2020 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS) | 978-1-7281-9290-1/20/$31.00 ©2020 IEEE | DOI: 10.1109/ANTS50601.2020.9342834

Fully Homomorphic Encryption based Privacy-Preserving Data Acquisition and Computation for Contact Tracing

Koushik Sinha*, Pratham Majumder†and Subhas K. Ghosh‡
*School of Computing, Southern Illinois University, Carbondale, USA
†Department of Computer Science and Engineering, CMR Institute of Technology, Bengaluru, India
‡Commonwealth Bank of Australia, Sydney, New South Eales, 2000, Australia.
Email: *koushik.sinha@cs.siu.edu, †pratham.m@cmrit.ac.in, ‡subhas.ghosh@cba.com.au

***Abstract*—For public health surveillance systems, privacy is a major issue in storing and sharing of personal medical data. Often, patients and organizations are unwilling to divulge personal medical data for fear of compromising their privacy because although the data may be encrypted, the encrypted values typically need to be first decrypted to perform any computation on the data. Unfortunately, such a barrier in easy sharing of data can severely hamper the ability to respond in a timely and effective manner to a crisis scenario, as evident in the case of the ongoing COVID-19 pandemic. To overcome this critical obstacle, we propose in this paper a novel privacy-preserving encryption mechanism for storage and computation of sensitive healthcare data. Our scheme is based on the use of a secure fully homomorphic encryption scheme, so that the required computations can be performed directly on the encrypted data values without the need for any decryption. The ability to execute queries or computation directly on encrypted data, without the need for decryption, is not present in any existing public-health surveillance system. We propose a novel computational model and also develop an algorithm for contact tracing with COVID-19 pandemic as a case study. We have simulated our proposed approach using the ElGamal encryption algorithm to check the correctness and effectiveness of our proposed approach. The results show that our proposed solution is effective in providing adequate security while supporting the computational needs for contact-tracing. Besides contact-tracing, our new data-encryption technique can have a much broader impact in the field of healthcare. By executing queries or computations directly on encrypted data, our innovative solution would make the sharing of data in healthcare-related research and industry significantly simpler and faster. The use of such a data encryption scheme to store and transmit sensitive healthcare data over a network can not only allay the fear of compromising sensitive information but also ensure HIPAA-compliance.**

***Index Terms*—Fully homomorphic encryption, Contact tracing, Healthcare, Privacy-preserving encryption.**

I. Introduction

The unprecedented pandemic situation created by the Covid-19 virus calls for urgent effort to devise effective means to contain its spread among the populace. As a result, since the outbreak of COVID-19, a plethora of contact-tracing apps have been launched [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [19] for timely detection of infection spread and thereby taking measures to contain it. It may be noted that such an approach will be useful for public health surveillance [13] to prevent the spread of infection of not only the COVID-19 virus, but any other such deadly virus. Development of any such contact tracing technique certainly depends on collection of appropriate data about the infected persons and people they have come in contact with. Public health surveillance systems can provide such data which can be used for contact tracing as well as prediction of spread and impact of the virus [1], [2], [14], [15], [16]. Unfortunately, due to stringent privacy laws (e.g., HIPAA) in most countries, owners of such databases are reluctant to share such sensitive data. Another major hurdles in obtaining data for contact tracing is the unwillingness of people in sharing their location history and medical records with either the government or contact-tracing apps for fear of their privacy being compromised and personal data being misused.

Contact-tracing solutions can be categorized as being either centralized or distributed. *Centralized models for contact tracing* are based on the assumption that a trusted entity will not misuse the sensitive data. While data from individuals may be collected in encrypted form, to make any sort of computation with these data elements for detecting the possibility of contacts or prediction of spread of the infection, the encrypted data elements need to be decrypted, and thus the anonymity of data might be compromised. Therefore, such a model based on simply trust cannot technologically prevent unethical practices of misusing sensitive data [17].

On the other hand, ethical issues arising from privacy leakage are also not altogether eliminated in *decentralized systems* [17], [26], [27], [28]. Even in a fully decentralized model, there exists some risk of identity-unmasking when someone in a room reports sick and a nearby proximity alert is generated for another person currently located in the same room [17].

Centralized solutions for contact tracing usually follow the PEPP-PT protocol [4], while decentralized solutions typically are based on the TCN [5] protocol or DP-3T [7], or Google/Apple's exposure notification service [8] built on PACT [9], [10]. All of these protocols are designed specifically for Bluetooth-based solutions. Regardless of the architecture or technology used, perception about trust and privacy has been the primary issues for low adoption of contact tracing solutions [19], [17], [20], [21], [22], [23], [24], [25]. For example, Healthy Together [1] and Care19 [2] have less than 2% of population coverage.

A. *Our Contribution*

In this paper, we have proposed for healthcare applications a novel secure computation scheme using a *fully homomorphic encryption* (FHE) algorithm [30], [31], [29] for encrypting sensitive data. FHE functions can perform computation directly on encrypted data, without need for decryption. We propose a computational model and develop an algorithm based on it for contact tracing. Our proposed algorithm also takes care of any secondary contacts, as experienced in case of Covid-19 infection, where a person may get the infection if he visits the same place within a very short time after the departure of a patient. Since the encrypted patient data need not be decrypted for the mathematical operations required for such contact tracing, the privacy of all patients will be preserved. As an example, we use the ElGamal algorithm [32] chosen from the class of FHE algorithms for this purpose. We simulate our proposed approach on a synthetic spatiotemporal location dataset (created using the timestamped locations obtained from the Google Timeline data of the authors) to show the correctness and effectiveness of our technique in detecting possible contact with some patients by a query-issuer. The ability to execute queries or computation directly on encrypted data, without the need for decryption, is not present in any existing public-health surveillance system. Therefore, our new data-encryption technique would have a much broader impact in the field of healthcare beyond just contact-tracing. By executing queries or computations directly on encrypted data, our innovative solution would make the sharing of data in healthcare-related research and industry significantly simpler and faster.

II. Formulation of the Contact Tracing Problem

Consider the application of FHE for the execution of query/analytics on a privacy-preserving patient-location-database for contact tracing with an arbitrary user's location data. A block diagram of a system using FHE is shown in the self-explanatory Fig. 1. In this paper, our goal is to focus on proving that sensitive personal data encrypted using our approach based on the ElGamal FHE algorithm would indeed be able to provide sufficiently strong security for a contact-tracing application while supporting the necessary computations needed for effective contact-tracing for a disease like COVID-19.

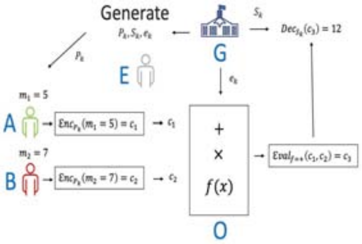


Fig. 1. Representative use of FHE

Let there be a set R of n records in the patient-location-time database given by $R = \{R_1, R_2, \cdots, R_n\}$ where each record $R_i, i = 1, 2, \cdots, n$ contains the location-time information of a patient P_i in terms of the locations $L_{ij}, j = 1, 2, \cdots, l_i$ visited by P_i along with his time of arrival T^a_{ij} at the location L_{ij} and time of departure T^d_{ij} from L_{ij}. Each location L_{ij} value would actually involve two coordinate values, e.g., (x_{ij}, y_{ij}) in two dimension or three coordinate values, e.g., (x_{ij}, y_{ij}, z_{ij}) in three dimension (when, for example, location in a multi-storeyed building is involved). It may be noted that a given location may be visited by P_i more than once at different time intervals, i.e., l_i is the number of distinct 3-tuples of (location, time of arrival, time of departure). Thus, the record R_i contains the following fields:

- i) P_i : A unique Id of patient P_i.
- ii) l_i : Number of distinct 3-tuples (location, time of arrival, time of departure).
- iii) A set of l_i 3-tuples $(L_{ij}, T^a_{ij}, T^d_{ij}), j = 1, 2, \cdots, l_i$ where the j^{th} 3-tuple specifies that the patient stayed at location L_{ij} during the time interval from T^a_{ij} to T^d_{ij}.

Symbolically, we represent R_i as $R_i = \{P_i, l_i, (L_{i1}, T^a_{i1}, T^d_{i1}), (L_{i2}, T^a_{i2}, T^d_{i2}), \cdots, (L_{i,l_i}, T^a_{i,l_i}, T^d_{i,l_i})\}$.

We assume that the information about the Id of each patient and his location-time information, i.e., P_i and each component of the 3-tuples $(L_{ij}, T^a_{ij}, T^d_{ij}), j = 1, 2, \cdots, l_i$, are all individually encrypted.

We further assume that there is a person Q who sends a query to the system regarding his possible contact with some patient at any location. That is, Q wants to know whether Q and some patient $P_i, i = 1, 2, \cdots, n$ visited the same location during some common time duration. It may be noted that the mobility information of Q will also be represented by a similar record structure as

$$\{Q, l_q, (L_{q1}, T^a_{q1}, T^d_{q1}), (L_{q2}, T^a_{q2}, T^d_{q2}), \cdots, (L_{q,l_q}, T^a_{q,l_q}, T^d_{q,l_q})\},$$

where each of the values $Q, L_{q1}, T^a_{q1}, T^d_{q1}, L_{q2}, T^a_{q2}, T^d_{q2}, \cdots, L_{q,l_q}, T^a_{q,l_q}, T^d_{q,l_q}$ will be encrypted.

Different situations of having a contact by Q with a patient P_i are illustrated by the scenarios 1-4 in Fig. 2. It may be noted that because of the inherent technological limitation, the widely used Bluetooth-based solutions all suffer from one fundamental problem, particularly for COVID-19, due to their inability of tracing contacts when someone may become infected by touching surfaces which a patient had touched recently [12]. We term this as secondary contact, as opposed to contacts when there is non-null intersection between some 3-tuples (location, time of arrival, time of departure) of a patient

P_i and the querying person Q. We illustrate this situation by scenario 5 in Fig. 2. However, we formulate below the computational problem for all these scenarios taken together.

The computational problem for contact tracing involves using only the encrypted values for $L_{i1}, T_{i1}^a, T_{i1}^d, L_{i2}, T_{i2}^a, T_{i2}^d, \cdots, L_{i,l_i}, T_{i,l_i}^a, T_{i,l_i}^d$ from the record R_i and the encrypted values for $L_{q1}, T_{q1}^a, T_{q1}^d, L_{q2}, T_{q2}^a, T_{q2}^d, \cdots, L_{q,l_q}, T_{q,l_q}^a, T_{q,l_q}^d$ to check if, for some given threshold values of ϵ, δ and γ, there exist any i, r and s such that the distance between L_{ir} and L_{qs} is less than equal to ϵ and any one of the following conditions is satisfied.

1) Overlap between a P_i and Q's presence at the same location:
   a) **Scenario 1**: $T_{qs}^d \geq T_{ir}^d \geq T_{qs}^a \geq T_{ir}^a$, and $T_{ir}^d - T_{qs}^a \geq \delta$.
   b) **Scenario 2**: $T_{ir}^d \geq T_{qs}^d \geq T_{ir}^a \geq T_{qs}^a$, and $T_{qs}^d - T_{ir}^a \geq \delta$.
   c) **Scenario 3**: $T_{qs}^d \geq T_{ir}^d \geq T_{ir}^a \geq T_{qs}^a$, and $T_{ir}^d - T_{ir}^a \geq \delta$.
   d) **Scenario 4**: $T_{ir}^d \geq T_{qs}^d \geq T_{qs}^a \geq T_{ir}^a$, and $T_{qs}^d - T_{qs}^a \geq \delta$.
2) **Scenario 5** (*Secondary contact*, i.e., no overlap in stay at the same location and same time, but Q arrived within time γ of some P_i leaving the location): $T_{qs}^d > T_{qs}^a > T_{ir}^d > T_{ir}^a$, with $T_{qs}^a - T_{ir}^d \leq \gamma$ and $T_{qs}^d - T_{qs}^a \geq \delta$.

In the next section, we discuss how these computations can be performed on the encrypted dataset using a fully homomorphic encryption algorithm in which the encryption function is monotonic, i.e., if a and b are two given real numbers with $a < b$, and their encrypted values are a' and b', respectively, then $a' < b'$.

III. Contact Tracing using Encrypted Data Values

Without loss of generality, we assume that the location values are represented in two dimension with the location L_{ir} of a patient P_i being denoted by (x_{ir}, y_{ir}), and the location L_{qs} of a person Q being denoted by (x_{qs}, y_{qs}). All these coordinate values will be encrypted so that the encrypted value of L_{ir} will be denoted by (x'_{ir}, y'_{ir}), and the encrypted value of L_{qs} will be denoted by (x'_{qs}, y'_{qs}). Since ϵ is usually very small, we make an approximation in checking the proximity between L_{qs} and L_{ir} within a distance of ϵ in original coordinate system by i) considering a square of side 2ϵ with the location L_{ir} as its center (instead of considering a circle of radius ϵ with its center at L_{ir}) and then ii) checking whether L_{qs} lies within this square. For small values of ϵ, the error in this approximation will be insignificant. We may note that the four vertices of this square will have the coordinate values $(x_{ir} - \epsilon, y_{ir} - \epsilon), (x_{ir} - \epsilon, y_{ir} + \epsilon), (x_{ir} + \epsilon, y_{ir} + \epsilon)$ and $(x_{ir} + \epsilon, y_{ir} - \epsilon)$, respectively. In the encrypted domain, this square will be converted to a rectangle defined by the coordinate values of its four vertices as $((x_{ir} - \epsilon)', (y_{ir} - \epsilon)'), ((x_{ir} - \epsilon)', (y_{ir} + \epsilon)'), ((x_{ir} + \epsilon)', (y_{ir} + \epsilon)')$ and $((x_{ir} + \epsilon)', (y_{ir} - \epsilon)')$, respectively where the $(')$ denotes the encrypted value of the corresponding coordinate. We then check whether the encrypted values of the coordinates (x'_{qs}, y'_{qs}) lie within the above rectangle in the encrypted domain for checking proximity within ϵ distance in unencrypted coordinate system.

For checking the overlapped time of presence of P_i and Q at the locations L_{ir} and L_{qs}, respectively, in the encrypted domain, we have to use the encrypted values $T_{ir}^{a'}, T_{ir}^{d'}, T_{qs}^{a'}$ and $T_{qs}^{d'}$ of $T_{ir}^a, T_{ir}^d, T_{qs}^a$ and T_{qs}^d, respectively. Along with this, we also need to have the encrypted values $(T_{ir}^d - \delta)'$, $(T_{qs}^d - \delta)'$ and $(T_{qs}^a - \gamma)'$ for detecting the overlaps in different scenarios as will be apparent from the discussions below.

Scenario 1 is identified by the condition $T_{ir}^{a'} \leq T_{qs}^{a'} \leq T_{ir}^{d'} \leq T_{qs}^{d'}$ and a contact will be reported in this scenario if $(T_{ir}^d - \delta)' \geq T_{qs}^{a'}$.

Scenario 2 is identified by the condition $T_{qs}^{a'} \leq T_{ir}^{a'} \leq T_{qs}^{d'} \leq T_{ir}^{d'}$ and a contact will be reported in this scenario if $(T_{qs}^d - \delta)' \geq T_{ir}^{a'}$.

Scenario 3 is identified by the condition $T_{qs}^{a'} \leq T_{ir}^{a'} < T_{ir}^{d'} \leq T_{qs}^{d'}$ and a contact will be reported in this scenario if $(T_{ir}^d - \delta)' \geq T_{ir}^{a'}$.

Scenario 4 is identified by the condition $T_{ir}^{a'} \leq T_{qs}^{a'} < T_{qs}^{d'} \leq T_{ir}^{d'}$ and a contact will be reported in this scenario if $(T_{qs}^d - \delta)' \geq T_{qs}^{a'}$.

Scenario 5 is identified by the condition $T_{ir}^{d'} < T_{qs}^{a'}$ and a contact will be reported in this scenario if $(T_{qs}^a - \gamma)' \leq T_{ir}^{d'}$ and $(T_{qs}^d - \delta)' \geq T_{qs}^{a'}$.

Thus, instead of a 3-tuple used for location coordinate, time of arrival and time of departure associated with each distinct location-time information in unencrypted domain, we need to have a 9-tuple with four coordinate values and five timing information in the encrypted domain, as described above.

We now describe below the Algorithm 1 for our proposed contact tracing approach called *PrivacyContactTrace* corresponding to the $(i, r)^{th}$-tuple of a patient P_i and the $(q, s)^{th}$-tuple of the querying person Q using the encrypted values.

IV. Simulation of the Proposed Approach

To analyze the effectiveness of our proposed privacy-preserving contact tracing method, we have created a synthetic spatiotemporal location dataset (R) that consists of all location related information over 24 hrs with ten COVID-19 positive patients and one querying person. To create this synthetic dataset, we used actual GPS-based timestamped location data obtained from Google Timelines of the authors. The detailed description of the dataset is given as follows:

1) Region coverage: 20 km × 20 km
2) Duraion: 24 hrs
3) Threshold distance (ϵ) = 2 m
4) Threshold contact duration (δ) = 15 min
5) Threshold time difference (γ) = 12 hr

A. *Selection of FHE algorithm*

For the purpose of user data encryption in the simulation process, we use ElGamal algorithm [32] which belongs to the class of fully homomorphic encryption algorithms. We have tested the randomness of its generated keys by using the National Institute of Standards of Technology (NIST) [33] test suite. The test parameter was designed with sequence of

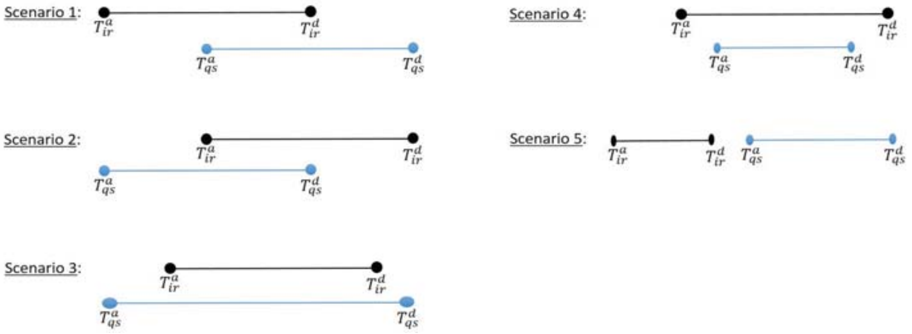


Fig. 2. Graphical representation of the five contact scenarios

```
Algorithm 1: PrivacyContactTrace
Input: $((x_{ir}-\epsilon)', (y_{ir}-\epsilon)'), ((x_{ir}-\epsilon)', (y_{ir}+\epsilon)'), ((x_{ir}+\epsilon)', (y_{ir}+\epsilon)'), ((x_{ir}+\epsilon)', (y_{ir}-\epsilon)')$
       and $(x'_{qs}, y'_{qs})$, $T^{a'}_{ir}, T^{d'}_{ir}, T^{a'}_{qs}, T^{d'}_{qs}$, $(T^d_{ir}-\delta)'$,
       $(T^d_{qs}-\delta)'$ and $(T^a_{qs}-\gamma)'$.
Output: $contact\_found$.
$contact\_found$ = FALSE;
if $(x'_{qs}, y'_{qs})$ is contained within the rectangle defined by $((x_{ir}-\epsilon)', (y_{ir}-\epsilon)'), ((x_{ir}-\epsilon)', (y_{ir}+\epsilon)'), ((x_{ir}+\epsilon)', (y_{ir}+\epsilon)'), ((x_{ir}+\epsilon)', (y_{ir}-\epsilon)')$ then
    if $T^{a'}_{ir} \leq T^{a'}_{qs} \leq T^{d'}_{ir} \leq T^{d'}_{qs}$ then
        /* Scenario 1*/
        if $(T^d_{ir}-\delta)' \geq T^{a'}_{qs}$ then
            $contact\_found$ = TRUE;
    if $T^{a'}_{qs} \leq T^{a'}_{ir} \leq T^{d'}_{qs} \leq T^{d'}_{ir}$ then
        /* Scenario 2*/
        if $(T^d_{qs}-\delta)' \geq T^{a'}_{ir}$ then
            $contact\_found$ = TRUE;
    if $T^{a'}_{qs} \leq T^{a'}_{ir} < T^{d'}_{ir} \leq T^{d'}_{qs}$ then
        /*Scenario 3 */
        if $(T^d_{ir}-\delta)' \geq T^{a'}_{ir}$ then
            $contact\_found$ = TRUE;
    if $T^{a'}_{ir} \leq T^{a'}_{qs} < T^{d'}_{qs} \leq T^{d'}_{ir}$ then
        /*Scenario 4*/
        if $(T^d_{qs}-\delta)' \geq T^{a'}_{qs}$ then
            $contact\_found$ = TRUE;
    if $T^{d'}_{ir} < T^{a'}_{qs}$ then
        /* Scenario 5*/
        if $((T^a_{qs}-\gamma)' \leq T^{d'}_{ir})$ && $((T^d_{qs}-\delta)' \geq T^{a'}_{qs})$ then
            $contact\_found$ = TRUE;
```

length = 10^6 bits and with number of sub-sequences as 10. The proportion of sequences passing a test depends entirely on the value of the significance level (α) used by NIST. For the default value of $\alpha = 0.01$ used for our analysis, the expected value of this proportion is 0.99 with a lower bound on the proportions as 0.96015. From test results shown in Table I, ElGamal algorithm passes all 15 statistical tests.

TABLE I
RANDOMNESS COMPARISON BETWEEN FHE AND NON-FHE SCHEMES

Test	ElGamal
Frequency	0.970
Block Frequency	0.977
Cumulative Sums	0.983
Runs	0.975
Longest Run	0.976
Rank	0.964
FFT	0.981
Non Overlapping Template	0.984
Overlapping Template	0.979
Universal	0.988
Approximate Entropy	0.971
Random Excursions	0.967
Random Excursions Variant	0.983
Serial	0.983
Linear Complexity	0.984

B. Simulation Results

Table II shows a summary of the simulation results on our datasets. The first column describes individual identities. The table heading L refers to the original (unencrypted) location of a person in two dimension, headings A, B, C and D refer to the coordinate values of the four vertices, respectively, of the rectangle in the encrypted domain around the position of a patient, headings T^a and T^d refer to the unencrypted arrival and departure times, respectively, of a person, while headings $T^{a'}$ and $T^{d'}$ refer to their corresponding values, respectively, in encrypted domain. The proximity between the querying person and a patient with respect to ϵ distance in the unencrypted domain is validated in the encrypted domain, as used by Algorithm 1. Different contact scenarios are shown with respect to timing information which are also verified in encrypted domain.

TABLE II
SIMULATION RESULTS

Human Identity	**Original Dataset**			**Encrypted Dataset**										**Results**
	L	T^a	T^d	**A**	**B**	**C**	**D**	(x'_q, y'_q)	$T^{a'}$	$T^{d'}$	$(T^d_{ir} - \delta)'$	$(T^d_{qs} - \delta)'$	$(T^a_{qs} - \gamma)'$	
P1	(28,21)	12:31	12:48	(221,208)	(221,214)	(228, 214)	(228, 214)	(223,211)	20:03	20:44	20:32	-	-	Contact found (Scenario 1)
Q	(27,21)	12:32	12:51						20:11	20:51				
P2	(105,46)	08:16	08:54	(715,319)	(715,389)	(765,389)	(765,319)	(738,363)	13:01	13:30	-	13:08	-	Contact found (Scenario 2)
Q	(106,45)	08:10	08:39						12:49	13:28				
P3	(5,76)	06:30	07:10	(79,548)	(79,589)	(86,589)	(86,548)	(81,550)	11:23	11:41	11:26	-	-	Contact found (Scenario 3)
Q	(5,74)	06:19	07:25						11:06	11:47				
P4	(40,32)	13:54	14:40	(298,256)	(298,260)	(303,260)	(303,256)	(300,259)	22:16	22:52	-	22:29	-	Contact found (Scenario 4)
Q	(42,32)	13:58	14:15						22:24	22:41				
P5	(110,45)	9:33	9:48	(752,311)	(752,317)	(758,317)	(758,311)	(755,315)	13:48	14:12	-	23:28	13:46	Contact found (Scenario 5)
Q	(112,45)	14:51	15:30						23:02	23:39				

V. INTEGRATION WITH VIRUS CONTACT MAP - A PUBLIC HEALTH SURVEILLANCE PLATFORM

The Virus Contact Map (VCM) platform is a privacy-preserving public-health surveillance system that is currently being developed at the Southern Illinois University, Carbondale, USA [34]. VCM currently integrated several different types of visualizations, and mobile-app based interfaces to upload data from patients and individuals interested in determining if they have been exposed to an infected person (henceforth called contact-testers). One of the visualizations generated by VCM is a map of a contact-tester's recent contacts with COVID-19 patients for the purpose of contact tracing by individuals interested in self-assessment, and health officials.

For the purpose of contact-tracing, VCM needs only two pieces of information about infected users: i) anonymized recent GPS location history (e.g., Google Maps), and ii) date of testing positive. All patient data in VCM is stored in a centralized database called *patient-data-store*. *We have currently integrated our proposed FHE-based encryption scheme with VCM to create a secure and privacy-preserving patient-data-store.* This privacy-preserving *patient-data-store* is compatible with our proposed *PrivacyContactTrace* algorithm to ensure that all contact-tracing computations can be done directly on the encrypted patient-data. VCM's *spatio-temporal contact-tracing engine* can detect not only contacts happening between people present at the same location at the same time, but also trace contacts who may become infected by touching a surface that an infected person had touched earlier (secondary contacts). To fully preserve the privacy of patients by avoiding the need for decrypting sensitive patient data, we have also integrated our proposed FHE-based *PrivacyContactTrace* algorithm with the *spatio-temporal contact-tracing engine.*

VCM employs a crowdsourced model of contact-tracing where contact-testers are in control of their data for contact tracing and exposure notification. For a contact-tester interested in self-assessment, only recent location history needs to be uploaded to the VCM platform, which is thereafter deleted immediately after generating the contact-tracing results. Fig. 3 shows a sample visualization of contact tracing results generated by VCM using our proposed FHE-based privacy-preserving contact-tracing approach. Each detected contact instance is color-coded based on a risk of infection calculated by VCM's *risk-assessment module.* By clicking a displayed contact, a contact-tester can see more details such as time and duration of contact, distance between them, and infection-risk of that contact.

VI. CONCLUSION

We have proposed a privacy-preserving approach for storage and computation of sensitive medical data using a fully homomorphic encryption (FHE) algorithm, so that all computations can be performed only on the encrypted data values, without the need for any decryption. We have developed a computation model along with an algorithm for tracing the contacts with any patient by executing the computations directly on encrypted spatiotemporal data values. The model also considers the scenario of contact tracing for secondary contacts, as observed with the COVID-19 infection. The proposed approach has been simulated with the ElGamal algorithm, which belongs to the class of FHE algorithms. The simulation results show that our proposed solution is effective in providing adequate security while supporting the computational needs for contact-tracing. To create a complete large-scale contact-tracing solution using such a FHE system, our future work will be to design a *multi-party computation protocol* [35], [36] where the *number of users is not known initially and changes over time* (i.e., users arrive asynchronously and hence cannot participate in standard threshold cryptography). Since the ability to execute queries or computation directly on encrypted data is not present in any existing public-health surveillance system, our proposed solution would have a much broader impact in the field of healthcare by making the sharing of data in healthcare-related research and industry significantly simpler and faster without violating HIPAA laws.

REFERENCES

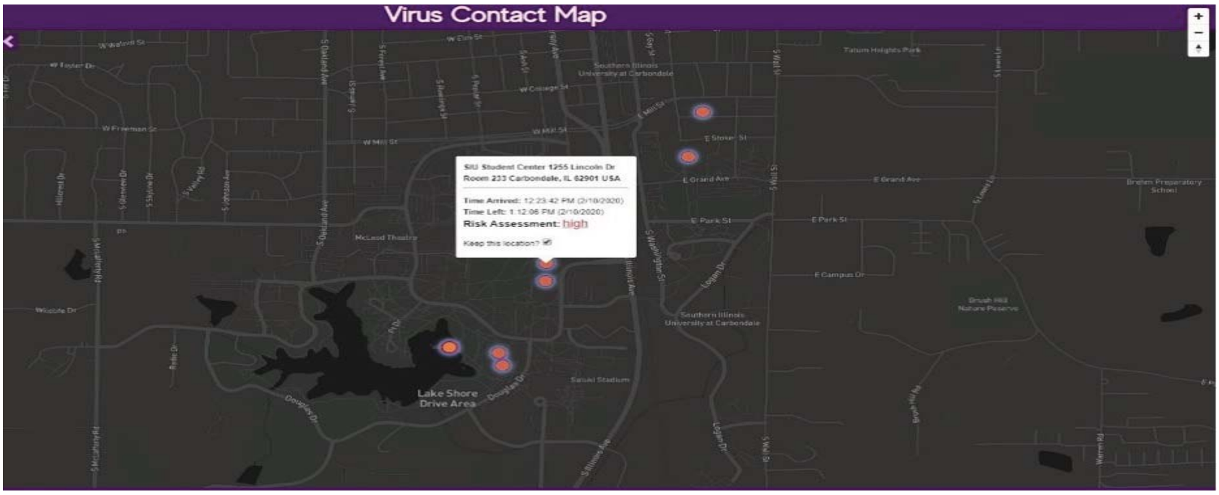
[1] Healthy Together, https://www.healthytogether.io/.

[2] CARE19, https://ndresponse.gov/covid-19-resources/care19.

[3] COVID-19 apps, https://en.wikipedia.org/wiki/COVID-19_apps.

[4] Pan-European Privacy-Preserving Proximity Tracing (PEPP-PT/PEPP), https://www.pepp-pt.org/.

[5] Temporary Contact Numbers (TCN) Protocol, https://en.wikipedia.org/wiki/TCN_Protocol.



SIU Student Center 1255 Lincoln Dr Room 233 Carbondale, IL 62901 USA

Time Arrived: 12:23:42 PM (2/10/2020)
Time Left: 1:12:06 PM (2/10/2020)
Risk Assessment: high

Keep this location? ☑

Fig. 3. Screenshot of VCM's contact-tracing result generated by our proposed privacy-preserving approach

[6] S. V. Arx et al., "Slowing the Spread of Infectious Diseases Using Crowdsourced Data," https://www.covid-watch.org/covid_watch_whitepaper.pdf.

[7] C. Troncoso et al., "Decentralized Privacy-Preserving Proximity Tracing (DP-3T)," https://arxiv.org/ftp/arxiv/papers/2005/2005.12273.pdf.

[8] "Exposure Notifications: Using technology to help public health authorities fight COVID-19," https://www.google.com/covid19/exposurenotifications/.

[9] R. L. Rivest et al., "PACT: Private Automated Contact Tracing," https://pact.mit.edu/.

[10] J. Chan et al., "PACT: Privacy-Sensitive Protocols and Mechanisms for Mobile Contact Tracing," ArXiv, abs/2004.03544, 2020.

[11] D. A. Drew et al., "Rapid implementation of mobile technology for real-time epidemiology of COVID-19," *Science*, https://science.sciencemag.org/content/early/2020/05/05/science.abc0473.full.pdf, 2020.

[12] J. Bay et al., "BlueTrace: A privacy-preserving protocol for community-driven contact tracing across borders," https://bluetrace.io/static/bluetrace_whitepaper-938063656596c104632def383eb33b3c.pdf.

[13] Centers for Disease Control and Prevention (CDC), "Introduction to Public Health," *Public Health 101 Series*, U.S. Department of Health and Human Services, 2014.

[14] Integrated Surveillance Information Systems/NEDSS, CDC, https://wwwn.cdc.gov/nndss/nedss.html.

[15] Illinois' National Electronic Disease Surveillance System (I-NEDSS), https://www.dph.illinois.gov/topics-services/diseases-and-conditions/infectious-diseases/infectious-disease-reporting.

[16] National Notifiable Diseases Surveillance System, CDC, https://www.cdc.gov/nmi/.

[17] "Governments Shouldn't Use 'Centralized' Proximity Tracking Technology," *Electronic Frontier Foundation*, https://www.eff.org/deeplinks/2020/05/governments-shouldnt-use-centralized-proximity-tracking-technology?fbclid=IwAR22AgKOhuDj75zxr66c5mxQcx0CT-lzGFaiiCQbVpjtIGz3o2DqdX9ZdNg.

[18] "Demonstrating 15 contact tracing and other tools built to mitigate the impact of COVID-19," https://techcrunch.com/2020/06/05/demonstrating-15-contact-tracing-and-other-tools-built-to-mitigate-the-impact-of-covid-19/.

[19] J. Li and X. Guo, "COVID-19 Contact-tracing Apps: a Survey on the Global Deployment and Challenges," ArXiv, abs/2005.03599, https://arxiv.org/ftp/arxiv/papers/2005/2005.03599.pdf,2020.

[20] H. Cho, D. Ippolito, and Y. W. Yu, "Contact Tracing Mobile Apps for COVID-19: Privacy Considerations and Related Trade-offs," Arxiv, abs/2003.11511, 2020.

[21] Z. Doffman, "COVID-19 Contact Tracing: Why Apple and Google Can't Make This Work," *Forbes*, https://www.forbes.com/sites/zakdoffman/2020/04/27/this-is-the-contact-tracing-worry-even-apple-and-google-cant-resolve/\#1311916a4510.

[22] M. M. Mello and C. J. Wang, "Ethics and governance for digital disease surveillance," *Science*, vol. 368, issue 6494, 2020, pp. 951-954.

[23] B. C. de Jong et al., "Ethical Considerations for Movement Mapping to Identify Disease Transmission Hotspots," *Emerging Infectious Diseases*, vol. 25, no. 7, July 2019.

[24] K. Denecke, "An ethical assessment model for digital disease detection technologies," *Life Sciences, Society and Policy*, Springer Open, vol. 13, no. 16, 2017, doi:10.1186/s40504-017-0062-x.

[25] M. J. Parker, C. Fraser, L. Abeler-Dörner, and D. Bonsall, "Ethics of instantaneous contact tracing using mobile phone apps in the control of the COVID-19 pandemic," *Journal of Medical Ethics*, May 2020, doi: 10.1136/medethics-2020-106314.

[26] S. Farell and D. J. Leith, "A Coronavirus Contact Tracing App Replay Attack with Estimated Amplification Factors," *School of Computer Science & Statistics (SCSS) Technical Report*, 2020.

[27] A. Berke, M.A.Bakker, P. Vepakomma, K. Larson, and A. Pentland, "Assessing Disease Exposure Risk with Location Data: A Proposal for Cryptographic Preservation of Privacy," ArXiv: Cryptography and Security, 2020.

[28] S. Altmann et al., "Acceptability of app-based contact tracing for COVID-19: Cross-country survey evidence," https://www.medrxiv.org/content/early/2020/05/08/2020.05.05.20091587, 2020.

[29] PALISADE Homomorphic Encryption Software Library, https://palisade-crypto.org/.

[30] Microsoft SEAL, https://www.microsoft.com/en-us/research/project/microsoft-seal/.

[31] HElib, https://homenc.github.io/HElib/.

[32] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, 31(4), pp.469-472.

[33] "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf.

[34] "SIU researcher's tool would improve tracking, avoidance of COVID-19 cases," https://news.siu.edu/2020/05/050520-virus-tracker.php, 2020.

[35] I. Damgård, V. Pastro, N. Smart and S. Zakarias, "Multiparty Computation from Somewhat Homomorphic Encryption. In Annual Cryptology Conference," *Springer*, pp. 643-662, 2012.

[36] M. Hastings, B. Hemenway, D. Noble, and S. Zdancewic. "SOK: General Purpose Compilers for Secure Multi-Party Computation." *Proc. 2019 IEEE Symposium on Security and Privacy* (SP), pp. 1220-1237, 2019.