Properties of contact tracing systems in low-income settings

Martin Hanzel
313710

Wouter Lueks
Supervisor

Autumn 2021

1 Introduction

The high virulence and transmissibility of the SARS-CoV-2 virus has highlighted the need for reliable methods of contact tracing in order to slow down the spread of the virus. Global efforts have gone into designing robust and privacy-preserving protocols such as DP3T [24], GAEN [18], Bluetrace [3], and Desire [5].

In general, digital contact tracing systems function by discovering and storing information about proximity contacts, then comparing these contacts to a list consisting of positive COVID-19 cases. If a user's proximity contacts in the recent past place the user at risk of infection, i.e. by prolonged exposure to a COVID-positive person, the user is notified. The most common way of discovering proximity contacts is the exchange of random tokens between users' devices over a short-range, wireless medium, such as Bluetooth.

Smartphones are very convenient devices for deploying digital contact tracing systems. They provide rich wireless communication functionality, support downloadable applications, and are ubiquitous in high-tech communities. However, in places where smartphones have not penetrated into daily life, including low-income areas, the utility of smartphone-based contact tracing systems is severely reduced. The COVID-19 pandemic naturally affects these places as well, and so there remains a need for contact tracing systems that do not rely on smartphones.

In a smartphone-less digital contact tracing system, we envision that every person is issued a compact digital "token", such as a wristband or keychain. These tokens would possess a radio and implement a digital contact tracing protocol similar to those in smartphone apps. Tokens would advertise over a short-range wireless medium such as Bluetooth or UWB (Ultra Wide Band) to discover proximity contacts, and connect to health authorities via local wide-area wireless networks to download positively-diagnosed cases and upload diagnosis results when necessary. We base the behaviour of this system on the decentralized DP-3T protocol.

The social and technical challenges of digital contact tracing in low-income settings are different than those which the scientific community has explored thoroughly in developed countries. In this report, we will explore how the privacy of contact tracing systems is affected by compromises in technology, infrastructure, and trust that are unique to low-income settings. We first re-visit the question of digital contact tracing systems, putting forth a list of properties that the system must have. Then, we present a threat model for the short-range and wide-area networking aspects of the system, and discuss possible vulnerabilities and mitigations.

2 Properties

At a high level, the properties of a smartphone-less contact tracing system are mostly identical to those in smartphone-based systems like DP3T [24].

We define the term **proximity contact** as follows: a proximity contact is a person who was present within a certain distance and for a certain minimum time to another person. Contact tracing systems identify and store proximity contacts, generating an exposure notification if at least one of a user's recent proximity contacts tests positive for COVID-19.

Two fundamental properties cover the guaranteed and accurate reception of exposure notifications:

Liveness/completeness If a person receives a positive COVID-19 diagnosis and notifies the contact tracing authority of this result, all recent proximity contacts should eventually, and within a reasonable time, receive an exposure notification.

Safety/authenticity A user is never notified if they have never been a proximity contact of a positively-diagnosed user. [1] In other words, exposure notifications cannot be "faked".

We extend the *liveness* property to cover denial of service:

No denial An adversary must not be able to deny or delay the correct and timely functioning of the contact tracing system for another user.

Similarly, we extend the *authenticity* property to cover attacks that do not lead to exposure notifications:

No impersonation It must be impossible for an adversary to impersonate another user.

[1] We assume that all users are dutiful and report positive diagnoses to the contact tracing authority

2.1 Privacy properties

Privacy properties ensure that as little as possible user information is revealed under normal functioning of the contact tracing system. Many of these properties follow from the DP3T whitepapers [19, 24] and the CrowdNotifier paper [16].

No location tracking It must not be possible for an adversary to learn a user's location history.

No diagnosis confirmation It must not be possible for an adversary to determine whether a particular user received a positive diagnosis.

No notification confirmation It must not be possible for an adversary to determine whether a particular user received an exposure notification.

No contact history confirmation It must not be possible for an adversary to learn the contact history of a user.

Hotspot privacy It must not be possible for an adversary to determine the exposure rate at a given location.

No healthy-state inference When all users are healthy (i.e. COVID-negative), an adversary must not learn any more information from observing the system than if the system were non-existent.

These properties are for an "ideal" contact tracing system. In reality, the *liveness* property contradicts some privacy properties in certain situations. For example, a user who receives a positive diagnosis and who remembers perfectly their own contact history will know who else will receive an exposure notification. Contact tracing systems have inherent risks that violate the ideal properties; Implementation details, political pressure, and concerns about trusting vendors can jeopardize the fundamental and privacy properties in even more ways [8, 19, 25, 26].

2.2 Non-functional properties

We also define several non-functional properties that relate to the user-friendliness and deployment of a contact tracing system, particularly in low-income or low-tech places.

No data collection Beyond initial setup, the system must not require input of user data in order to function fully.

Autonomy Besides carrying some physical beacon and ensuring that it is charged and turned on at all times, the system should require no intervention or interaction by the user to function.

Handiness The physical beacon should not be cumbersome to carry for a full day.

Battery life The beacon should be able to function for at least one full day of regular use, and ideally much longer.

Infrastructure The system should use readily-available or off-the-shelf infrastructure solutions, e.g. for wireless communications.

Low resources The system should have a low bandwidth and compute power requirements. It should function on low-power hardware and on dense, bandwidth-limited networks.

The last two properties stem from the fact that not all users in a low-income setting would have smartphones, and that wireless infrastructure may be antiquated, underdeveloped, or overburdened.

3 Threat model

A contact tracing system operates over three separate media:

1. A short-range advertising medium, on which users' devices exchange data to discover proximity contacts;
2. A wide-area wireless medium that devices use to connect to a "backbone" network;
3. A backbone network such as the Internet that connects all user devices and necessary infrastructure.

We consider threats on the first two media.

3.1 Short-range advertising

The core of our contact tracing system lies in beacons advertising their presence by transmitting over a short-range, wireless medium such as Bluetooth or Ultra Wide Band (UWB). Adversaries may eavesdrop on these transmissions or inject their own, but are nonetheless limited to the useful range of the medium, forcing adversaries to have a physical presence in order to attack targets.

We classify adversaries on this medium on two axes:

1. **Level of access** to the medium. Adversaries with *physical layer access* may directly observe modulated signals over the air. This level of access requires purpose-built receivers or software-defined radios (SDRs). Adversaries with *application layer access* may only observe messages through a high-level API, as is the case with a smartphone application, for example. Application-layer adversaries cannot directly attack or exploit the physical medium itself.
2. **Scale of deployment** of attacks. Adversaries working on a large scale can carry out (possibly concurrent) attacks across a large segment of the population or over a wide geographical area. Adversaries on a small scale are limited to a small physical area. This area may be stationary or mobile.

3.1.1 Application layer, small scale

Small-scale, application-layer adversaries are the weakest adversaries we consider. Such adversaries are assumed to have access to off-the-shelf, widely-available consumer hardware such as smartphones or laptops. They are also sufficiently tech-savvy to observe or reverse-engineer the protocols and applications in use by the contact tracing system.

Small-scale adversaries are able to eavesdrop on transmissions, inject their own traffic, replay traffic, deny service by flooding the medium, and collect data based on transmissions made only a short distance away. They do not have a global view of events beyond the transmission range. Application-layer adversaries potentially can link two sequential transmissions by signal strength and transmission time, but the probability of linking two transmissions decreases with greater time between those transmissions.

Examples of adversaries include: a curious and tech-savvy user who uses a wireless adapter to observe transmissions; an actor who floods the airwaves with bogus messages, denying service to other users in that area; or a vengeful individual who transmits malicious (known/suspected COVID-positive identifiers) to a specific target.

3.1.2 Application layer, large scale

Large-scale adversaries extend the above with a global perspective. They have somehow deployed a network of transmitters/receivers across a wide area, allowing them to target large segments of the population across different demographic groups. The adversary is limited to harvesting application-visible data (such as ephemeral identifiers or MAC addresses) but can potentially observe them over a larger geographical or temporal space, exposing the possibility of mass location tracking.

An active adversary can perform relay and replay attacks at the application layer, wherein they *relay* received data in real-time over long distances through a tunnel, or *replay* some observed data at a later time. Either of these attacks violates the *no impersonation* property if the relayed message is accepted as valid by the receiver and the message was not intended to be forwarded (e.g. in a gossiping circumstance). Two-way handshakes and distance bounding may provide a degree of protection by rendering invalid relayed/replayed messages.

An example of a large-scale, application-layer adversary would be a malicious smartphone app developer, who publishes an attractive, cheap/free app that is installed by many people, such as a game or social network. Worryingly, such apps are pre-installed on many Android devices (e.g. Facebook) and applications may update their permissions, including the permission to access Bluetooth, without notifying the user [7].

3.1.3 Physical layer, small scale

Small-scale, physical layer adversaries have access to hardware capable of capturing physical signals or link-layer frames.

Physical layer adversaries can perform radio fingerprinting, allowing them to link two transmissions from the same device. The likelihood of this attack is invariant to the time between transmissions, but subject to some luck and ambient temperature conditions [11, 15]. Ironically, radio fingerprinting on a small scale can be more effective than at a global scale, since there are fewer ambiguous signals and receivers are more likely to be stationary and in indoor, temperature-controlled environments.

Examples of adversaries are individuals, businesses, landlords, abusive partners, etc. who purchase an SDR to track targets via radio fingerprinting on their own properties.

3.1.4 Physical layer, large scale

Large-scale, physical layer adversaries are the most powerful and problematic. They have a global view of physical or link-layer transmissions across a wide segment of the population. There are many examples of such adversaries, and many pathways for a sufficiently motivated and well-financed actor to become one, with varying degrees of scale:

- **Hardware vendors** (particularly those of wireless networking adapters or appliances, which are ubiquitous in buildings and consumer electronics) can incorporate SDRs in their products. The privacy risk of hardware vendors becoming adversaries is tempered by the business risk of becoming exposed as an eavesdropper.
- **Infrastructure operators** can install eavesdropping devices on their properties. This kind of adversary is not necessarily a telecommunications operator — transit companies can eavesdrop within their vehicles or stations; utility providers can install listening devices on utility poles; building owners can install SDRs throughout their properties; governments can install devices at roadways, parks, voting places, and other such "hotspots".
- **Bribers** can collude with or hoodwink location owners to install SDRs on-premises, constructing a network of receivers at chosen locations. Of particular concern are politically-minded organizations [9], who are not necessarily global adversaries on their own but must collude with others to further their agenda.
- Similarly, **state-level adversaries** can compel parties to install listening devices.

3.1.5 Hackers

We can also consider the possibility of *hackers* who compromise devices and use them to eavesdrop. The level of access depends on the kind of compromised device. A wireless access point, for example, may expose physical or link-layer

characteristics, but an IoT device probably falls in the application layer access category.

3.2 LoRaWAN

Contact-tracing devices must possess some connectivity to a larger network, at minimum to carry out the following tasks: *downloading* data to update its database of positively-diagnosed cases and to synchronize its clock, and *uploading* its own identifiers or keys in the case that its owner receives a positive diagnosis.

We consider as our primary choice of wide-area network LoRaWAN [2], a long-range, low-power, wireless network aimed at IoT (Internet of Things) systems. We choose LoRaWAN for its high availability even in low-income settings, as well as its ability to be geographically scaled by volunteer radio operators. LoRaWAN acts an analogue to cellular data networks in smartphone-based contact tracing systems such as GAEN. [3] However, in contrast to cellular networks, LoRa and LoRaWAN are less power-demanding, easier to interface with, easier to scale, and service a much larger geographical area.

3.2.1 How does LoRaWAN work?

In this section, we summarize how LoRaWAN works in a few paragraphs. For a more in-depth description of LoRaWAN's architecture and protocols, see the official specifications [14, 21].

LoRaWAN specifies a wireless network infrastructure from the link to the session layer. *End Devices* (EDs; the "tokens" owned by users) are connected to a network in a star-of-stars topology. EDs communicate wirelessly with one or more *gateways*, which are appliances operating exclusively on the physical layer. Gateways proxy messages between EDs and a *Network Server* (NS), which terminates the link layer. Two other servers are involved here: a *Join Server* (JS) which handles EDs joining and leaving the network; and an *Application Server* (AS), which provides application-level services such as databases or message brokers. Multiple applications may run on the same LoRaWAN network — the NS routes data between EDs and the AS to which the EDs are provisioned. Links between gateways and the three servers are over IP.

The NS-JS-AS triad is centralized. The NS and possibly JS are controlled by a network operator, and can be considered as analogues of cellular network exchanges. The gateways can be owned by the network operator, but can also be set up by volunteers running SDRs with custom software. An implementation based on GNU Radio exists [22]. Finally, ownership of the AS depends on the particular application. The AS may be run on the application owner's own hardware, in the cloud, or on the network operator's infrastructure. The AS may

[2] https://lora-alliance.org/about-lorawan/

[3] A note on vocabulary: LoRa is a low-power modulation technique for radio signals, specifying the physical networking layer; LoRaWAN is the specification for higher layers, specifying point-to-point links, sessioning, and other high-level concerns.

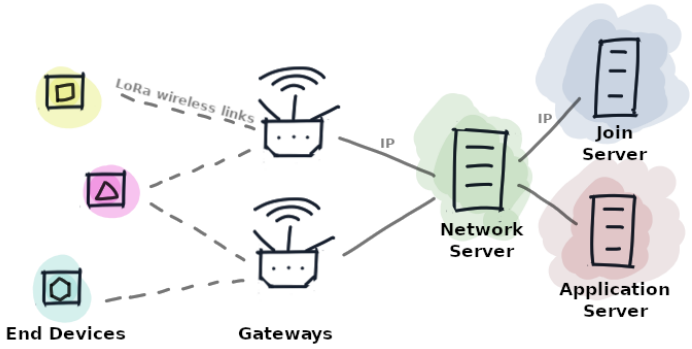

Figure 1: Architecture of a LoRaWAN network.

run custom or third-party software. [4] LoRaWAN is flexible in the deployment of these three servers.

Each ED must be provisioned on the network and to a specific application before joining. When an ED joins the network, symmetric keys are derived that protect the integrity and confidentiality of messages. A device address `DevAddr` is also leased that functions like a session-specific MAC address. LoRaWAN supports two join procedures:

Over The Air Activation (OTAA) is the more common procedure. An ED is provisioned with a globally unique ID `DevEUI` and random application key `AppKey`, which are known to the ED, NS, JS, and the proper AS [5]. When joining, the ED provides its `DevEUI`, then both the ED and network derive a set of symmetric session keys from the shared `AppKey` and other public or transmitted material. `AppKey` is never transmitted. The ED is also assigned a unique `DevAddr` during this step. Both the ED and network contribute random key material during the join procedure.

Activation By Personalization (ABP) is the second provisioning method, where each ED is provisioned with session keys and `DevAddr` ahead of time. The ED is limited to interacting with the specified network and may not re-authenticate or ask for a new session.

LoRaWAN has many potential adversaries. We discuss each below.

3.2.2 Adversaries with access to secrets

When an ED joins the network by OTAA, it derives a set of session keys from a parameter `AppKey`, which is known by the ED and the network. If an ED's `AppKey` is compromised, backward and forward secrecy is lost for that device and

[4] Chirpstack is an example of an off-the-shelf LoRaWAN solution: https://www.chirpstack.io/

[5] in LoRaWAN 1.0. Version 1.1 introduces separate keys for the Network and Application servers which are provisioned in a similar fashion.

an eavesdropper can derive any session keys by observing the joining procedure. `AppKey` is never transmitted over the air, though an adversary may recover one by physically compromising a device, or by leaking them from the database of EDs. `AppKey`s are specific to an ED, so a compromised `AppKey` affects the security of only that device. An adversary can use a stolen `AppKey` to create a fraudulent session and impersonate that user.

Network operators, hosting providers, and third-party software vendors may have access to a network's or application's ED database, and therefore `AppKey`s. Further, in version 1.0 of LoRaWAN, the network operator must possess all `AppKey`s since the NS participates in the activation procedure. Version 1.1 introduces a separate per-device `NwkKey` (Network Key) which is shared between an ED and an NS avoiding the need to share `AppKey` with the network operator.

3.2.3 Passive listeners

A *passive listener* adversary is one who sets up one or more radios and eavesdrops on LoRa traffic. This adversary does not transmit.

LoRa is capable of transmitting over a long range, realistically around 10 kilometers. It is easy for an adversary to install a cheap radio and listen to LoRa traffic over a very wide area. LoRaWAN is also demonstrated to be tolerant to occlusion, so a receiver placed in a crowded place (for example, a city centre) may receive signals from a long distance in spite of buildings or geography in the way [10]. A passive adversary can also effectively determine the position of a transmitter by multilateration with multiple receivers or directional antennae.

LoRaWAN frames include device- or session-specific identifiers in unencrypted headers. A passive listener who collects and stores transmissions may link two packets and reconstruct the ED's location history. EDs activated using ABP are most vulnerable — the device has a `DevAddr` for its entire lifetime. EDs activated using OTA are leased a `DevAddr` which lasts as long as its session. OTA devices can leave and re-join the network at any time, but the joining procedure leaks the ED's `DevEUI`, which is globally unique and cannot be changed without re-provisioning the device. An eavesdropper can correlate `DevEUI` and `DevAddr` by observing packet timings or having prior knowledge of the `AppKey`. There appears to be no way for an ED to appear anonymous to a LoRa network, or untraceably change its address.

LoRaWAN networks can multiplex many applications over the same network. When there are several applications in use, an adversary cannot know to which application an ED belongs, and so cannot with certainty infer membership of an ED or its owner in contact tracing. However, if observable behaviours of EDs vary greatly between applications, the adversary can infer membership with some confidence. This may be the case if EDs belonging to other applications are not mobile (e.g. in a "smart city"), or if the influx rate of new EDs into the LoRaWAN network is very high when a contact tracing application is first introduced.

A passive listener who records traffic may also be able to exploit flaws in LoRaWAN's cryptographic procedures to recover message plaintexts [1, 12, 27].

LoRaWAN version 1.1 resolves some vulnerabilities from version 1.0 but is not perfectly secure [4].

Radio fingerprinting may enable tracking users. Robyns *et al.* [20] demonstrated that radio fingerprinting is effective on LoRa transmitters, with classification accuracy of 59% to 99% on 16 devices with identical chipsets using a supervised learning model. Classification accuracy on devices with varying chipsets is nearly 100%. The authors also mention a zero-shot method that can identify unknown transmitters. A semi-supervised or unsupervised approach may also be viable as for Bluetooth [11]. It is unclear whether a LoRa fingerprinting method scales to hundreds or thousands of devices in a small area.

Because LoRaWAN message headers are unencrypted, eavesdroppers plainly see which devices are transmitting/receiving, and can harvest metadata on encrypted messages. When a user receives a positive COVID-19 diagnosis, she can upload data to the contact tracing authority through her device. Event-driven uploads may allow adversaries to infer which uploads are more likely to be real among dummy uploads [13, 23]. An adversary can exploit location information whenever an upload takes place to strengthen their inference — e.g. uploads will most likely be performed at a user's home or at a testing centre. A scenario might also be possible where an adversary intersects the location histories of users suspected to be infected so as to reveal locations where infections may have occurred at a higher-than-normal rate, violating the *hotspot privacy* property. To protect against such attacks, EDs must make dummy uploads where dummies are perfectly indistinguishable from real uploads, which is a challenging task.

3.2.4 Active listeners

Active listeners extend passive listeners with the ability to transmit signals.

A simple attack for an active listener to perform is radio jamming. The LoRa physical layer and LoRaWAN have few protections against targeted jamming [2]. The success rate of jamming varies in experimental studies but nearly 100% jamming efficiency is possible under realistic scenarios using cheap hardware [1, 2, 17]. It is sufficient for an adversary to change a single bit in the message to fail a CRC check on the receiver, and LoRa's low data rate renders it easy to detect messages and corrupt them before a frame is done transmitting. Further, since LoRaWAN message headers are unencrypted, an adversary can target messages sent or received by chosen victims.

The effectiveness of jamming is reduced in regions with overlapping gateways, as only one gateway needs to receive the intact message for the transmission to succeed. Duplicate messages from different gateways are filtered by the NS. There is a privacy-availability trade-off here — dense networks consisting of "official" and third-party LoRaWAN gateways increase availability while enabling mass eavesdropping or more accurate location-harvesting, while sparse networks are vulnerable to jamming.

Active listeners may also exploit several security and cryptographic vulnerabilities in the LoRaWAN protocol to de-synchronize the state between EDs and

the NS, establish false sessions, or recover plaintexts of messages [1, 4, 12, 27].

3.2.5 Gateway operators

LoRaWAN networks can be built up by volunteer operators who set up cheap gateway devices. This low barrier to entry allows easily extending LoRaWAN networks to country-wide scales without the need for an infrastructure operator to achieve full wireless coverage over large areas [6]. LoRaWAN gateways operate on the physical layer, bridging local EDs to the LoRaWAN Network Server. Gateways communicate with the NS over IP, but can use any medium to do so, including wireless mobile networks or satellite links, and so are not geographically limited to places with wired Internet service.

An operator of a LoRaWAN gateway does not have much more adversarial power over a passive or active listener. A gateway only forwards bits between EDs and the NS — it cannot normally decrypt or alter messages. A gateway can deny service by refusing to deliver frames, but this causes no more damage than wireless jamming described above. Since EDs do not establish sessions with gateways themselves, and frames can reach more than one gateway, the denial effect of a malicious gateway is mitigated as long as there exists another, correct gateway within range.

3.2.6 Network operator/server hosts

Because of LoRaWAN's centralized architecture, the network operator is a powerful, global, and internal adversary. A network operator controls the NS, but depending on the particular deployment, it may also control the JS and AS. If servers are hosted on third-party infrastructure (e.g. in the cloud), the infrastructure owner also becomes an internal adversary.

LoRaWAN messages may be cryptographically protected on the application layer, but a network operator can harvest metadata, read unencrypted headers, send control packets, or update routing tables to attack certain groups of people.

The NS can estimate the location of an ED if it knows the location of the gateway that forwards a message. Despite LoRa's large cell size, location estimation can be narrowed down if the server sees duplicate messages from other nearby gateways. The ability of volunteers to extend a LoRa network increases this adversary's location-harvesting power at no cost to itself.

As mentioned in Section 3.2.2, network operators may have access to secrets that may allow them to break confidentiality and integrity protections for individual EDs.

3.2.7 Practical LoRa

LoRaWAN, as officially specified, is severely flawed from a privacy-preserving perspective. Its centralized architecture requires placing a large amount of trust in infrastructure operators. The LoRa physical layer is prone to eavesdropping and jamming. Worst of all, there is no way for an ED to appear anonymous

on a LoRaWAN network — interacting with the network as intended by the protocol always leaks some identifier that can be linked to a particular device, allowing ED locations to be tracked.

The EDs that we envision in our contact tracing system are cheap physical tokens without a rich user interface. Unlike in smartphone-based contact tracing systems, a user may have no way to declare a positive COVID-19 diagnosis through their device, especially if they do not own another internet-connected device which could pair with the token. In such cases, how are positive diagnoses uploaded to the health authority? One way is for the data to be uploaded at the testing centre. If data is encrypted with the health authority's public key before leaving the device, the testing centre learns no more information than if they were also a passive listener. This method sounds reasonable for rapid antigen tests, but it is risky to ask people to return to the testing centre after a positive diagnosis from a slower test, such as PCR.

In decentralized contact tracing, EDs also need to periodically download identifiers from positively-diagnosed individuals so that exposure notifications can be determined. Unfortunately, LoRaWAN lacks a true broadcast capability where a message can be transmitted to all devices. LoRaWAN does support multicasts, but EDs should periodically announce themselves to the nearest gateway so that the NS can maintain routing tables and send multicast messages only to necessary gateways. This method leaks users' locations at a constant interval. Workarounds might be possible, for example installing immobile "dummy" EDs in a grid to force the NS to multicast everywhere, but these methods need to be verified empirically. Further, to receive multicasts, EDs must operate in a special receiving mode (i.e. LoRaWAN Class B mode), which consumes more power and requires all devices to have approximately synchronized clocks.

It is possible to deviate from the established LoRaWAN protocol and retain only the LoRa physical layer, replacing the link and network layers with a custom protocol designed to preserve privacy. Such an approach would be incompatible with existing LoRaWAN networks and would require deploying custom gateways to provide coverage. Existing LoRaWAN gateways may be augmented with such functionality in software. Volunteer gateway operators could help scale the system and potentially receive an incentive from local governments for doing so. Though there is a start-up cost involved, the benefits to privacy may be worth it, especially in high-risk, high-density areas like city centres.

4 Conclusion

It has been known since the beginning of the COVID-19 pandemic that digital contact tracing systems cannot provide perfect privacy [24]. Many weaknesses were identified when the first systems were introduced, and more have been discovered and characterized since then.

Existing digital contact tracing systems work by exchanging ephemeral IDs over a short-range medium such as Bluetooth or UWB. Ensuring the privacy of these systems on the software layer has proven challenging enough, but wireless

media also demonstrate flaws on the physical layer that can be exploited to identify and track users.

As a wide-area networking protocol, LoRaWAN displays concerning flaws that render it unsuitable as a secure and privacy-preserving wireless medium. Lack of anonymous communication with the network allows adversaries to link transmissions and track users' locations. Poorly-implemented cryptographic primitives can be exploited to break confidentiality and integrity. Moreover, LoRaWAN depends on critical infrastructure being operated by a single, central authority. These flaws may be somewhat mitigated by building a new protocol atop the LoRa physical layer, but concerns about physical-layer tracking methods such as radio fingerprinting remain.

While it is possible for local adversaries to invade the privacy of contact tracing system users in a small neighbourhood, a greater concern is large, wealthy actors such as governments or companies, which can deploy wide-reaching networks of eavesdropping devices and collect data on many users at once. Arguably, we are presented with a similar problem in developed societies through tracking of smartphones, even before the pandemic. However, low-income settings are decidedly more low-tech, and it remains to be decided whether introducing a digital contact tracing system poses a proportionally greater privacy risk to the intended audience, especially when technical restrictions such as the choice of wide-area wireless network are at odds with maximizing privacy.

References

[1] Emekcan Aras et al. "Exploring the Security Vulnerabilities of LoRa". In: *3rd IEEE International Conference on Cybernetics, CYBCONF 2017, Exeter, United Kingdom, June 21-23, 2017*. IEEE, 2017, pp. 1–6. DOI: 10.1109/CYBConf.2017.7985777. URL: https://doi.org/10.1109/CYBConf.2017.7985777.

[2] Emekcan Aras et al. "Selective Jamming of LoRaWAN using Commodity Hardware". In: *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Melbourne, Australia, November 7-10, 2017*. Ed. by Tao Gu, Ramamohanarao Kotagiri, and Huai Liu. ACM, 2017, pp. 363–372. DOI: 10.1145/3144457.3144478. URL: https://doi.org/10.1145/3144457.3144478.

[3] Jason Bay et al. "BlueTrace: A privacy-preserving protocol for community-driven contact tracing across borders". In: (Apr. 2020). URL: https://bluetrace.io/static/bluetrace_whitepaper-938063656596c104632def383eb33b3c.pdf.

[4] Ismail Butun, Nuno Pereira, and Mikael Gidlund. "Analysis of LoRaWAN v1.1 security: research paper". In: *Proceedings of the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects, SmartObjects@Mobihoc 2018, Los*

Angeles, CA, USA, June 25, 2018. Ed. by Pietro Manzoni et al. ACM, 2018, 5:1–5:6. DOI: 10.1145/3213299.3213304. URL: https://doi.org/10.1145/3213299.3213304.

[5] Claude Castelluccia et al. "DESIRE: A Third Way for a European Exposure Notification System Leveraging the best of centralized and decentralized systems". In: *CoRR* abs/2008.01621 (2020). arXiv: 2008.01621. URL: https://arxiv.org/abs/2008.01621.

[6] *Ultra-Wide Band dongles for contact tracing*. Workshop on Digital Contact Tracing in Low and Medium Income Countries. Zürich, Oct. 26, 2021.

[7] Paul-Olivier Dehaye and Joel Reardon. "Proximity Tracing in an Ecosystem of Surveillance Capitalism". In: *WPES'20: Proceedings of the 19th Workshop on Privacy in the Electronic Society, Virtual Event, USA, November 9, 2020*. Ed. by Jay Ligatti et al. ACM, 2020, pp. 191–203. DOI: 10.1145/3411497.3420219. URL: https://doi.org/10.1145/3411497.3420219.

[8] Paul-Olivier Dehaye and Joel Reardon. *SwissCovid: a critical analysis of risk assessment by Swiss authorities*. 2020. arXiv: 2006.10719 [cs.CR].

[9] Rosario Gennaro, Adam Krellenstein, and James Krellenstein. "Exposure Notification System May Allow for Large-Scale Voter Suppression". In: Real World Crypto 2021. Jan. 2021. URL: https://www.krellenstein.com/adam/get/exposure-notification.2020-09-02.pdf.

[10] Ross Gilson and Michael Grudsky. *LoRaWAN Capacity Trial in Dense Urban Environments*. Tech. rep. URL: https://info.semtech.com/hubfs/machineQ_LoRaWAN_Capacity_Trial-2.pdf (visited on 11/20/2021).

[11] H. Givehchian et al. "Evaluating Physical-Layer BLE Location Tracking Attacks on Mobile Devices". In: *2022 IEEE Symposium on Security and Privacy (SP) (SP)*. Los Alamitos, CA, USA: IEEE Computer Society, May 2022, pp. 507–521. DOI: 10.1109/SP46214.2022.00030. URL: https://doi.ieeecomputersociety.org/10.1109/SP46214.2022.00030.

[12] Frank Philipp Hessel. "LoRaWAN Security Analysis: An Experimental Evaluation of Attacks". MA thesis. Technical University of Darmstadt, Aug. 2019. URL: https://tuprints.ulb.tu-darmstadt.de/17550/1/LoRaWAN-Security-Analysis.pdf.

[13] Patrick Leu et al. "I Send, Therefore I Leak: Information Leakage in Low-Power Wide Area Networks". In: *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks, WiSec 2018, Stockholm, Sweden, June 18-20, 2018*. Ed. by Panos Papadimitratos, Kevin R. B. Butler, and Christina Pöpper. ACM,

2018, pp. 23–33. DOI: 10.1145/3212480.3212508. URL: https://doi.org/10.1145/3212480.3212508.

[14] *LoRaWAN 1.1 Specification*. Tech. rep. Jan. 2015. URL: https://lora-alliance.org/resource_hub/lorawan-specification-v1-1/.

[15] Norbert Ludant et al. "Linking Bluetooth LE & Classic and Implications for Privacy-Preserving Bluetooth-Based Protocols". In: *42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021*. IEEE, 2021, pp. 1318–1331. DOI: 10.1109/SP40001.2021.00102. URL: https://doi.org/10.1109/SP40001.2021.00102.

[16] Wouter Lueks et al. "CrowdNotifier: Decentralized Privacy-Preserving Presence Tracing". In: *Proc. Priv. Enhancing Technol.* 2021.4 (2021), pp. 350–368. DOI: 10.2478/popets-2021-0074. URL: https://doi.org/10.2478/popets-2021-0074.

[17] Ivan Martinez, Philippe Tanguy, and Fabienne Nouvel. "On the performance evaluation of LoRaWAN under Jamming". In: *12th IFIP Wireless and Mobile Networking Conference, WMNC 2019, Paris, France, September 11-13, 2019*. Ed. by Mérouane Debbah et al. IEEE, 2019, pp. 141–145. DOI: 10.23919/WMNC.2019.8881830. URL: https://doi.org/10.23919/WMNC.2019.8881830.

[18] *Privacy-preserving contact tracing - Apple and Google*. URL: https://covid19.apple.com/contacttracing.

[19] The DP-3T Project. "Privacy and Security Risk Evaluation of Digital Proximity Tracing Systems". In: (Apr. 2020). URL: https://github.com/DP-3T/documents/blob/08e9c145dabfe26907afd66e0973aceb4e4b44f7/Security%5C%20analysis/Privacy%5C%20and%5C%20Security%5C%20Attacks%5C%20on%5C%20Digital%5C%20Proximity%5C%20Tracing%5C%20Systems.pdf.

[20] Pieter Robyns et al. "Physical-layer fingerprinting of LoRa devices using supervised and zero-shot learning". In: *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks, WiSec 2017, Boston, MA, USA, July 18-20, 2017*. Ed. by Guevara Noubir, Mauro Conti, and Sneha Kumar Kasera. ACM, 2017, pp. 58–63. DOI: 10.1145/3098243.3098267. URL: https://doi.org/10.1145/3098243.3098267.

[21] N. Sornin et al. *LoRaWAN Specification*. Tech. rep. Jan. 2015. URL: https://resources.lora-alliance.org/technical-specifications/lorawan-specification-v1-0.

[22] Joachim Tapparel et al. “An Open-Source LoRa Physical Layer Prototype on GNU Radio”. In: *21st IEEE International Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2020, Atlanta, GA, USA, May 26-29, 2020*. IEEE, 2020, pp. 1–5. DOI: 10.1109/SPAWC48557.2020.9154273. URL: https://doi.org/10.1109/SPAWC48557.2020.9154273.

[23] DP-3T Team. *Best Practices Operational Security for Proximity Tracing*. Tech. rep. June 2020. URL: https://github.com/DP-3T/documents/blob/08e9c145dabfe26907afd66e0973aceb4e4b44f7/DP3T%5C%20-%5C%20Best%5C%20Practices%5C%20for%5C%20Operation%5C%20Security%5C%20in%5C%20Proximity%5C%20Tracing.pdf.

[24] Carmela Troncoso et al. “Decentralized Privacy-Preserving Proximity Tracing”. In: *CoRR* abs/2005.12273 (2020). arXiv: 2005.12273. URL: https://arxiv.org/abs/2005.12273.

[25] Serge Vaudenay. “Analysis of DP3T”. In: *IACR Cryptol. ePrint Arch.* (2020), p. 399. URL: https://eprint.iacr.org/2020/399.

[26] Serge Vaudenay and Martin Vuagnoux. *Analysis of SwissCovid*. Tech. rep. 2020. URL: https://lasec.epfl.ch/people/vaudenay/swisscovid/swisscovid-ana.pdf.

[27] Xueying Yang et al. “Security Vulnerabilities in LoRaWAN”. In: *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation, IoTDI 2018, Orlando, FL, USA, April 17-20, 2018*. IEEE Computer Society, 2018, pp. 129–140. DOI: 10.1109/IoTDI.2018.00022. URL: https://doi.org/10.1109/IoTDI.2018.00022.