Tracking the Invisible: Privacy-Preserving Contact Tracing to Control the Spread of a Virus

Didem Demirag[1](✉) and Erman Ayday[2,3]

[1] Concordia University, Montreal, QC, Canada
d_demira@encs.concordia.ca
[2] Case Western Reserve University, Cleveland, OH, USA
exa208@case.edu
[3] Bilkent University, Ankara, Turkey

Abstract. Today, tracking and controlling the spread of a virus is a crucial need for almost all countries. Doing this early would save millions of lives and help countries keep a stable economy. The easiest way to control the spread of a virus is to immediately inform the individuals who recently had close contact with the diagnosed patients. However, to achieve this, a centralized authority (e.g., a health authority) needs detailed location information from both healthy individuals and diagnosed patients. Thus, such an approach, although beneficial to control the spread of a virus, results in serious privacy concerns, and hence privacy-preserving solutions are required to solve this problem. Previous works on this topic either (i) compromise privacy (especially privacy of diagnosed patients) to have better efficiency or (ii) provide unscalable solutions. In this work, we propose a technique based on private set intersection between physical contact histories of individuals (that are recorded using smart phones) and a centralized database (run by a health authority) that keeps the identities of the positively diagnosed patients for the disease. Proposed solution protects the location privacy of both healthy individuals and diagnosed patients and it guarantees that the identities of the diagnosed patients remain hidden from other individuals. Notably, proposed scheme allows individuals to receive warning messages indicating their previous contacts with a positively diagnosed patient. Such warning messages will help them realize the risk and isolate themselves from other people. We make sure that the warning messages are only observed by the corresponding individuals and not by the health authority. We also implement the proposed scheme and show its efficiency and scalability via simulations.

1 Introduction

A pandemic, which typically occurs due to uncontrollable spread of a virus, is a major threat for the mankind. It may have serious consequences including

An extended version of this work is available at: https://arxiv.org/pdf/2003.13073v2.pdf.

© Springer Nature Switzerland AG 2020
J. Garcia-Alfaro et al. (Eds.): DPM 2020/CBT 2020, LNCS 12484, pp. 240–249, 2020.
https://doi.org/10.1007/978-3-030-66172-4_15

people losing their lives and economical devastation for countries. To decrease the severity of such consequences, it is crucial for countries to track the spread of a virus before it becomes widespread.

Main threat during such a spread is the individuals that had close contact with the carriers of the disease (i.e., people carrying the virus before they are diagnosed with the disease or before they start showing symptoms). Thus, it is very beneficial to identify and warn individuals that were in close contact with a carrier right after the carrier is diagnosed. If a country can identify who had close contact with the already diagnosed patients, by sending warnings to its citizens and telling them to self-quarantine themselves, the spread of the virus can be controlled. By doing so, individuals that receive such warnings (that they had close contact with one or more diagnosed patients) can take self-measures immediately. This is also economically preferred instead of completely shutting down a country.

However, implementing such an approach is not trivial due to privacy reasons. First of all, due to patient confidentiality, identities of diagnosed patients cannot be shared with other individuals. Similarly, healthy individuals do not want to share sensitive information about themselves (e.g., their whereabouts) with the authorities of the country.

In this work, we propose a privacy-preserving technique that allows individuals receive warnings if they have been in close proximity of diagnosed patients in the past few weeks (that is determined based on the incubation period of the virus). We propose keeping the (physical) contact histories of individuals by using communication protocols in their smart phones. These contact histories are then used to determine if an individual was in close contact with a diagnosed patient in the past few weeks, and if so, the individual receives a warning. In order to do this in a privacy-preserving way, the proposed system uses private set intersection (PSI) on the background as the cryptographic building block (between the local contact histories of the individuals and database keeping the identities of the diagnosed patients). Even though PSI consumes more resources, it provides more privacy for the involving parties. By utilizing PSI, we aim to mitigate the privacy vulnerabilities of existing schemes.

The proposed scheme guarantees that (i) identities or the whereabouts of the diagnosed patients are not revealed to any other individuals, (ii) contact histories of the individuals are not shared with any other parties, (iii) warning received by an individual (saying that they were in close proximity of a diagnosed patient) is only observed by the corresponding individual and no one else, and (iv) the individual that receives a warning can anonymously share their demographics with the healthcare officials only if they want to. Furthermore, we also propose an extension of the proposed scheme against malicious individuals that may try to tamper their local contact histories in order to learn the diagnosis of some target individuals. We also implement and evaluate the proposed technique to show its efficiency and practicality.

The rest of the paper is organized as follows. In the next section, we summarize the related work. In Sect. 3, we describe the proposed solution in detail.

In Sect. 4, we implement and evaluate the proposed scheme. Finally, in Sect. 5, we conclude the paper.

2 Related Work

The importance of disease surveillance (without considering the privacy) has been studied by [4,5]. Privacy considerations in existing contact tracing mobile apps for COVID-19 are addressed in [6]. Authors show that none of the existing apps and schemes (except for private messaging systems which lack scalability) can protect the privacy of diagnosed patients and other exposed individuals at the same time.

Most of the current proposed systems are Bluetooth-based. Covid Watch [2] is an open source project that relies on Bluetooth for contact tracing and also uses anonymized GPS data to detect high-risk areas. Furthermore, Epione [13] is a PSI-CA based system, where a new semi-honest PSI-CA primitive for asymmetric sets is used. While Epione is very similar to our proposed work, we published our initial idea [8] on arxiv simultaneously with Epione.

A privacy-preserving Bluetooth protocol for contact tracing is proposed in [3]. Users receive the information about diagnosed people from a diagnosis server and the matching is done locally on the user's device. In a similar system proposed in [14], users keep their local contact histories by broadcasting their ephemeral, pseudo-random IDs from their smart phones and recording the IDs of other users that are in close proximity. The risk of a person for contracting the disease is computed locally on their phone. However, this system is not robust against a malicious user that may try to identify the infected individuals by observing (or modifying) their contact history.

Some apps, including [10,11] track individuals' location and save it in a local database. However, such an approach compromises location privacy of diagnosed patients as location information is gathered by the health authorities. As opposed to these approaches, in this work, we propose a scheme that protects the privacy of diagnosed individuals as well as the healthy ones.

Some other works incorporate different building blocks as well. In [12], a secure multiparty computation-based approach is proposed. Another solution utilizes trusted execution environment (TEE) [1]. Here, users share encrypted location history and also the test status. The system uses private computation to determine the people who were in contact with a diagnosed person. Finally, in [15], authors introduce BeepTrace, which is a blockchain-enabled approach.

3 Proposed Solution

In this section, we first introduce our system and threat models and then, we describe the proposed solution in detail.

3.1 System Model

The proposed system includes healthy (or not yet diagnosed) individuals with smart phones, diagnosed patients for the disease, and a health authority (e.g., ministry of health or NIH). Individuals interact with the database of the health authority. In the following, we will describe the proposed scheme assuming a single database for the health authority. For the sake of generality, one can also assume multiple local databases for the health authority (e.g., located in different geographical regions).

The health authority keeps the identities of the diagnosed patients and it does not want this information to be learnt by other parties. Individuals keep their local (physical) contact histories in their smart phones and they do not want this information to be observed by other parties (including the health authority). Also, when an individual receives a warning about a contact with a diagnosed patient, the individual wants to make sure that no other party can observe this warning.

3.2 Threat Model

We consider a semi-honest attacker model for the parties that involve in the protocol. That is, each party in the system follows the protocol honestly but they may be curious to learn sensitive information of the other parties. On the other hand, individuals may try to learn the identities of diagnosed patients or the health authority may try to learn the contact histories of the individuals. As we will discuss in detail later, the proposed algorithm protects the parties against these threats. Finally, we assume all communications between parties (between smart phones of two individuals or between an individual and the database of the health authority) are encrypted, and hence robust against eavesdroppers.

In the following, we provide a list of possible attacks against the proposed system.

1. A curious user trying to infer contact information or diagnosis belonging to a target user.
2. A curious user trying to introduce fake contacts in the contact list,
3. The server colluding with a curious user to infer contact information or diagnosis belonging to another target user.

3.3 Keeping the Contact History at Local Devices

Each individual keeps a vector in their local smart phone for their physical contact history. When an individual A spends some amount of time within the close proximity of an individual B, their phone records the ID of person B. For IDs of the individuals, we propose using the hash of their User ID (UID), which is generated by the application. To measure the proximity between the individuals, we propose using the Bluetooth signals on their devices. Thus, to add individual A as a contact, B needs to spend at least t seconds within r radius of A. This

process is also illustrated in Fig. 1. As a result of this interaction, the new record in the contact history of A is the ID of B. Each individual may also separately keep the time and the duration of the contact (to have more insight about their risk, as will be discussed later).

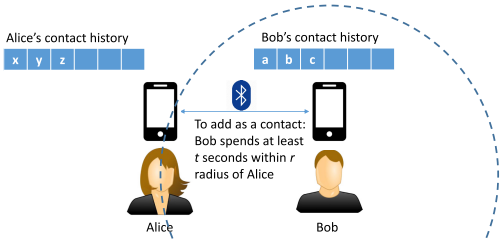

Fig. 1. Keeping and updating local contact histories of the individuals.

It has been shown that the strengths of Bluetooth signals can be used to approximate the distance between two devices [9,16]. Alternatively, one can also use (i) just the Bluetooth coverage (e.g., when two devices are in the range of each other for more than t seconds, they can update their local contact histories with each others' IDs), (ii) GPS information (when two devices are within their Bluetooth coverage, they can exchange GPS information to measure their distance more accurately and update their local contacts if they spend more than t seconds within a close range of each other), or (iii) NFC signal coverage (when the devices are within NFC signal coverage of each other, which is about 3 feet maximum, for more than t seconds, they can update their local contact histories with each others' IDs). Since this part is not the main contribution of the paper, we do not go into the details of establishing the contact histories.

It is important to make sure that an individual cannot add a contact in their contact list aiming to learn whether a target person has positive diagnosis or not. For instance, knowing the UID of a target person, an attacker may construct its local contact list only from the ID of that target, and hence learn the diagnosis of the target. To prevent such an attack, we consider two options: (i) make sure the local contact histories of individuals are stored in such a way that data cannot be accessed or modified by the individuals (e.g., the local contact history can be encrypted in the device by the key of the health authority or the contact history can be stored in a storage for which the individual does not have read/write permission). Or, (ii) each new contact B of an individual A also includes a digital signature that is signed by a centralized authority (e.g., the telecom operator). To do so, if individuals A and B spend a certain amount of time within close proximity of each other (measured as discussed before), they both send the contact request to the operator, the operator signs and sends back the signed contact record to both parties, and each party keeps the contact records and the corresponding signature together. This way, an attacker cannot fake new contacts in its local contact history. The validity of these signatures are

then verified when the local contact history of an individual is compared with the diagnosed patients in the health authority's database.

Using either of these techniques, the developed algorithm makes sure that the contact history cannot be tampered by the individual. Note that if the storage of the local contact list would be an overhead, it is also possible to store the local contacts of an individual at a cloud server (encrypted by individual's key) and update the local contacts periodically. Current Bluetooth based solutions, such as [3] do not let the users access their contact histories (that is stored on their mobile devices). Even if the user can access their contact history, tampering with this list can be avoided by using digital signatures and encryption, as explained in this section.

3.4 Keeping the IDs of Diagnosed Patients at a Centralized Database

When an individual is diagnosed (e.g., by a hospital) with the disease, the User ID (UID) of the positively diagnosed patient is stored in the database of the health authority (e.g., ministry of health), as shown in Fig. 2. It is important to note that only the hospital and the health authority know the ID of the diagnosed individual.

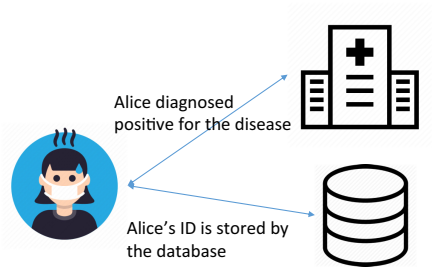

Fig. 2. Updating the database of the health authority with the IDs of the diagnosed patients.

3.5 Private Set Intersection to Identify the Individuals at Risk

The application on an individual's smart phone sends queries to the health authority's database following a random schedule. This schedule can be determined by the system to avoid an overload to the database. It is also important not to allow the individual to send queries at any time in order to control the system's bandwidth.

An individual A uses their contact history to query the database of the health authority and the goal is to identify whether there is an intersection between the local contact history of the individual and the IDs of the diagnosed patients

in the authority's database (as shown in Fig. 3). Size of this intersection reveals the number of diagnosed people with whom A had been in close proximity in the previous a few weeks. As the size of the intersection increases, the risk of individual A being infected also increases. The proposed algorithm provides the result of this intersection to individual A as a warning message. Using the warning, the individual may take early precautions (e.g., have a test or quarantine themselves).

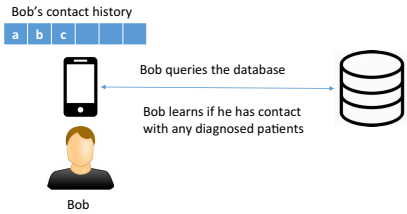

Fig. 3. Privacy-preserving interaction between an individual and the database of the health authority.

To compute this intersection in a privacy-preserving way, we use the private set intersection cardinality (PSI-CA) protocol, in which parties that are involved in the protocol obfuscate their inputs (sensitive information) and compute the result of this intersection. Eventually, only individual A learns the result of the intersection and the health authority does not learn any information about the contact history of individual A or the result of the intersection. We also make sure that individual A does not learn anything about the database content of the health authority (e.g., IDs of diagnosed patients).

Here, we explain the details of the proposed PSI-CA based protocol between an individual (client) and the health authority (server). As input to the protocol, client has its local contact list and server has the list of positively diagnosed patients.

Client masks its input with the random exponent $R_c^{'}$ and obtains the list of a_i-s and computes $X = g^{R_c}$ (X is similar to an ElGamal public key). Client sends the list of a_i values and X to the server.

Server permutes its input list and applies $H(.)$ on the list. Server masks a_i values with its random exponent $R_s^{'}$, shuffles the resulting list, and computes $Y = g^{R_s}$, which is a public-key like value. Server creates the list of ts_j-s by applying the one-way function $H(.)$ over the multiplication of X^{R_s} and exponentiation of hs_j-s to random value $R_s^{'}$. Server sends shuffled and masked a_i values, Y, and ts_j-s to the client.

As the last step, client does the matching between the list of ts_j-s that it received from the server and its own list of tc_i-s. tc_i-s are obtained by applying the one-way function $H(.)$ over the multiplication of Y^{R_s} and the shuffled a_i-s, which are stripped of the random value $R_c^{'}$. At the end of the protocol, client

only learns the cardinality of the intersection. At the last step, a notification is generated based on the output of PSI-CA.

3.6 Further Steps to Track the Spread

Once an individual receives a warning as a result of the proposed algorithm, they can choose to (i) provide (anonymous) information back to the health authority to help the authority to track the spread and/or (ii) share their local contact history with the health authority to get further information about their risk.

In (i), the individual shares their demographics, the size of intersection they obtain as a result of the proposed algorithm, and their location with the health authority. Using such information received from different individuals, the health authority can have a clear idea about how the virus spreads in the population. In (ii), the health authority, using the local contact history of the individual and further details about the contacts of individual (duration and location of each contact), can provide a more detailed risk information to the individual (e.g., if the duration of the contact with a diagnosed patient is long, then the risk of individual also increases). As discussed before, such contact details (i.e., duration or location) are not used in the proposed privacy-preserving algorithm; they can be collected and kept by the individual and may be shared with the health authority to get more insight about the risk.

4 Evaluation

We implemented and evaluated the proposed privacy-preserving search algorithm. We ran our experiments on macOS High Sierra, 2.3 GHz Intel Core i5, 8 GB RAM, and 256 GB hard disk. We used MD5 as the hash function to hash the UIDs of individuals in the local contact lists and in the health authority's database. We used the implementation of PSI-CA in [7], in which q and p are 160 and 1024 bits, respectively. We ran each experiment for 20 times and reported the average.

We show the results of the evaluation of PSI-based solution in Tables 1 and 2. In Table 1, we set the size of client's local contact list to $1,000$ and vary the size of server's database. In Table 2, we set the size of server's database to $100,000$ and vary the size of client's local contact list. Our results show that the online phase of the protocol can be efficiently completed by the parties even when the input sizes of both parties are significantly large. Note that the server does not need to run the offline part of the algorithm for each client separately. Instead, the server can use the same offline computation during its interaction with every client. Also, a client can conduct its offline steps as it generates its local contact list.

Complexity of the proposed algorithm is linear in the size of the two sets. Let the cardinality of server's set be w and client's set be v. Client performs $2(v+1)$ exponentiations with $|q|$-bit (short) exponents modulo $|p|$-bit and v modular multiplications. Server performs $(v+w)$ modular exponentiations with

Table 1. Offline and online run-times for PSI-based protocol (in milliseconds) at the client (individual) and server (health authority) with varying size for server's database. Size of client's contact list is set to 1,000.

Size of server's database	Offline time (ms)	Online time (ms)
1,000	Client: 210.6	Client: 100.85
	Server: 388.05	Server: 107.5
10,000	Client: 201.2	Client: 978.7
	Server: 2213.5	Server: 1003.8
100,000	Client: 202.95	Client: 9766.1
	Server: 20054.1	Server: 9925.6
1,000,000	Client: 202.4	Client: 96685.8
	Server: 194289.6	Server: 98631.1

Table 2. Offline and online run-times for PSI-based protocol (in milliseconds) at the client (individual) and server (health authority) with varying size for client's local contact list. Size of server's database is set to 100,000.

Size of client's contact list	Offline time (ms)	Online time (ms)
10	Client: 2.7	Client: 9852.95
	Server: 20012.75	Server: 9560.8
100	Client: 22.6	Client: 9854.2
	Server: 20218.1	Server: 9968.65
1,000	Client: 202.3	Client: 9817.5
	Server: 20448	Server: 9979.75
10,000	Client: 1990.4	Client: 9787.45
	Server: 20246.45	Server: 9970.65

short exponents and w modular multiplications. The resulting communication overhead is $2(v+1)\ |p|$-bit and w κ-bit values, where κ is the security parameter. It can be deduced from these results that the protocol does not incur a significant overhead for a smart phone. Security and correctness of the proposed algorithm depends on the security and correctness of the original PSI-CA algorithm. We refer to [7] for details.

5 Conclusion

In this paper, we have proposed a privacy-preserving technique to control the spread of a virus in a population. The proposed technique is based on private set intersection between physical contact histories of individuals (that are recorded using smart phones) and a centralized database (run by a health authority) that keeps the identities of the positively diagnosed patients for the disease. We have shown that individuals can receive warning messages indicating their previous contacts with a positively diagnosed patient as a result of the proposed technique. While doing so, neither of the parties that involve in the protocol obtain any sensitive information about each other. We believe that the proposed scheme can efficiently help countries control the spread of a virus in a privacy-preserving way, without violating privacy of their citizens.

References

1. COVID-19 self-reporting with privacy (2020). https://github.com/enigmampc/safetrace. Accessed 11 Apr 2020
2. COVID watch (2020). https://www.covid-watch.org/article. Accessed 11 Apr 2020
3. Privacy-preserving contact tracing (2020). https://www.apple.com/covid19/contacttracing. Accessed 10 Apr 2020

4. Bhatia, S., et al.: Big brother is watching-using digital disease surveillance tools for near real-time forecasting. Int. J. Infect. Dis. **79**, 27 (2019)
5. Carneiro, H.A., Mylonakis, E.: Google trends: a web-based tool for real-time surveillance of disease outbreaks. Clin. Infect. Dis. **49**(10), 1557–1564 (2009)
6. Cho, H., Ippolito, D., Yu, Y.W.: Contact tracing mobile apps for COVID-19: privacy considerations and related trade-offs. arXiv preprint arXiv:2003.11511 (2020)
7. De Cristofaro, E., Gasti, P., Tsudik, G.: Fast and private computation of cardinality of set intersection and union. In: Pieprzyk, J., Sadeghi, A.-R., Manulis, M. (eds.) CANS 2012. LNCS, vol. 7712, pp. 218–231. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35404-5_17
8. Demirag, D., Ayday, E.: Tracking and controlling the spread of a virus in a privacy-preserving way. arXiv preprint arXiv:2003.13073 (2020). https://arxiv.org/pdf/2003.13073v1.pdf
9. Jung, J., Kang, D., Bae, C.: Distance estimation of smart device using Bluetooth. In: The Eighth International Conference on Systems and Networks Communications, pp. 13–18 (2013)
10. Moyal, O.S.: "Hamagen" Application – Fighting the Corona Virus, March 2020. https://medium.com/proferosec-osm/hamagen-application-fighiting-the-corona-virus-4ecf55eb4f7c. Accessed 26 Mar 2020
11. Raskar, R., et al.: Apps gone rogue: maintaining personal privacy in an epidemic. arXiv preprint arXiv:2003.08567 (2020)
12. Reichert, L., Brack, S., Scheuermann, B.: Privacy-preserving contact tracing of COVID-19 patients. IACR Cryptol. ePrint Arch. **2020**, 375 (2020)
13. Trieu, N., Shehata, K., Saxena, P., Shokri, R., Song, D.: Epione: lightweight contact tracing with strong privacy. arXiv preprint arXiv:2004.13293 (2020)
14. Troncoso, C., et al.: Decentralized privacy-preserving proximity tracing. https://github.com/DP-3T/documents/blob/master/DP3T%20White%20Paper.pdf. Accessed 3 Apr 2020
15. Xu, H., Zhang, L., Onireti, O., Fang, Y., Buchanan, W.B., Imran, M.A.: BeepTrace: blockchain-enabled privacy-preserving contact tracing for COVID-19 pandemic and beyond. arXiv preprint arXiv:2005.10103 (2020)
16. Zhou, S., Pollard, J.K.: Position measurement using Bluetooth. IEEE Trans. Consum. Electron. **52**(2), 555–558 (2006)