Poster: Privacy-Preserving Epidemiological Modeling on Mobile Graphs

Daniel Günther
guenther@encrypto.cs.tu-darmstadt.de
Technical University of Darmstadt

Marco Holz
holz@encrypto.cs.tu-darmstadt.de
Technical University of Darmstadt

Benjamin Judkewitz
benjamin.judkewitz@charite.de
Charité-Universitätsmedizin

Helen Möllering
moellering@encrypto.cs.tu-darmstadt.de
Technical University of Darmstadt

Benny Pinkas
benny@pinkas.net
Bar-Ilan University

Thomas Schneider
schneider@encrypto.cs.tu-darmstadt.de
Technical University of Darmstadt

Ajith Suresh
suresh@encrypto.cs.tu-darmstadt.de
Technical University of Darmstadt

ABSTRACT

Over the last two years, governments all over the world have used a variety of containment measures to control the spread of COVID-19, such as contact tracing, social distance regulations, and curfews. Epidemiological simulations are commonly used to assess the impact of those policies before they are implemented in actuality. Unfortunately, their predictive accuracy is hampered by the scarcity of relevant empirical data, concretely detailed social contact graphs. As this data is inherently privacy-critical, there is an urgent need for a method to perform powerful epidemiological simulations on real-world contact graphs without disclosing sensitive information.

In this work, we present RIPPLE, a privacy-preserving epidemiological modeling framework that enables the execution of a wide range of standard epidemiological models for any infectious disease on a population's most recent real contact graph while keeping all contact information private locally on the participants' devices. Our theoretical constructs are supported by a proof-of-concept implementation in which we show that a 2-week simulation over a population of half a million can be finished in 7 minutes with each participant consuming less than 50 KB of data.

CCS CONCEPTS

• **Security and privacy → Cryptography**; **Privacy-preserving protocols**.

KEYWORDS

Decentralized Epidemiological Modeling; Privacy; Private Information Retrieval; COVID-19; Trusted Execution Environments

CCS '22, November 7–11, 2022, Los Angeles, CA, USA

ACM ISBN 978-1-4503-9450-5/22/11.
https://doi.org/10.1145/3548606.3563497

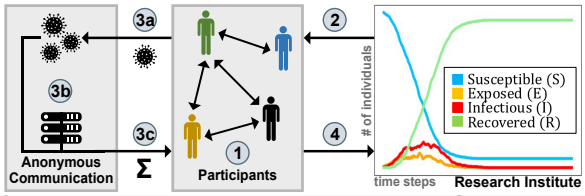
ACM Reference Format:
Daniel Günther, Marco Holz, Benjamin Judkewitz, Helen Möllering, Benny Pinkas, Thomas Schneider, and Ajith Suresh. 2022. Poster: Privacy-Preserving Epidemiological Modeling on Mobile Graphs. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22), November 7–11, 2022, Los Angeles, CA, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3548606.3563497

1 INTRODUCTION

The current pandemic significantly impacted people's daily lives, posing significant challenges such as increased mental illness, domestic abuse cases, and many more [7]. Governments worldwide have taken various steps in the last two years to restrict the spread of the virus to save human lives and keep the economic system working. Those range from closing institutions, such as schools, to country-wide lockdowns. Despite these courageous efforts, the global number of infections skyrocketed, and COVID-19 claimed far too many lives. Aside from highly lethal diseases like COVID-19, many other infectious diseases have emerged and have had a significant impact on human life over time. In the context of COVID-19, contact tracing apps are being used worldwide to notify contacts of potential infections [2]. Unfortunately, there is a fundamental limitation to contact tracing: It only notifies contacts of an infected person *after* the infection has been detected, i.e., typically after a person develops symptoms, is tested, receives the test result, and can connect with contacts. Recent studies [6, 10] have concluded that contact tracing must be supplemented with multiple additional measures to control disease spread effectively.

Epidemiological modelling allows us to predict the spread of an infectious disease in the *future* and has received a lot of attention [3, 11]. It allows for assessing the effectiveness of containment measures by mathematically modelling their impact on the spread. As a result, it can be an extremely valuable tool for governments to select effective containment measures [9]. With access to detailed information about a population's size, density, transportation, and health care system, epidemiological modelling could accurately forecast disease transmission in various situations [1]. Especially

precise, up-to-date information about movements and physical interactions in space and time is crucial for precisely forecasting transmission and the impact of various control measures before being implemented [5]. However, data on personal encounters is very scarce; thus the impact of containment measures can only be approximated so far [1]. This lack of data is primarily because encounter data has generally been acquired by surveys, which do not accurately reflect reality [5], e.g., random encounters in public transport or shopping malls. Moreover, social interaction patterns change over time and sometimes even rapidly as we have seen with social distancing measures, rendering collected contact information outdated. None of the existing data permits realistic simulations of the actual person-to-person social contact graph. From a modeller's perspective, epidemiologists would ideally like access to a population's complete physical interaction graph. Furthermore, the right for privacy reflected in strict data protection regulations of liberal states makes accurate tracking of interpersonal contacts unacceptable.



① Mobile apps collect anonymous encounter tokens. ② RI begins the simulation by providing initialization parameters. ③a Participants securely upload infection likelihood to servers. ③b Servers securely compute cumulative infection likelihood per participant. ③c Participants retrieve their cumulative infection likelihood. ④ The aggregate results (#S,#E,#I,#R) are sent to the Research Institute.

Figure 1: Overview of RIPPLE Framework.

To address the issue of obtaining the most recent contact data while protecting individuals' privacy, we present RIPPLE, a practical privacy-preserving framework for epidemiological modelling that allows precise simulations of disease spread based on current participant data while taking into account deployed control measures and without leaking any information about individuals contacts. RIPPLE provides a privacy-preserving method for collecting real-time physical encounters and can compute arbitrary compartment-based epidemiological models on the most recent contact graph in a privacy-preserving manner. RIPPLE can be used to investigate the effect of containment measures not only for COVID-19, but for *any* infectious diseases.

2 THE RIPPLE FRAMEWORK

This paper introduces RIPPLE, a framework for expanding the scope of privacy research from contact tracing to epidemiological modelling. RIPPLE uses a fully decentralised system similar to the federated learning paradigm [8] to achieve high acceptance and trust in the system and to motivate many participants to join the system to generate representative contact information. All participant data, such as encounter location, time, distance, and so on, are kept locally on the participants' devices.

RIPPLE comprises of p participants, denoted collectively by $\mathcal{P}$, a research institute RI who is in charge of the epidemiological simulations, and a set of MPC servers C responsible for anonymous communication among the participants. We assume that the research institute and MPC servers follow the *semi-honest* security model, which means they correctly follow protocol specifications while attempting to gather additional information. RIPPLE is divided into four phases as shown in Fig. 1: i) Token Generation, ii) Simulation Initialization, iii) Simulation Execution, and iv) Result Aggregation. Our framework can be applied to any compartment-based epidemiological modelling of any infectious disease.

Protocol RIPPLE

① - Token Generation

- $\mathcal{P}_i \in \mathcal{P}$ executes $\mathcal{F}_{\text{gen}}$ all the time (on its mobile device), collecting encounter data of the form (r_e, m_e) with $e < E_i^{\max}$.

② - Simulation Initialization

- $\mathcal{P}_i \in \mathcal{P}$ receives $\text{param}_{\text{sim}}$ from RI and locally sets $I_i^1 = I_i^{\text{init}}$.

③ - Simulation Execution

For each simulation step $s \in [N_{\text{step}}]$, $\mathcal{P}_i \in \mathcal{P}$ execute the following:

- Filter out encounters using $\text{param}_{\text{sim}}$ to obtain encounter set $\mathcal{E}_i^s$.
- For each token $r_e \in \mathcal{E}_i^s$, compute the infection likelihood $\delta_i^{r_e}$ locally using the formula from RI.
- Invoke $\mathcal{F}_{\text{esim}}$ with the input $\{\delta_i^{r_e}\}_{r_e \in \mathcal{E}_i^s}$ and obtain $\Delta_i^s = \sum_{r_e \in \mathcal{E}_i^s} \hat{\delta}_i^{r_e}$.
- Update the infection class I_i^s using Δ_i^s and the guidelines from RI.

④ - Result Aggregation

For each simulation step $s \in [N_{\text{step}}]$, execute the following:

- $\mathcal{P}_i \in \mathcal{P}$ prepares $\{v_i^1, \ldots, v_i^{N_{\text{inf}}}\}^s$ with $v_i^k = 1$ if $I_i^s = \text{class}_{\text{inf}}^k$ and $v_i^k = 0$ otherwise, for $k \in [N_{\text{inf}}]$.
- Invoke $\mathcal{F}_{\text{agg}}$ with inputs $\{v_i^1, \ldots, v_i^{N_{\text{inf}}}\}^s$ to enable RI obtain the tuple $\{C_{\text{inf}}^1, \ldots, C_{\text{inf}}^{N_{\text{inf}}}\}^s$ where $C_{\text{inf}}^k = \sum_{\mathcal{P}_i \in \mathcal{P}} v_i^k$ for $k \in [N_{\text{inf}}]$.

Figure 2: RIPPLE Framework (for one simulation setting).

RIPPLE comprises of two methods for achieving privacy-preserving epidemiological modelling, each of which caters to a different use case. The first is $\text{RIPPLE}_{\text{TEE}}$, which assumes that each participant's mobile device has a Trusted Execution Environment (TEE). The second method, $\text{RIPPLE}_{\text{PIR}}$, eliminates this assumption by utilising cryptographic primitives such as Private Information Retrieval (PIR). We summarize our contributions as follows:

(1) We present RIPPLE, the *first* privacy-preserving framework to perform epidemiological modelling on contact information stored on mobile devices.
(2) RIPPLE formalises the notion of *privacy-preserving* epidemiological modelling and defines privacy requirements in the presence of both semi-honest and malicious participants.
(3) For epidemiological simulations using real-world contact data acquired with participants' mobile devices, we present two techniques – $\text{RIPPLE}_{\text{TEE}}$ and $\text{RIPPLE}_{\text{PIR}}$, that combine anonymous communication techniques with either TEEs or PIR and anonymous credentials.
(4) We propose PIR-SUM, which allows clients to download the sum of τ database entries without learning the values of individual entries or revealing which entries were requested.

Entities	Protocol	Population (p)									
		1K	10K	50K	100K	500K	1M	2M	5M	10M	20M
Participants in $\mathcal{P}$ (in KB)	$\text{RIPPLE}_{\text{TEE}}$	16.00	16.00	16.00	16.00	16.00	16.00	16.00	16.00	16.00	16.00
	$\text{RIPPLE}_{\text{PIR}}$: $\text{PIR}^{\text{I}}_{\text{sum}}$	51.63	62.42	69.97	73.22	80.77	84.02	87.27	91.56	94.81	98.06
	$\text{RIPPLE}_{\text{PIR}}$: $\text{PIR}^{\text{II}}_{\text{sum}}$	3.45	3.49	3.52	3.53	3.56	3.57	3.59	3.60	3.62	3.63
Servers in C (in GB)	$\text{RIPPLE}_{\text{TEE}}$	0.02	0.19	0.96	1.92	9.60	19.20	38.40	96.00	192.00	384.00
	$\text{RIPPLE}_{\text{PIR}}$	0.01	0.10	0.48	0.96	4.80	9.60	19.20	48.00	96.00	192.00

Table 1: Communication costs per simulation step in our RIPPLE instantiations.

(5) We demonstrate the practicality of RIPPLE by a detailed performance evaluation. Our findings indicate that our protocol can scale up to millions of participants. For instance, a simulation of 14 days with 1 million participants can be completed in less than half an hour.

Fig. 2 summarises the phases of the RIPPLE framework in the context of a single simulation setting. Note that multiple simulations can be executed in parallel.

3 EVALUATION

We evaluate the computation and communication efficiency of the simulation phase of our two RIPPLE protocols. The simulations can ideally be done overnight while mobile phones are charging and have access to a high-bandwidth WiFi connection. We run the benchmarks on the server-side with three servers with Intel Core i9-7960X CPUs@2.8 GHz and 128 GB RAM connected with 10 Gbit/s LAN and 0.1 s RTT. The client is a Samsung Galaxy S10+ with an Exynos 9820@2.73 GHz and 8GB RAM. We instantiate our protocols in RIPPLE with $\kappa = 128$ bit security. A typical simulation step would be one day such that 14 steps simulate two weeks.

Communication Costs. We consider participant sizes, denoted by p, ranging from thousand (1K) to twenty million (20M) to analyse the scalability of our protocols. Tab. 1 summarises the communication costs incurred by each participant as well as the communication servers (C) for one simulation step in a specific simulation. It should be noted that one simulation step includes all protocol steps, beginning with participants locally computing their infection likelihood (δ) and ending with them obtaining their cumulative infection likelihood (Δ) for that step.

	Per Simulation Step			Per Simulation ($N_{\text{step}} = 14$)		
	Message Generation (in ms)	PIR Queries (in ms)	Output Computation (in ms)	Message Generation (in sec)	PIR Queries (in sec)	Output Computation (in sec)
$\text{RIPPLE}_{\text{TEE}}$	80.00	-	3040.00	1.12	-	42.56
$\text{PIR}^{\text{I}}_{\text{sum}}$	0.30	11.73	4.8e-2	4.26e-3	0.16	6.72e-4
$\text{PIR}^{\text{II}}_{\text{sum}}$	0.30	3.0e-3	4.8e-2	4.26e-3	4.2e-5	6.72e-4

Table 2: Average participant computation times per simulation step distributed across various tasks. Values are obtained using a mobile for a population of 500K.

Computation Costs. Tab. 2 summarizes the computation time with respect to a participant $\mathcal{P}_i$ for a two-week simulation over a half-million population. For a 14-day simulation with a population of half a million, $\mathcal{P}_i$ in $\text{RIPPLE}_{\text{TEE}}$ needs approximately 43.7 seconds of computation time to perform the encryption and decryption tasks and may require additional time for the remote attestation procedure, which is not covered in our benchmarks. $\mathcal{P}_i$'s computation time in $\text{RIPPLE}_{\text{PIR}}$, on the other hand, is significantly lower and is at most 5 milliseconds for the case of $\text{PIR}^{\text{II}}_{\text{sum}}$, while it increases to around 165 milliseconds for the case of $\text{PIR}^{\text{I}}_{\text{sum}}$.

Our benchmarking using the proof-of-concept implementation demonstrated the RIPPLE framework's viability for real-world adaptation. One of the key benefits of our approaches is that participants have very little work to do. The system's efficiency can be increased with appropriate hardware and optimized implementations. RIPPLE is provably privacy-preserving by construction. We refer the reader to the full version [4] for more elaborate details of our work.

Code availability. Available at `DOI: 10.5281/zenodo.6595449`.

Acknowledgements. This project received funding from the ERC under the European Union's Horizon 2020 research and innovation program (grant agreement No. 850990 PSOTI). It was co-funded by the DFG within SFB 1119 CROSSING/236615297 and GRK 2050 Privacy & Trust/251805230, and by the BMBF and the HMWK within ATHENE.

REFERENCES

[1] David Adam. 2020. Special report: The simulations driving the world's response to COVID-19. *Nature* (2020).
[2] Nadeem Ahmed, Regio A Michelin, Wanli Xue, Sushmita Ruj, Robert Malaney, Salil S Kanhere, Aruna Seneviratne, Wen Hu, Helge Janicke, and Sanjay K Jha. 2020. A Survey of COVID-19 Contact Tracing Apps. *IEEE Access* (2020).
[3] Giulia Giordano, Franco Blanchini, Raffaele Bruno, Patrizio Colaneri, Alessandro Di Filippo, Angela Di Matteo, and Marta Colaneri. 2020. Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. *Nature Medicine* (2020).
[4] Daniel Günther, Marco Holz, Benjamin Judkewitz, Helen Möllering, Benny Pinkas, Thomas Schneider, and Ajith Suresh. 2022. Privacy-Preserving Epidemiological Modeling on Mobile Graphs. (2022). https://doi.org/10.48550/arXiv.2206.00539
[5] Petra Klepac, Adam J Kucharski, Andrew JK Conlan, Stephen Kissler, Maria L Tang, Hannah Fry, and Julia R Gog. 2020. Contacts in context: large-scale setting-specific social mixing matrices from the BBC Pandemic project. *MedRxiv* (2020).
[6] Dyani Lewis. 2020. Where Covid contract-tracing went wrong. *Nature* (2020).
[7] Dominika Maison, Diana Jaworska, Dominika Adamczyk, and Daria Affeltowicz. 2021. The challenges arising from the COVID-19 pandemic and the way people deal with them. A qualitative longitudinal study. *PloS One* (2021).
[8] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *International Conference on Artificial Intelligence and Statistics*.
[9] Robin N. Thompson. 2020. Epidemiological models are important tools for guiding COVID-19 interventions. *BMC Medicine* 18, 1 (2020), 152.
[10] Paul Tupper, Sarah P. Otto, and Caroline Colijn. 2021. Fundamental Limitations of Contact Tracing for COVID-19. *FACETS* (2021).
[11] Tijana Šušteršič, Andjela Blagojević, Danijela Cvetković, Aleksandar Cvetković, Ivan Lorencin, Sandi Baressi Šegota, Dragan Milovanović, Dejan Baskić, Zlatan Car, and Nenad Filipović. 2021. Epidemiological Predictive Modeling of COVID-19 Infection: Development, Testing, and Implementation on the Population of the Benelux Union. *Frontiers in Public Health* 9 (2021).