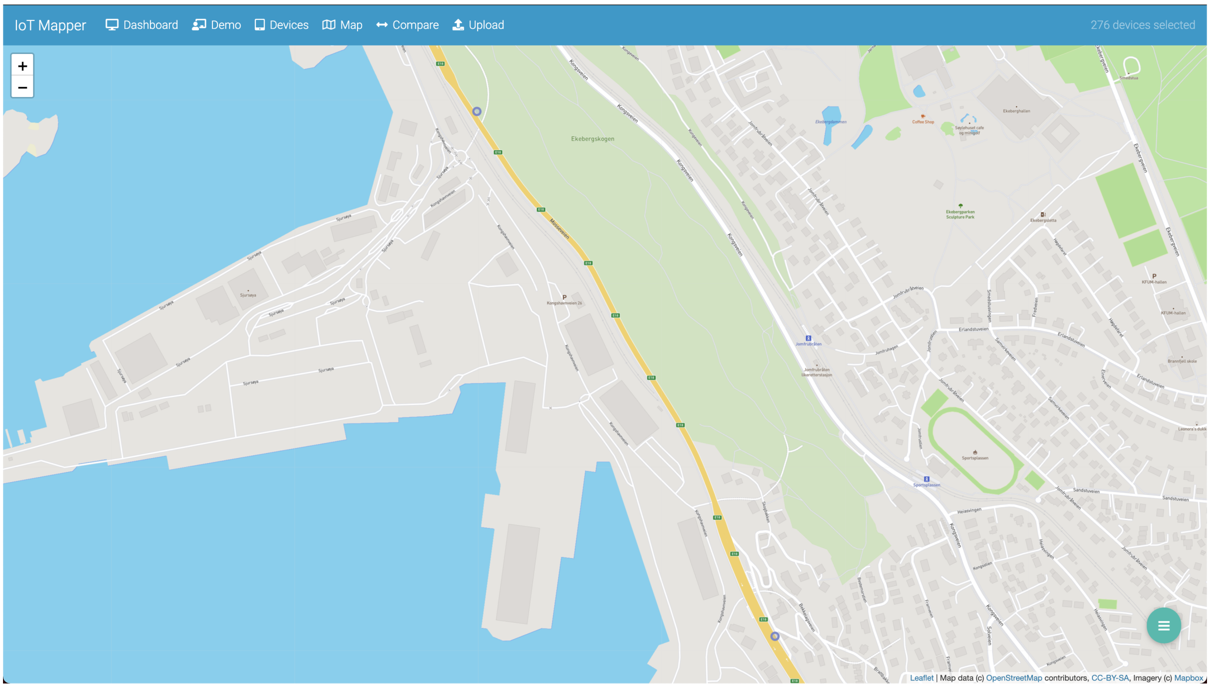


**Figure 7.9:** Experiment 4 Peugeot

### "Highway controller"

This device appears to be a Bluetooth accessory for cars. It was scanned at two separate days and two separate locations, suggesting that the same car was scanned twice independently.

In addition to those three devices, there are various other devices scanned at different locations. These include electric scooters, Wi-Fi access points, and many devices with no SSID/name. Most of the devices in the selection only appeared twice, meaning it is challenging to extract meaningful information about paths taken and patterns in the devices.

## 7.4.2 Discussion

The fact that this method allows for finding various devices in different locations at different times shows that it might be possible to find paths taken and patterns in people's movement through this method, given a larger dataset. The fact that devices like phones, cars, and electric scooters appear in the selection after filtering also makes it seem like a viable method for tracking individuals' movements. If an adversary were to scan for a longer period of time with several devices spread out in an area, they might be able to find precise paths taken by individuals and patterns in their daily lives.

Even though the method seems quite viable, there are some problems an adversary might face. The problem of tying a device to a person is maybe the biggest. Even if the adversary is able to find precise paths taken by a device, they still need a way to find out whom the device belongs to for it to be a privacy risk. If the adversary finds patterns in where a device appears over the course of a week, however, they

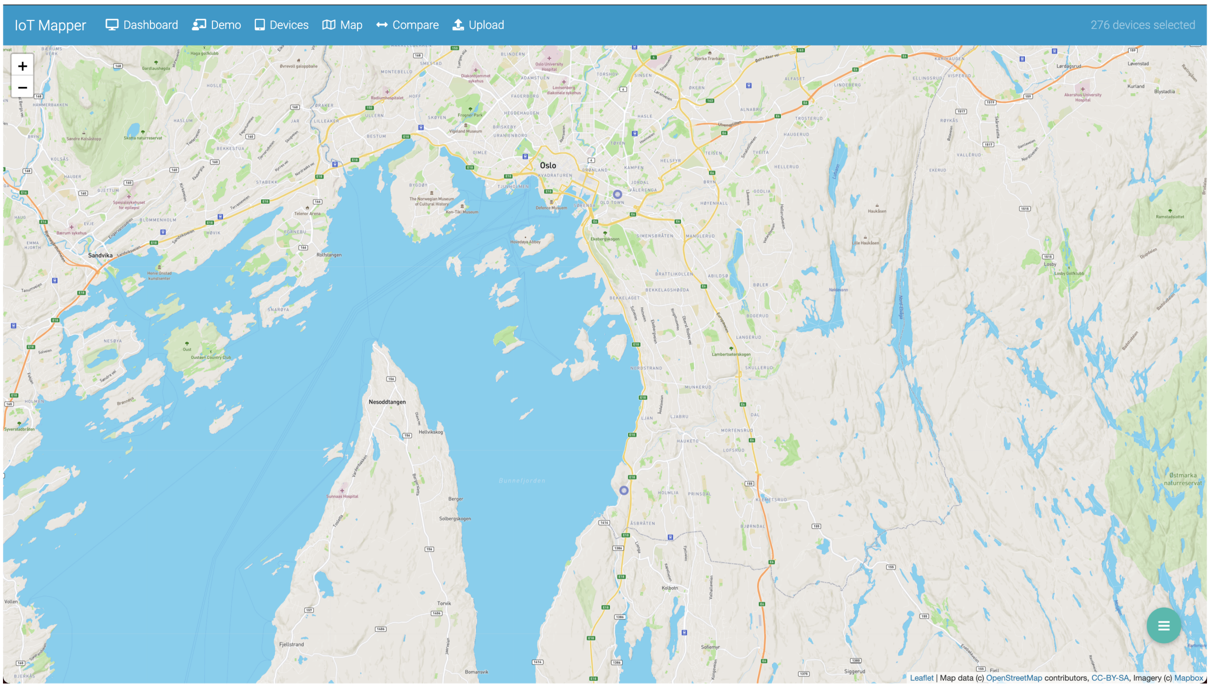


**Figure 7.10:** Experiment 4 Highway controller

might be able to intercept the person at a location the person regularly visits to find the identity of the person. This means this method could be a substantial privacy risk if used in combination with other methods.

More research on this topic is needed to say anything with certainty. However, it seems this method could potentially be a substantial privacy risk.

# Chapter 8

# Conclusion

## 8.1 Summary

Wi-Fi and Bluetooth devices are everywhere these days, making it increasingly important to be aware of privacy risks related to these technologies. These devices constantly broadcast signals that can be picked up by unwanted actors. Apps and programs that are freely available to anyone can be used to scan and analyze these signals. By developing a prototype for a system that can store and map the results of these scans, it can be demonstrated what privacy risks related to these devices are exploitable by an adversary.

The proposed prototype is web-based to be easily accessible across different devices. It accepts CSV files following the format specified by the Wigle app. The functionality includes the ability to view all the scanned devices on a map or in a table, adding or subtracting devices in the current selection of devices, trilateration of coordinates, and more.

By conducting experiments using this prototype, it is demonstrated that certain methods can be used to obtain private information about people. This includes what kind of devices are in a house, if there are visitors at a house, if someone living in a given house has visited a specific location, and patterns in how devices move on a larger scale. The first three experiments show that it is possible to use Wi-Fi and Bluetooth scanning to obtain this information. However, it is hardly practical compared to other methods of obtaining the same information. However, the last experiment showed that it is possible to detect patterns in the movement of devices in an area given a large enough dataset.

## 8.2 Research questions revisited

### 8.2.1 RQ1

*What information are personal devices giving out to third parties through wireless communication?*

Personal devices are constantly broadcasting their presence through Wi-Fi and Bluetooth. Because these signals do not travel very far, the approximate location of the devices can be found through the GPS of the scanning device. This can be made even more accurate through the use of trilateration. Modern phones have implemented measures to limit this by only broadcasting their presence while they are currently searching for Wi-Fi access points or Bluetooth accessories. Despite this, the devices they are connected to, like headphones and car stereos, often lack this functionality.

### 8.2.2 RQ2

*How can information from personal devices be exploited?*

The information broadcasted from personal devices can be used to detect if someone is home, what devices are present in their house, if they have visitors, where they have visited, and how they move in a large area. This is all dependent on very favorable conditions for an adversary, however, and is likely not any more useful than other methods, the exception being how someone moves in a large area. This could potentially be a substantial privacy risk if an adversary is able to acquire a large enough dataset.

### 8.2.3 RQ3

*How can users best understand how their privacy is affected by the internet of things?*

The privacy risks revolving around the internet of things can be demonstrated with a tool that lets the user scan for devices themselves and see how much information can be extracted.

### 8.2.4 RQ4

*What other risks besides privacy risks are present with the internet of things?*

Besides privacy risks, there is also the risk of property loss, should an adversary use the signals coming from household electronics to decide if a house is worth breaking into. Other than that, this thesis did not encounter any other non-privacy risks.

## 8.3 Future work

### 8.3.1 Probe requests

As stated in section 2.3, Wi-Fi probe requests can contain a list of SSIDs for the networks a device is configured to join. This seems like a substantial privacy risk. Since the Wigle app does not record probe requests, this thesis could not investigate this further. Developing a system for capturing these probe requests and visually demonstrating the privacy risks involved could give valuable insights for the public into this issue.

### 8.3.2 Larger datasets

The privacy risk that appears the largest was related to scanning for a long period of time in a large area. This thesis demonstrated that many devices appear in multiple locations at different times in a dataset gathered from a single device. Scanning for devices in multiple locations simultaneously over a longer period of time could potentially reveal a substantial privacy risk.

# Bibliography

[1] PwC. *Internet of Things - gir deg muligheten til å styre taklampen fra mobilen.* 2020. URL: https://www.pwc.no/no/teknologi-omstilling/digitalisering-pa-1-2-3/internet-of-things.html (visited on 03/05/2020).

[2] Statista. *Forecast end-user spending on IoT solutions worldwide from 2017 to 2025*. 2020. URL: https://www.statista.com/statistics/976313/global-iot-market-size/ (visited on 03/05/2020).

[3] ITU. *Internet of Things Global Standards Initiative*. 2020. URL: https://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx (visited on 05/05/2020).

[4] Margaret Rouse. *internet of things (IoT)*. 2020. URL: https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT (visited on 05/05/2020).

[5] Imed Romdhani, Riad Abdmeziem and D. Tandjaoui. 'Architecting the Internet of Things: State of the Art'. In: July 2015.

[6] TechTerms. *NIC Definition*. 2020. URL: https://techterms.com/definition/nic (visited on 08/05/2020).

[7] OmniSecu. *What is NIC (Network Interface Card)*. 2020. URL: https://www.omnisecu.com/basic-networking/what-is-nic-card-network-interface-card.php (visited on 08/05/2020).

[8] Wi-Fi Alliance. *Discover Wi-Fi*. 2020. URL: https://www.wi-fi.org/discover-wi-fi (visited on 24/05/2020).

[9] Paal E. Engelstad. *IEEE 802.11 WiFi*. Universitetet i Oslo. 2020. URL: https://folk.uio.no/yanzhang/IN5030-2020/WLAN.pdf (visited on 24/05/2020).

[10] *Wi-Fi Channels, Frequencies, Bands & Bandwidths*. Electronics notes. 2020. URL: https://www.electronics-notes.com/articles/connectivity/wifi-ieee-802-11/channels-frequencies-bands-bandwidth.php (visited on 24/05/2020).

[11] *WiFi channels explained*. Minim. 2018. URL: https://www.minim.co/blog/wifi-channels-explained (visited on 24/05/2020).

[12] *WLAN (IEEE 802.11) capture setup*. Wireshark. 2020. URL: https://wiki.wireshark.org/CaptureSetup/WLAN#Promiscuous_mode (visited on 24/05/2020).

[13] AnandKumar Sukumar. *How is the BSSID derived from the Access Point ethernet MAC address?* 2014. URL: https://community.arubanetworks.com/browse/articles/blogviewer?blogkey=ca6f2be7-6974-4134-bed7-507db1f90deb (visited on 28/05/2021).

[14] Wi-Fi Alliance. *What are passive and active scanning?* 2020. URL: https://www.wi-fi.org/knowledge-center/faq/what-are-passive-and-active-scanning (visited on 25/05/2020).

[15] O'Reilly. *Chapter 4. 802.11 Framing in Detail*. 2020. URL: https://www.oreilly.com/library/view/80211-wireless-networks/0596100523/ch04.html (visited on 25/05/2020).

[16] David Akin and Jim Greier. *CWAP Certified Wireless Analysis Professional Official Study Guide (Exam PW0-205) First Edition*. Brandon A. Nordin, 2004. ISBN: 0-07-225585-4.

[17] Jennifer Bray and Charles F. Sturman. *Bluetooth: Connect Without Cables Second Edition*. Prentice Hall PTR, Prentice-Hall, Inc. Upper Saddle River, NJ 07458, 2002. ISBN: 0-13-066106-6.

[18] Akash Peshin. *What Is The Range Of Bluetooth And How Can It Be Extended?* ScienceABC. 2018. URL: https://www.scienceabc.com/innovation/what-is-the-range-of-bluetooth-and-how-can-it-be-extended.html (visited on 27/05/2020).

[19] Guevara Noubir. *Bluetooth*. Northeastern University. 2020. URL: http://www.ccs.neu.edu/home/noubir/Courses/CSU610/S06/bluetooth.pdf (visited on 28/05/2020).

[20] McAfee. *What is Wardriving?* 2014. URL: https://www.mcafee.com/blogs/consumer/identity-protection/wardriving/ (visited on 26/02/2021).

[21] Alison Grace Johansen. *Wardriving: What it is and how to help protect your network*. 2020. URL: https://us.norton.com/internetsecurity-id-theft-wardriving-what-it-is-and-how-to-help-protect-your-network.html (visited on 26/02/2021).

[22] WiGLE. *Frequently Asked Questions*. 2021. URL: https://wigle.net/faq (visited on 26/02/2021).

[23] Vue.js. *Introduction*. 2020. URL: https://vuejs.org/v2/guide/#What-is-Vue-js (visited on 21/10/2020).

[24] Vue.js. *vue*. 2020. URL: https://github.com/vuejs/vue (visited on 21/10/2020).

[25] OpenStreetMap. *About*. 2021. URL: https://www.openstreetmap.org/about (visited on 26/02/2021).

[26] Leaflet. *Leaflet*. 2021. URL: https://leafletjs.com/ (visited on 26/02/2021).

[27] Cambridge Dictionary. *Privacy*. In: *dictionary.cambridge.org dictionary*. URL: https://dictionary.cambridge.org/dictionary/english/privacy (visited on 16/04/2021).

[28] María-Teresa Gil-Bazo. 'The Charter of Fundamental Rights of the European Union and the Right to be Granted Asylum in the Union's Law'. In: *Refugee Survey Quarterly* 27.3 (Sept. 2008), pp. 33–52. ISSN: 1020-4067. DOI: 10.1093/rsq/hdn044. eprint: https://academic.oup.com/rsq/article-pdf/27/3/33/6926942/hdn044.pdf. URL: https://doi.org/10.1093/rsq/hdn044.

[29] Gilad Rosner and Erin Kenneally. 'Privacy and the Internet of Things: Emerging frameworks for policy and design'. In: *Rosner, Gilad and Kenneally, Erin, Privacy and the Internet of Things: Emerging Frameworks for Policy and Design (June 7, 2018). UC Berkeley Center for Long-Term Cybersecurity/Internet of Things Privacy Forum*. 2018.

[30] Valeros Verónica and Sebastián García. 'How bluetooth may jeopardize your privacy. An analysis of people behavioral patterns in the street.' In: Dec. 2013.

[31] Mohd Ismifaizul Mohd Ismail, Rudzidatul Dziyauddin, Shafiqa Samsul, Nur Azmi, Yoshihide Yamada, Fitri Yakub and N.A. Ahmad. 'An RSSI-based Wireless Sensor Node Localisation using Trilateration and Multilateration Methods for Outdoor Environment'. In: (Dec. 2019).

[32] A. Tillekens, N. Le-Khac and T. T. Pham Thi. ‘A Bespoke Forensics GIS Tool’. In: *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*. 2016, pp. 987–992. DOI: 10.1109/CSCI.2016.0189.

[33] Serena Zheng, Noah Apthorpe, Marshini Chetty and Nick Feamster. ‘User perceptions of smart home IoT privacy’. In: *Proceedings of the ACM on Human-Computer Interaction* 2.CSCW (2018), pp. 1–20.

[34] Luiz Oliveira, Daniel Schneider, Jano De Souza and Weiming Shen. ‘Mobile Device Detection Through WiFiProbe Request Analysis’. In: (2019).

[35] Miranda Blogg, Conor Semler, Manu Hingorani and Rod Troutbeck. ‘Travel time and origin-destination data collection using Bluetooth MAC address readers’. In: *Australasian transport research forum*. Vol. 36. 2010.

[36] WiGLE. *WiGLE CSV Format*. 2021. URL: https://api.wigle.net/csvFormat.html (visited on 26/02/2021).

[37] Mozilla. *IndexedDB API*. 2021. URL: https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API (visited on 11/05/2021).