A survey of privacy and secure computational protocols

IAN MARTINY, University of Colorado Boulder

People are now beginning to focus on their privacy. This has lead to people being more aware of the tools they use and what data they allow third-parties access to. People are still interested in the services being provided but wish to keep their sensitive data private. There are tools that allow this type of privacy ranging from anonymous communication tools to secure computations in the cloud which use cryptographic techniques to leverage private information to complete computations. In this paper we examine the different methods that tools can use to preserve anonymity. In particular we look at anonymous communication, private information retrieval (PIR), zero knowledge proofs, secure computation, and differential privacy.

Additional Key Words and Phrases: Zero-knowledge Proofs, differential privacy, multiparty computation, secure computation

ACM Reference Format:
Ian Martiny. 2018. A survey of privacy and secure computational protocols. 1, 1 (October 2018), 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Recently individuals are more aware of their private data, specifically when used by third-parties. In light of the Cambridge Analytica scandal with Facebook [7, 13] users are more aware of the private data they are giving to third parties, and are beginning to realize the effect their data can have.

However, users still value certain tools and services that require data. To reconcile this contradiction various projects have worked on methods of delivering results on private data.

In this paper we analyze various techniques which are commonly used in tools with privacy in mind. We examine tools which allow for private contact discovery [3, 4], private analysis of data [6], private computation of data [2], anonymous communication [14], and verifiable private computation [1, 11]. Additionally, we review the concepts of secure computation [8, 15] and differential privacy [5].

The rest of this paper is structured as follows: Section 2 will discuss tools which focus on privacy and anonymity, Section **??** will discuss the notion of secure computation on private data, as well as examine the notion of verifiable computation in private settings, Section **??** will explore the concept of differential privacy, and finally Section **??** will conclude.

2 PRIVACY

In this section we analyze some of the tools which focus on privacy and anonymity for average users. Specifically we examine the problems of contact discovery (how to users of a service discovery whether contacts also use a service), presence discovery (whether users are currently online), and anonymous communication.

Author's address: Ian Martiny, University of Colorado Boulder, Boulder, CO, ian.martiny@colorado.edu.

Manuscript submitted to ACM

2.1 Private Information Retrieval

We briefly introduce the notion of Private Information Retrieval (PIR) as it crops up in many of the future topics. There are two main classes of PIR, Information Theoretic (IT-PIR) and Computational (C-PIR).

2.2 Contact discovery

Contact discovery is the process of clients of a service discovering whether any of their contacts also use the same service.

In non-private settings this is very simple to achieve, the client provides contact details to the service which then queries its database and returns who is available on the service. This method requires each client to reveal their whole contact list to the service. Allowing the service to track metadata on every client, as well as leaking non-clients phone numbers.

Signal systems has gone through the process of trying to handle contact discovery in a private way [9]. As a text messaging service offering end-to-end encryption it is necessary for Signal to be aware of which contact are also Signal clients. They address the difficulties of contact discovery, a natural first attempt is to simply hash contacts and send them to the service to verify hashes. Unfortunately the pre-image space of these hashes is small (there are only 10 billion possible phone numbers, many of which are not valid, due to area codes) which allows to off-line attacks by the service after receiving a client's contact list. Other possible options for contact discovery exist including using Private Information Retrieval (PIR), bloom filters, and Private Set Intersection (PSI), though these solutions cause services to make trade-offs due to either large overhead in communication sizes or in computation costs. Signal has recently implemented a more private contact discovery method leveraging Intel's Software Guard Extensions (SGX) [10]. The high-level implementation is that clients make a secure connection to the Signal server and perform a remote attestation that the code they are running is the published open-source code, and then use SGX's trusted enclave to perform an intersection where the server is sent an encrypted form of the clients contacts (which are only decrypted and readable in the trusted enclave).

However, recent attacks on SGX [12] have demonstrated the the trusted enclave should not be so trusted. This leads to the natural question of "are there better methods of contact discovery?".

As stated above, the use of PIR or PSI as a method of private contact discovery has large overhead which is not usable for large services. However, recent work [4] has combined these two concepts to lessen the overhead. Using a distributed point function, clients can generate an information theoretic private information retrieval (IT-PIR) request from 2-4 separate servers, afterward this request is completed by using a Private Equality Test (PEQ) between the client's contact information and the database query. This method scales better than previous methods in that a client with 1024 contacts can perform private contact discovery with a server of 67 million clients in 1.36 seconds and 4.28MiB of communication.

REFERENCES

[1] Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, Eran Tromer, and Madars Virza. 2013. SNARKs for C: Verifying program executions succinctly and in zero knowledge. In *Advances in Cryptology–CRYPTO 2013*. Springer, 90–108.

[2] Peter Bogetoft, Dan Lund Christensen, Ivan Damgård, Martin Geisler, Thomas Jakobsen, Mikkel Krøigaard, Janus Dam Nielsen, Jesper Buus Nielsen, Kurt Nielsen, Jakob Pagter, et al. 2009. Secure multiparty computation goes live. In *International Conference on Financial Cryptography and Data Security*. Springer, 325–343.

[3] Nikita Borisov, George Danezis, and Ian Goldberg. 2015. DP5: A private presence service. *Proceedings on Privacy Enhancing Technologies* 2015, 2 (2015), 4–24.

[4] Daniel Demmler, Peter Rindal, Mike Rosulek, and Ni Trieu. 2018. PIR-PSI: Scaling Private Contact Discovery. (2018).

[5] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*. Springer, 1–19.

[6] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. ACM, 1054–1067.

[7] Kevin Granville. 2018. Facebook and Cambridge Analytica: What You Need to Know as Fallout Widens. *The New York Times* (Mar 2018). https://www.nytimes.com/2018/03/19/technology/facebook-cambridge-analytica-explained.html

[8] Yehida Lindell. 2005. Secure multiparty computation for privacy preserving data mining. In *Encyclopedia of Data Warehousing and Mining*. IGI Global, 1005–1009.

[9] Moxie Marlinspike. 2014. The Difficulty Of Private Contact Discovery. *Signal Blog* (Jan 2014). https://signal.org/blog/contact-discovery/

[10] Moxie Marlinspike. 2017. Technology preview: Private contact discovery for Signal. *Signal Blog* (Sep 2017). https://signal.org/blog/private-contact-discovery/

[11] Bryan Parno, Jon Howell, Craig Gentry, and Mariana Raykova. 2013. Pinocchio: Nearly practical verifiable computation. In *2013 IEEE Symposium on Security and Privacy*. IEEE, 238–252.

[12] Jo Van Bulck, Marina Minkin, Ofir Weisse, Daniel Genkin, Baris Kasikci, Frank Piessens, Mark Silberstein, Thomas F Wenisch, Yuval Yarom, and Raoul Strackx. 2018. Foreshadow: Extracting the keys to the Intel SGX kingdom with transient out-of-order execution. In *Proceedings of the 27th USENIX Security Symposium. USENIX Association.*

[13] Chris Welch. 2018. More people are taking Facebook breaks and deleting the app from their phones. *The Verge* (Sep 2018). https://www.theverge.com/2018/9/5/17822736/facebook-break-delete-app

[14] David Isaac Wolinsky, Henry Corrigan-Gibbs, Bryan Ford, and Aaron Johnson. 2012. Dissent in Numbers: Making Strong Anonymity Scale.. In *OSDI*. 179–182.

[15] Andrew C Yao. 1982. Protocols for secure computations. In *Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on*. IEEE, 160–164.