IoTrace: A Flexible, Efficient, and Privacy-Preserving IoT-enabled Architecture for Contact Tracing

Pietro Tedeschi, Spiridon Bakiras, and Roberto Di Pietro
Division of Information and Computing Technology
College of Science and Engineering
Hamad Bin Khalifa University, Doha - Qatar
Email: {ptedeschi, sbakiras, rdipietro}@hbku.edu.qa

***Abstract*—Contact tracing promises to help fight the spread of COVID-19 via an early detection of possible contagion events. To this end, most existing solutions share the following architecture: smartphones continuously broadcast random beacons that are intercepted by nearby devices and stored into their local contact logs. In this paper, we propose an IoT-enabled architecture for contact tracing that relaxes the smartphone-centric assumption, and provide a solution that enjoys the following features: (i) it reduces the overhead on the end-user to the bare minimum—the mobile device only broadcasts its beacons; (ii) it provides the user with a degree of privacy not achieved by competing solutions—even in the most privacy adverse scenario, the solution provides k-anonymity; and, (iii) it is flexible: the same architecture can be configured to support several models—ranging from the fully decentralized to the fully centralized ones—and the system parameters can be tuned to support the tracing of several social interaction models. What is more, our proposal can also be adopted to tackle future human-proximity transmissible diseases. Finally, we also highlight open issues and discuss a number of future research directions at the intersection of IoT and contact tracing.**

I. INTRODUCTION

One thing is clear about the COVID-19 pandemic declared in March 2020: despite the release of few vaccines, the fight against the virus could still last for years—due to the required global mass production, untested efficacy at scale, expected delays in distribution, and the very same virus polymorphic capabilities. Indeed, the initial battles gained against the virus have been later lost, with the "second wave" ravaging the world as of November 2020 [1].

The initial, dramatic spread of COVID-19 prompted individual states and international organizations to implement drastic measures to "flatten the curve" of the pandemic [2], [3]. *Digital contact tracing* is one of the most promising technological solutions, and its premise is quite intuitive: leverage the user's smartphone to keep track of other users nearby (called *contacts*) [4]. Then, if a contact has a positive diagnosis for the coronavirus, the user is notified to take precautionary measures, such as testing or self-quarantine. The most prominent approach to contact tracing is to have each mobile device broadcast pseudo-random beacons via its Bluetooth Low Energy (BLE) interface. These beacons are then received and recorded by other users within the BLE transmission range. Alternatively, solutions like Israel's Hamagen [5] adopt the Global Navigation Satellite System (GNSS) for localization and proximity tracing.

A watershed difference in contact tracing applications lies in the *reconciliation* process, i.e., the identification of "infected" beacons inside a user's contact list that signal possible contagion events. To one extreme, *centralized solutions* require all users to share their beacons and/or contact lists with the health authorities, who perform the reconciliation process and notify the exposed users. To the other extreme, *decentralized solutions* do not collect any information from the mobile devices. Instead, when a user is diagnosed as positive, the app releases the user's beacons to the authorities, which are then distributed to all the other users in the system. As such, the app is responsible for the reconciliation process, by matching the released beacons against the stored contact logs.

This generic contact tracing framework raises some concerns about the *usability* of the solution, and opens up the Pandora's box of *privacy* and *security* issues. The cited dimensions, other than being critical on their own, could also thwart the widespread adoption of contact tracing, making it irrelevant in fighting the pandemic. Indeed, Oxford researchers have calculated that, to be effective, contact tracing apps must be actively used by at least 60% of the population [6]. To reach the above goal, a more usable and privacy-preserving solution would have the potential to attract more active users, thus increasing the effectiveness of contact tracing.

In terms of usability, the main challenges are related to the energy efficiency and computational cost of the contact tracing app. For example, one of the common criticisms against existing applications is the diminished smartphone battery life. While some energy is consumed on the periodic transmission of the device's beacon, the main factor behind battery drain is the continuous scanning of the Bluetooth channel for beacons transmitted by the surrounding devices.

When it comes to privacy and security, the scientific community has started debating the issue from the very beginning [7]. The most recurrent threats are user de-identification and user tracking. In particular, an eavesdropper can identify a user as positive to the disease, by cross-referencing the "infected" beacons published by the authorities with the beacons acquired via eavesdropping. The same data may also allow

an adversary to track the locations that a positively diagnosed individual has visited. This is a clear violation of the General Data Protection Regulation (GDPR) laws in the EU and, in any case, a serious threat that could hinder the adoption of the contact tracing application.

Contributions. Motivated by the above observations, we introduce IoTrace, a contact tracing solution that relies on a distributed, flexible, and lightweight IoT-based infrastructure. IoTrace imposes minimal overhead on the user's smartphone, while providing strong privacy guarantees not available in competing proposals. Specifically, IoTrace relaxes the requirement for smartphones to receive the beacons issued by other devices in their proximity. This translates into considerable savings in energy consumption and computational/storage costs. Further advantages are that the IoT infrastructure is fully distributed, heterogeneous, and pervasive. Distribution and heterogeneity do help security [8], while pervasiveness would assure efficient and accurate contact tracing. The reconciliation mechanism is fully tunable and could range from a completely decentralized solution to a centralized one. Finally, it is worth noting that the proposed solution would work for any type of *human2human* transmissible disease—just tailoring the application parameters, such as the time of exposure needed to trigger the precautionary measures.

II. Related Work

Several contact tracing applications have been developed in the last few months. In the following paragraphs, we provide a brief introduction of the state-of-the-art approaches and also present a quantitative comparison in terms of user privacy and performance.

BlueTrace [9]. BlueTrace is an open-source protocol that is utilized in Singapore's TraceTogether app. It adopts the BLE technology, where devices exchange their ephemeral IDs (i.e., beacons) via broadcast and log all encounters in their history logs. When a user is diagnosed as positive, his/her history logs are sent to a central authority, using a secure connection. Even though BlueTrace leverages the decentralized architecture, the ephemeral IDs are generated by the central authority and distributed to the individual devices. As such, the reconciliation function and exposure notification are performed at a centralized location, i.e., BlueTrace is considered a *hybrid* solution. The main cryptographic primitive involved in the computation of the ephemeral IDs is AES-256-GCM.

DP-3T [10]. A large consortium of European researchers, comprising numerous universities and institutions, proposed the Decentralized Privacy-Preserving Proximity Tracing protocol that leverages BLE technology to track and log encounters with other users. The contact logs are never transmitted to a central authority, but they are stored only on the client's device. When a user tests positive, his/her ephemeral IDs are transmitted to the central authority. The IDs are generated with symmetric key protocols, such as HMAC-SHA-256 and AES-128-CTR. Finally, the project is completely open-source.

Apple/Google [11]. Similar to DP-3T, Apple and Google agreed on a decentralized protocol for contact tracing, based on BLE technology. The contact tracing logs do not contain any private information, and ephemeral IDs are only stored on the user's device. From the cryptographic perspective, they adopt HMAC-SHA-256 and AES-128. Note that, Apple/Google is not a complete contact tracing solution; instead, the companies released the exposure notification API as open-source to allow public health authorities to develop their own mobile applications. For example, *Immuni* [13] is the Italian State-sponsored official contact tracing app that leverages the Apple/Google framework.

Hamagen [5]. Hamagen was developed by Israel's Ministry of Health to monitor the COVID-19 pandemic. It allows the identification of positive patients and people who came in contact with them. Hamagen continuously monitors and logs the user's GPS coordinates on the device (requiring no interaction with other devices). After a user tests positive, and if he/she gives prior consent, their location data is transmitted to the Ministry of Health. All devices periodically download the up-to-date location data and compare them against their own GPS history logs.

PEPP-PT [12]. The Pan-European Privacy-Preserving Proximity Tracing protocol adopts BLE to discover and store locally the ephemeral IDs of devices that are in proximity. Similar to BlueTrace, it uses the hybrid architecture by having the health authorities generate the users' beacons. As such, a centralized server collects and processes the contact logs from infected users, and performs the reconciliation process in a centralized manner. The main cryptographic algorithm they employ is AES. This approach also adopts the open-source paradigm.

Solutions Comparison. Table I presents a quantitative comparison of these state-of-the-art protocols for a variety of metrics, such as privacy and operational cost. In our analysis, we consider the health authorities as trusted entities. Otherwise, centralized and hybrid protocols cannot offer any meaningful level of privacy. In terms of health status privacy, decentralized protocols fail to protect the identity of the infected users, which is a violation of numerous health privacy acts, such as HIPAA and GDPR. Specifically, DP-3T and Apple/Google disclose all the ephemeral IDs that belong to the infected users, which allows an adversary to infer with certainty whether a known ID (i.e., person) has contracted the virus. As for hybrid solutions (BlueTrace and PEPP-PT), they only reveal the user's contact logs and are, thus, more privacy-preserving. However, the ephemeral ID of the infected individual might be inferred from its absence within a cluster of IDs with the same time/location tags. Hamagen is a GPS-based solution, so it reveals the infected user's entire location history. While the identity of the user may not be immediately clear, background knowledge can be applied to link the published trajectories to a specific individual.

Regarding location privacy, both the decentralized and hybrid protocols offer excellent privacy to the users who never test positive. This is due to the unidirectional flow of information, i.e., the devices only download data from the central authority's server without ever uploading any data of their own. However, a user who tests positive has to disclose some relevant information to the central server. Usually, the cited disclosure involves publishing ephemeral IDs, contact

Table I
COMPARISON OF STATE-OF-THE-ART REPRESENTATIVE SOLUTIONS. n: CONTACT LIST SIZE, l: NUMBER OF STORED LOCATIONS, f: TX FREQUENCY, NONE: –, LOW: ⋆, MEDIUM: ⋆⋆, HIGH: ⋆ ⋆ ⋆. FOR IOTRACE, SOME METRICS INCLUDE TWO RATINGS THAT CORRESPOND TO THE BASIC: ▲ AND PRIVACY-ENHANCED: ■ VERSIONS.

Features	BlueTrace [9]	DP-3T [10]	Apple/Google [11]	Hamagen [5]	PEPP-PT [12]	IoTrace	
Wireless Technology	Bluetooth	Bluetooth	Bluetooth	GPS	Bluetooth	Bluetooth	
Open-Source	Yes	Yes	Yes	Yes	Yes	Yes	
Architecture (C/D/H)	H	D	D	D	H	C ▲	D ■
RF Energy Consumption (mJ/min)	$\approx 1.23 \cdot 10^3$	$\approx 1.21 \cdot 10^3$	$\approx 1.21 \cdot 10^3$	$\approx 2.19 \cdot 10^3$	$\approx 1.21 \cdot 10^3$	≈ 3.2760	
Security Level (Crypto)	⋆ ⋆ ⋆	⋆ ⋆ ⋆	⋆ ⋆ ⋆	N/A	⋆ ⋆ ⋆	⋆ ⋆ ⋆	
Health Status Privacy	⋆	–	–	⋆	⋆	⋆⋆	⋆ ⋆ ⋆
Location Privacy (w.r.t Positive)	–	–	–	–	–	–	⋆ ⋆ ⋆
Location Privacy (w.r.t Negative)	⋆ ⋆ ⋆	⋆ ⋆ ⋆	⋆ ⋆ ⋆	⋆ ⋆ ⋆	⋆ ⋆ ⋆	⋆ ⋆ ⋆	
Device Storage Requirements (B)	$\approx n \cdot 140$	$\approx n \cdot 24$	$\approx n \cdot 16$	$\approx l \cdot 10$	$\approx n \cdot 30$	0	
Crypto Computational Cost (ms)	0	≈ 24.8973	≈ 30.2039	0	0	≈ 23.3652	
Broadcast TX Overhead (B)	$f \cdot 140$	$f \cdot 24$	$f \cdot 31$	0	$f \cdot 30$	$f \cdot 16$	

logs, or GPS coordinates, unfortunately leading, among others, to a complete compromise of the geographic locations that the user has visited in the near past.

To assess the performance of the discussed solutions in a quantitative manner, we considered the Bluetooth SoC nRF51822 and the GPS SiP nRF9160 (for Hamagen) hardware platforms. We first estimated the energy consumption related to the RF operations (TX and RX), using the platforms' operational specifications, such as voltage and current consumption. For the BLE-based protocols, we assumed a beacon broadcast interval of 500 ms, and a duty cycle of 50% for the scanning function. The energy consumption of each approach is computed as the integral of power over time. For Hamagen, we assumed continuous scanning in low-power mode. As presented in Table I, IoTrace is orders of magnitude more efficient than the competing approaches, because it does not need to scan the Bluetooth channel for broadcasted beacons. As per the crypto operations for generating the ephemeral IDs, they are very cheap for all protocols, necessitating ≈ 30 ms to generate the IDs for an entire day (on a Cortex M0 CPU). However, IoTrace is considerably more lightweight, as it does not need to store and actively update a contact list. For the same reason, IoTrace sports the lowest storage requirement.

III. THREAT MODEL

In this work, we consider a powerful eavesdropping adversary that is capable of collecting all beacons transmitted by the users. The adversary is equipped with a powerful antenna, which can be either a regular Bluetooth handheld device or a Software Defined Radio (SDR) that is operated through a laptop/smartphone running an SDR-compatible software tool. Additionally, the attacker tags every beacon with a timestamp and the geographic location where it was recorded. As a result, the adversary has a global view of all communications and can pinpoint every beacon to a unique point in space and time, although the beacon cannot be linked to a specific user. We also consider a more involved eavesdropping adversary that is able to get close to a target victim, in order to record beacons that belong to the victim with a very high probability (i.e., there are no other devices in the vicinity, or the adversary uses a directional antenna). Such an adversary is only interested in identifying beacons that are associated with one or more unique individuals.

Finally, we embrace a standard assumption in the literature: the adversary runs in polynomial time and is unable to break the cryptographic protocols (such as symmetric encryption and hashing) that generate the pseudo-random beacons. Based on the aforementioned adversarial model, we consider two types of privacy attacks against the contact tracing system:

- **Location privacy attack:** In this attack, the adversary's objective is to track the movements of one or more users through the collected beacons.
- **Health status privacy attack:** Here, the objective is to correctly infer whether one or more *known* users have contracted the COVID-19 virus.

IV. EDGE CONTACT TRACING WITH IOT DEVICES

The novelty of IoTrace lies in the deployment of IoT devices that support the contact tracing tasks at the network's edge, complementing the individual mobile devices. In what follows, Section IV-A introduces the IoTrace architecture, and Section IV-B describes the contact tracing protocol and the corresponding message flow in the context of a centralized architecture. Section IV-C discusses an alternative fully distributed approach that also provides a high level of privacy with the use of public-key cryptography.

A. System Architecture

The entities involved in the IoTrace architecture are the following:

- *User.* A user carrying a smartphone device that runs our contact tracing app. The app simply transmits BLE beacons (pseudo-random ephemeral IDs) that are received by the deployed IoT devices. The transmitted beacons are also stored locally on the device for verifying proximity to other users. Unlike existing approaches, the app only operates in transmit-mode, i.e., it does not collect BLE beacons from other devices.
- *Totem.* This is an IoT smart device, equipped with a BLE transceiver that collects the beacons transmitted from the users' devices. We also assume that the totem maintains a secure intranet connection to the central authority,

where it forwards all the received beacons—in a fashion that could span from batch mode to real-time. In our terminology, we call these beacons *negative*, i.e., they belong to users who have not tested positive. From a practical perspective, a totem could be a simple low-end device, like a Raspberry Pi.

- *Hospital.* This is a medical center that tests users who may possibly have a COVID-19 infection. If a user tests positive, the health professionals are permitted to access his/her mobile device and forward the stored beacons to the central authority. We call these beacons *positive*.
- *Central authority.* This is a trusted party, whose role is to collect the positive and negative beacons sent by the corresponding hospitals and totems. It is assumed to be always online and ready to provide an updated list of beacons that belong to users who had a close contact with an infected user. In a real scenario, this role can be played by the *Ministry of Health*.

As shown in Fig. 1, the proposed architecture can be adopted in open spaces, like parks, or in closed spaces, like shopping malls and offices.

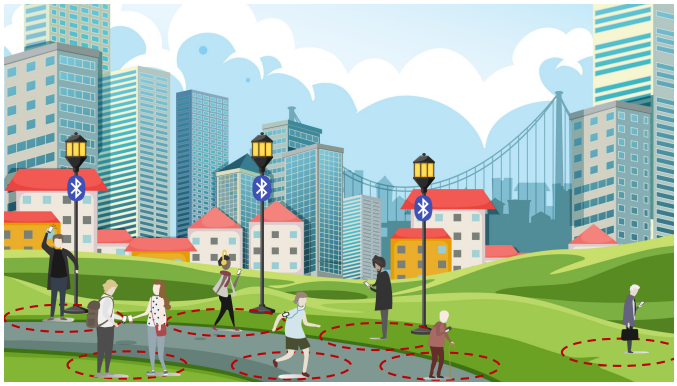
B. Protocol Message Flow

IoTrace's protocol is illustrated in Fig. 2 and summarized below:

- Let us assume a generic time-slot t_i. At the beginning of the time-slot, the transmitting user (Alice) generates a pseudo-random BLE beacon, according to some cryptographic primitive, such as AES-128 encryption.
- After collecting all beacon(s) within time-slot t_i, the totem forwards them to the central authority. The authority stores each beacon as a tuple <*totem-ID, time-slot, beacon*> on its long-term memory.
- Let us assume Alice is diagnosed as positive at an authorized hospital. An authorized health official will access Alice's mobile application to send her recent beacons (e.g., from the last two weeks) to the central authority.
- Consider one of Alice's positive beacons that was transmitted at time-slot τ. The central authority will identify all the negative beacons within time-slots $\tau \pm \epsilon$ (at that particular totem), where ϵ is a time window that depends on the broadcast frequency of the beacons. The list of all negative *and* positive beacons is published online.
- Finally, Bob downloads from the central authority the list published at the previous step and checks whether his own beacons are in the list. If there is a match, and it is sustained for an amount of time sufficient to declare a potential contagion risk (set by the health authorities), Bob is notified by the app of this possibility.

The data flow that summarizes the above described operations is depicted in Fig. 3.

Compared to previous approaches in the literature, this basic version of IoTrace already provides better protection of the users' health status privacy, since both the positive and the negative beacons are disclosed by the central authority. As a result, IoTrace provides k-anonymity [14] in terms of health status privacy. That is, if k beacons are published on behalf of a single totem, each beacon has a $1/k$ chance of being the positive one. As per the location privacy guarantees, they are identical to existing decentralized solutions, such as DP-3T and Apple/Google. However, IoTrace has a clear advantage in terms of operational cost for mobile devices.



(a) Open space: Park.

(b) Closed space: Shopping mall.

(c) Closed space: Office.

Figure 1. IoTrace infrastructure in different environments: open and closed spaces. A totem is represented by the Bluetooth icon. BLE beacons are transmitted from the smartphones to the IoT totems.

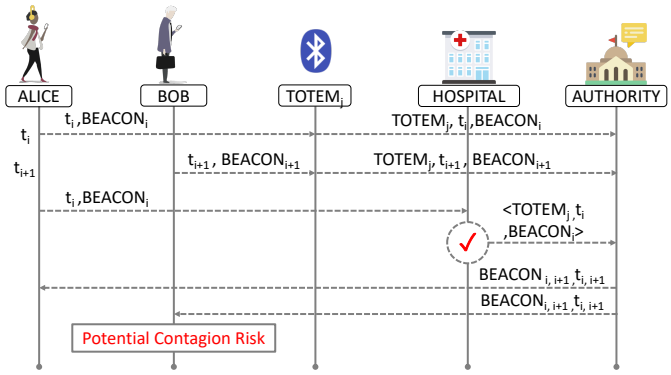


Figure 2. Sequence diagram of the IoTrace protocol. The authority marks Alice as infected by reporting an alert of Potential Contagion Risk to Bob.

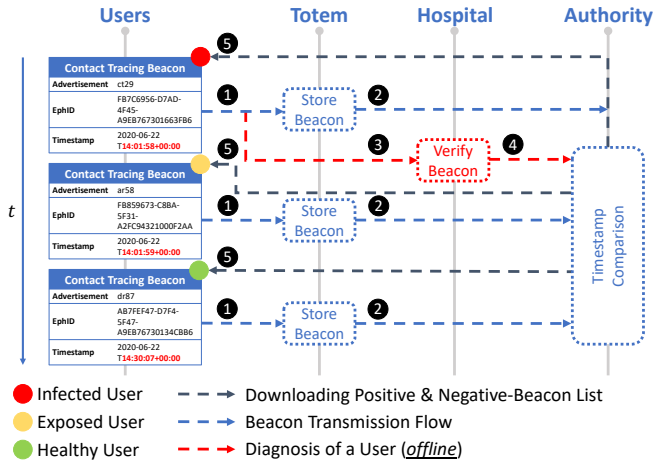


Figure 3. The IoTrace protocol data flow diagram. An exposed user becomes yellow if he/she is close to an infected user based on the BLE beacon's timestamp.

C. A Privacy-Enhanced Solution

We will now show how to significantly enhance the privacy under IoTrace, while leveraging the same architecture. The first improvement is related to the centralized storage of all beacons. To this end, IoTrace can operate in a fully decentralized mode, i.e., the totems will store the received beacons locally, without sending them to the central authority. When a user tests positive for COVID-19, the central authority will forward the positive beacons to all totems and, in turn, the totems will send back to the central authority all negative beacons that fall within the predetermined time window from a positive one. This approach preserves the privacy guarantees and operational costs for mobile devices while removing the inherent risks of centralized storage.

Our second improvement comes with increased computational and power consumption costs for the mobile devices, but results in a contact tracing solution which is secure against eavesdropping adversaries. The key observation is that the IoT infrastructure is relatively static, so it is easy to store on each mobile device the list of all totem IDs, along with their public-key certificates. Then, instead of transmitting their beacons in cleartext, the mobile devices will first exchange a symmetric key with the totem using the locally stored certificate, and then send their beacons encrypted with that key. The totem will locally decrypt the beacons, and the protocol will continue as described. Consequently, when the positive/negative beacons are published by the central authority, an eavesdropper cannot link them to a particular totem and time-slot. Overall, this latter solution is very flexible, allowing individual users to trade off more privacy with a higher computing cost.

V. CHALLENGES AND ROAD AHEAD

Contact tracing is, in essence, a surveillance-type application. As such, the security and privacy of the entire system are of paramount importance. In the following sections, we describe the challenges that must be addressed, to make edge contact tracing a secure and privacy-preserving solution.

A. Security Considerations

Edge Security. In the proposed architecture, an IoT device (totem) represents the edge component between the mobile devices and the hospital/authority. Hence, a research direction relevant to our solution, but also of general interest to the IoT domain, arises from the need to reduce the required computations, for instance, adopting lightweight cryptographic protocols to meet the intended security and privacy goals. The most obvious concern with regards to the security of the proposed architecture is the exposure of the totems to physical attacks, due to their being unattended. As a result, no sensitive information, such as user beacons or private keys, should be stored in plaintext format. To solve the cited issue, data at rest could be encrypted with the public key of the central authority. Furthermore, the totem should utilize a secure enclave to perform the necessary cryptographic operations, and all beacons (even when encrypted, as suggested above) should be erased as soon as they are received by the trusted authority. For the case of the fully distributed architecture where the data are stored locally at the totems, additional measures should be implemented to harden their security.

Replay and Relay Attacks. These are active attacks where the adversary eavesdrops on the broadcast beacons and then replays those beacons to many other (even far away) totems. The objective of these attacks is to generate a large number of false contacts such that, if one individual tests positive, the disclosure of his/her beacons will trigger many false positive alerts. Such attacks can be addressed in two different ways. First, the beacon generation protocol may incorporate certain cryptographic protocols to thwart replay attacks. Second, the trusted authority can analyze the collected data and identify fraudulent beacons, e.g., the same beacon appearing in two distant locations in a not time-congruent manner.

B. Privacy Considerations

Linkage and Profiling. Contact tracing protocols and applications bring with them several privacy concerns, e.g., the misuse of the collected data at the trusted authority, under the centralized and hybrid models. Indeed, a malicious insider

with access to all beacons, locations, timestamps, and contact lists, can extract sensitive information about the underlying individuals (such as locations visited, routes, social contacts, etc.). Our proposed architecture makes such attacks less feasible by design, since users do not submit their own contact lists. Instead, all the beacons are aggregated at the distributed totems, which makes it much harder for an adversary to track individuals. Still, an interesting research direction would be to quantify the privacy leakage under the centralized edge contact tracing architecture.

Eavesdropping. Eavesdropping is a passive attack where an adversary with the ability to eavesdrop at a large scale can simply record most beacons that are broadcast by the users. When the list of positive/negative beacons is published, the adversary can identify all the locations that the infected user has visited. This is an attack that none of the existing contact tracing protocols can defend against. To this end, a possible research direction would be to design secure two-party protocols (between the trusted authority and a user) that allow users to blindly match their beacons against the server's beacon list (which will not be published). Note, however, that our proposal is able to thwart such attacks when the user employs the public-key of the totem to bootstrap a secure channel with the totem itself—as described in Section IV-C.

C. Technology Considerations

Localization Accuracy. Most technologies adopted for contact tracing rely on the Received Signal Strength Indicator (RSSI). With the help of a radio-propagation model, this feature is useful in estimating the distance between the transmitter and the receiver nodes. Unfortunately, several factors can affect the accuracy of distance estimation, including radio noise, obstacles, multipath reflection and shadowing effects, or environmental factors like rain, temperature, and humidity. Therefore, Bluetooth RSSI may produce a large number of false positives and false negatives. To this end, alternative features like Angle-of-Arrival, Time Difference of Arrival, and Time of Arrival should be investigated. Furthermore, thanks to the vast amount of available data, AI algorithms could be employed on the edge devices to improve the localization accuracy of the Bluetooth technology.

Communication Technologies. While BLE is the de facto choice for all contact tracing solutions in the literature, we believe that more research is needed on different communication technologies. In particular, Ultra-wideband (UWB) carriers as well as acoustic channels and ultrasonic sound waves could be employed to improve the accuracy, privacy, and reliability of proximity tracing [15].

D. Social Considerations

Accessibility. IoTrace shifts a significant portion of the energetic and computational costs of contact tracing to the IoT edge devices and/or the centralized server. As a result, the corresponding mobile application can be easily deployed on low-cost devices that would otherwise be unable to participate in the contact tracing network. This will increase considerably the accessibility of the solution to the general public.

Usability. The usability of existing solutions is primarily hindered by the shortened battery life that users experience. We have shown that energy consumption under IoTrace is reduced by multiple orders of magnitude. Additionally, the reconciliation process is mostly performed at the health authorities and/or IoT devices. As such, we argue that IoTrace's mobile app would be extremely lightweight, and therefore, would not affect the user's experience—hence increasing the chance of adoption.

Trust. In addition to usability, trust (or the lack of) is the deciding factor that discourages people from actively using existing contact tracing apps. To this end, IoTrace's superior privacy guarantees could motivate more users to install and actively use the app. Furthermore, by releasing the app's code as open-source, we can further ease the public's concern with respect to privacy and security.

E. Limitations

The major limitation of IoTrace is the cost to deploy, operate, and maintain the IoT infrastructure. Indeed, the IoT devices must be connected to a fixed power supply and have access to a cellular/cable network infrastructure, in order to communicate with the health authorities. As such, we envision that a practical implementation would employ cheap, Raspberry Pi like devices, that would cost somewhere between \$10–\$20 each. For $100,000$ devices, the cost would rise to a couple of million dollars, which is very reasonable for a large city. We should emphasize that IoTrace would only be deployed in crowded areas, such as shopping malls, public transportation venues, airports, stadiums, parks, etc. Additionally, the government may offer incentives to individual business owners to install and maintain their own IoT devices, thus expanding the range of IoTrace's network.

Despite the cited costs, IoTrace has the following advantages that make it a very attractive solution for contact tracing: (i) significant energy savings for the mobile devices, as they can operate in transmit-only mode; (ii) superior privacy guarantees; (iii) better proximity tracing accuracy stemming from a moderately dense deployment of IoT sensors (improved localization with techniques like triangulation and trilateration); (iv) reduced computational and storage requirements for the mobile devices, allowing the app to work seamlessly on cheap devices; (v) flexibility on behalf of the health authorities, because IoTrace does not enforce any constraint on the distance (or duration) that qualifies a digital encounter as a legitimate contact.

VI. Conclusion

In this paper, we proposed IoTrace, a novel IoT-based architecture for contact tracing, that addresses some of the most important limitations of existing solutions: it provides a balance between the level of privacy for the different user categories; it reduces the overhead on the end-user device in terms of energy consumption and computational cost; it enhances location privacy; and, it is scalable and flexible—allowing to accommodate different contact tracing models,

from the purely decentralized to the centralized one. We believe that the novelty of the proposal, as well as its striking properties and flexibility, has the potential to pave the way for further research.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers, that helped improving the quality of the paper. This publication was partially supported by awards NPRP 11S-0109-180242 from the QNRF-Qatar National Research Fund, a member of The Qatar Foundation. The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of the QNRF.

REFERENCES

[1] John Hopkins University, "Coronavirus Resource Center," https://coronavirus.jhu.edu/, November 2020, (Accessed: 2021-1-1).
[2] D. Shu Wei Ting *et al.*, "Digital technology and COVID-19," *Nature Medicine*, vol. 26, no. 4, pp. 459–461, Mar. 2020.
[3] J. M. Cecilia *et al.*, "Mobile crowdsensing approaches to address the COVID-19 pandemic in Spain," *IET Smart Cities*, vol. 2, no. 2, pp. 58–63, 2020.
[4] L. Garg *et al.*, "Anonymity Preserving IoT-Based COVID-19 and Other Infectious Disease Contact Tracing Model," *IEEE Access*, vol. 8, pp. 159 402–159 414, 2020.
[5] Israeli Health Ministry. (2020) Hamagen. (Accessed: 2021-1-1). [Online]. Available: https://govextra.gov.il/ministry-of-health/hamagen-app/download-en/
[6] Q. Tang, "Privacy-Preserving Contact Tracing: current solutions and open questions," *IACR Cryptol. ePrint Arch.*, vol. 2020, p. 426, 2020.
[7] L. Baumgärtner *et al.*, "Mind the GAP: Security & Privacy Risks of Contact Tracing Apps," *arXiv e-prints*, Jun. 2020.
[8] Y. Lu *et al.*, "Internet of Things (IoT) Cybersecurity Research: A Review of Current Research Topics," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2103–2115, 2019.
[9] J. Bay *et al.*, "BlueTrace: A privacy-preserving protocol for community-driven contact tracing across borders," *Government Technology Agency-Singapore, Tech. Rep*, 2020.
[10] "Decentralized Privacy-Preserving Proximity Tracing: Overview of Data Protection and Security," https://github.com/DP-3T/documents/blob/master/DP3T%20White%20Paper.pdf, 2020, (Accessed: 2021-1-1).
[11] Apple Google. (2020) Privacy-Preserving Contact Tracing. (Accessed: 2021-1-1). [Online]. Available: https://www.apple.com/covid19/contacttracing
[12] PEPP-PT Team. (2020) Pan-European Privacy-Preserving Proximity Tracing. (Accessed: 2021-1-1). [Online]. Available: https://www.pepp-pt.org/
[13] Italian Ministry of Health, "Immuni," https://www.immuni.italia.it/, June 2020, (Accessed: 2021-1-1).
[14] J. Wang *et al.*, "Achieving Personalized k-Anonymity-Based Content Privacy for Autonomous Vehicles in CPS," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4242–4251, 2020.
[15] M. Caprolu *et al.*, "Short-Range Audio Channels Security: Survey of Mechanisms, Applications, and Research Challenges," *IEEE Communications Surveys Tutorials*, pp. 1–1, 2020.

BIOGRAPHIES

Pietro Tedeschi is PhD Student at HBKU-CSE. He received his Master's degree with honors in Computer Engineering at Politecnico di Bari, Italy. He worked as Security Researcher at CNIT, Italy, for the EU H2020 SymbIoTe. His research interests cover security issues in UAVs, Wireless, IoT, and Cyber-Physical Systems.

Spiridon Bakiras is associate professor of cybersecurity at HBKU-CSE. His research interests include Security and Privacy, Applied Cryptography, and Spatiotemporal Databases. He held teaching and research positions at Michigan Technological University, the City University of New York, the University of Hong Kong, and the Hong Kong University of Science and Technology. He is a recipient of the U.S. National Science Foundation CAREER award.

Roberto Di Pietro, ACM Distinguished Scientist, is full professor of Cybersecurity at HBKU-CSE. His research interests include Distributed Systems Security, Wireless Security, OSN Security, and Intrusion Detection. In 2011-2012 he was awarded a Chair of Excellence from University Carlos III, Madrid. In 2020 he received the Jean-Claude Laprie Award for having significantly influenced the theory and practice of Dependable Computing.