DESIRE: A Practical Assessment

The DP3T Consortium

Version 1.0
13 May 2020

Several proximity tracing apps have been proposed to aid in combating the COVID-19 epidemic. Solutions based on location data such as GPS are known to present major privacy risks; they also tend to fail in indoor environments. Bluetooth-based solutions are currently the only viable option that can meet legitimate expectations for privacy and effectiveness. These solutions install a secret key in an app, generate ephemeral broadcast tokens from this key and broadcast these tokens via Bluetooth. Other phones with the app receive these tokens and store them (together with time information and received signal strength). If a person tests positive, their key or tokens are used to identify users at risk, that is, users who have been sufficiently close to the person for a sufficient amount of time.

One set of proposals (e.g., TraceTogether, PNTK, and ROBERT) takes a centralized approach in generating keys and making risk decisions: this requires uploading all received tokens to a central database. In a decentralized solution such as DP3T, the key is generated in the app and the received tokens never leave the phone. Infected users upload their keys to a central server; all phones download these keys and make the risk decision locally on the phone. There are privacy concerns with respect to central server solutions: the contact graph of all users is known to the server, hence a breach of the server results in a massive loss of highly sensitive data; moreover, the collection of central data increases the risks of function creep and abuse of the information. A decentralized approach avoids function creep, at the cost of a modest increase of vulnerability to local attacks.

The DESIRE proposal for proximity tracing is positioned by its designers as an optimal tradeoff between centralized and decentralized contact tracing. By combining the received beacons in a cryptographic way (a so-called Diffie-Hellman exchange) into a contract trace (called a PET token), security against some local attacks is improved. Decisions on infection risks of users are still made in a centralized manner, but the server cannot link individual PET tokens of infected users. The server knows the sets of PET tokens received by non-infected users but only linked to a pseudonym. The DESIRE proposal intends to hide the social graph.

In this note, we explain why we believe that DESIRE is a very interesting *academic* proposal, but not a practical solution that reconciles the decentralized and centralized approach. As a general comment, the DESIRE document does not propose concrete parameters for frequency of exchanges, communication, or storage that would allow a full feasibility analysis.

Radio interface

A first concern is that the DESIRE proposal uses a Diffie-Hellman exchange to combine two tokens into one PET token. For cryptographic reasons, this increases the length of each broadcast token from 128 to 256 bits. This seems like a minor technical detail, but transmitting 256 bits rather than 128 bits over Bluetooth Low Energy (BLE) in a reliable manner is substantially more difficult. A 128-bit message fits into a single passive advertisement packet, the receiver simply needs to listen. Longer messages require active advertisements in which the sender advertises a message, the receiver requests it, and the sender then transmits the actual message.

This exchange requires at least three different packets be reliably sent and received, which increases the load on the battery. In order to ensure that the packets are exchanged during a contact, both phones need to be scanning for packets for the duration of the contact. Currently, iOS guarantees that a phone's BLE radio wakes up only for 2 seconds every 5 minutes (see Apple/Google Exposure Notification BLE). Furthermore, due to the implementation of BLE radios, sending different packets instead of repeatedly broadcasting the same packet increases the power consumption as the application CPU must be woken up.

Moreover, the probability of correct reception of a single beacon is 30%–70% (depending on the distance and signal strength). This means that 2 packets are received correctly with a probability of 9–49% (for 3 packets this probability is 2.7–34%). Lower probabilities require a larger number of attempts and thus a higher battery drain. These percentages are observed in lab conditions, with two devices exchanging beacons and with no other contact-tracing devices beaconing in the vicinity. This raises questions about the feasibility of the DESIRE protocol in settings with more than a few people (devices) — note that the DESIRE document indicates that it is intended for use in "crowded places" where "manual contact tracing does not work".

The implementation of a reliable DH exchange for proximity tracing will require significantly higher BLE radio usage and phone energy consumption than the current Google/Apple Exposure Notification API. It is not clear if OS manufacturers and users would be willing to accept this. Users are sensitive to battery drain and OS manufacturers have other functions running on BLE beaconing. Users may disable this functionality, or their phones may run out of battery power, reducing app uptake.

Scalable and trusted anonymous channel

The solution hides the social graph at the server, but there are many open questions about the feasibility, efficacy and practicality of the approaches required for this solution. In DESIRE, the server only sees encounters of pseudonymous participants and partial graphs. This property is achieved by uploading the PET tokens of infected users over an anonymous

communication channel. The documentation proposes three ways to implement such a channel:

> **Mixnets:** mixnets were proposed by Chaum in the 1980s to hide the identity of the sender of a message. In spite of extensive research, there are ***no sufficiently large-scale deployed mixnets***. The largest network for low latency traffic is Tor: it consists of about 6000 nodes and a few million users. It is unclear how to engineer and deploy a new mixnet that will scale to tens of millions of users. Open questions include: how many nodes are needed, who will run these nodes, and what is needed in terms of processing power, bandwidth and latency? Moreover, it is unclear how to set the parameters of such a network in order to achieve desired latency and anonymity properties.
>
> **Proxies:** this corresponds to a mixnet with a single node that is fully trusted to perform the mixing. This places a very high level of trust in this single node, which brings us very close to a centralized solution in terms of trust. The DESIRE documentation suggests that the network address translation system (NAT) of the mobile network could be sufficient. However, this does not take into account IPv6, which does not have a NAT, and the fact that a NAT does not hide timing information. Moreover, this seems to place a very high level of trust in mobile operators.
>
> **Trusted hardware**: the centralized server can perform the mixing in a trusted hardware module. Replying on trusted hardware raises many issues. How are software and hardware implemented and verified? How can one be sure that trusted hardware is not replaced by other hardware during deployment? Is this trusted hardware secure against side channel attacks, which are common in the literature but normally outside the threat model? Can the software and hardware vendor be trusted to not collude with government-level adversaries? Can the required performance be achieved with the existing trusted hardware? If one uses trusted hardware for privacy protection, would it not be more appropriate to run the more efficient centralized protocol fully in trusted hardware?

In all three approaches, PET tokens need to be mixed to ensure that PET tokens from individual users cannot be linked. If the mixing is imperfect or if the server performing the mixing is compromised, information starts to leak about the (partial) pseudonymous graph. It is unclear how much information leakage is acceptable, in particular since one can expect that a powerful attacker may be able to interfere with the anonymous communication channel and may also have additional side information on a subset of infected users and contacts. It seems also that upload of dummy PET tokens will be needed to protect individual users. Again, it is unclear how many are sufficient and what the performance impact is of these dummy uploads. Finally, high levels of anonymity typically require a high latency. However, one of the benefits of proximity tracing with an app is that the tracing is performed

faster. It is unclear whether the high latency requirements of the anonymous channel would not interfere with the main goal of that app, that is, faster detection of users at risk.

Overall, an anonymous communication channel for proximity tracing is a complex, large-scale system for which it is difficult to specify requirements to match the expected level of privacy under realistic attack models. Moreover, ***these systems are not currently available***. Even if we had specific requirements, it would be very challenging to build such a system either due to the high level of trust in a single server or because of the complex tradeoffs between reliability, anonymity, bandwidth, and latency for a large-scale solution.

Which privacy properties are achieved

Independent of the properties of the anonymous channel discussed in the previous point, the DESIRE proposal leaks information to the central server about the number of contacts of users at risk. While this seems a minor issue, it may lead to users being identified as violating social distancing rules. The server only knows a pseudonym, but these users are expected to contact the health infrastructure so the anonymity set may be rather small. Moreover, violations of social distancing rules may also lead to deanonymization requests on the trusted intermediary such as the proxy or the mixnet.

It is unclear which other information the DESIRE proposal leaks about users if additional side information is available or if very few users are infected. This seems very hard to quantify and requires thorough analysis.

Interoperability

It is very difficult to see how the DESIRE proposal could be made interoperable with existing centralized or decentralized proposals. By changing the amount of information sent over Bluetooth, it seems unlikely that a DESIRE app could interoperate with a DP3T or ROBERT app. Moreover, the information stored in the central server is also very different from other protocols and it is not clear that it could be sent to other decentralized or even centralized solutions. In particular for DP3T, any interoperability abandons all privacy benefits derived from local decisions, which has been critical to public legitimacy in many countries. Finally, the choice of a centralized decision mechanism implies that the Google/Apple API for exposure notification cannot be used, which closes the door for a realistic and interoperable implementation around the world.

Encryption of information on the central server

The DESIRE protocol adds additional protection of the data stored on the central server. The data of non-infected users (including their PET tokens) is encrypted under a secret key stored on the phones of the users. This key is sent to the user and subsequently deleted on the

server. For each exposure test, the key is sent back to the server for temporary usage. This is a neat trick to reduce the attack surface.

However, it is unclear how effective this trick would be in a realistic setting. It relies on trust in the server (it would be very easy for the server to retain these keys surreptitiously and difficult to spot such an attack in an audit). Moreover, one has to implement the server in such a way that this key is not captured accidentally in a normal backup process. For example, if the server is running in a virtual machine, the whole machine could be swapped out, resulting in the encryption keys ending up on secondary storage. Finally, a sophisticated attacker who can steal data on the server can install malware that over time exfiltrates both data and the keys from the server.

Unclear benefits of centralized decisions

While there are clear risks involved in centralized decisions, the DESIRE proposal presents several arguments for the benefits of centralized decisions that are not fully convincing. A first argument is flexibility: centralized decision processes are easier to update. In a decentralized protocol, smart phones download on a regular basis (a few times per day) the keys of infected users. It is very easy to augment these downloads with updated (authenticated) parameters for the decision algorithm. Note that the DESIRE proposal also plans to include a method to update on the phone a similar parameter, the thresholds to store an encounter.

A second argument is the ability to *measure* and *control* the number of at-risk users. In a decentralized system, users notified of risk will contact health authorities, resulting in a similar monitoring feature. Moreover, even if a small fraction of the users collaborate by voluntarily participating in epidemiological research, this research data will also be able to estimate the number of at-risk users.

One can wonder whether it is necessary to control the number of at-risk users informed of their risk. If this number is growing faster than expected, one should probably take other measures than informing only a subset of the at risk users (and leaving the rest to expose other people). But if such a control would be needed, decentralized systems can also support it. Phones could report their risk scores to the server, which can then decide on the appropriate threshold levels. If some prioritization is required, one can also introduce several thresholds in the decentralized solution and give different advice depending on the risk score. Again, these thresholds could be updated on a regular basis if that would be needed.

Authorization scheme

The DESIRE proposal contains an authorization scheme to limit the number of app registrations from a single phone. It is unclear how effective the proposed mechanism would be for rooted phones.

Overall conclusion

The DESIRE proposal contains some interesting ideas that open new directions in research on proximity tracking solutions. However, some of the design assumptions are not realistic for current mobile phone Bluetooth radios and some essential building blocks such as anonymous communication networks have not achieved sufficient maturity for large-scale deployment in a timely manner. This leads to serious reservations on the scalability and the feasibility of the overall proposal. More research is needed to specify the full requirements on the building blocks and to model their interactions in order to fully understand the privacy and performance tradeoffs. Our experience with complex protocols such as Tor have taught us that it can take many years before we understand these tradeoffs in sufficient detail.