DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Developing a Scalable, Secure, and Privacy-preserving Platform for Collection, Aggregation, and Analysis of Mobility Data

Dinh Le Khanh Duy

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Developing a Scalable, Secure, and Privacy-preserving Platform for Collection, Aggregation, and Analysis of Mobility Data

Entwicklung einer skalierbaren, sicheren, privatsphäre-erhaltenden Plattform zur Sammlung, Aggregation und Analyse von Mobilitätsdaten

Author:	Dinh Le Khanh Duy
Supervisor:	Prof. Dr.-Ing. Jörg Ott
Advisor:	Doan Trinh Viet
Submission Date:	15.03.2020

I confirm that this master's thesis in informatics is my own work and I have documented all sources and material used.

Munich, 15.03.2020 Dinh Le Khanh Duy

Abstract

blablablbalblalbalbal

Contents

1. Introduction

1.1. Motivation

There are currently round 7.7 billion people living on earth with a declining growth rate at currently 1.1 percent per year. The UN is predicting that the population growth is going to be sinking steadily in the next decades, but despite that, the population is increasing until 2100. They estimate that the earth will hit 10.9 billion at the end of the century 2100.

It is inevitable, that big cities and metropoles around the globe are growing denser year by year. According to the United Nations, urban areas around the globe have been growing denser year by year. While the same can be said for rural areas because of the general growth rate of the population, the influx of people moving into big cities is much higher compared to the countryside. Depending on how certain factors evolve over time, such as climate change, autonomous driving and other technology, it could decelerate or even accelerate growth.

To solve issues stemming from the increase in housing, traffic and infrastructure, urban planning is one of the most important tools. Improving traffic control requires knowledge of the traffic flow and inefficiencies that are causing traffic jams. Knowing local population trends, movement patterns and other factors could benefit planning housing and energy infrastructure. There are only a handful of companies that hold the monopoly on the much needed data, but acquiring them is expensive and cause problems with data protection laws. Publicizing the data without anonymizing it, severely infringes on the privacy of the originator of the data. But the sole act of anonymizing the data is not enough. The use of inference attacks may make it possible to associate data and re-identify the individual as L. Sweeney has proven.

To solve this, we can leverage the widespread availability of mobile computing power, and create a platform that provides the data on a need-to-know basis. Today, most smartphones are equipped with several sensors, including a GPS sensor, pedometer and accelerometer, which can be used to collect mobility data. Many applications are utilizing those sensors to implement location based services and games, for instance Uber and Pokemon GO, but there is a lot of controversy around products of that kind. "If you're not paying, you're the product" is a quote that is often said around data driven applications. It applies to a lot of free services offered in exchange for your data. Google Maps, for example, provides a free navigation service, whilst collecting your sensory data, such as GPS and accelerometer, to collect traffic information. Facebook provides a free social media platform for people to connect, while using it to sell targeted advertisement using your data.

So publicizing big anonymous data sets has its limits and data collection itself is a hot topic in today's media.

1.2. Research Questions

We try to find a balance between the usability of data, the privacy preservation of the user and the trust between the participants.

RQ1: What are the benefits and drawbacks in Simon van Endern's architecture?

We examine his result and analyze his architecture. Something about weakpoints.

RQ3: What improvements are possible?

His work shows a minimal viable product on which we can expand further investigations. We investigate what further data can be aggregated and how the original architecture can be improved both in security and scalability.

RQ4: Is it feasible to create a decentralized mobile network for the purpose of aggregating data?

There has been a lot of advancement in decentralization and we look into possible frameworks than can be implemented to enable a decentralized network.

1.2.1. RQ5: Is it possible to remove trust?

1.3. Contributions

We will reexamine the results found by Simon van Endern and while focusing on the scalability of the crowdsourcing platform and the privacy analysis of aggregation for more specic mobility data, we will expand on his original idea.

First, we will review related work concerning risks and solution to sensitive data and pertinent anonymization techniques. After that, we will propose our approach to solve the mentioned problems. In chapter 4, we explain our implementation approach and decisions. Chapter 5 documents our test setup and evaluate our results. Lastly, we will draw a conclusion on our work and discuss further improvements.

2. Related Work

2.1. Attack Vectors

Working with sensitive data brings out multiple risks that have to be assessed before actions can be taken. We compile possible attack vectors and how they have been solved in other systems.

2.1.1. Centralized

To improve efficiency, centralized information systems have always been a go-to infrastructure. The advantages of simple deployment, ease of maintenance and less bureaucracy will always be an incentive for big companies and governments to consider it. In 1965, the US Social Science Research Council proposed a National Data Center for the purpose of storing all data in a central location for statistical data analysis. In the end, the plans for the system were shot down because of the lack of privacy protection.

Today, there are several big tech companies, that are collecting information about their users and storing them in central databases. But centralization has a fatal flaw for protecting privacy. The collection of sensitive data in one location pose a high risk to their originator. Centralized databases that store private information are one of the weak points that have been on constant attack. In 2019, there have been over 5000 data breaches with almost 8 billion records exposed in the first three quarters, 33% more compared to the number reported in the first nine months of 2018. Around 10% of the breaches originated from the inside, from accidental leaks to malicious publications.

2.1.2. Inference

If we strip away all sensitive information, there remains a risk of private data leaking by re-identification, reconstruction and tracing. The most harmful problem is the danger of re-identification. This attack enables malicious actors to deduce identity information using other publicly available data sets or auxiliary knowledge. L. Sweeney was able to re-identify former Massachusetts governor William Weld by linking medical records from the Group Insurance Commission and the voter registration list. The goal of reconstruction is to determine sensitive data from a data set using publicly available information. Tracing on the other hand is the ability to identify if an individual is present in a data set or not.

2.1.3. Location Tracking

In regards to location data, the ability to infer the home and work location pose both risks for privacy and life and limb. Research has shown that it is possible to deduce the home address of an individual and even his work place using historical location data. Being able to analyze the movement pattern and predict the presence of a person in a certain location may put his or her life in danger.

2.2. Countermeasures

To implement a secure platform that is not so vulnerable to the problems proposed in the sections above, we look into methods and design decisions to prevent them.

2.2.1. Distributed and Decentralized

The dangers of data breaches originate from outside the companies as well as inside and their cause range from poorly implemented security to lax security policies to human error. Figure [X] depicts two more alternatives in place of a centralized architecture: distributed and decentralized.

When the internet was implemented as Advanced Research Projects Agency Network (ARPANET) in 1969, it was a decentralized network of computers scattered across the United States. With the adoption of the TCP/IP in 1982, it became the internet, an interconnected network of networks. In 1989, Tim Berners-Lee introduced the world wide web as a read-only means of accessing information from other computers and the commercialization turned it into the centralized web we know today.

In the last decade, we have seen a rise in attempts to reorganize the internet. This trend amassed attention in 2009, when the creator under the pseudonym Satoshi Nakamoto created Bitcoin, "A Peer-to-Peer Electronic Cash System" leveraging blockchain technology, followed by the Ethereum platform in 2013.

The goal of decentralization is the separation of power from a single instance, here for instance is to take away the control over data from monopolies.

Distribution takes decentralization a step further, by eliminating the central control. All instances have the same power.

One of the protocols that arose from this niche is the InterPlanetary File System (IPFS). IPFS is a peer-to-peer hypermedia protocol to make the web more decentralized and distributed.

Using distributed storage removes the single point of vulnerability and lowers the the effort-to-reward balance and thus might deter malicious actors to try to steal data.

2.2.2. Homomorphic encryption

Another way to ensure that anonymity is provided, is when the data is not readable. If a malicious actor manages to steal private data while it is still encrypted, he won't be able to infer identity or additional information, unless he manages to decrypt it beforehand. This protects that intermediate parties can't use data mining to infringe on sensitive data.

Homomorphic encryption enables the arithmetic operations on an encrypted data set. They are separated into three categories:

- **Fully Homomorphic Encryption (FHE)** allow multiple arbitrary operations, but have a lot of overhead and thus are expensive computationally.
- **Somewhat Homomorphic Encryption (SWHE)** support only selected operations to a limited number of times and are computationally more feasible.
- **Partially Homomorphic Encryption (PHE)** enables one type of operation any number of times.

2.2.3. Differential privacy

Differential privacy can be used to prevent statistical databases from leaking private information. An algorithm is differential private when it disables tracing attacks, meaning that an output doesn't show signs if a particular individual is present in it or not.

2.2.4. k-Anonymity

The above sections have shown, that if a data set seems anonymous by itself, quasi-identifiers, such as ZIP code, birth date or sex, still enable a malicious actor to link an individual to his data. One of the countermeasures to this problem, is providing k-anonymity. This is achieved when every query for a quasi-identifier returns at least k results. A quasi-identifier is an identifier or a combination of non-identifying attributes, that can used to link with external data sources to create new identifiers. For this, P. Samarati and L. Sweeney suggest the use of generalization and suppression. Former is realized by expanding certain attributes into ranges. For instance instead of assigning the real age, an age range is used. This results in a loss of accuracy, but higher degree of anonymity and thus privacy. For the latter, there are two methods of suppression. Attribute suppression removes attributes from the data set, reducing the number of possible quasi-identifiers by lowering the number of possible combinations. Record suppression on the other hand deletes entire entries in the data set to take out unique entries that don't meet the criteria of k-anonymity.

2.2.5. Spatial Cloaking

3. Design

3.1. Trust, Usability and Anonymity

In any platform, we expect a certain level trust from each participant, may it be from the service provider or the service consumer. How this trust is created can have many different sources. But the main reason a person or organization can be trusted is accountability. Companies that have a strong market presence can be trusted because they have been present for a longer period of time and they can't just disappear over night, thus they can be held accountable for their actions. Today, Microsoft, Google, Amazon and Facebook for instance, are trusted to a certain degree. But this trust is not invulnerable. Trust can be damaged by hiding important information. Keeping the public from being notified about severe data breaches and selling sensitive data to third parties without their knowledge and permission has strained the public trust in recent years.

On the other hand, we as consumers require privacy. If we can guarantee anonymity however, we can also guarantee privacy, as re-identification should be impossible at that point. As chapter 2 has shown, it is a hard task to anonymize big data sets, as quasi-identifiers can be used to reconstruct data using linking attacks. So one way to achieve anonymity, would be to strip away all attributes that could be used in another data set.

Here we hit a wall with the usability of the data itself. Cynthia Dwork says "de-identified data isn't", meaning that either that the data itself is not de-indentified because of possible reconstruction or the data is not data anymore because it is not useful for analysis. Data itself is only as useful as its relationships.

So there is a need to find a way to find a balance between trust, usability and anonymity. As trust research itself is a very broad subject, we'll focus on the anonymity of the crowd and usability of their collected data.

3.2. Different Design

As discussed before, there are advantages and disadvantages to a centralized or decentralized or distributed system. We explore how we can implement one of the choices and if they would be feasible in a theoretical point of view.

3.2.1. Simon van Endern

The chosen architecture implemented in his thesis by Simon van Endern was with a centralized approach with a distributed storage solution. The model can be seen in figure [X].

In his proposal, he eliminates the need for a central database, by collecting and storing raw data directly on the crowds's smartphone, creating a distributed database. This gives each participant a lot of control over their own data, just by deleting the application they can opt-out of future data analyses.

His original idea, the mobile phones would use peer-to-peer technology to forward the aggregation request and finally send the data to the server. But because of the lack of available technology, he opted to use the central server as an intermediary to send from mobile phone to mobile phone. To keep the data confidential and ensure anonymity from the server, he uses RSA and AES encryption. To forward the data to the next device, he implemented a polling solution. The application periodically polls the server for a new aggregation request targeted at the device. If one is present, it fetches the request over the REST API and after adding its own data posts it to the server.

In the finite scope, he managed to implement three aggregation types:

- Average number of steps of a day over all participants.
- Average time spent on an activity (walking, running, in a vehicle or on a bicycle).
- Average number of steps of a participant during the test period.

In his implementation, we see a few flaws, that we try to improve in this thesis.

3.2.2. Expanded Approach

Simon van Endern's original idea as is can't be scaled because of the nature of its forwarding chain. Adding n new devices creates a linear time complexity of $\mathcal{O}(n)$ and space complexity of $\mathcal{O}(n \log n)$.

4. Implementation

4.1. Technology Stack

```
{
  "date": 0,
  "lat": 0,
  "lon": 0,
  "radius": 0
}
```

```
{
  "type": 0,
  "start": 0,
  "end": 0,
  "lat": 0,
  "lon": 0,
  "radius": 0
}
```

```
{
  "date": 0,
  "accuracy": 0,
  "anonymity": 0,
  "lat": 0,
  "lon": 0,
  "radius": 0
}
```

```
{
  "start": 0,
  "end": 0,
  "lat": 0,
  "lon": 0,
  "radius": 1
}
```

Figure 4.1.: helloworld.cpp

4.2. Decentralized Approach

4.2.1. IPFS

4.2.2. Limitations

4.3. Centralized Approach

4.3.1. First Approach

4.3.2. Second Approach

API

NodeJS

Android

Benefits and Sacrifices

5. Performance and Evaluation

5.1. Test Environment

5.2. Performance

5.2.1. Accuracy

5.2.2. Data Consumption

5.3. Privacy Evaluation

5.3.1. Collected Data

6. Conclusion

6.1. Research Questions

6.2. Limitations

6.3. Future Work

6.3.1. Additional Information

6.3.2. Decentralization and Blockchain

6.3.3. Reproducibility

7. Introduction

Use with pdfLaTeX and Biber.

7.1. Section

Citation test (with Biber) [**latex**].

7.1.1. Subsection

See Table 7.1, Figure 7.1, Figure 7.2, Figure 7.3, Figure 7.4, Figure 7.5.

Table 7.1.: An example for a simple table.

A	B	C	D
1	2	1	2
2	3	2	3

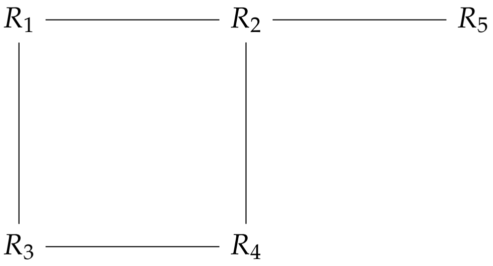

Figure 7.1.: An example for a simple drawing.

This is how the glossary will be used.

Donor dye, ex. Alexa 488 ($\mathrm{D_{dye}}$), Förster distance, Förster distance (R_0), and k_{DEAC}. Also, the TUM has many computers, not only one Computer. Subsequent acronym usage will only print the short version of Technical University of Munich (TUM) (take care of plural, if needed!), like here with TUM, too. It can also be –> hidden[1] <–.

[(Done: Nevertheless, celebrate it when it is done!)]

[1] Example for a hidden TUM glossary entry.

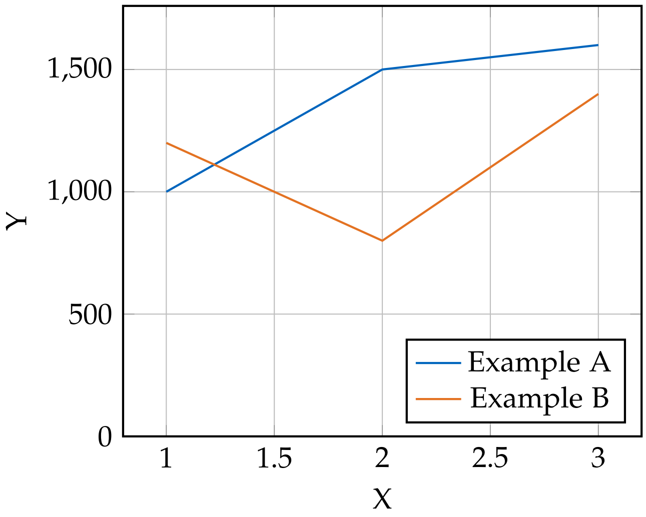

Figure 7.2.: An example for a simple plot.

```sql
SELECT * FROM tbl WHERE tbl.str = "str"
```

Figure 7.3.: An example for a source code listing.

Figure 7.4.: Includegraphics searches for the filename without extension first in logos, then in figures.

Figure 7.5.: For pictures with the same name, the direct folder needs to be chosen.

(a) The logo.

(b) The famous slide.

Figure 7.6.: Two TUM pictures side by side.

8. Second Introduction

Use with pdfLaTeX and Biber.

A. General Agenda

If there are several additions you want to add, but they do not fit into the thesis itself, they belong here.

A.1. Detailed Addition

Even sections are possible, but usually only used for several elements in, e.g. tables, images, etc.

B. Figures

B.1. Example 1

✓

B.2. Example 2

✗

List of Figures

List of Tables