ConTra Corona: Contact Tracing against the Coronavirus by Bridging the Centralized–Decentralized Divide for Stronger Privacy

Short Summary

Wasilij Beskorovajnov[1], Felix Dörre[2], Gunnar Hartung[2], Alexander Koch[2], Jörn Müller-Quade[2], and Thorsten Strufe[2]

[1] FZI Research Center for Information Technology
lastname@fzi.de
[2] Karlsruhe Institute of Technology
firstname.lastname@kit.edu

This is a summary, sketching the main points of our proposal, please refer to https://eprint.iacr.org/2020/505 [BDH+20] for a full description.

1 A Hybrid Approach

Digital contact tracing using smartphone apps based on Bluetooth Low Energy (BLE) may help to keep the SARS-CoV-2 pandemic in control. While the public debate and early proposals focus mostly on "centralized" vs. "decentralized" solutions[3] for digital contact tracing, our approach can be classified as hybrid, combining the advantages of both, centralized and decentralized solutions. While our protocol makes use of several (central) servers, these servers cannot learn the identities of infected users or persons who have been in contact with infected users. At the same time, users receiving a notification about an encounter with an infected person cannot deduce the infected person's identity.

Our key idea is to introduce a distinction between short-lived *public identifiers* (which are broadcast locally via BLE) and long-term *secret/warning identifiers* (used for publishing warnings). In our protocol, one of the central servers maintains a lookup table, mapping each short-term identifier to its respective long-term warning identifier. This server is oblivious of the mapping, though, since the long-term identifiers are encrypted with a public key of another server. Importantly, and in contrast to the centralized solutions, the contact history stays local to the phone. Additionally, we introduce a server architecture which features a strict separation of different tasks. Overall, our protocol combines the privacy advantages of both worlds, i.e. *protection against a centralized actor*, as well as *protection of infected users' identities*.

[3] Note that the terminology of this debate is slightly misleading, as schemes utilizing private set intersection (PSI) can be realized with the help of a central server architecture. Following the arguments of this debate such schemes would be flagged as insecure, although they may provide a high level of security and privacy.

2 Key Security Features

Separation of Duties. Our protocol makes use of a strict server separation concept to mitigate the threat to users' privacy posed by data collection on centralized servers. In our protocol, the medical professional uploads signed and encrypted public identifiers (without learning them) to a dedicated *matching server*. This server then does a lookup of the respective secret identity (registered via a *submission server*), which are then published (after decryption and deduplication) by a *warning server*. This separation does not lead to a significant increase of computation on the side of the smartphone.

Note that these servers have to be operated by independent, distinct organisations.[4] Thus, in order to compromise the server architecture to the full extent, multiple institutions would have to cooperate maliciously.

Protection of Infected Users' Identities. When a user A has met an infected user B and receives a warning, A only learns the *day* on which the encounter took place. In contrast, for the DP3T approach [TPH+20], a user might be able to determine the time of encounter much more precisely, e.g. with a precision down to several minutes (depending on the epoch length). This might enable A to deduce B's identity, and presents a serious risk to B's privacy. In order to prevent a user from registering multiple accounts and/or using a distinct warning identifier per encounter, we employ a certain anonymous e-token dispenser for rate limiting of submission server uploads [CHK+06].

Hiding Multiple Exposures. If users were to receive distinct warnings for every encounter with an infected user, they would be able to infer the identity of the infected user by counting the warnings and comparing them to how often they met with specific individuals. Our protocol makes use of a deduplication of warnings by the central servers, further reducing the risk to infected users' privacy.

Proving a Warning. Our protocol enables warned users to securely prove the fact that they have been warned, e.g. towards medical professionals who administer tests for COVID-19. Without this feature, anybody who is curious about their status but not at risk could get tested, e.g., by showing a screenshot of a warning from someone else's smartphone – which would be unacceptable in times of restricted testing resources.

Anonymous Communication. Our protocol uses TOR to prevent linking values received by the servers to IP addresses of their senders.

Proximity Discovery Robustness. Our protocol leverages secret sharing to ensure that the contact discovery process takes the contact duration into account.

[4] For the case of Germany such institutions may be the Robert Koch Institute (RKI), the Federal Ministry of Health (BMG) or the Data Privacy Agencies of the federal states.

3 ConTra Corona: Protocol Overview

Any data communicated throughout the protocol does not contain identifying information. However, to protect against a linking of this information to e.g. the sender's IP address, we assume all communication to servers to proceed via TOR (or a proxy). See Figure 1 for an overview of our protocol.

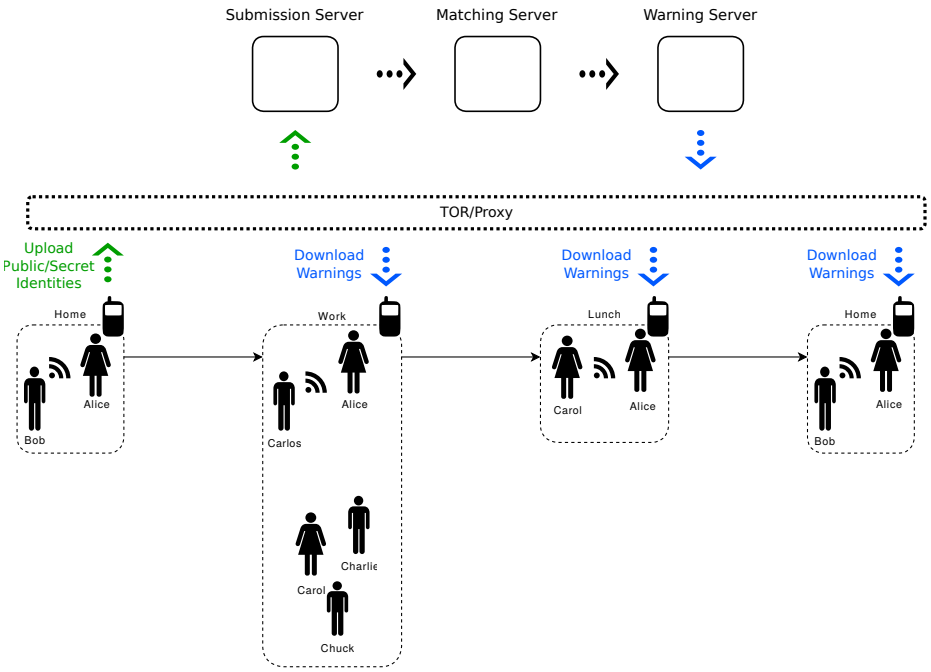

Fig. 1: Overview of the application's infrastructure. The figure depicts different possible scenarios during the day: In the morning, Alice uploads her daily public/secret identities (Section 3.2) to the submission server, and periodically checks for warnings (Section 3.5) published by the warning server. Throughout the day, while she is in proximity to Bob, Carlos and Carol, the application automatically exchanges public identifiers with their phones (Section 3.3).

3.1 Setup Phase

Registration. Initially, the user is required to register – which is to ensure that each person has only one instance of the smartphone app (per device/phone contract). This registration can be performed by proving the possession of a valid phone number via an SMS challenge. After a successful registration the application receives an anonymous e-token dispenser, which can be used to authenticate an upload of a warning identifier later on.

Furthermore, the user enters some of his personal information, e.g. her full name, to her app instance. This information is not used in the registration process and will only be relevant when the user would like to prove that she has received a warning. This information is cryptographically *hidden* within the warning identifiers (generated in a later step).

3.2 Upload of Daily Lookup Table Information

The application generates ephemeral cryptographic data, which will be valid during the day and freshly generated by the beginning of the next day: 1) a warning identifier, as well as 2) a set of public and secret identities. The secret identities are different encryptions of the warning identifier, which can be decrypted by a warning server, and the public identities are hashes of the secret identities. The public identities are broadcasted by the application via BLE, while the secret identities remain known solely to the application and parts of the server pipeline.

The application generates an e-token for the current day from its token dispenser and uses it to upload the set of public and secret identities to the submission server, which gathers a global upload batch of multiple applications, shuffles the entries and forwards the complete batch to the matching server.

3.3 Proximity Discovery

The application continuously broadcasts secret shares of the public identities. The broadcasts of individual public identities are overlapping such that there are enough shares of a public identity in any potential contact interval. Once enough shares of other users' public identities have been received, the application reconstructs and records the respective public identity.

3.4 Issuing a Warning After a Positive Test

The doctor has an authorized and authenticated web interface for the matching server. After determining the time period the participant has been infectious the doctor uses the web interface for generating a TAN. Finally, the participant is either given a print with the TAN and the time period on it or has the TAN and time period communicated over the phone. After providing this information to the application, the recorded public identities from this time period are uploaded to the matching server together with the TAN. The matching server verifies the TAN and performs a lookup for matched public identities and forwards the list of according secret identities to the warning server. The warning server can now decrypt the secret identities and removes duplicates from the resulting list of warning identifiers. The list is published for everyone to see.

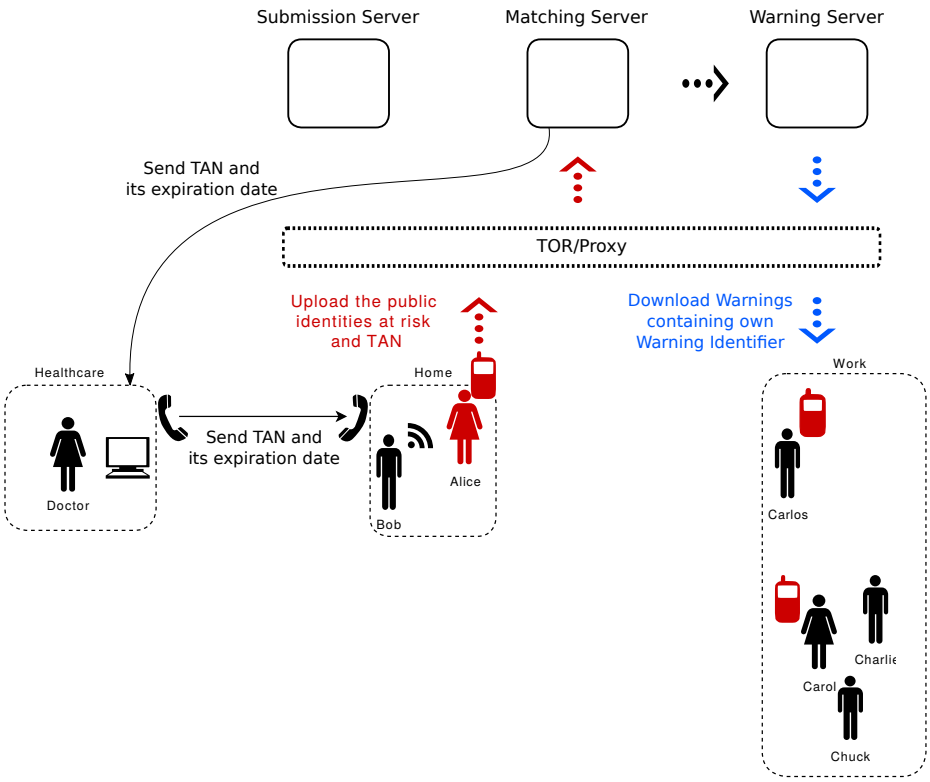

Fig. 2: Information flow upon issuing a warning. When the doctor is informed about a positive test, she obtains a new TAN from the matching server, to communicate it to COVID-19-positive Alice. Then, using this TAN, Alice uploads all public identities she observed during her (approximation of the) infectious period (Section 3.4). The application regularly downloads the list of warnings from the warning server, and checks it for warning identities of the owning user. In the case of Carlos and Carol, this check will turn out to be positive (Section 3.5).

3.5 Exposure Status Request

Warning Status. The application regularly downloads the list warning identities from the warning server and checks whether any of its own warning identifiers are contained in the list. In this case, the user is informed about having had an encounter with an infected person.

Proving a Warning. The user presents her warning identity and her name to the medical professional and conducts a zero-knowledge proof showing the warning identity is a valid commitment to her name.The medical professional verifies the proof and the user's name, and checks the warning identity has been published by the warning server. If all of these conditions are met, the medical professional accepts the proof.

References

[BDH+20] W. Beskorovajnov, F. Dörre, G. Hartung, A. Koch, J. Müller-Quade, and T. Strufe. *ConTra Corona: Contact Tracing against the Coronavirus by Bridging the Centralized–Decentralized Divide for Stronger Privacy*. Apr. 29, 2020. Cryptology ePrint Archive, Report 2020/505.

[CHK+06] J. Camenisch, S. Hohenberger, M. Kohlweiss, A. Lysyanskaya, and M. Meyerovich. "How to win the clonewars: efficient periodic n-times anonymous authentication". In: *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006*. Ed. by A. Juels, R. N. Wright, and S. D. C. di Vimercati. ACM, 2006, pp. 201–210. DOI: 10.1145/1180405.1180431.

[TPH+20] C. Troncoso, M. Payer, J.-P. Hubaux, M. Salathé, J. Larus, E. Bugnion, W. Lueks, T. Stadler, A. Pyrgelis, D. Antonioli, L. Barman, S. Chatel, K. Paterson, S. Capkun, D. Basin, J. Beutel, D. Jackson, B. Preneel, N. Smart, D. Singelee, A. Abidin, S. Gürses, M. Veale, C. Cremers, R. Binns, and C. Cattuto. *Decentralized Privacy-Preserving Proximity Tracing*. Apr. 12, 2020. URL: https://github.com/DP-3T/documents/raw/master/DP3T%20White%20Paper.pdf.