Proximity Tracing Approaches - Comparative Impact Analysis

Antoine Boutet, Nataliia Bielova, Claude Castelluccia, Mathieu Cunche, Cédric Lauradoux, Daniel Le Métayer, Vincent Roca

Proximity Tracing Approaches
Comparative Impact Analysis

PRIVATICS team, Inria, France[1]

version 1.0, April 30, 2020

Summary

Although they address the same problem, the so-called “centralized” and “decentralized” approaches to COVID-19 proximity tracing rely on different threat model assumptions. The goal of this document is to analyze the impact of these two options in terms of privacy, security and reliability.

The main objective of the decentralized approach is to protect users against a malicious server or a state-level adversary and to prevent the leak of sensitive data due to attacks or negligence at the server side. Therefore, the role of the server is reduced as much as possible, and the exposure verification is performed on the user device. In contrast, the centralized approach puts more emphasis on the protection of users against other malicious users trying to infer who is infected. Hence, the role of the server in the centralized approach is more important, including the verification of exposure.

This design choice involves different privacy risks:

- The decentralized approach provides many opportunities to malicious or curious users (through wide scale and undetectable attacks or during normal usage) to infer the identity of infected users or to monitor specific areas. These privacy risks coming from users (e.g., neighbors) can easily lead to abuses as well as stigmatization and harassment of diagnosed users. On the positive side, the server learns little information about users.
- In the centralized approach, in contrast, the capability of users to learn who is infected is drastically limited. This better protection however comes at the cost of relying on a server which is able to learn some information about users.

[1] The authors are grateful to Nicolas Anciaux and Benjamin Nguyen for their comments on an earlier version of this document.

Law enforcement agencies and third parties colluding with the server are sources of risk in both approaches but they do not concern the same population. Only infected users who consent to declare themselves are concerned in the decentralized approach, while these risks concern all users (infected or not) in the centralized approach. However, the likelihood of these risks needs to be assessed and balanced, as all other risks, with the potential benefits of these applications in the fight against COVID-19. To this respect, the centralized approach can bring added value because the health authority is aware of the number of exposed people and can use it both for statistical purposes and to easily adjust the risk calculation algorithm (to decide if a user should be classified as “at risk”).

Figure 1 provides an overview of the risks discussed in this document for the two approaches, with respect to curious or malicious users (left), and a malicious server, a State-level adversary and a server colluding with a third party (right). These risks are classified according to their severity and likelihood. Each of these risks is presented and analyzed in the core of the document.

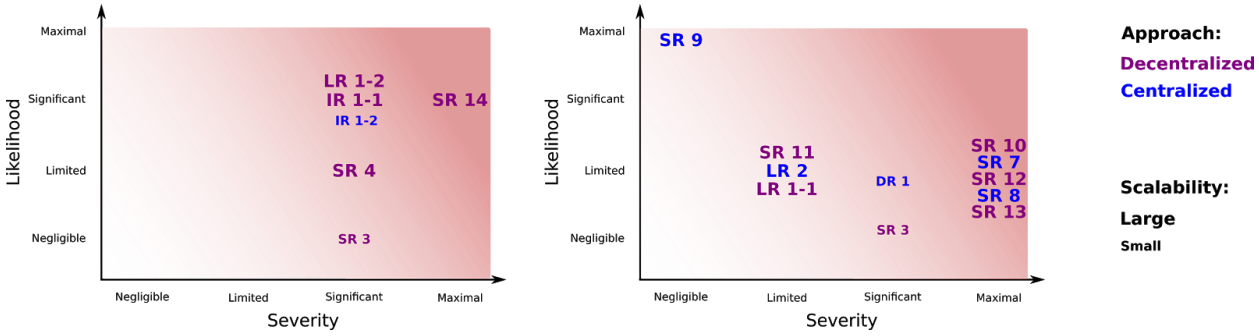

Figure 1: Severity and likelihood of feared events with respect to curious and malicious users (left) and a curious server, a State-level adversary, and a server colluding with a third party (right) for centralised and decentralized approaches.

1. Introduction

Covid-19 proximity tracing applications should provide the most useful functionalities to limit the spread of the virus in a secure, and reliable way. Another key factor influencing the adoption of these applications is to ensure the protection of the privacy of its users. These requirements are essential to justify the legitimacy of their deployment and their acceptability by citizens who should be free to install them as well as to uninstall them at any time. We do not discuss the effectiveness of these proximity tracing applications here, as they have to be parameterized and assessed in collaboration with epidemiologists. We rather focus on privacy, security and reliability issues.

2. Methodology

Many different variants of proximity tracing applications have been proposed in different countries during the last months. In Europe, strong emphasis is put on GDPR compliance and the application of the "privacy by design" approach has led to the emergence of two main approaches [1,2], sometimes called "centralized'' and "decentralized''. These terms essentially refer to the component where the risk assessment is performed (i.e., on a central server or decentralized on the devices) and are not, strictly speaking, related to the architecture since both options involve a central server. These terms are defined more precisely below. We also stress the fact that we do not discuss specific features of existing proposals (which may still evolve) but focus only on the fundamental impacts of the "decentralized" versus "centralized" approach. The goal of this document is to identify the benefits and limitations of the two approaches following a scientific approach in order to provide an objective account of the situation. As discussed in conclusion, considering that each option has its pros and cons, the choice between the two is a matter of setting priorities between the different impacts, which is a political rather than technical decision.

3. Assumptions

We assume that the single purpose of the application is to allow users to know if they have been in close contact with other users that have been diagnosed positive. Some of these applications could be integrated into wider systems providing further functionalities such as feeding anonymous databases for epidemiological research. We do not consider these extra functionalities here for a better separation of concerns.

The expected usage of the application is to have it running permanently (at least in all situations where the user may be in the vicinity of other people). In addition, when a user receives an "exposed" (or "at risk") status, he is supposed to follow the official recommendations (e.g., to be tested or isolated). When the user is diagnosed positive, the application should be able to provide the necessary information to the server to ensure that users who have been in contact with him are informed[2]. However, users do not have any obligation to behave as expected. For example, they can switch off their Bluetooth interface or the application at any time or decide not to declare their status after being diagnosed positive. The impact of such behaviour will essentially be on the effectiveness of the application which falls outside the scope of this document.

[2] The discussion on whether the user is required to share some information or whether it's a user's voluntary choice is open to legal requirements (for example, it might be possible only upon user's consent). We don't discuss such requirements here and leave this for future legal analysis.

We use the following terminology on our assessment:

- A user is *infected* if she is a carrier of the COVID-19 virus.
- A user is *diagnosed* if she has been tested and diagnosed COVID-positive.
- A user is *exposed* if she has received a notification that she has been in close proximity to some diagnosed user.
- An *exposure status* for a user is “true” if the user is exposed, and “false” otherwise.
- An *exposure verification* is a procedure to decide whether a given user is exposed or not.
- Pseudonyms refer to the identifiers broadcast by the application. As the validity of a pseudonym is limited in time, the same application uses multiple different pseudonyms.
- Contact pseudonyms refer to the pseudonyms that an application has received from nearby applications.

4. Existing Proposals

The two main approaches to proximity tracing followed in Europe, sometimes called “centralized’’ and “decentralized’’, fundamentally differ on the place where the risk assessment is performed (to decide if a user should be considered at risk or not). In the decentralized approach, the risk assessment is performed locally, on the user’s phone upon reception from the server of pseudonyms corresponding to infected users. In the centralized approach, the risk assessment is performed on a central server when it receives the pseudonyms of exposed users. The main features of the two approaches are summarized below.

Decentralized risk assessment

- Diagnosed users declare their infection status to the server by sending all pseudonyms they used during the contagious window.
- The server is used only to relay these infection declarations to all users. It stores only transient information.
- The exposure verification operation is performed locally, on users’ devices.

Centralized risk assessment

- Diagnosed users send the pseudonyms of their contacts (during the contagious window) to the central server.
- The exposure verification operation is performed on the server.
- Users contact the server to obtain their exposure status.

For instance, the ROBERT [2] system relies on a centralized risk assessment approach, while DP-3T [1] and the Apple/Google proposal [4] rely on a decentralized risk assessment approach. Existing proposals may reveal other differences but they are not necessarily inherent to the “centralized” / “decentralized” dichotomy discussed here.

5. Risk Sources and Threat Models

In this section, we introduce the risk sources ("adversaries" or "attackers" in the security terminology) that we consider in this document and their associated threat models. We consider the following risk sources: regular user, tech-savvy user, eavesdropper, health authority, backend and State-level adversary. Our definitions of threat models are consistent with the ones proposed in [1] and [2] (threat models associated with each risk source are recalled in Annex 1).

Although they address the same problem, the centralized and decentralized approaches to COVID-19 proximity tracing rely on different threat model assumptions. The main goal of the decentralized approach is to protect users against a malicious server or a state-level adversary and to prevent the leak of sensitive data due to attacks or negligence at the server side. Therefore, the role of the server is reduced as much as possible, and the exposure verification is performed on the user device. However some risks can remain as shown in the next section.

In contrast, the centralized approach puts more emphasis on the protection of users against other malicious users trying to infer who is infected. Hence, the role of the server in the centralized approach is more important including the verification of exposure. This design choice involves different privacy risks as shown in the following section.

6. Impact Analysis of the Two Approaches

Following the CNIL (the French Data Protection Authority) terminology [5,6], we distinguish:

- the **risk sources** with their associated threat model (a.k.a. attackers or adversaries in the security terminology)
- the **feared events** (use of personal data by a risk source that can have impacts on the privacy of data subjects)
- the **impacts** of feared events on data subjects
- the **severity** of impacts (in the severity scale proposed by the CNIL [6], the severity depends on the significance of the consequences and how difficult it would be for subjects to overcome them, which leads to a four level scale: negligible, limited, significant and maximum)
- the **threats** that can bring about these feared events (sequence of actions carried out by a risk source and leading to a feared event)
- the **feasibility** of the threats (which depends on the weaknesses of the system and the technical means and expertise of the risk source)
- the **likelihood** of the threats (which depends on the feasibility of the threat and its motivation)
- possible **mitigations** of the threats

We assume that the server is operated by an entity called “the authority”. For the sake of simplicity, we do not consider the risks associated with the medical centers (or hospitals) that can be involved in the infection declaration process, assuming that they can be trusted. If it is deemed to be useful, these entities can be included in a later version of the document. For each architectural option, we discuss the feasibility of the threats and possible mitigations. As stated above, we do not enter into the more subjective discussion about the relative importance of the impacts and priority setting, which is a political rather than technical decision.

Last but not least, we do not discuss here other key issues that are largely independent of the choice between the centralized and decentralized approach, such as accountability, time limitation (de-installation) or the effectiveness of the use of Bluetooth technology in this context. Finally, the reported severity does not take into account the possible mitigations.

6.1. Risks shared between centralized and decentralized approaches

Some risks are shared by the centralized and decentralized approach while others are specific to one approach. We use the risk taxonomy proposed by the DP-3T group [3], which includes the following risks that are applicable to both approaches:

- Inherent risks of proximity tracing systems
    - IR 1: Identify infected individuals
    - IR 2: Prevent notifications
- Risks of practical BLE-based systems
    - GR 1: Cause false alarms through BLE range extensions
    - GR 2: Cause false alarms through active relays
    - GR 3: Identify location with infected people present
    - GR 4: Disrupt contact discovery
    - GR 5: Tracking a Bluetooth enabled device
    - GR 6: Reveal usage of the contact tracing app
- Risks of networked systems
    - NR 1: Network identities reveal data about infected patients
    - NR 2: Traffic analysis reveals data about infected patients
- Risks of systems that store observed Bluetooth identifiers
    - SR 1/SR 2: Reveal social interactions and recompute risk score through local phone access

Since the above risks apply to both approaches, we do not discuss them in this document. Interested readers can refer to [3] for a full description. There are two exceptions though, IR 1 and GR 2, for which we do not follow [3], as explained below.

Revisiting IR 1: Identify infected individuals

IR 1 is presented as an inherent risk in any proximity tracing system against certain attacks (some basic examples involving few users are given in [3]). However, the scale of this risk is very different in the two approaches.

- In the decentralized approach, an adversary (e.g., a user) can exhaustively find who is infected or not among all the users that he has been close to (this action is inexpensive and undetectable). In addition, as described in [3]:

  > "In decentralized systems in which infected people share their identifier, there is an easier way for an attacker to learn, when she was in close proximity to an infected person, without creating multiple accounts. The attacker can simply match the set of infected identifiers against each of her recorded Bluetooth identifiers to determine when she was in contact with an infected person and use this information to reveal the identity of the infected."

  Therefore, an adversary is able to identify all diagnosed users he has been close to during a time window corresponding to a period of contagiousness. The sharing or publication of this information can lead to the stigmatization and harassment of all diagnosed users.

- In the centralized approach, in contrast, when the user is notified that she was in close proximity to an infected person, this user only knows that at least one encountered person has been diagnosed. Although a user is able to re-identify the infected individual if she has met only one person, this re-identification task is much harder otherwise. For example, one way to carry out this attack would be to create an instance of the application (registered on the server) for each encountered person, which is much more costly to deploy.

Therefore, risk IR 1 "Identify infected individuals" has a very large scalability in the decentralized approach.

To make a clear distinction between the scalability of this attack in the two approaches, we revisit the definition of this risk in the proposed taxonomy by using the following definition:

- **IR 1-1: Identify all infected individuals among encounters,** when the adversary is able to find diagnosed users among all persons he has encountered during a period corresponding to a contagious period. The attacker proceeds by collecting pseudonyms of each person encountered, and then correlating this list of pseudonyms with the list of infected users' pseudonyms published by the authority to determine when she was in

contact with an infected person and use this information to reveal the identity of the infected. This attack concerns only the decentralized approach (e.g., DP3T) and is not possible in the centralized approach (the source of exposure cannot be established as soon as a user has more than one encounter[3]). By extension, this risk also covers the detection that some targeted users have been diagnosed.
Source risk: Tech-savvy users
Feasibility: Easy
Scalability: Large (all infected users)
Impact: Stigmatization and harassment of diagnosed users
Severity: Significant
Likelihood: Significant
Possible mitigations: D3PT v2 proposed to associate a coarse-grain time information to each observed pseudonym. Using a TPM to protect data and the execution of the application on the phone could also mitigate this risk.

- **IR 1-2: Identify a targeted infected individual:** this is a specific case of IR 1-1 which is also possible in centralized approaches when the set of encounters of the user is limited to the target only. This attack can easily be carried out by turning on the bluetooth interface when in presence of the targeted individual, alone, then turning it off. In the ROBERT protocol, the procedure requires a dedicated application instance to the targeted individual (as the adversary becomes at risk).
  Source risk: Regular users
  Feasibility: Requires to dedicate an instance of the application (e.g., on a different user device)
  Scalability: Small (the user requires to solve a proof of work for each application instance)
  Impact: Stigmatization and harassment of a diagnosed user
  Severity: Significant
  Likelihood: Significant
  Possible mitigations: improvement of the proof of work (e.g., more CAPTCHAs) or probabilistic notifications as proposed in ROBERT in the centralized approach ; none for the decentralized approach (inherent risk)

In case of several colluding adversaries, or several applications running under the control of one adversary, the scalability of the **IR 1-2** risk becomes higher in the centralized approach. However, in this case the feasibility of this risk is much lower because it requires either several attackers or the presence of several devices.

Indeed, creating multiple accounts is inexpensive and undetectable in the decentralized approach -- the attacker can simply rotate accounts on the same device. This capability makes it

[3] Indeed, the authority provides a binary answer (at risk or not) and does not provide any information that could be used to identify the source of exposure.

easier for the attacker to de-anonymize a user. In the centralized approach, this de-anonymization attack puts the adversary at risk which implies the need to register another account. The cost of this operation would depend on the adopted countermeasures (e.g., proof of work, or anonymous tokens delivered from an trusted party).

Therefore, the risk of identifying infected users is a major intrinsic threat for the decentralized approach, in the sense it is easily achievable, scalable, inexpensive and undetectable. This threat is likely to be an obstacle to GDPR compliance given the ease of the attack. It also creates risks with respect to the credibility of the proximity tracing service.

Although risk IR 1-2 cannot be totally excluded with the centralized approach (e.g., in ROBERT), it involves an explicit action of the user and does not scale. It should be noted that finding that a single contact is diagnosed positive (which can naturally happen with an isolated elderly person who has a single contact, during his daily meal delivery for instance) is already possible during manual inquiries held by epidemiologists to identify contacts.

6.2. Risks specific to the decentralized approach

In this section, we list the risks specific to the decentralized approach. We reuse some of the risks proposed in [3], and we also add new risks that have not been previously explicitly identified (we mark them with “[new]” below).

LR 1-1: Linkability of infected identifiers on the server [new].
This risk happens when an adversary learns which pseudonyms belong to the same infected person. This linkability can be learned by the server if multiple broadcasted identifiers are associated to the same upload/network identifier. This risk has been identified but not named in [3].
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all infected users)
Impact: Link pseudonyms belonging to the same infected person on the server
Severity: Limited
Likelihood: Limited
Proposed Mitigations: To mitigate this risk, D3PT proposed to use a proxy and to send each identifier in different connections.

LR 1-2: Linkability of infected identifiers by users [new].
This risk happens when an adversary can compute all pseudonyms used by an infected user from the declaration of infection published by the server. Consequently, the adversary is able to learn which pseudonyms belong to the same infected person. This risk has been identified but not named in [3].
Source risk: Tech-savvy users

Feasibility: Easy
Scalability: Large (concern all infected users)
Impact: Link pseudonyms belonging to the same infected person by users
Severity: Significant
Likelihood: Significant
Proposed Mitigations: This risk can be mitigated by limiting the lifetime of seed used to generate pseudonyms or by using a data structure which aggregates all pseudonyms as proposed in D3PT v2.

SR 3: Location tracing after local phone access [3]
This risk of location tracing stems from the fact that phones locally store their own pseudonyms in order to provide them to the server in case of infection. An adversary can obtain this list of pseudonyms through physical access or remotely with the aid of a malware running on the phone. The adversary could then track the future and past location of the user with the aid of Bluetooth scanners deployed in an area.
Source risk: Tech-savvy users, State
Feasibility: Easy
Scalability: Small (need a physical or remote access to the phone and deployed Bluetooth scanners)
Impact: Tracking of users (diagnosed or not)
Severity: Significant
Likelihood: Negligible
Possible mitigations: To mitigate the risk, the phone can store its pseudonyms in an encrypted format where only the server is able to decrypt them (i.e., using the private key of the server). In this case, law enforcement could also require that users give them access to their mobile phone and ask the server the required key to decrypt pseudonyms.

SR 4: Location tracing of a targeted infected individual [3]
Adversaries that are able to link pseudonyms that belong to the same infected individual can leverage this information to track a user's path over the contagious period. This tracing is limited to the contagious window for which the infected individual shares her pseudonyms and for the duration for which pseudonyms become linkable. As this risk is a direct result of the risk **LR 1-2**, the same mitigations can be applied. This risk was called in [3] as **"**Location tracing of infected individuals", but we decided to make it more explicit by adding the word "targeted" to make a clear distinction from the risk **SR 11** below.
Source risk: Tech-savvy users
Feasibility: Easy
Scalability: Large (the adversary can monitor as many areas as he wants)
Impact: Tracking of diagnosed users
Severity: Significant
Likelihood: Limited
Proposed Mitigations: see **LR 1-2**

SR 10: Tracking of all infected users [new]
This risk is a direct result of **LR 1-1** where an adversary is able to learn which pseudonyms belong to the same infected person. Law enforcement agencies or third-parties colluding with the server which are able to observe pseudonyms broadcast by the application in public places are also able to track and monitor all infected users across time.
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all infected users)
Impact: Mass surveillance
Severity: Maximum
Likelihood: Limited
Possible mitigations: See **LR 1-1**. D3PT v3 also partially mitigates the capacity of an adversary by collecting broadcast pseudonyms at scale by spreading a pseudonym over multiple broadcasted messages.

SR 11: Re-identification of all infected users [new]
Direct result of **SR 11**. It is well known that location tracing leads users re-identification with few observations. Consequently, third parties colluding with the server which is able to observe pseudonyms broadcast by the application in public places could be able to re-identify all infected users.
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all infected users)
Impact: Mass surveillance
Severity: Limited
Likelihood: Limited
Possible mitigations: see **SR 11**.

SR 12: Reveal social interactions / collocation between all infected users [new]
Similarly to **SR 10**, any third-party colluding with the server, which is able to observe the broadcast pseudonyms in public places at scale is also able to monitor colocation information about infected users. This colluding is consequently able to **r**econstruct social interaction graphs between infected users. These risks were not acknowledged in [3] and are a direct result of **LR 1-1**.
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all infected users)
Impact: Mass surveillance
Severity: Maximum
Likelihood: Limited
Possible mitigations: To mitigate the risk, the mitigation of **LR 1-1/LR 1-2** can be used, however the colluding entity is still able to detect any infected users.

SR 13: Monitoring attack [new]
The decentralized approach can be used by an adversary to monitor the Bluetooth message in an area and to count the number of infected persons every day. This feature enables a curious user to perform statistics on the infection spreading in this area. This feature also enables the adversary to study the infection spread in areas of interest very easily. For example, a curious user just needs to deploy a phone running the application in a passing corridor (for instance) in the target area. This application will capture all the broadcast pseudonyms of all persons passing by during the study period. The adversary is then able to count how many people are infected in the monitored area. Note that the adversary can also re-identify the users by installing a camera and performing **SR 12**, described above. This simple attack (reachable by a regular user only leveraging a phone running the application) illustrates a serious weakness of the decentralized approach systems where the proximity testing is performed locally by the devices. This attack is concerning for several reasons: (1) if the group of people concerned is small (the scanner is installed discreetly in an office), data is almost personal; (2) it can reveal weaknesses in the key target organisations (companies, institutions, departments, units, etc.); (3) it enables the adversary to monitor the infection status of an area of interest, for instance the building where the attacker is living. If this information is made public, this could (1) enable a curious user to monitor its neighbors (2) make employees unwilling to go to their workplace or harm the reputation of an entity; (3) designate some location as blackspot or area to be avoided. This risk revisits GR3 for the decentralized approach which has a scale much larger than for the centralized approach resulting from the difference between **IR1-1** and **IR1-2**.
Source risk: Tech-savvy users, Server, State
Feasibility: Easy
Scalability: Large (the adversary can monitor as many areas as he wants)
Impact: Targeted surveillance / Mass surveillance
Severity: Maximum
Likelihood: Significant for Tech-savvy users, Limited for Server, State
Possible mitigations: see **SR 11** (if source risk is the server or the state) or **LR 1-2** (if source risk is a regular user of a tech-savvy user)

6.3. Risks specific to the centralized approach

In this section, we list the risks pertinent to only the centralized approach. We reuse some of the risks from [3], and we also propose new risks that have not been previously explicitly identified (we mark them with “[new]” below).

LR 2: Linkability of identifiers on the server [3]
If the server generates pseudonyms, it is able to learn pseudonyms belonging to the same person.
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all users)

Impact: Link pseudonyms belonging to the same infected person
Severity: Limited
Likelihood: Limited
Proposed Mitigations: none

DR 1: Risk of data breaches and data leaks [new]
An exploitation of a security vulnerability on the server can lead to revealing the pseudonyms belonging to the same person and the associated exposure risk. This risk has been identified but not named in [3], we decide to name it as DR 1.
Source risk: Server, State
Feasibility: Difficult (need to compromise the server)
Scalability: Small
Impact: Reveal all information stored on the server at the time of the data breach
Severity: Significant
Likelihood: Limited
Possible mitigations: To mitigate the risk, the information storage (or capability to retrieve it) has to be minimal. For instance, the mapping between permanent and temporary pseudonyms can be stored (or computed) on a separate server (potentially secured through TEE). The information stored on the server could also be encrypted with a key shared between the server and the App (as proposed in keyless ROBERT).

SR 7: Location tracing through access to a central server [3]
Due to the linkability of pseudonyms on the server (i.e., **LR 2**), law-enforcement is able to trace the location of any user of the system given Bluetooth observations.
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all users)
Impact: Mass surveillance
Severity: Maximum
Likelihood: Limited
Possible mitigations: none

SR 8: Reconstructing social interaction graphs [3]
The server can learn information about the social interactions of infected individuals from their contact pseudonyms. More precisely, as infections increase and contact pseudonyms are sent to the server, it could learn and reconstruct a graph of social interactions by analyzing the timestamp associated with each contact. For instance, if Alice is diagnosed COVID-positive and sends its contact pseudonyms (Bob at t0 and Charlie at t1) to the server, and then bob get infected later and sends its contact pseudonyms (containing Alice at t0, and Davis at t3), the server is able to reconstruct partial social graph.
This risk has been identified in [3], and declined in several sub related risks such as SR 5 (Reveal social interactions to the server) and SR 6 (Reveal colocation information about

infected individuals to the server). As these risks are included in SR 8, we decide to leave them out.
Source risk: Server, State
Feasibility: Easy (as contact pseudonyms are sent to the server, the adversary has to build a graph relying on both the pseudonyms of the contact and the associated timestamps)
Scalability: Large (rely on contact pseudonyms of all infected users)
Impact: Reveal social interactions between users (infected or not)
Severity: Maximum
Likelihood: Limited
Proposed Mitigations: To mitigate the risk, as proposed in ROBERT, infected users must send their contact pseudonyms in a random order through a proxy and each contact has to be sent through a different connection to break links between both the infected user and their contacts, and contacts together. To improve trust in this proxy, a trusted execution environment (e.g., Intel SGX) can be used to implement it. However, the timestamps associated with each contact can still reveal interaction between users.

SR 9: Reveal at-risk status to a central server [3]
By construction, the server learns which pseudonyms are at risk.
Source risk: Server, State
Feasibility: Easy
Scalability: Large (concern all non infected users)
Impact: Stigmatization of exposed users
Severity: Negligible
Likelihood: Maximal
Possible mitigations: None

7. Comparative Analysis

To provide a comparative overview of the two approaches, we summarize the risks according to their source in the following tables. The first one reports the privacy risks considering regular users as risk sources, and the additional risks related to tech-savvy users. The second table reports the privacy risks considering the server as a risk source and the additional risks related to the potential collusion of the server with law enforcement agencies (or subpoena).

Risk source	Decentralized	Centralized
Regular users	**IR 1-2** (Identify a targeted infected individual)	**IR 1-2** (Identify a targeted infected individual)
Tech-savvy users	**IR 1-1** (Identify all infected individuals among encounters) **LR 1-2** (Linkability of infected identifiers from users) **SR 3** (Location tracing after local phone access) **SR 4** (Location tracing of a targeted infected individual) **SR 13** (Monitoring attack)	

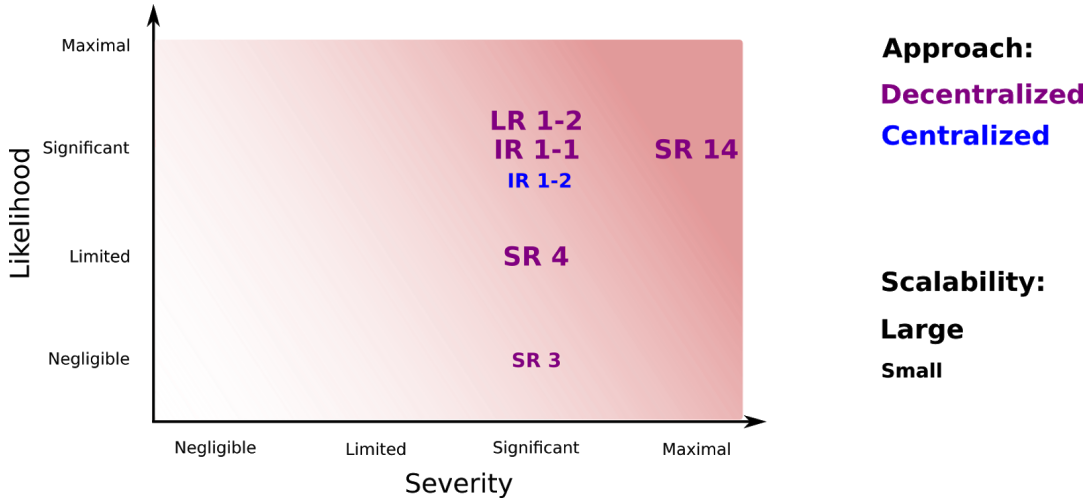

Figure 2: Severity versus likelihood of feared events from curious and malicious users (i.e., regular and tech-savvy) for centralised and decentralized approaches.

By design, the privacy risks **from curious or malicious users** are substantially different in the decentralized and centralized approach. As shown in Figure 2, the decentralized approach introduces many privacy risks by allowing users with technical skills to trace and re-identify infected users as well as to monitor targeted areas. This risk has a significant likelihood and potentially irreversible consequences on infected users. The most preeminent impact of all these risks is the stigmatization and harassment of diagnosed users, especially if the information is made public.

In contrast, the centralized approach drastically reduces the capability of any user in the system to learn information about infected users, users can only get information about herself from the server.

Risk source	Decentralized	Centralized
Server	**LR 1-1** (Linkability of infected identifiers on server)	**LR 2** (Linkability of infected identifiers) **SR 8** (Reconstructing social interaction graphs) **SR 9** (Reveal at-risk status to a central server) **DR 1** (Risk of data breaches and data leaks)
State (law enforcement agencies)	**SR 3** (Location tracing after local phone access) **SR 10** (Tracking of all infected users) **SR 11** (Re-identification of all infected users) **SR 12** (Reveal social interactions / colocation between all infected users) **SR 13** (Monitoring attack)	**SR 7** (Location tracing through access to a central server)

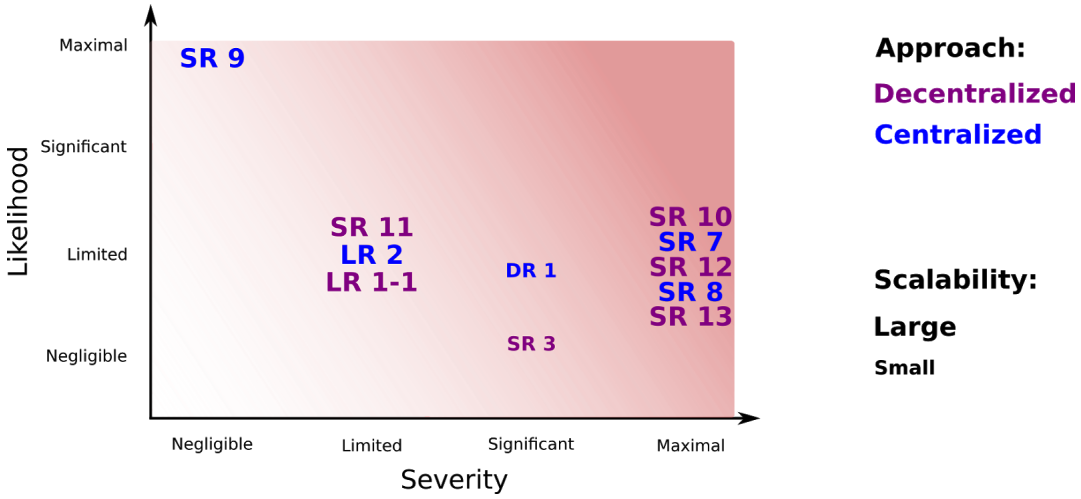

Figure 3: Severity versus likelihood of feared events from a curious server, a State-level adversary, or a server colluding with a third party for centralised and decentralized approaches.

By design, **the server** is not used in the same way in both approaches. In the decentralized approach, the server is just in charge of relaying pseudonyms to the users, while in the centralized approach the server handles the generation of the pseudonyms and computes the risk exposure. The role of the server in the centralized approach exposes the system to inherent risk such as data leak (Figure 3). In addition, as the server is able to link all pseudonyms belonging to the same person, the contact pseudonyms sent from infected users can be leveraged to infer social relationships between users (as explained in SR 8). In contrast, in the

decentralized approach, the server can only link all pseudonyms belonging to the same infected user.
Finally, with regard to the server colluding with a party controlling Bluetooth scanners, the only difference comes from the population targeted to the risks. The privacy risks concern only the infected users in the decentralized approach, while these risks concern all users (infected or not) in the centralized approach. However, the likelihood of these risks have to be considered.

8. Conclusions

The choice between the centralized and the decentralized approach leads to different privacy impacts for proximity tracing systems. This document presents these impacts from the technical perspective in order to enlighten the debate and the political decision to deploy or not such a solution and, if so, to choose among different technical options. As shown in the previous section, a key factor to decide upon available technical solutions is the level of trust that can reasonably be placed in the server and in users. Another essential aspect to consider is the benefit of the system to limit the spread of the virus. In this report, we have not discussed the effectiveness of proximity tracing applications, which would deserve a specific study. It should nevertheless be pointed out that the centralized approach may be more appropriate for health authorities because the server is aware of the number of exposed people, which can be used both for statistical purposes and to adjust the risk calculation algorithm (to decide if a user should be classified as “exposed”).

As a conclusion, it should also be stressed that the technical measures discussed in this document should be complemented by strict accountability measures. In particular, the server should be regularly audited by an independent body with all technical resources and access rights necessary to ensure that the trust placed in the server is not breached. This should be a key requirement for the deployment of such systems for both approaches, in particular when they follow the centralized approach.

References

[1] DP-3T - Decentralized Privacy-Preserving Proximity Tracing - version April 12th, 2020, https://github.com/DP-3T

[2] ROBERT - Robust and privacy-preserving proximity tracing - version 1.0, April 19th, 2020, https://github.com/ROBERT-proximity-tracing/documents

[3] Privacy and Security Risk Evaluation of Digital Proximity Tracing Systems - The DP-3T Project - 21 April 2020,

https://github.com/DP-3T/documents/blob/master/Security%20analysis/Privacy%20and%20Security%20Attacks%20on%20Digital%20Proximity%20Tracing%20Systems.pdf

[4] Google / Apple initiative, Privacy-Preserving Contact Tracing, https://www.apple.com/covid19/contacttracing/

[5] Privacy Impact Assessment - Methodology, CNIL, February 2018.

[6] Privacy Impact Assessment - Knowledge Bases, CNIL, February 2018.

Annex 1

Regular user
A typical user of the system that will not look at any information not available via the App UI nor will they try to tamper with it. They behave normally and will not change their movement patterns in any way to learn more. The majority of users fall into this category.

Tech-savvy user (Blackhat/Whitehat hacker, NGOs, Academic researchers, etc.)
This user has access to the system via the mobile App. Can set up (BT, WiFi, and Mobile) antennas to eavesdrop locally. Can decompile/modify the app. Can have access to the backend source code.

- (White hacker) Will investigate the App code, the information in the phone, and will look at what information is exchanged with the server (using an antenna or software installed on the phone, e.g., Lumen) or broadcast via Bluetooth (passive). Will publish this information and can create a backslash on public acceptance of the App.
- (Malicious) Can DOS the system (targeted or system-wide), tamper with authenticity (e.g., generate false contagion alerts), generate fake contact events, etc.

Eavesdropper (Internet Service Provider, Local System administrators, Bluetooth sniffer)
Can observer network communication (i.e., source and destination of packages, payload, time) and/or Bluetooth BLE broadcast messages.

- (Network adversary) Can use observed network traffic to determine the state of a user (e.g., whether they are at-risk, infected, etc.)
- (Local Bluetooth BLE Sniffer) Can observe local Bluetooth broadcasts (possibly with a big antenna to cover a wider area) and try to trace people.

Health authority
Receives information about at-risk people as a result of the proximity process. It is in personal contact with infected people and will reach out to at-risk individuals. It can combine knowledge about infected individuals with proximity tracing (and other background knowledge) to learn more about infected, at-risk, and non-exposed individuals.

Backend
Can access all data stored at the servers and query data from the mobile apps within the content provider operational scope. They could also change the code of their backend software and the code of the mobile apps. We assume they will not modify the mobile app because doing so would be detectable. They can combine and correlate information, request information from apps, combine with other public information to learn (co-)location information of individuals.
State-level adversary (Law enforcement, intelligence agencies)

They have the combined capabilities of the Tech-savvy user and the eavesdropper. Additionally can obtain subpoenas that give them capabilities of the health authority or the backend. They may want to obtain information about the population, but also target particular individuals. They may be interested in past information (what is already stored) or future information (that will enable target-tracing in the future).