Introduction

The Internet is vast, and in many ways it is an amazing piece of technology. It connects people from all over the world. It lets us do banking, watch movies, listen to music, read newspapers, receive education from home, search for information, and enjoy social interaction with all of our friends. The Internet plays a very important role in our society today. Most of what we are, what defines us as individuals, has at some point passed over the Internet. How many steps we walked today, our weight, travel schedule, our grades and achievements, our emails, our taste in music, our relationships, all of our digital conversations, our attitude towards friends, our mood day-by-day, and the entirety of our economic data. All the data needed to tell who and what we are is on the Internet.

Our information has never before been so exposed. We rely on the Internet in every-day life, many tasks carried out today would not be possible without it. And while we need to transmit sensitive data over the Internet, we also need to to maintain our personal integrity – we can not let the internet degrade us below what is decent with respect to our privacy. Already in 1890 Warren and Brandeis thought that technology was spreading information too fast for privacy to be maintained. In their work "The right to privacy" they respond to the technological advancement of the camera and the increasing proliferation of newspapers making use of photography. In this work, they also gave us the first definition of privacy, as *"right to be let alone"* [24].

> *Instantaneous photographs and newspaper enterprise have invaded the sacred precincts of private and domestic life; and numerous mechanical devices threaten to make good the prediction that "what is whispered in the closet shall be proclaimed from the house-tops."*
>
> – Warren and Brandeis

Though devices are no longer mechanical but instead digital, we still struggle with privacy in much the same way. Today we have much more data than images that we need to protect, and information spreads in a completely incomparable velocity as compared to 1890. What is the impact of cameras and newspapers next to that of smartphones and social media? Given that we have the same concerns, but the technology impact is tremendously much greater I can not help but to raise the question of whether Warren and Brandeis exaggerated the impact then, or if we do not fully appreciate the threat of the modern age.

I think we rely more heavily on personal integrity than we realize in our day-to-day life. The United Nations lists privacy as article 12 of the Universal Declaration of Human Rights by the United Nations [22]. In fact, it is easy to get the feeling that the free world is not likely to continue to function in the event of serious degradation of privacy. Our behavior is known to be more constrained if we know that we are observed [25, 8], making us less likely to speak out of turn, to deviate from the norm, to innovate, and to call out miss-behavior. I would even go so far as to say that a democratic society, as far as we see them today, would be very hard-pressed to exist in a world without privacy. If ballots are not private, can minorities really vote according to their hearts' desire?

I personally do not think there is any doubt that we need to work on maintaining privacy – not only in the short term, but for a sustainable society in the long run – but we also can not cease exposing our information to the Internet. It is therefore important that we are able to use current services on the Internet privately. We should be able to exchange messages without fear of having our privacy violated, but at the same time we need to enjoy not only the services we see today, but also the ever more complex applications of the future.

This thesis focuses on exploring the privacy of a specific piece of information – our *location*. In this work, we use a technique called *secure multi-party computation* (SMC) [26]. Using SMC, we give guarantees that when location information is sent to services on the Internet, it remains impossible for the service provider to read the data without the consent of the end user.

1 Privacy and Confidentiality

Most of this thesis talks about privacy. Privacy is a subjective matter, and what is considered private information is often different in different cultures. Computer scientists often talk about confidentiality instead, which has a more objective and precise meaning. Whether or not a system provides privacy for the user's data is not something we can prove mathematically – but whether the data is confidential is an objective fact. As an example, tax statements in Sweden are public documents. Being a swede, my privacy is therefore not violated by the fact that my income is not confidential, as this is inherently accepted in the Swedish society. We can construct a system which keeps tax statements confidential, but to construct a system which respects the users' privacy we may have to take into consideration whether they were brought up in Sweden or some other country. In this thesis, as is common in the literature within computer security, the words privacy and confidentiality are sometimes used interchangeably – when talking about the privacy of a users, we mean the confidentiality of their data.

In public media we sometimes see arguments raised against privacy, as it for instance hampers the effectiveness of law enforcement. Law enforcement may need to have surveillance on known criminals to maintain safe living conditions. However, for the scope of this thesis more privacy is always considered better than less privacy.

A large portion of the techniques proposed in academic literature to preserve privacy make use of pragmatic approaches such as simple transformations of the data to hide the most important characteristics, commonly referred to as *obfuscation.* In my opinion, these largely fall short to provide rigorous privacy guarantees. In contrast, this thesis focuses on approaches using SMC in order to protect a user's information, ensuring that the service is *unable* to intrude on privacy, rather than *unlikely* as may be the effect of obfuscation techniques.

Location Privacy Location privacy was defined as *"the ability to prevent other parties from learning one's current or past location"* by Beresford and Stajano in 2003 [2]. Later, a more precise definition was presented by Duckham and Kulik in 2006, who called it *"a special type of information privacy which concerns the claim of individuals to determine for themselves when, how, and to what extent location information about them is communicated to others"* [9]. This definition by Duckham and Kulik captures subtle nuances of location information that can be disclosed: *when*, *how* and *what.* It is reasonable that a user is more concerned about disclosure of their recent location updates than older information. How information is released is also of importance, for instance a service may allow the owner of location data to decline certain location requests rather than always automatically dispatching information about their location. Finally, what information is disclosed, and in what detail, is naturally of interest – a user may be fine with disclosing which city they are currently in, but not willing to disclose whether they are in the hospital.

Some researchers have argued that many users are willing to give away their location information [1, 6]. In my opinion this fact does not in any way lessen the importance of location privacy, it simply shows that users are keen to use the service. Since it has hitherto not been possible to easily deploy a privacy-preserving Location-Based Service (LBS), I conjecture that such approaches will be increasingly popular as they apply to more and more services.

I would argue that if privacy is inherently guaranteed by the technology, many audiences would likely be willing to provide more of their private data. Thus, these techniques open new venues for operating on private data, and enables *more private data to be collected.* They would need to trust only the technology, instead of the service provider. Using SMC, and thus allowing more private data to be used while making sure that less data is known, could give very powerful platforms in the future.

2 A Gentle Introduction to Secure Multi-party Computation

Existing cryptographic techniques have for a long time made information *unreadable* except towards intended parties [23, 27]. SMC is a separate strain of research which recently picked up much momentum and makes information *usable* while still being unreadable [18, 17, 11]. Traditional cryptography handles static information. It allows us to send and receive information privately by encrypting communication when surfing the web, reading emails and using instant messaging. It also allows us to store data privately by encrypting our hard drives.

With traditional cryptography, the data that can be decrypted is exactly the data that was encrypted. With SMC, we can encrypt some piece of data, and perform computations on the ciphertext, to decrypt something else. As an example, let us imagine three users Alice, Bob and Claire. Assume that Alice and Bob can encrypt data, but only Claire can decrypt. Alice and Bob each has a secret number a and b, respectively. Alice and Bob wants to let Claire now the sum of their numbers. To achieve this, Alice encrypts her number, say $a = 2$, sends it to Bob, who can add his number to the ciphertext, say $b = 3$. Now Claire can decrypt the number 5, while Alice and Bob retains the privacy of their secret inputs.

The above example is not a very useful application compared to most services we see on the internet, but as we will see later in the thesis, SMC can be used for real-world applications as well. This means that maintaining privacy no longer implies the necessity to remove functionality – Alice and Bob can keep a and b secret while still allowing Claire to learn the sum. We can focus on achieving the functionality of an application, rather than on telling the service provider all information needed in the necessary computation. Companies behind social networks wants to *target adds* towards you, they do not necessarily need to *know* your private data. A good example of how these cryptographic techniques can be applied is shown by Bogetoft et al. [5], where they present results from an experiment for sugar beet farmers in Denmark. The farmers sold their crops in a privacy-preserving manner with the help a cryptographic auctioning system. The sales were processed without disclosing any bids, except the final one, to any party.

The three distinct areas that currently dominate the SMC scene are *homomorphic encryption* (HE) [18, 11], *garbled circuits* (GC) [26] and *secret sharing* (SS) [21]. Homomorphic encryption schemes commonly support only computation of either multiplication or addition [17], and are then called partially homomorphic. However, *fully homomorphic encryption* (FHE) schemes also exist [11], which can compute both additions and multiplications given only ciphertexts. Unfortunately, FHE is much less efficient than HE schemes which are only additively or multiplicatively homomorphic.

Homomorphic encryption can compute arithmetic integer operations in constant time with respect to the size of the operands. Garbled circuits is often faster for more complex functions where variable size is small and fixed. There is active and accelerating research in both fields, and which solution performs best is typically application-dependent. Secret sharing normally requires an honest majority (for instance three parties with at most one is misbehaving), but is for many applications the most efficient approach to SMC. Secret sharing has been shown to be suitable for several real-life scenarios as exemplified through usages of the ShareMind Project [4, 13] where this level of trust is acceptable and efficiency is paramount.

SMC can be integrated in a system in different ways depending on which of FHE, HE, SS and GC is used. FHE is a great solution for cloud computing scenarios, as the party holding the private key does not need to be active during computations. For partially HE and GC, it is usually not the case that the party receiving the output has to do less work than if the protocol is run in the plaintexts, though it has been demonstrated that GC can be used to speed up computations for some applications [7]. For partially HE and GC, it is common that the function is computed by a party that also is providing inputs. SS is mostly suitable when the involved stakeholders in a system are fixed, and where they inherently are reluctant to collude. This could be the setting of a set of governmental institutions, for instance. When applying SMC for location privacy HE is a good choice, as geometric computations are carried out using many arithmetic operations, which is why this is the choice made for most works included this thesis.

Homomorphic Encryption Several papers in this thesis focuses on how to use HE. HE schemes are a subset of public-key encryption schemes. Public-key cryptographic systems are asymmetric, where only the holder of the private key can decrypt and anyone who holds the public key can encrypt. The private key must remain secret, while the public key can be published and considered globally known. Some other cryptographic systems are symmetric, which means that the same key is used both for encryption and decryption, but these are not important for this thesis.

HE allow for a user who does not hold the private key (and thus cannot decrypt the data) to compute functions on the ciphertexts, which have *predictable meaning* in the plaintexts. The most canonical example is school-book RSA. Given a private key k and a public key K, encryption works by exponentiation, a message m is encrypted by computing m^K, resulting in a ciphertext c. Given two ciphertexts c_1 (encrypting m_1) and c_2 (encrypting m_2), any party can compute the ciphertext $c_3 = (m_1 * m_2)^K$ by simply multiplying c_1 and c_2.

The second and third paper of this thesis uses additively homomorphic encryption. Given two ciphertexts encrypted using an additively homomorphic cryptographic system, such that c_1 is the encryption of m_1, and c_2 is the encryption of m_2, one can compute another ciphertext c_3 encrypting m_1+m_2. Using additively homomorphic encryption, it is also possible to compute the multiplication if one plaintext is known to the evaluator. To compute $c_1 \cdot c_2$ while knowing that c_2 encrypts m_2, one adds c_1 to itself m_2 times, computing $\sum_0^{m_2} c_1$.

3 Contributions

This section outlines the contributions presented later in the thesis. The thesis contains six separate papers, which all follow Alice and Bob as they try to communicate different functions of location data with different privacy guarantees.

The first paper proposes a concise privacy-preserving protocol for proximity testing, called *InnerCircle*. InnerCircle is a building block used in several of the following papers. InnerCircle only gives privacy guarantees if the attacker is honest, in the sense that they follow the protocol. This attacker model, called *semi-honest*, is a normal setting when the adversary cannot easily change the source code of the running program. The second paper provides a new primitive, called *BetterTimes*, which can be used in InnerCircle and many other protocols to allow them to tackle stronger attackers, called *malicious*, who are also able to deviate from the intended protocol flow.

Another drawback of InnerCircle is that it only considers what privacy guarantees are achieved during a single invocation of the protocol. But when included in an application, the protocol is likely to run many times. This is addressed with the *MaxPace* policy, which tackles malicious adversaries while giving privacy guarantees even as the protocol is rerun.

The fourth paper utilizes the Internet of Things (IoT) to enable stronger authentication. This is done by aggregating the location data of all devices believed to be carried by the user. While such a system may be run by any trusted third party, we also provide a privacy-preserving version utilizing similar techniques as those used in InnerCircle.

The fifth paper studies how InnerCircle as implemented to be used in an Android app compares to existing popular Android applications found on Google Play.

The last paper of the thesis considers a larger functionality than proximity testing, that of ridesharing. The paper studies what patterns users which may enjoy ridesharing may follow, and shows two separate tracks to detect ridesharing opportunities, one via an extension of InnerCircle, and the other via novel primitive we call threshold private set intersection.

3.1 Decentralized Privacy-Preserving Location Proximity

The first paper of this thesis focuses on the problem of location proximity, where principals are willing to reveal only whether they are within a certain distance from each other. The principals are privacy-sensitive, not willing to reveal any further information about their locations, nor the distance.

Privacy-sensitive location information of end users is commonly sent to the LBS in plaintext, trusting a third party is to handle principals' locations. However, due to privacy concerns it is better to avoid trusting third parties. We therefore take the road of making the data computationally unobtainable, encrypting it with a private key known only to the user whom the data concerns.

Many simple approaches to location privacy are based on obfuscating a principal's position. Such techniques often decrease the usability of the service due to the introduction of inaccurate results. These approaches lead to false positives and false negatives. In some cases over 66% of reported positives can be false [16] A major challenge to be addressed is to provide precise results without unnecessary information disclosure.

Homomorphic encryption is an apt tool to give control of location data to the principal whose location is being measured. It allows for them to encrypt the data before dispatching it to the LBS, while still enjoying the service normally. By using SMC, a user may control *what* information is disclosed even after the information has been sent away from devices controlled by the user.

InnerCircle is a privacy-preserving protocol for location proximity requiring merely one round-trip. InnerCircle allows for only the general proximity of a principal, with a radius r, to be disclosed while maintaining privacy of each principal's input. In contrast to most of the related work, we fully dispense with any third parties while maintaining a precise result, yielding no false positives or negatives at all. Further, InnerCircle benefits highly from parallelization in contrast to much previous work which gives better efficiency than existing approaches for realistic parameters.

Statement of Contribution This paper was co-authored with Martìn Ochoa and Andrei Sabelfeld. All authors contributed equally to the technical development and writing of the material.

This paper was published in the proceedings of the 13th IEEE conference on Privacy, Security and Trust (PST 2015).

3.2 Privacy-assured Outsourced Multiplications

This paper is a more theoretical, high-level contribution, presenting a system to compute any arithmetic formula in a privacy-preserving manner using an additively homomorphic encryption scheme. An arithmetic formula can be seen a directed graph where each node is an operation, each source is an input and where there is a single sink. This is illustrated in Figure 1.

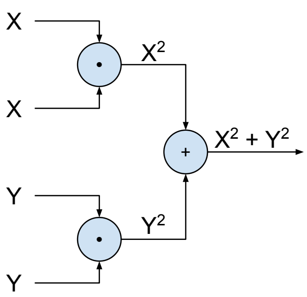

Fig. 1. An arithmetic formula computing $X^2 + Y^2$

A good example of an arithmetic formula that is frequently seen in the literature is to compute the (squared) euclidean distance between two coordinates. The technique is leveraged by a range of protocols within privacy-preserving biometric authentication and privacy-preserving LBS [10, 20, 14, 19, 28]. This can be used to compare feature vectors to find users with similar interests, to compare eigenvectors when comparing images, or for LBS.

The squared euclidean distance is shown in Equation 1 (where x_A and y_A are inputs from principal A and x_B and y_B are inputs from principal B). Recalling that Bob can add and multiply by scalar numbers, one can separate inputs from both parties, such that e.g. A can send three ciphertexts $\alpha = 2x_A, \beta = 2y_A, \gamma = x_A^2 + y_A^2$ to B to let B calculate the distance homomorphically.

$$D = x_A^2 + y_A^2 + x_B^2 + y_B^2 - (2x_A x_B + 2y_A y_B) \tag{1}$$

However, a problem arises if A decides to send three ciphertexts such that $\gamma \neq \left(\frac{\alpha}{2}\right)^2 + \left(\frac{\beta}{2}\right)^2$. In this case, A can trick B into computing another function than the euclidean distance between (x_A, y_A) and (x_B, y_B) which causes unwanted leakage of information.

The novelty in the paper is a privacy-assured multiplication protocol, called BetterTimes. Using BetterTimes, a system for arbitrary arithmetic formulas is proposed, which allows us to let Alice send (x_A, y_A) directly, instead of the three ciphertexts computed from her coordinates. This system can be applied to upgrade much existing work from being secure only against honest-but-curious adversaries to being secure in the malicious adversary model. The approach approach is evaluated using a prototypical implementation. The results show that the added overhead of our approach is small compared to insecure outsourced multiplication.

Statement of Contribution This paper was co-authored with Martìn Ochoa and Andrei Sabelfeld. All authors contributed equally to the technical development and writing of the material.

This paper was published in the proceedings of the 9th LNCS conference on Provable Security (ProvSec 2015).

3.3 Speed-Constrained Location Queries

Combining the two previous papers, we can construct a protocol to test the proximity of Alice and Bob while preserving privacy even in the case when Alice is misbehaving in any arbitrary manner (technically, she is a malicious adversary). However, when considering real-world applications of location proximity, we see that even with protections against malicious adversaries *in a single run* of the protocol, we are still vulnerable to adversaries that re-run the protocol in order to learn Bob's location. This is usually referred to as a multi-run attacker or continuous querying.

To mitigate these concerns we develop MaxPace, a general policy framework to restrict proximity queries based on the speed of the requester. We demonstrate the boost of privacy by comparative bounds on how the knowledge about the users' location changes over time. The effectiveness of the policy is illustrated in Figure 2, where and unconstrained attacker locates Bob (who's position is marked by a star) in three attempts, while an attacker under the MaxPace policy needs nine attempts.

MaxPace applies to both a centralized setting, where the server can enforce the policy on the actual locations, and a decentralized setting, dispensing with the need to reveal user locations to the service provider. The former has already found a way into practical location-based services. For the latter, we develop a protocol using techniques from both InnerCircle and BetterTimes, which also incorporates the speed constraints in its design. We formally establish the protocol's privacy guarantees and benchmark our prototype implementation to demonstrate the protocol's practical feasibility.

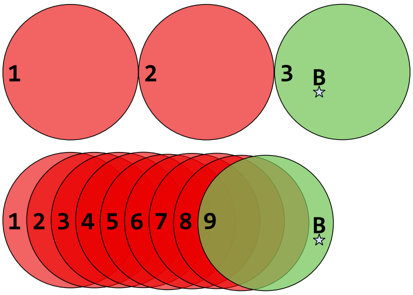

Fig. 2. Different protocols

Statement of Contribution This paper was co-authored with Martìn Ochoa and Andrei Sabelfeld. All authors contributed equally to the technical development and writing of the material.

This paper was published in the proceedings of the IEEE conference on Communications and Network Security (CNS 2016).

3.4 Location-enhanced Authentication using the IoT

User location can act as an additional factor of authentication in scenarios where physical presence is required, such as when making in-person purchases or unlocking a vehicle. This paper proposes a novel approach for estimating user location and modeling user movement using the Internet of Things (IoT). The goal is to utilize the scale and diversity of devices in the IoT to estimate the user's location robustly.

We leverage the increasing number of IoT devices carried and used by them and the smart environments that observe these devices. We also exploit the ability of many IoT devices to "sense" the user. Correct estimation of a user's location can be used to stop adversaries from using compromised user credentials (e.g., stolen keys, credit cards, passwords, etc.) in arbitrary physical locations. An example is given in Figure 3, where a user's devices are observed in their home at 8:00 AM, in a coffee shop at 8:15 AM, and where the user tries to enter their office building at 8:35 AM. The user has left their tablet device at home, but since there are more devices with the user near the office than at home, the system will detect the user's true position in the office. If instead, for instance, the credit card would be in the coffee shop, and all the user's devices in the office, a purchase should not be allowed.

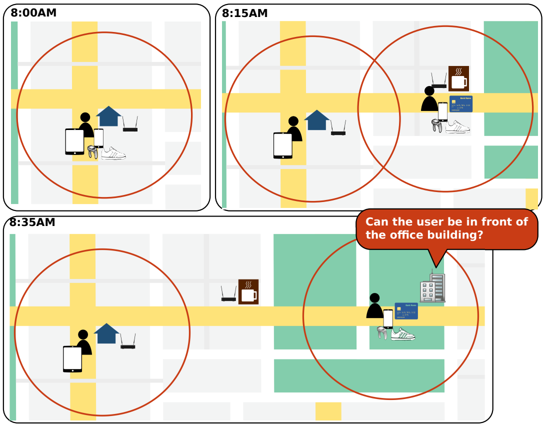

Fig. 3. An example of location-enhanced authentication

To demonstrate how effective and how efficient the approach is, a concrete system was developed, called Icelus. Experiments with Icelus shows that it exhibits a smaller false-rejection rate than for instance smartphone-based location-based authentication and it rejects attackers with few errors (i.e., false acceptances). Icelus collects location and activity data from IoT devices to model user movement and location. Icelus can run as a service on a device of the user, such as a smarthome hub, or it can be hosted in the cloud. To collect data, it organizes the various devices in a hierarchy, so that the ones with Internet connectivity can relay the data of the ones without to the system. Third-party systems can also provide data by directly connecting to Icelus or indirectly by forwarding notifications of certain events (e.g. the use of a credit card at a location, an entry in the user's calendar, etc.). To alleviate privacy concerns, we also develop a privacy-preserving extension of the protocol used in Icelus that allows us to operate purely on distances, without revealing the actual locations of individual devices. At the core of the extension is a secure multi-party computation protocol that leverages additively homomorphic encryption and blinding.

Statement of Contribution This paper was co-authored with Ioannis Agadakos, Dimitrios Damopoulos, Georgios Portokalidis and Andrei Sabelfeld. I contributed with the privacy-preserving architecture and implementation. All authors contributed equally to the writing of the material.

This paper was published in the proceedings of the 32nd Annual Conference on Computer Security Applications (ACSAC 2016).

3.5 Privacy-Preserving Location-Proximity for Mobile Apps

While, as seen above, there has been much recent progress by the research community on developing privacy-enhancing mechanisms for LBS, their evaluation has often focused on the privacy guarantees, while the question of whether these mechanisms can be adopted by practical LBS applications has received limited attention. This paper studies the applicability of privacy-preserving location proximity protocols in the setting of mobile apps. We categorize popular social location-based apps and analyze the trade-offs of privacy and functionality with respect to privacy-enhancing enhancements. To investigate the practical performance trade-offs, we present an in-depth case study of an Android application that implements InnerCircle, a state-of-the-art protocol for privacy-preserving location proximity. This study indicates that the performance of the privacy-preserving application for coarse-grained precision is comparable to real applications with the same feature set.

Statement of Contribution This paper was co-authored with Simonas Stirbys, Omar Abu Nabah and Andrei Sabelfeld. Omar, Simonas and I contributed to the technical development during the project. All authors contributed equally to the writing of the material.

This paper was published in the proceedings of the 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP 2017).

3.6 Privacy-Preserving Ridesharing

Location-based services have revolutionized transportation business, as witnessed by the success of Uber, Lyft, BlaBlaCar, and the like. From a privacy point of view, these services leave much to be desired. The location of the user is shared with the service, opening up for privacy abuse, as in some recently publicized cases [3].

To mitigate such privacy concerns in ridesharing applications, the last paper of the thesis presents PrivatePool, a model for privacy-preserving ridesharing. While primitives like proximity-testing are rather easy to define, ridesharing is a “large” concept. We focus on scenarios more aligned with car-pooling approach taken by BlaBlaCar, rather than the taxi-like structure like that of Uber. We formalize the case when two users specify an origin and a destination of a trip, and they want to find out whether they can share a ride in a privacy-preserving manner. Our resulting model is rather complex, and creating even a privacy-insensitive system to match rides in this manner proves rather impractical. Instead, we focus on two corner cases. The first is when both the origins and the destinations for the two users are close, which we call endpoint-matching. The second is when a large portion of the routes overlap, which we call intersection-based matching. Intersection-based matching can be useful in many cases even if the endpoints are very far apart. The two situations are depicted in Figure 4, where the left image shows end-point matching and on the right-hand side we can see intersection-based matching.

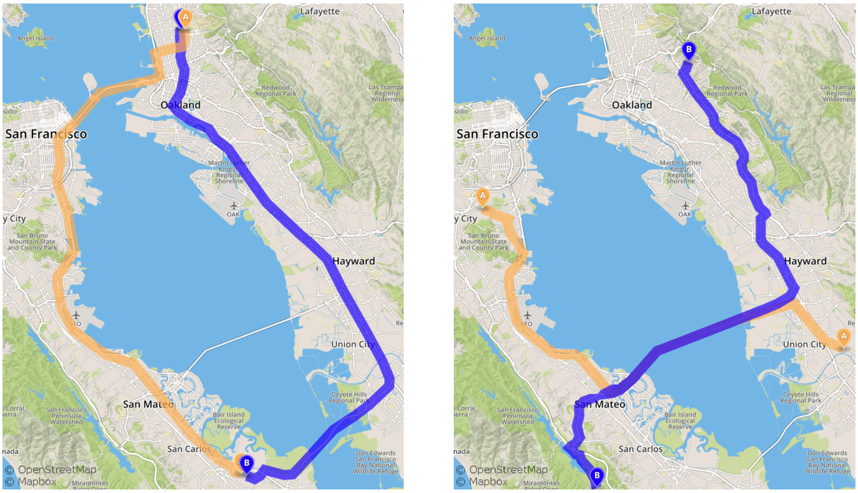
A
B
Lafayette
Angel Island
Oakland
San Francisco
Redwood Regional Park
Las Trampas Regional Wilderness
Martin Luther King Jr Regional Shoreline
OAK
y City
San Bruno Mountain State and County Park
Hayward
Union City
Coyote Hills Regional Park
San Mateo
Bair Island Ecological Reserve
San Carlos
Don Edwards San Francisco Bay National Wildlife Refuge
San Francisco Peninsula Watershed
© OpenStreetMap
© Mapbox

Fig. 4. End-point matching and route intersection

The paper presents secure multi-party computation techniques for endpoint and intersection-based matching that allow the protocols to be run without trusting third parties. At the same time, the users learn of a ride segment they can share and nothing else about other users' location. For endpoint matching, we build on InnerCircle to create an SMC protocol to detect if both of the endpoints are sufficiently close. For intersection-based matching, we created a novel cryptographic technique, called threshold key encapsulation (T-KEM). We plug T-KEM into existing solutions for privately computing the intersection of two sets, one held by each user. This fits nicely in our case, if the two sets are defined by the points traversed during their trips.

Statement of Contribution This paper was co-authored with Claudio Orlandi and Andrei Sabelfeld. All authors contributed equally to the technical development and writing of the material.

This paper will be published in the proceedings of the 30th IEEE Computer Security Foundations Symposium (CSF 2017).

4 Future work

There are still much to do in the interest of achieving better location privacy. While many of the cryptographic techniques used in this thesis can be called practical out-of-context, a fully-fledged system built on top of SMC needs more work before it is useful in practice.

On the one hand, practically-oriented research results are needed to show how to use SMC in general without information leakage from a running system. As SMC works on the application layer, it is oblivious to information on other levels such as IP addresses etc. Each user currently needs to trust the environment which is running the application both on their machine and on the machine which they are communicating with, and they need to trust the service that is distributing the binary for the application they are running.

On the other hand, further foundational work is needed in terms of scalability to large numbers of users. The state of the art is making great leaps in this direction, such that is now possible to efficiently compute a fixed function of many users [15]. However, state-of-the art SMC protocols are only efficient for a limited number of users for cases like ours where each party wants to evaluate a different function, as the question "who can *I* share a ride with?" is context-sensitive.

On another note, I also feel there is a need for more work from the software-engineering community. It is very different to debug an application with and without access to the values stored in each variable. We need to adapt processes and tools, to discuss things such as logging, backup, and many other issues which are more-or-less solved in the traditional setting also in the encrypted domain.

5 Conclusions

The thesis presents usages of SMC within location-based services and an augmentation of additively homomorphic encryption to add a privacy-guaranteed multiplication functionality. All of these serve to some extent in giving the far end of the information exchange more control over data disclosure, moving away from centralized architectures relying on trust and achieving a higher level of privacy. Trust in third parties is through these techniques reduced, and the owner of data can be more confident that it is handled and used as they intend.

References

1. Ahern, S., Eckles, D., Good, N., King, S., Naaman, M., and Nair, R. Over-exposed?: privacy patterns and considerations in online and mobile photo sharing. In *CHI* (2007), M. B. Rosson and D. J. Gilmore, Eds., ACM, pp. 357–366.
2. Beresford, A. R., and Stajano, F. Location privacy in pervasive computing. *IEEE Pervasive Computing 2*, 1 (2003), 46–55.
3. Bessette, C. Does Uber Even Deserve Our Trust? http://www.forbes.com/sites/chanellebessette/2014/11/25/does-uber-even-deserve-our-trust/, Nov. 2014.
4. Bogdanov, D., Laur, S., and Willemson, J. Sharemind: A framework for fast privacy-preserving computations. In *Computer Security - ESORICS 2008, 13th European Symposium on Research in Computer Security, Málaga, Spain, October 6-8, 2008. Proceedings* (2008), S. Jajodia and J. López, Eds., vol. 5283 of *Lecture Notes in Computer Science*, Springer, pp. 192–206.
5. Bogetoft, P., Christensen, D. L., Damgård, I., Geisler, M., Jakobsen, T. P., Krøigaard, M., Nielsen, J. D., Nielsen, J. B., Nielsen, K., Pagter, J., Schwartzbach, M. I., and Toft, T. Secure multiparty computation goes live. In *Financial Cryptography* (2009), R. Dingledine and P. Golle, Eds., vol. 5628 of *Lecture Notes in Computer Science*, Springer, pp. 325–343.
6. Brush, A. J. B., Krumm, J., and Scott, J. Exploring end user preferences for location obfuscation, location-based services, and the value of location. In *UbiComp* (2010), J. E. Bardram, M. Langheinrich, K. N. Truong, and P. Nixon, Eds., ACM International Conference Proceeding Series, ACM, pp. 95–104.
7. Carter, H., Mood, B., Traynor, P., and Butler, K. R. B. Secure outsourced garbled circuit evaluation for mobile devices. In *USENIX Security* (2013), S. T. King, Ed., USENIX Association, pp. 289–304.
8. Diener, E., Fraser, S. C., Beaman, A. L., and Kelem, R. T. Effects of deindividuation variables on stealing among halloween trick-or-treaters. *Journal of personality and social psychology 33*, 2 (1976), 178.

9. DUCKHAM, M., AND KULIK, L. Location privacy and location-aware computing. *Dynamic & mobile GIS: investigating change in space and time 3* (2006), 35–61.
10. ERKIN, Z., FRANZ, M., GUAJARDO, J., KATZENBEISSER, S., LAGENDIJK, I., AND TOFT, T. Privacy-preserving face recognition. In *Privacy Enhancing Technologies* (2009), I. Goldberg and M. J. Atallah, Eds., vol. 5672 of *Lecture Notes in Computer Science*, Springer, pp. 235–253.
11. GENTRY, C. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009* (2009), M. Mitzenmacher, Ed., ACM, pp. 169–178.
12. GHORBANI, A. A., TORRA, V., HISIL, H., MIRI, A., KOLTUKSUZ, A., ZHANG, J., SENSOY, M., GARCÍA-ALFARO, J., AND ZINCIR, I., Eds. *13th Annual Conference on Privacy, Security and Trust, PST 2015, Izmir, Turkey, July 21-23, 2015* (2015), IEEE.
13. GUROV, D., LAUD, P., AND GUANCIALE, R. Privacy preserving business process matching. In Ghorbani et al. [12], pp. 36–43.
14. HALLGREN, P. A., OCHOA, M., AND SABELFELD, A. Innercircle: A parallelizable decentralized privacy-preserving location proximity protocol. In Ghorbani et al. [12], pp. 1–6.
15. KERSCHBAUM, F. Adapting privacy-preserving computation to the service provider model. In *Proceedings of the 12th IEEE International Conference on Computational Science and Engineering, CSE 2009, Vancouver, BC, Canada, August 29-31, 2009* (2009), IEEE Computer Society, pp. 34–41.
16. MASCETTI, S., FRENI, D., BETTINI, C., WANG, X. S., AND JAJODIA, S. Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies. *VLDB J. 20*, 4 (2011), 541–566.
17. PAILLIER, P. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding* (1999), J. Stern, Ed., vol. 1592 of *Lecture Notes in Computer Science*, Springer, pp. 223–238.
18. RIVEST, R. L., ADLEMAN, L., AND DERTOUZOS, M. L. On data banks and privacy homomorphisms. *Foundations of secure computation 32*, 4 (1978), 169–178.
19. SADEGHI, A.-R., SCHNEIDER, T., AND WEHRENBERG, I. Efficient privacy-preserving face recognition. In *ICISC* (2009), D. Lee and S. Hong, Eds., vol. 5984 of *Lecture Notes in Computer Science*, Springer, pp. 229–244.
20. SEDENKA, J., AND GASTI, P. Privacy-preserving distance computation and proximity testing on earth, done right. In *ASIACCS* (2014), S. Moriai, T. Jaeger, and K. Sakurai, Eds., ACM, pp. 99–110.
21. SHAMIR, A. How to share a secret. *Commun. ACM 22*, 11 (1979), 612–613.
22. Universal declaration of human rights, Dec. 1948. Available: http://www.ohchr.org/EN/UDHR/Documents/UDHR_Translations/eng.pdf (accessed 2017-07-06).
23. TURNER, S., AND POLK, T. Prohibiting Secure Sockets Layer (SSL) Version 2.0. RFC 6176 (Proposed Standard), Mar. 2011.

24. Warren, S. D., and Brandeis, L. D. The right to privacy. *Harward Law Review 4*, 5 (December 1890), 193–220.

25. Webb, E. J., Campbell, D. T., Schwartz, R. D., and Sechrest, L. *Unobtrusive measures: Nonreactive research in the social sciences*, vol. 111. Rand McNally Chicago, 1966.

26. Yao, A. C. Protocols for secure computations (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982* (1982), IEEE Computer Society, pp. 160–164.

27. Ylonen, T., and Lonvick, C. The Secure Shell (SSH) Protocol Architecture. RFC 4251 (Proposed Standard), Jan. 2006.

28. Zhong, G., Goldberg, I., and Hengartner, U. Louis, lester and pierre: Three protocols for location privacy. In *Privacy Enhancing Technologies, 7th International Symposium, PET 2007 Ottawa, Canada, June 20-22, 2007, Revised Selected Papers* (2007), N. Borisov and P. Golle, Eds., vol. 4776 of *Lecture Notes in Computer Science*, Springer, pp. 62–76.