Calculs distribués et sécurisés pour le cloud personnel

Riad Ladjel

université
PARIS-SACLAY

Secure Distributed Computations for the Personal Cloud

Calculs Distribués et Sécurisés pour le Cloud Personnel

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat: Informatique
Unité de recherche: Université Paris-Saclay, Inria, Inria Saclay Île-de-France, 91120, Palaiseau, France
Référent: : Université de Versailles-Saint-Quentin-en-Yvelines

Thèse présentée et soutenue à Paris-Saclay, le 08/12/2020, par

Riad Elarbi LADJEL

Composition du jury

Aline CARNEIRO VIANA Directrice de recherche, Inria Saclay	Présidente
Philippe BONNET Professeur, IT University of Copenhagen	Rapporteur & examinateur
David GROSS-AMBLARD Professeur, Irisa	Rapporteur & examinateur
Anne CANTEAUT Directrice de recherche, Inria Paris	Examinatrice
Mélanie CLÉMENT-FONTAINE Professeur, UVSQ	Examinatrice
Ludovic MÉ Professeur, Inria Rennes - Bretagne Atlantique	Examinateur

Direction de la thèse

Nicolas ANCIAUX Directeur de recherche, Inria Saclay	Directeur de thèse
Philippe PUCHERAL Professeur, UVSQ	Co-encadrant
Guillaume SCERRI Maître de conférences, UVSQ	Co-encadrant

Thèse de doctorat

NNT: 2020UPASG043

Acknowledgements

While getting this Ph.D. has been an important and gratifying achievement, the people who accompanied, helped and supported me over the past four years are what I will remember the most from this period.

First of all, I would like to thank my supervisors, Nicolas, Philippe and Guillaume. Without them, getting this Ph.D. would have been impossible. I thank them for the help they provided to me whenever I needed it. Each in their own way helped form me into the person I am today with their guidance and knowledge they helped imbibe in me. Beyond the scientific aspect, I also thank them for all the times they have encouraged me and cheered me up when I was losing my strength.

Of course, I would like to thank my beloved wife, Khadija, who was by my side all along my journey, for the best and the worst. She endured and supported me every time I needed it without ever complaining.

I would also like to thank my family who supported me by all the means at their disposal, I have always been able to count on their constant help.

I would like to thank Aydogan, Dimitris, and Julien L., with whom I have shared unforgettable moments, exciting and rewarding adventures and misadventures on a personal and professional level. I will always remember the lively and "barely" politically correct debates we had around a good meal, the Raclettes, Pizzas or Sushi evenings and all the evenings spent at the office working (or not). I will particularly remember all the times they stayed late at the office just to provide me with support and assistance. Meeting them was undoubtedly one of the best things during my PhD.

I would like to thank the other members of the Petrus team, past and present, who have in their own way contributed to my success: Luc, the super "Garant technique" who made sure that my defense went smoothly. Paul, whose seriousness and good humor have always inspired me. Laurent, the talented engineer who never refuses to help. Robin, Ludo and Julien M., the "promising" next generation of doctoral students, with whom I shared very good moments, and are the initiators of the "doctoral seminars" which are undoubtedly of great help in difficult moments of blockages. Iulian, Floris and Mariem, with whom I did not work directly, however, they were never stingy with advices and suggestions. Sébastien with whom I worked for several months and who taught me a lot about law. Not forgetting of course Emmanuelle and Régine, who accompanied me for my "very, very numerous" administrative procedures with incredible efficiency.

I would like to thank my friends Salim, Mohamed B. and Feriel for their proofreading of my manuscript, for the advices they gave me for my presentation and slides and for all the times they were by my side to cheer me up or to change my mind.

Finally, I would like to thank all the people that I have not mentioned but have undeniably contributed to my success.

Contents

List of Figures

Chapter 1

Introduction

Whether they come from smartphones, connected devices, sensors or smart meters the volume of generated and exchanged data is growing exponentially. In 2018, 33 Zettabytes (which represent 10^{12} Gigabytes) of data were produced all over the world. This enormously large volume of data continues to grow every day. The International Data Corporation (IDC) predicts that this number is expected to reach 175 Zettabytes in 2025 [111]. With an estimated generated revenue of 203 billion euros in 2020, this large amount of economically valuable data is, unsurprisingly, a gold mine for the people holding it. The World Economic Forum compares it to "the new oil" [59].

Traditionally, these data are collected and stored in centralized servers by large corporations (Google, Amazon, Facebook, insurance companies, etc.). This massive collection and centralisation of data allows the crossing of data from millions of users. Thanks to the very efficient big data algorithms developed during the last few decades, ranging from simple statistical analysis (groupings, aggregation) and automatic information search (automatic classification, rule discovery), to learning (based for example on neural networks), companies are now able to offer tailor-made services directly inspired by user behaviour, which increases productivity, ergonomics and usefulness. Thus, crossing data from multiple individuals is of utmost personal and societal interest.

Unfortunately, lately, this traditional model has shown its limitations. Indeed, centralization suffers from many drawbacks. Public awareness of the dangers posed by the data monopoly orchestrated by the Web giants began in 2013 when the whistle-blower Edward Snowden shed light on one of the biggest scandals of the 21st century [107]. Snowden revealed that the American government, through its intelligence agencies, was conducting massive surveillance of individuals with the complicity of data holders. However, this is not the only problem that centralization suffers from. In 2017 a report published by Cracked Labs [39] reveals how the different web companies share and pool the personal data of their users collected directly or indirectly and how this astronomical amount of data is used to create extremely accurate profiles containing sensitive and intrusive personal information of millions of individuals. In its report [39] Cracked Labs states that "The profiles that data brokers have on individuals include not only information about education, occupation, children, religion, ethnicity, political views, activities, interests and media usage, but also about someone's online behaviors such as web searches. Additionally, they collect data about purchases, credit card usage, income and loans, banking and insurance policies, property and vehicle ownership, and a variety of other data types. Data brokers also calculate scores that predict an individual's

possible future behavior, with regard to, for example, someone's economic stability or plans to have a baby or to change jobs". The result of this massive profiling is the manipulation of individuals that can range from simply influencing their shopping habits, to more worrying issues such as manipulation of public opinion by even going so far as to influence the results of an election. It was typically the case with the 2016 American elections, where the Cambridge Analytica scandal [34] revealed how the elections were influenced after analyzing the profiles of millions of Facebook users and using the information learned to influence the vote of the targeted individuals.

Moreover, data breaches are another element that undermines the centralized model. Whether they are intentional (misuse, malicious attack), or just by negligence (data leakage, mismanagement), these data breaches result in the leakage of a large amount of data. And their number is increasing more and more. Indeed, an attack against a server containing millions of records represents a big win for the attackers as the benefit-to-cost ratio is very high. Among the thousands of yearly breaches, one can cite Facebook's one, which in 2019 exposed 540 million user records on amazon's cloud servers due to poor security [119]. The same year Microsoft accidentally exposed 250 million customer service records [30]. The all-time record is held by Yahoo which suffered an attack, starting from 2013, that exposed 3 billion user accounts [105].

The result of this situation is that users lose control over their own data. These threats point the need for personal platforms which allow their users to collect, manage and share their own data. This is the essence of the self-data movement. For all these reasons many voices are calling for a reconsideration of the current architecture of the Web, including the founder of the Web himself. In 2018, Tim Berners Lee published an open letter [24] denouncing the monopoly of a few majors on the collection of personal data, he says in particular "the Web has evolved into an engine of inequity and division; swayed by powerful forces who use it for their own agendas." Thanks to smart disclosure initiatives, the new web that he describes in his open letter is no longer a dream or an impossible utopia.

The smart disclosure program started in 2010 with the blue button initiative which allows patients to download their personal health data by simply clicking on a "blue button". The former president of the United States, Barack Obama, said in September 2011 during the opening of an open government partnership event in New York city "We have developed new tools called 'smart disclosures' so that the data we make public can help people make health care choices, help small businesses innovate, and help scientists achieve new breakthroughs" [98]. The blue button initiative was so successful that it paved the way for other initiatives like the green button for personal energy usage data and the red button for personal educational data. The same initiatives have been proposed in Europe, first at a national level for each country, such as MiData [90] (energy, financial, telecommunications and retail data) in Great Britain or MesInfos [89] in France, then at a broader level, within the European Union, with the General Data Protection Regulation (GDPR) [99], and in particular its data portability prerogative. The data portability allows users to access their personal data from the companies or government agencies that collected them. In the French official journal the data portability is defined as "the data subject shall have the right to receive the personal data concerning him or her, which he or she has provided to a controller, in a structured, commonly used and machine-readable format and have the right to transmit those data to another controller without hindrance from the controller to which the personal data have been provided". This is clearly a big step to give users back control over their personal data and empower them. But it is not enough to help users escape from a captive ecosystem. Indeed,

the users need a technical solution which allows them to store, manage, share and exploit these retrieved data. This is exactly what personal data management systems (PDMS) also called personal clouds offer.

Personal data management system solutions are flourishing. Their goal is to empower users to leverage their personal data for their own good. They provide a way for individuals to store all their digital environment in the same place. This opens the way to new value-added services that were not possible with the centralized model. Indeed, users are now able to cross their data collected from different sources (e.g., crossing bank statements with shopping history or health records with data from connected watches, etc.). The different PDMS solutions will be reviewed thoroughly in Chapter 2.

While storing data, previously scattered over different silos, in PDMSs increases user control over them, collaborative uses of data are often overlooked in this context. However, as said above, the benefits derived from crossing data belonging to multiple individuals are considerable and has both personal and social advantages in many areas (healthcare, banking, smart cities, social assistance, etc.). For example, computing statistics or clustering data for an epidemiological or sociological study, training a neural network to organize bank records into categories or predicting diagnoses according to medical symptoms. A user may want to share her GPS position to have accurate traffic prediction [84], or her medical records to train a shared neural network so that it can detect several diseases [42, 103]. She may also want to adapt her energy contract based on her actual consumption without jeopardizing her privacy [92]. A naive approach to this problem is to send personal data to a trusted third party who will perform said collaborative computations. But as shown above, the "trusted third party" assumption is strong and unrealistic knowing all the threats against the centralized model. Moreover, sending personal data to a third party means loosing control over them and thus forsake one of the major advantage of the decentralized model.

The goal of this thesis is to overcome this unrealistic trust assumption and propose a computing framework that *allows the crossing of personal data of multiple individuals/PDMSs* and ensures them sovereignty over their data and the ability to make informed and independent choices. This raises two questions:

1. *How to preserve the trust of individuals on their PDMS while engaging their data in a distributed process that they cannot control?*
2. *How to guarantee the honesty of a computation performed by a myriad of untrusted participants?*

Answering these questions requires establishing mutual trust between all parties in a distributed computation. On the one hand, any (PDMS) participant must get the guarantee that only the data required by the computation are collected and that only the final result of the computation he consents to contribute to, is disclosed (i.e., none of the collected raw data can be leaked). On the other hand, the querier must get the guarantee that the final result has been honestly computed, with the appropriate code, on top of genuine data. Besides this, to have a practical interest the framework must be:

- ***Generic***, meaning that the framework is able to compute arbitrary functions, from simple statistics to complex machine learning algorithms.
- ***Scalable***, meaning that the framework can be run over a large number of participants (e.g. tens of thousands) without a deterrent overhead.

- ***Decentralized***, meaning that the computations are executed at the edge of the network, directly within the participant's devices.

Our contributions are the following:

1. We propose a generic and scalable secure decentralized computing framework which allows the crossing of personal data of multiple individuals/PDMS and provides the expected mutual trust and computation honesty properties. We qualitatively and quantitatively evaluates the scalability and security of the solution on practical use cases (group-by queries, k-means clustering).

2. We propose a solution to hide data-dependant communications that may leak information from attackers observing the network traffic. We quantify formally the privacy level provided by our solution.

3. We propose a concrete application of our framework in the medical-social field and we demonstrate the practicality of the solution through a real case-study conducted over 10.000 patients in the healthcare field and evaluate it in terms of security, performance and societal impact.

4. We define and formalize the personal agency, a product of the social sciences which forms the basis of individual empowerment, in the Personal cloud context and analyze to which extent the personal agency is achieved in current models. Finally, we show how our framework achieves it.

This thesis is organised in seven chapters. The current chapter introduces our work and gives the general context.

Chapter 2 introduces the concepts necessary to understand the contributions of the thesis and to position them with respect to the state of the art. In a first step, we will draw a panorama of the different families of PDMS solutions and show why current solutions cannot answer our objectives, notably the ability to perform computations crossing data of multiple individuals. We will then study the different existing techniques that are used in the literature to perform distributed computations and evaluate the possibility of applying them to our context. Finally, we will present the third topic related to our work, the use of secure hardware to perform computations in a database context.

In Chapter 3, we will first define and formalize the problem we are addressing. We then propose a framework that satisfies all the above objectives under the assumption that the communication patterns are hidden to the adversaries. Finally, we will assess the effectiveness of the framework and evaluate its security through a mix of real implementation and simulations. This chapter is based on a work [77] published and presented in the 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications / 13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE) in 2019[1] and presented in APVP'19[2] and BDA'19[3].

In Chapter 4, we propose a mechanism to remove the anonymous communication patterns assumption. We formally prove the robustness of the proposed solution against powerful

[1]https://forumpoint2.eventsair.com/QuickEventWebsitePortal/trustcom19/tc19
[2]https://project.inria.fr/apvp2019/programme/
[3]https://bda.liris.cnrs.fr/

colluding adversaries able to observe all the communication patterns and show that the information leakage is negligible even if all the participants except one collude to infer the data of the last one. This chapter is based on an ongoing work.

In Chapter 5, we present an application of the framework in the medical-social field using an on-going deployment of a PDMS on a district-wide basis and we assess the practicality and the adaptability of the framework even in constrained environments where the bandwidth is limited. This chapter is based on a work [76] published and presented in the 28th International Conference on Information Systems Development (ISD2019)[4] and [78] which is an extention of [76] published in Transactions on Large-Scale Data and Knowledge-Centered Systems journal volume XLIV (Special Issue on "Data Management - Principles, Technologies and Applications")[5].

Chapter 6 presents a work done in collaboration with lawyers. We will show how the properties of secure distributed computations can have a concrete interest for the individual in terms of empowerment, in line with the work accomplished by the European Union with the introduction of the right to data portability. We will particularly show how our solution can lead to strong user empowerment. This Chapter is based on a work done in collaboration with lawyers and published in the Global Privacy Law Review[6].

Finally, Chapter 7 concludes this thesis by summarizing the main contributions and giving some interesting directions for future work.

[4]https://isd2019.isen.fr/

[5]https://www.irit.fr/tldks/volumes/

[6]http://www.kluwerlaw.com/journals/global-privacy-law-review/

Chapter 2

Background Knowledge and Related Works

Contents

In this chapter, we present the three main topics related to our work: in Section 2.1 we review the main families of *Personal Data Management Systems (PDMS)* and show to which extent they tackle the challenges identified in the introduction. In Section 2.2 we present the classical approaches used in the database context to perform distributed computations: homomorphic encryption, secure multi-party computation, gossip-based protocols and local differential privacy. Finally, in Section 2.3 we present the last topic related to our work, computations using secure hardware.

2.1 Personal Data Management Systems

The *Personal Data Management Systems (PDMS)* [8], *Personal Information Management Systems* [1], *Personal Data Servers* [5] and *Personal Data Stores* [45] are all different names given to the same paradigm, the *Personal Cloud*. A personal cloud is a set of software and/or hardware solutions that allow their owners to gather their whole digital content, in a single place, stored and managed under their control.

In this section we will give a global overview of the different PDMS solutions' families as specified in [8]. For each family we will show the main features and the trust model. At the end of the section, we will discuss to which extent the current solutions tackle the challenges identified in the introduction.

2.1.1 Online Personal Clouds

Most of the existing personal cloud solutions fall under this category. Many industrial products such as: BitsAbout.Me [25], CozyCloud [40], Digi.me [47], Nextcloud [95], Meeco [87] or Perkeep [104] and even governmental programs like MesInfos.fing.org [89] in France or MyData.org [94] in Finland are representative of this family. Such solutions offer their users the ability to store their personal data in a central server managed by the provider. The users can access their data through the Internet. These solutions propose three features:

- ***Data collectors.*** The solutions cited above offer the possibility to automatically collect the users' personal data from the companies and administrations hosting them. This is made possible thanks to smart disclosure initiatives (e.g. Blue and Green Button, MyStudentData, etc. in the U.S.) and new regulations such as the GDPR in Europe. The data collectors use users' credentials to connect to the online services and fetch the latest updates to the personal cloud owner. For instance, CozyCloud implements this feature thought the *CozyCollect* application which allows the connection to different services, including banks, insurance companies, energy companies, and so on.

- ***Cross-computations.*** As the complete digital life of the individual is stored within the same location (instead of spread among many different databases held by the companies that generated/gathered them in first place), new computations involving data from different sources are made possible. For example, Cozy cloud, through *Cozy Banks*, offers the users the possibility to manage and monitor all their different bank accounts at the same place.

- ***Trusted storage.*** Within the cloud provider's infrastructure, users' data are compartmentalized, and the users can only access and perform computations on their own data. This logical separation guarantees a trusted storage. Some solutions allow the storage of users' encrypted data in different locations rather than in the provider's server which gives a higher level of protection.

Trust model. Even if there are some differences on how the gain of users' trust is achieved, the common point of different online personal cloud solutions is the promises they make to users. In particular, the cloud providers ensure that they will not observe nor disclose users' personal data, and these data are not exploited for anything not consented by the owner. The three main arguments put forward are: (i) the use of the best practices in terms of security standards

(e.g. using cryptographic primitives to protect data at rest, encrypt communication channels using SSL, etc.) (ii) the second argument is the establishment of legally binding contracts (e.g. having servers in location with high level of legal privacy protection) and a business model depending on the trust that users have in the provider (i.e. it is not in their interest to lose users' trust as this will inevitably lead to bankruptcy) and (iii) the last argument is the auditability and/or the accessibility of their code. Many of the online personal cloud solutions have an open source code that can be checked and verified by the community. For the remaining ones their code is audited by specialized companies.

These approaches rely on strong hypotheses in terms of security: (i) the PDMS provider and its employees are assumed to be fully-honest, and (ii) the PDMS code as well as all applications running on top of it must be trusted. This is critical in a centralized context exacerbating the Benefit-to-Cost ratio of an attack. On the other hand, collective computations are simplified by the data centralization but the security of such processing remains an issue.

2.1.2 No-knowledge Personal Clouds

To overcome the limitations inherent to online personal cloud solution that comes from the strong assumption of trust in the cloud provider, some solutions (e.g. SpiderOak [120] or Sync [121] and to a certain extent Digi.me mentioned above) propose an architectural variation which consists in encrypting the data stored in the cloud. These kinds of solutions provide two features:

- ***Secure storage.*** Unlike the online personal cloud solutions where the storage is trusted thanks to logical separation, in no-knowledge personal cloud solutions the storage is *secure* thanks to the encryption of data on disk. The personal cloud owner has the responsibility to store and manage the encryption keys elsewhere and the cloud provider has never access to the encryption keys.
- ***Secure backup.*** In a digital world, the risk of data loss due to, for example, mishandling or a malicious act by an attacker (e.g. ransomware) is non negligible. No-knowledge personal cloud solutions offer a secure point-in-time recovery that allows users to restore all their data at a given date in the past.

Trust model. The no-knowledge personal cloud solutions protect against: (i) a malicious or honest-but-curious cloud provider who tries to leak/observe users' data or uses the data outside of the owner's consent, (ii) an attacker who compromises the provider's servers to gain access to raw users' data and, (iii) a user device failure or corruption (e.g. ransomware).

The price to pay for this increase of security is the difficulty to develop advanced (local or distributed) services on top of no-knowledge personal clouds, reducing their use to a robust personal data safe. Indeed, if the users want to use their data to feed an application, they first need to download the whole content of their personal cloud, decrypt it and feed the application with the desired data.

2.1.3 Home Cloud Software Solutions

In online personal cloud solutions, the user's entire digital content is stored in a central server owned by the cloud provider. These solutions provide a high data utility but present two

major drawbacks: (i) the cloud provider may be corrupted and uses user data without their consent and (ii) a higher probability of a massive leak of data in case of an attack. The no-knowledge cloud solution addresses these limitations but sacrifices data utility. An alternative is proposed by the *home cloud software* solutions (e.g., OpenPDS [45], DataBox [66]). The personal data are managed at the extremities of the network (e.g., within the user's equipment) to circumvent the security risks of data centralization. Hence, queries on a user's data can be computed locally and only the result (not the raw data) is sent back to the querier. The features of the home cloud software family are:

- ***Trusted storage.*** The data storage is considered trusted because it is stored at the edge of the network, in user devices. For example, in OpenPDS, users accumulate their data in stores located in their smartphones/computers and can access, explore and share these data using a privacy-preserving framework.
- ***Cross-computations.*** As for online personal cloud solutions, the user's entire digital life is stored in the same location. This opens the way to new value-added computations. The *safe answer* system proposed by OpenPDS allows to answer queries while returning only the final result instead of the raw data used.
- ***Data dissemination.*** This feature is a direct consequence of the two previous ones. Indeed, the data are under user control and the users are granted with frameworks and tools allowing them to explore, share and manage their data as they want.

Trust Model. These solutions implicitly assume two statements: (i) the user device at the edge is trusted and cannot be tampered with and (ii) the framework and applications are trustworthy. But no serious nor formal guarantees are given to underpin these assumptions.

2.1.4 Home Cloud Plugs

The next family of personal cloud is a variant of the *home cloud software*. The users are equipped with a dedicated box that can store terabytes of data and run a server. This solution alleviates the burden of administering a server on the individual's device and logically isolates the user's computing environment from the box. As an example of this family, we can cite CloudLocker [41], MyCloud [93], Helixee [97] and many personal NAS solutions. The main features are:

- ***Trusted storage and backup.*** The dedicated hardware is plugged on the individuals' home internet gateway and is either connected to an external drive or it directly integrates it. The data are stored encrypted and the encryption keys are held by the plug. Usually, the users can access their plug through a central server which acts as a DNS and stores the IP addresses of each plug. This central server also acts as a backup server where data are stored encrypted and can be restored at any moment.

Trust Model. As for home cloud software, the home cloud plugs solutions protect against massive leaks as no central server has access to raw data. However, a strong security assumption about the hardware and the software is made. Unlike home cloud software, this assumption is supported by the fact that no other software than the one running on top of the plug can be installed. But still no formal guarantees are provided.

In conclusion, home cloud software and plugs focus on the trusted storage and backup features and an access to personal data at any moment. They, however, typically do not focus on security nor on the data related functionalities as providing these functionalities requires extending the trusted computing base.

2.1.5 Tamper Resistant Personal Server

Research projects such as Personal Data Server [5] or Trusted Cells [7] propose an enhancement for the home cloud plugs family by adding a tamper-resistant element (e.g. a chip) to the hardware. This tamper-resistant element embeds a minimal trusted computing base that may be formally proven secure and acts as a DBMS. The features of this family are:

- ***Secure storage.*** The database is embedded within the secure element, it inherits its security properties. An external flash memory (e.g. μ-SD cards) is used to store the encrypted data, while the encryption keys and the metadata are stored within the secure element.
- ***Secure cross-computations.*** Same as for the storage, the secure cross-computations come from the fact that the DBMS inherits the security properties of the secure hardware.
- ***Secure distributed computations.*** In [126, 124] algorithms based on Trusted Cells [7] are proposed to achieve secure distributed computations by relying on an untrusted central server leveraging its high computation capabilities. The data are encrypted or anonymized and then sent to this server which performs partial computations on them.

Trust model. The trust in this type of solution is achieved through: (i) the tamper-resistance of the hardware which makes software and hardware attacks highly difficult to perform (ii) the embedded DBMS is minimalist which makes its administration easy even for non expert users. However, the trust in the distributed computation is weaker. Indeed, using a central server introduces some vulnerabilities. The server is considered as a *malicious adversary* having *weakly malicious intents* [22]. In other words, the server may try to cheat, as long as it cannot be detected as this would be against its business model (i.e. cheating will cause financial damages).

This family seems to be the one fulfilling most of the desired features but still, it is poorly extensible, notably for the part of cross-computations and distributed computations. The limited computing power of tamper-resistant hardware forces the trade-off of privacy for utility (by, for example, resorting to a central server with more computing power). Moreover, the proposed algorithms to perform the aforementioned distributed computations are ad-hoc, making the generic computation objective difficult to achieve.

2.1.6 Conclusion

Many conclusions can be drawn from this analysis. First, some features (storage, backup) are common the all PDMS families with some variation on how they are achieved. But unfortunately, none of the proposed solutions cover the whole data life cycle as specified in [8] especially for the distributed computation part. However, as seen in the introduction, distributed computations are highly important nowadays since it is the cornerstone of big

personal data processing. Second, apart from the last solution, the trusted computing base is so big that it is hard to prove it formally. Combining the different solutions to take advantage of the benefits of each one is impossible because of the heterogeneity of the architectures and trust models. Finally, none of the proposed solutions tackles our objectives stated in the introduction. The tamper resistant personal server solution is the closest but is far from being generic or scalable to tens of thousands of users. The solution is to increase its computing power and propose adapted algorithms.

2.2 Secure Distributed Computations Schemes

In this section, we will present the different approaches in the literature for performing secure distributed computations. There are four main approaches: (i) Using *Homomorphic Encryption*, (ii) Secure *Multi-Party Computations (MPC)*, (iii) *Local Differential Privacy* and (iv) *Gossip-Based Protocols*. For each approach, we will present the main idea behind, some notable works, and explain why it does not achieve our objectives.

2.2.1 Homomorphic Encryption

The principle behind homomorphic encryption was introduced by Rivest, Adelman and Dertouzos in 1987 under the appellation *Privacy Homomorphisms* [112]. The idea was to allow computation over encrypted data. In other terms, let $E(a)$ and $E(b)$ the encrypted values of respectively the message a and the message b. Where E is a private homomorphism. Let $d = E(a) \oplus E(b)$, we have $D(d) = a \odot b$ where D is the decryption function associated to E and $\oplus, \odot$ are two (different or no) operations. After decades of research, the community came up with interesting results that can be summarized in three categories [2]:

- ***Partially homomorphic encryption*** are encryption schemes that are homomorphic for one operation. A lot of existing encryption schemes are partially homomorphic such as [65, 102] which are homomorphic for the addition or [113, 54] which are homomorphic for the multiplication.
- ***Somewhat homomorphic encryption*** schemes allow both addition and multiplication but for a limited number of times. Like [29] which allows an unlimited number of additions and one multiplication. In [28] they propose a way to query a private database using somewhat homomorphic encryption, but the set of possible operations is limited.
- ***Fully homomorphic encryption*** schemes are homomorphic for both addition and multiplication for an unlimited number of times. *Gentry* [62] was the first to propose a fully homomorphic encryption scheme in his thesis dissertation. Gentry showed that any function may be computed over encrypted data. His work was followed by many others (for example [130, 33]) mostly inspired by Gentry's framework.

Partially and somewhat homomorphic encryption cannot be used to meet our genericity requirement. Fully homomorphic encryption schemes are incompatible with our scalability requirement, even if huge improvements were made since Gentry's work, the computational cost remains too high to be practical in real world scenarios. As an example, in [36] the authors showed that it takes 5 minutes to encrypt an AES block.

2.2.2 Secure Multi-Party Computation

The goal of Secure Multi-Party Computations (MPC) is to allow n users $u_1, u_2, \ldots, u_n$ to *jointly* compute an *arbitrary* function $f(x_1, x_2, \ldots, x_n)$ over their *private inputs* $x_1, x_2, \ldots, x_n$ without learning anything more than the final result. For example, when computing the maximum salary of a group, it is possible to deduce that all the other salaries are lower than the maximum one but nothing else is revealed about the actual salaries.

The MPC problem was first introduced by Yao [133] in 1982 when he proposed a solution to answer the question "Two millionaires wish to know who is richer, how can they do so without revealing their wealth ?". Decades of research were made on this topic to find the most efficient solution. Many different approaches have been proposed to solve the MPC problem. However, all these approaches can be classified into three main categories:

Garbled circuits based

The first paper of this category *How to Generate and Exchange Secrets* [134] was proposed by Yao in 1986 for the case of two parties. The idea was later generalized to n parties by Goldreich, Micali and Wigderson [64]. The principle behind is the secure evaluation of boolean circuits. Indeed, any arbitrary function can be written as a composition of logical gates. Thus, if we have a solution that securely evaluates a logical gate, one can evaluate any function. Yao's idea follows four steps:

- Suppose that Alice and Bob want to compute a function. They collaborate to transform the function into a booelan circuit (or one of the parties does it and discloses it to the second party).
- One of the parties (say Bob) garbles the circuit, encrypts it and sends it to Alice together with his encrypted inputs.
- Alice needs to garble her inputs. To this end, she needs Bob's help as he is the only one who knows the encryption key. This operation is made possible thanks to *"1 out of 2 oblivious transfer"* [58].
- Alice has everything now to evaluate the circuit. She does so and reveals the final result to Bob.

Trust model. This approach guarantees a *computational security*[1] for any $t < n$ honest-but-curious colluding attackers. The bottleneck is the data oblivious transfer step.

Secret Sharing based

The building block of the second category of MPC protocols such [35, 23] is the *Shamir's secret sharing scheme* [117] also known as *(t,n)-threshold scheme*. The goal is to share a secret among n parties in such a way that the combination of any t or more shares is required to recompute the secret, while knowing less than t shares provides no information about the secret. Shamir's scheme works as follows:

[1] A cryptosystem is said to be computationally secure if it cannot be broken with the current computer technology within a reasonable amount of time.

- Bob wants to share his secret s over n parties with a threshold of $t+1$ parties to disclose the secret. He generates a polynomial of degree t : $P(x) = \alpha_t x^t + \ldots + \alpha_1 x + s$ where α_i are randomly selected in a finite field $\mathbb{F}_q$, where q is any prime power with $q > s$ and $q > n \geq t$.
- Bob evaluates P in n different points (i.e. $P(x_1), P(x_2)...., P(x_n)$). Note that the points need to be different from 0 as $P(0) = s$. Each couple $(x_i, P(x_i))$ is a share. Each share is sent to one party.
- When the secret needs to be rebuilt, $t + 1$ parties need to collaborate to reconstruct the polynomial using the polynomial interpolation. To reveal the secret, the polynomial needs to be evaluated in 0.

This simple but elegant principle is used to perform MPC. It is trivial to notice that having a set of t shares $(x_i, P_1(x_i))$ from a polynomial $P_1(x)$ and t shares $(x_i, P_2(x_i))$ from a polynomial $P_2(x)$ and using the sum of the shares $(x_i, P_1(x_i) + P_2(x_i))$, allows one to rebuild the polynomial corresponding to the sum of the two polynomials $P_1(x) + P_2(x)$. Using this property, it is possible to perform secure addition. However, multiplication is tricky. In [23] the authors proposed a solution that performs the multiplication without jeopardizing the security. Note that multiplications are costly in term of communication compared to additions which do not require any.
Trust model. Secret Sharing based category offers an information-theoretic security[2] for any $t < \frac{n}{2}$ honest-but-curious colluding attackers.

Fully Homomorphic Encryption based

The last category of MPC is based on the fully homomorphic encryption presented above in Section 2.2.1. Some notable works for this category are [43, 38]. The main idea is the following:

- A pair of encryption and decryption keys is generated. The encryption key is published to all the parties and the decryption key is shared using a threshold scheme among all the parties.
- Each party encrypts its input using the encryption key and sends it to the other parties.
- Each party can now compute the function over the encrypted inputs.
- Finally, the parties collaborate to decrypt the final output.

Trust model. FHE based category offers an information-theoretic security for any $t < n$ honest-but-curious colluding attackers or any $t < \frac{n}{3}$ active attackers.

Conclusion

The MPC research field is very active, many improvements were made during the last decades and some practical implementations were proposed. For example in *Secure Multiparty Computation Goes Live* [27] Danish farmers used MPC protocols to agree on the price of sugar beets. The proposed protocol in [79] can join medium size databases with 100k rows in a

[2] An unconditional security that does not depend on any assumption

few minutes using the *Sharemind* framework [26]. MPC objective (i.e. allowing to compute a generic function over private data while protecting the privacy of the inputs) is typically one of our objectives, it however does not scale to a large number of users. Moreover, most MPC protocols assume honest-but-curious users.

Note that some ad-hoc MPC protocols were proposed to solve some specific problems such as *secure sum* [118], *dot-products* [70], *private matching and set intersection* [60], etc. Some of these protocols can scale to a large number of participants but they are by nature not generic. Typically, MPC adaptations to distributed databases contexts, like SMCQL [20], either support only few tens of participants or are limited to specific database operations.

2.2.3 Local Differential Privacy

Another technique to perform a distributed computation is *local differential privacy* which is an adaptation of the global model proposed by Dwork in [51]. The general idea is as follows, having two datasets $\mathcal{D}_1$ and $\mathcal{D}_2$ which differ only in one row (i.e. the cardinality of $\mathcal{D}_1 - \mathcal{D}_2$ is 1). If $\mathcal{O}$ is the output of a certain query, the probability that $\mathcal{O}$ comes from $\mathcal{D}_1$ is almost equal to the probability that $\mathcal{O}$ comes from $\mathcal{D}_2$.

Unlike the other anonymization techniques (i.e. *k-Anonymity* [115], *l-Diversity* [86] or *t-Closeness* [80]), differential privacy applies on the process and not the data (i.e. by analyzing the data one cannot say if it is differentially private or not). Moreover, the protection provided by differential privacy is stronger.

The principle behind local differential privacy takes its origin from the *randomized response* [132] where the idea is to give means to eliminate the bias introduced in surveys by introducing some randomness which protects the individual answers. For example, we want to have statistics about an illegal behaviour (say drug addiction). We want participants to answer the question "Do you take drugs?" instead of saying "yes" or "no", each participant toss a coin without revealing the outcome:

- If the result is heads, the participant answers the truth.
- If the result is tails, the participant tosses another coin and answers "yes" if it is heads and "no" if it is tails.

It is easy to see why this is better for the respondents. Indeed, when a participant says "yes", one cannot distinguish if it is the truth or if it is a random answer. The privacy level can be adapted depending on how sensitive the collected data are by, for example, using an unfair coin.

The power of differential privacy comes from the fact that it is no longer necessary to know the attacker's capabilities. Thus, differential privacy guarantees that: (i) an attacker will not get any information about the individuals of the database and (ii) no matter what is the prior knowledge of the attacker, the privacy guarantees still hold. However, even if there are some real world implementations of differential privacy such as the US Census Bureau [85] for publishing statistics, Uber [72] for enabling a private query interface over user data for employees, Google [56] to compute aggregations over users' data and Apple [122] with its private collection of emojis, this adoption remains rare. The major obstacle is to design algorithms that give an acceptable level of privacy and acceptable utility.

Local differential privacy suffers from three drawbacks. First, implementing algorithms that can compute an arbitrary function of users' private data is not possible. Second, the

accuracy of the final result cannot be guaranteed (most of differential privacy solutions introduce noise over the data). Finally, differential privacy has shown its limitation when it comes to computing a function over many attributes [96].

2.2.4 Gossip-Based Protocols

Another approach is the use of an adaptation of gossip protocols to perform distributed computations. The initial goal of gossip protocols is not to provide privacy preserving computation schemes but to disseminate information in a peer-to-peer network. They work the same way as a rumor spread (i.e. information/data are transmitted first from a node to another node. Then, the two nodes holding the information spread it to another two nodes. This process is repeated until the information is spread all over the network).

Several works adapted gossip protocols to perform privacy preserving operations such as distributed filtering [31], clustering [4], anonymous content dissemination [32], aggregations [71]. The adapted distributed algorithms work on fragmented data exchanged among nodes, and noise is added to provide differentially private communication patterns that reveal data content.

Gossip protocols scale well but are not generic in terms of possible computations. Moreover, they consider an honest-but-curious threat model and cannot be adapted to reach all of our goals.

2.2.5 Conclusion

As seen in this section, classical techniques to perform computations in a distributed database context cannot be used to reach our objectives. They are either not able to compute generic functions (local differential privacy, gossip-based protocols) or not scalable for a large number of participants (homomorphic encryption, secure multi-party computation). Moreover, none of the above solutions consider the computation integrity objective nor the limited data collection.

2.3 Secure Hardware Based Distributed Computations

The last topic related to our work, is the use of dedicated secure hardware to securely compute over data. There are various types of secure hardware, embedded chips like the ones used in credit cards, SIM cards used in phones, encrypted hard-drives, secure co-processors and hardware security modules (HSM). Each technology has varied objectives ranging from accelerated processing and secure storage, to isolated execution and trusted computing. They are generally used in specific use cases and provide ad-hoc solutions to some existing problems. In the next subsection we will briefly describe the limitations of the aforementioned solutions with respect to our context. We will then discuss a more promising secure hardware technology called *Trusted Execution Environment (TEE)* which best fits our context.

2.3.1 Traditional Secure Hardware

Several works propose the use of specific secure hardware in databases context. These works can be sorted into two categories: (i) mono-user setting where the computations and/or data are controlled by the same controller. In other words, the data are held by (or belong to) the

same entity managing the computations done over them and (ii) multi-users setting where data are scattered over many users that manage the computations.

Mono-user setting

TrustedDB [17], Oblivious Query Processing [13] and Cipherbase [12, 110] make use of a secure co-processor to offer a secure query evaluation. They basically split the query processing into two parts, one executed on the untrusted part and the other executed within the secure co-processor. TrustedDB embeds a full SQL lite DBMS within the secure part while in Cipherbase, the secure co-processor is used only for cryptographic operations and expression evaluation. The encryption keys are stored in the secure part and used to decrypt the data and encrypt the result. The processing of the sensitive data is done inside the secure part, but due to limited storage capacity of the hardware, data are stored encrypted in an untrusted storage and are sent to the secure part only when needed. These solutions are centralized by nature and do not match our context.

PicoDBMS [109] and MILo-DB [9] are two DBMS designed to run in highly constrained environments (small RAM, slow write capabilities, etc.). Both use the tamper-resistance of smartcards to implement a full secure DBMS able the handle most traditional database queries (selections, joins, aggregations, etc.). The encryption keys are stored within the smartcard while the encrypted data are stored in an untrusted storage (e.g. flash memory). Both solutions are designed for a mono-user context but they can be used to execute simple queries in a multi-user context (as seen next).

Multi-users setting

Decentralized solutions based on secure hardware have also been proposed for aggregate queries. For example, in [125, 126] the authors propose a protocol to perform global computations such as SQL aggregates using a specific secure hardware [11] and architecture [7]. In their architecture, each node is equipped with a *Trusted Data Store (TDS)* with limited computing resources, storage, and low availability. A central entity called *Supporting Server Infrastructure(SSI)* is used to exchange encrypted messages between nodes and store intermediate results. Unlike the TDS, the SSI has high computing resources and availability. The TDSs are considered honest while the SSI is honest-but-curious. Based on the same technology as in [7] and a similar architecture the authors of [124] propose a way to perform privacy-aware mobile participatory sensing. Both solutions suffer from the same drawbacks: (i) the honest-but-curious central entity is not compatible with our setting, (ii) all the nodes share a common key used to encrypt data transiting between nodes, which means that if the security of one node is compromised, the privacy of the whole system fails and (iii) due to the low computing resources, the proposed protocols can execute a limited number of tasks which clashes with our genericity objective.

Conclusion

Ad-hoc secure hardware cannot be used in our context. The major obstacles are either their limited computing power, which prevent the design of generic and scalable protocols to perform secure and distributed computations, or the unrealistic assumption of equipping all participants with specific secure hardware such as secure co-processor.

2.3.2 TEE as Game-changer

The emergence of *Trusted Execution Environments (TEE)* [114] definitely changes the game. The following definition of TEE is given in [114]:

"*Trusted Execution Environment (TEE)* is a tamper- resistant processing environment that runs on a separation kernel. It guarantees the authenticity of the executed code, the integrity of the runtime states (e.g. CPU registers, memory and sensitive I/O), and the confidentiality of its code, data and runtime states stored on a persistent memory. In addition, it shall be able to provide remote attestation that proves its trustworthiness for third-parties. The content of TEE is not static; it can be securely updated. The TEE resists against all software attacks as well as the physical attacks performed on the main memory of the system. Attacks performed by exploiting backdoor security flaws are not possible."

In other words, a TEE is a secure area inside a main processor. It runs in parallel with the operating system. It combines tamper-resistant hardware and software components to provide integrity and confidentiality guarantees for arbitrary computations on sensitive data. More precisely TEEs provide three main security properties:

- *Isolation* for the code they execute. This means that a code being executed inside a TEE cannot be influenced by anything (user environment/OS) outside of the secure area.
- *Confidentiality* of the data within the TEE. This means that an attacker (or even the TEE owner) cannot leak the data processed inside the TEE (except the inputs and the outputs).
- *Remote attestation* that is a mechanism which enables the proof of the identity of the code running inside a TEE [6]. Attestation abstractly is a cryptographic hash of the running code together with its return value, signed with the secret key of the TEE. It allows for performing remote computations inside TEEs while obtaining integrity guarantees on the result.

Compared to ad-hoc secure hardware, TEE are now omnipresent in end-user devices like PCs (e.g., Intel's Software Guard eXtention (SGX) in Intel CPUs since the Skylake version in 2015), mobile devices (e.g., ARM's TrustZone in ARM processors equipping smartphones and set-top boxes) and dedicated platforms (e.g., TPM combined with CPU or MCU). All these solutions provide the three properties mentioned above with different levels of performance.

In the same way as with traditional secure hardware, works using TEEs in databases context can be sorted into the same two categories: mono-user and multi-users.

Mono-user setting

Several works use TEEs to provide security in a single database context. EnclaveDB [108] proposes a high performance database engine that uses the properties of TEE to guarantee confidentiality, integrity and freshness of data and queries even with a malicious database administrator and/or compromised operating system. The sensitive data are stored within secure enclaves together as compiled queries. Other works focus on secure key value store such as [123] which proposes an SGX-based log-structured merge tree key value store that ensures integrity, completeness and freshness. The proposed solution uses protected memory buffers outside the secure part to circumvent the limited enclave memory and optimize updates. SPEICHER [16] and ShieldStore [74] are two other secure key value stores based on SGX.

Secure indexes were also proposed using TEEs. HardIDX [61] leverages SGX enclaves to securely search over outsourced and encrypted data while maintaining high query performance. HardIDX implements only the search function inside secure enclaves and thus can be used as an efficient and encrypted database index. Oblix [91] is another search index over encrypted data. Unlike HardIDX, Oblix combines oblivious access techniques with the TEEs to hide the access patterns and prevent information leakage.

TEEs are also used in the distributed databases context. For example, Opaque [135] is a distributed data analytics platform implemented on SparkSQL that makes the access pattern oblivious. ObliDB [57] is another solution providing oblivious access pattern. VC3 [116] implements map reduce using Intel SGX. The mappers and reducers are executed in separate enclaves. VC3 ensures the security of the code and the processed data and provides integrity guarantees to the controller using the different properties of TEEs. Communications between enclaves are encrypted but the access pattern may leak information. M2R [49] and Observing and Preventing Leakage in MapReduce [100] propose a solution to overcome the access pattern leakage in VC3 by hiding communication patterns between mappers and reducers. The anonymity of inputs is ensured by adding a shuffle step between mappers and reducers. In [106], the authors propose another solution to execute map reduce using SGX, their proposal is close to VC3 one but proposes an easier and fault-safe solution that uses Lua to implement a lightweight MapReduce framework.

Most of these works have a unique controller, as opposed to our setting where no unique individual is supposed to be in control of the computation. Additionally, most of the time this controller also provides the data to be computed on. This greatly simplifies the problem as the same controller verifies all enclaves and organizes the computation.

Multi-users setting

The closest works to ours are the ones falling into this category. Ryoan [69] provides a framework for building a network of SGX backed sandboxes for executing "software as a service" computations. However, only the master enclave is supposed to obtain guarantees on the computation, instead of propagating trust in the whole system. Hence, the objectives of the computation are fundamentally distinct from ours. Authors in [101] propose privacy-preserving multi-party machine learning algorithms. They propose an adaptation of five machine learning algorithms that prevents the exploitation of side channels attacks by using a library of data-oblivious primitives. To prevent the leakage due to external data access, they propose to randomize the data and always access all the data. Their solution focuses on machine learning algorithms and is then not applicable in our context where one of the objectives is to compute arbitrary functions. In SEP2P [83] the authors propose a P2P personal data processing. Their goal is to provide a protocol able to select a list of random users to execute a query. They use distributed hash tables and CSAR protocol [15] to ensure that the selection process includes at least one honest user such that the whole random list is trusted. DISPERS [82] extends SEP2P to the query evaluation. The authors proposed a way to split and distribute the execution of a query to a set of randomly selected actors. Both works are targeted to the PDMS context but the main differences with our work is that: (i) they do not consider the integrity of the computation which is one of our main objectives and (ii) in their architecture, the users are not able to choose to participate to a computation or not, the only choice they have is to be part of the whole system or not.

2.3.3 Conclusion

TEEs are able to compute arbitrary functions over sensitive data (decrypted on the fly in protected areas) while guaranteeing data confidentiality (the execution cannot be observed outside of the secure part) and providing an integrity attestation (proof that the code executed in a remote enclave is genuine). This opens up new ways of doing secure distributed processing with the hope of reconciling genericity and scalability.

Unfortunately, TEEs are far from providing a complete solution on their own. Indeed, TEEs have been primarily designed to delegate the execution of a given code to an untrusted server in the cloud. Building similar security guarantees in a decentralized setting with thousands of participants running different pieces of code is a brand new challenge.

Moreover, while TEEs tamper-resistance makes attacks highly difficult and costly, it does not eradicate them completely. The authors of [10] classify these attacks into two categories: (i) *Attacks based on speculative execution* like Spectre [75] and Foreshadow [128], these attacks need to be fixed by the hardware manufacturer and (ii) *side channel attacks* [131], the TEE in this case behaves in a "sealed glass proof" mode [127], i.e., the confidentiality property is compromised, but the isolation and attestation properties still hold. These attacks are however complex to perform and usually require physical access to the TEE, which prevents large scale attacks. Unfortunately, TEEs corrupted by side-channel attacks cannot be detected by honest ones as their behavior is still the correct one and it should be taken into account when designing protocols using TEEs.

Chapter 3

Manifest-Based Framework (MBF)

Contents

This chapter is based on a work [77] published and presented in the 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications / 13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE) in 2019[1] and presented in APVP'19[2] and BDA'19[3].

In this chapter, we propose a generic and scalable secure framework for organizing the computation at the edge of the network that answers the two initial questions stated in the

[1]https://forumpoint2.eventsair.com/QuickEventWebsitePortal/trustcom19/tc19
[2]https://project.inria.fr/apvp2019/programme/
[3]https://bda.liris.cnrs.fr/

introduction (i) *How to preserve the trust of individuals on their PDMS while engaging their data in a distributed process that they cannot control?* and (ii) *How to guarantee the honesty of a computation performed by a myriad of untrusted participants?*

In our setting, each participant holds her personal data in a PDMS and is equipped with a trusted execution environment. We leverage the properties of TEEs to achieve our goals.

3.1 Problem Formulation

3.1.1 Architecture and Trust Model

The trust model considered in our context stems from the decentralized nature of the targeted infrastructure and the properties of the TEEs introduced in Section 2.3.2 (i.e. *isolation*, *confidentiality* and *remote attestations*).

- ***Untrusted user devices.*** No credible security assumptions can be made on the execution environment running on widely open personal devices (PC, laptop, home box, smartphone, etc.) managed by non-experts. We thus consider that the device OS and applications can be corrupted.
- ***Untrusted infrastructure.*** We also consider the communication infrastructure as untrusted. At this point we make the assumption that the communication flow incurred by the computed algorithm is made *data independent*, i.e., that personal data cannot be inferred by observing the communication pattern among participants. In Chapter 4 we will provide a solution on how to make the communication flow *data independent*.
- ***Large set of trusted TEEs, small set of corrupted TEEs.*** We assume that each individual owns a TEE-enabled device hosting his personal data (i.e., his PDMS). This is definitely no longer fantasy considering the omnipresence of ARM's TrustZone or Intel's SGX on most PC, tablets and smartphones. As explained in Section 2.3.2, a small subset of TEEs could have been corrupted by malicious participants (potentially colluding) to break their confidentiality with side-channel attacks.
- ***Trusted computation code.*** We consider that the code distributed to the participants has been carefully reviewed and approved beforehand by a regulatory body (e.g., an association or national privacy regulatory agency). But the fact that the code is trusted does not imply that its execution behaves as expected. Indeed, some malicious users may try to participate to the computation with another code that disclose more information that it should.
- ***Trusted citizen identity.*** We consider that citizens have been assigned a private/public key by a trusted (e.g., governmental) entity (e.g., as used today for paying taxes online). This prohibits attackers generating multiple identities with the objective to massively contribute to a computation to isolate a small set of participants and infer their data.

3.1.2 Problem Statement

The problem can be formulated as follows: how to translate the trust provided to the computation code by the regulatory body into a mutual trust between all parties participating

to the computation under the presented trust model? To solve this problem, the following properties need to be satisfied:

- ***Mutual trust.*** Assuming that the declared code is executed within TEEs, mutual trust guarantees that: (1) only the final result of the computation can be disclosed, i.e., none of the raw data of any participant is leaked and the result is honestly computed as certified by the regulatory body, (2) only the data strictly specified for the computation are requested from the participant PDMSs, (3) the computation code is generic and makes it possible to verify that any collected data is genuine[4].
- ***Local assurance of validity.*** The querier and each involved participant must be able to monitor *locally* (i.e., on its own, without relying on a central trusted party) that the computation is being performed in compliance with the code declaration, by *all* other participants. If any honest participant detects a validity violation, an error is produced and the computation stops without producing any other (partial) result.
- ***Resilience to side-channel attacks.*** Assuming a small fraction of malicious and potentially colluding participants involved in the computation with corrupted TEEs, our framework must (1) guarantee that the leakage remains circumscribed to the data manipulated by the sole corrupted TEEs, (2) prevent the attackers from targeting a specific intermediate result (e.g., sensitive data or data of targeted participants) and (3) maximize the Cost-to-Benefit ratio of an attack. Note that this is the best we can do assuming that the code manipulates clear data and that side channel attacks can be performed. In addition, the means to achieve resilience should maintain the communication flow independent of the data being processed (i.e., attack resiliency should not affect the *data independence*).
- ***Genericity and scalability.*** To have a practical interest, the solution must finally: (1) be generic enough to support any distributed computations (e.g., from simple aggregate queries to advanced machine learning computations) and (2) scale to a large population (e.g., tens of thousands) of individuals.

3.2 Mutual Trust

To provide the *mutual trust* property, we propose adopting a manifest-based approach. As described in Figure 3.1.

3.2.1 Global Overview of the Framework

Our framework is conducted in three steps :

Step1: logical manifest declaration. We call *Querier* an entity (e.g., a research lab, a statistic agency or a company, acting as a data controller in the GDPR sense) wishing to execute a treatment over personal data. The Querier specifies a *Logical Manifest* describing the computation to be performed, namely: its purpose, the source code of the operator to be run at each participant, the distributed execution plan materializing the data flow between

[4]Assuming data genuineness can be actually verified by the running code in any way (e.g., thanks to a digital signature).

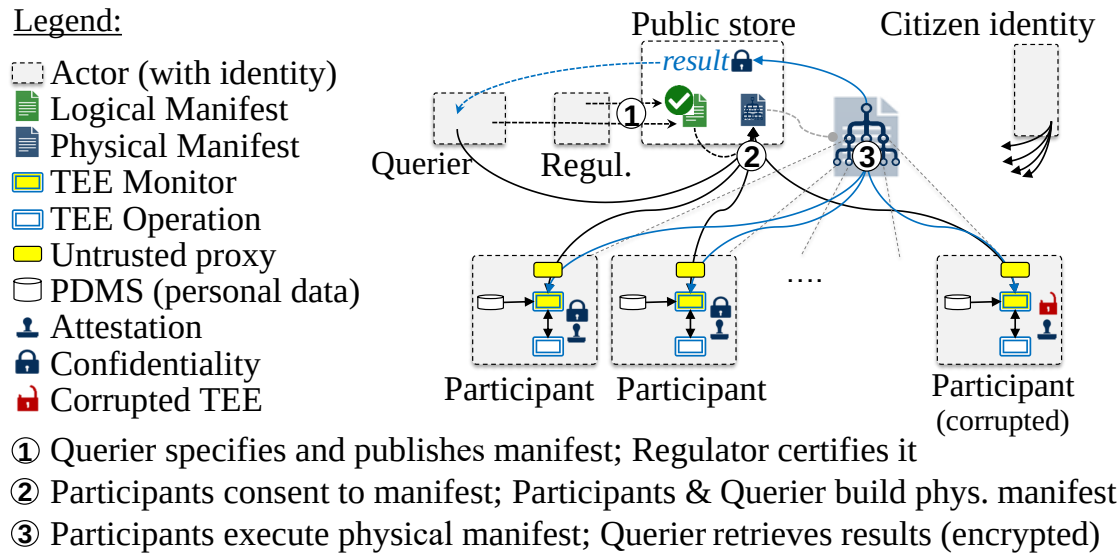


Figure 3.1: Manifest-based distributed computation

operators and a set of privacy rules to be fulfilled, including data collection rules and expected number of participants. The Querier submits this logical manifest to a Regulatory body which certifies its compliance with the expected privacy practices. The certified logical manifest is then published in a public manifest store where it can be downloaded by individuals wishing to participate. Example 1 illustrate a logical manifest (deliberately naive for the sake of simplicity) for a group-by query implemented using a MapReduce-like framework.

Example 1. *(GroupBy manifest)*

```
Purpose:
    Compute the mean quantity of anxiolytic prescribed to employees group
    by employer
Operators:
    mapper source code
    reducer source code
Distributed execution plan and dataflow:
    number of mappers: 10.000
    number of reducers: 100
    any mapper linked to all reducers
Collection rules:
    SELECT employer_name FROM Job;
    SELECT sum(qty)FROM Presc
    WHERE drugtype = 'anxiolytic';
Querier Public key: Rex2%ÃźHj6k7âĂę
Manifest signature : dF$3s1£
```

Step2: physical manifest construction. Once certified, the manifest can be viewed as a logical distributed query plan (participants are not yet identified). When a sufficient num-

ber of potential participants consent to contribute with their data, a *Physical Manifest* is collectively established by the TEEs of all participants (according to our trust model, each participant is equipped with a TEE). A physical manifest assigns an operator to each participant. As detailed in Section 3.4.1, this step is critical for *resilience to side-channel attacks*, by prohibiting corrupted participants from selecting specific operators in the query plan for malicious purpose.

Step3: physical manifest evaluation. Each participant downloads the physical manifest (or the subpart allocated to him). The participant's TEE initializes an enclave to execute his assigned operator and establishes communication channels with the TEEs of other participants supposed to exchange data with him (according to the manifest distributed execution plan). The participants then contributes his personal data to the operator and allows the computation to proceed. Once all participants have executed their task, the end-result is delivered to the querier.

3.2.2 Assessment of the Mutual Trust

Let us introduce the following definitions in order to analyze how *mutual trust* is achieved.

Definition 1 (Distributed Execution Plan)**.** *A distributed execution plan DEP is defined as a directed graph* (V, E) *where vertices* V *are couples* $(op_i, a_j) \in OP \times A$ *with* OP *the set of operators to be computed and* A *the set of computing agents, and edges* E *are couples* $(< op_i, a_j >, < op_k, a_l >)$ *materializing the dataflow among operators, namely the transmission by* a_j *to* a_l *of* op_i *output. For any* $v_i \in V$*, we denote by* $Ant(v_i)$ *(resp.* $Succ(v_i)$*) the antecedents (resp. successors) of* v_i *in the DEP, that is the vertices linked to* v_i *by a direct incoming (resp. outgoing) edge.*

This representation of distributed execution plans is generic enough to capture most distributed data-oriented computations. Based on this definition, we can introduce the notion of logical manifest.

Definition 2 (Logical Manifest)**.** *A logical manifest* LM *is as a tuple* $< PU, DEP, CR, N >$*, with* PU *the textual purpose declaration,* DEP *a distributed execution plan,* CR *the collection rule applied at each participant and* N *the expected number of participants.*

The CR declaration translates the limited collection principle enacted in all legislations protecting data privacy (i.e., no data other than the ones strictly necessary to reach the declared purpose PU will be collected). We assume that this declaration is done using a basic assertional language (e.g., a subset of an SQL-like language) easily interpretable by the Regulatory body on one side and easily translatable into the specific query language of any PDMSs on the participant's side. For the sake of simplicity, we assume that the data queried at each participant follow the same scheme (if it is not the case, it is basically a matter of translating the collection rules in different schemes). N plays a dual role: it represents both a significance threshold for the Querier wrt. the declared purpose and a privacy threshold for the Regulatory body wrt. the risk of reidentification of any individual in the final result. The notion of physical manifest can be defined as follows:

Definition 3 (Physical Manifest)**.** *A physical manifest* PM *is a tuple* $< LM, P, F, Q_{CR} >$ *such that: (1) function* $F : LM.DEP.A \mapsto P$ *assigns agents to the participants* P *contributing*

to the computation of LM; (2) F is bijective, so that a given participant cannot play the role of different agents and each agent is represented by a participant; (3) any query $q_i \in Q_{CR}$ is the translation for participant p_j of the collection rule LM.CR into the query language of his PDMS.

Definition 4 (PM valid execution)**.** *An execution of a physical manifest* PM *is said valid if the execution has not deviated in any manner from what is specified in* LM*, i.e., (1) the operators in LM.DEP.OP are each executed by the TEE of the participant designated by F while respecting the dataflow imposed by LM.DEP.E, (2) the TEE of any participant p_i queries its host with q_i, (3) N different participants contribute to the computation and (4) all data exchanged between the participants' TEEs are encrypted with session keys.*

Lemma 1. *Under the hypothesis* H1 *that the execution of a* PM *is valid and* H2 *that no TEE have been corrupted, the* mutual trust *property is satisfied.*

We postpone to Section 3.3 how to achieve hypothesis *H1* and to Section 3.4 the countermeasures suggested in the case hypothesis *H2* does not hold.

Proof. The three conditions in *mutual trust* definition given in Section 3.1.2 hold by construction. First, condition (1) is satisfied because *H1* guarantees that each operator in $DEP.OP$ is executed within a TEE, and *H2* and the TEE's confidentiality property ensure that no data can leak other than the input and output of each $DEP.OP$. Encrypting the data exchanges between each vertex v_i and $Ant(v_i)$ and $Succ(v_i)$ in DEP with a session key ensures the confidentiality of the global execution of $PM.DEP$. The final result is itself sent encrypted to the Querier so that no raw data other than the final result can leak all along the execution. Second, condition (2) stems from the fact that each participant p_i is presented with q_i which is a translation of $LM.CR$. The honest execution of q_i over p_i's PDMS remains however under the participant's responsibility who selected it to protect his personal data. Regarding condition (3), *H1* and *H2* again guarantee the integrity of the global execution of $PM.DEP$. Note that this guarantee holds even in the presence of corrupted TEEs since side-channel attacks on TEEs may compromise the confidentiality of the processing but not the isolation property. It immediately follows that any check integrated in the operator code can be faithfully performed on cleartext data, thus ensuring genericity. □

Compared to state of the art solutions, our manifest-based approach holds the capacity to reconcile security with genericity and scalability. First, the TEE confidentiality property can be leveraged to execute the computation code at each participant over cleartext genuine data. Second, the shape of the DEP and then the resulting number of messages exchanged among participants, directly results from the distributed computation to be performed. Hence, conversely to MPC, homomorphic encryption, Gossip or Differential privacy approaches, no computational constraints compromising genericity nor performance constraints compromising scalability need to be introduced in the processing for security reasons.

3.3 Local Assurance of Validity

Once mutual trust is ensured, one needs to ensure that each participant gets the assurance that the computation was performed as expected. Ideally, this means that the computation should behave as if all participants could continuously monitor all the others, i.e., check all

operator computations, ensuring correctness of the sent/received data at each step, and abort the whole process if any misbehavior happens. This is formalized in Definition 5.

3.3.1 Definitions and Naive Solution

At this stage, we assume that the execution plan has been produced by an arbitrary function, assigning a position i in the execution plan to each participant (the strategy for performing this assignment is discussed in Section 3.4). We also assume that the local code executed by a participant either terminates successfully or explicitly returns an error.

Definition 5 (locally checkable execution)**.** *The execution of a distributed execution plan* DEP *is said locally checkable if for any participant $p_j \in PM.P$, either (1) p_j's view of the partial execution up to p_j's role is valid or (2) p_j returns an error and no data is ever transmitted to other participants.*

An immediate consequence of Definition 5 is that, for any locally checkable execution, either a global result is produced if the execution is valid or no intermediate values is ever leaked. It follows that a protocol guaranteeing locally checkable executions for a *DEP* exactly provides local assurance of validity as any deviation from the normal execution would result in an invalid execution and would therefore result in an error at the participant's level.

As participants execute code in TEEs, a naive way to satisfy Definition 5 is to instrument the code of each operator in order to make sure that before sending out any (partial) result the code gets approval from all other participants. While this solution trivially satisfies our goal of local assurance of validity, the communication overhead with a large number of participants is overwhelming.

3.3.2 Proposed Solution

In order to overcome the aforementioned problem, we leverage the fact that using the TEE mechanisms and attestation, one can rely on checks made within other participant's TEEs. In our architecture, the foundation of local checkability is the decomposition of the code running at each participant in a generic TEE monitor and a specific TEE computation code. The objective of this distinction is to avoid the need for any participant to recompile the code running on the other participants and compute its hash to evaluate the validity of the requested remote attestations. The execution at each participant then works as follows: (1) untrusted code executed on the local host, called *untrusted proxy* in Figure 3.2, creates a TEE enclave and launches the TEE monitor code inside this enclave, (2) the *TEE monitor*, the role of which is to interpret the manifest and drive the local execution, creates a second enclave to launch the TEE computation code corresponding to the operator assigned to the participant in the execution plan. Note that all of the scheduling is performed by the untrusted proxy, in particular waking up TEE monitors as they are needed for the computation.

The TEE monitor code is identical for each participant, so that its hash is known by everyone. This code is minimal, can be easily formally proved and is assumed trusted by all participants. This lets us consider the manifest LM as data, including the code of the local operator to be computed, let each local TEE monitor check the integrity of this data and then attest the other participants (antecedents and successors in the execution plan) to the genuineness of the TEE computation code. Antecedents and successors can easily check in turn the validity of the received remote attestation by checking only the genuineness of

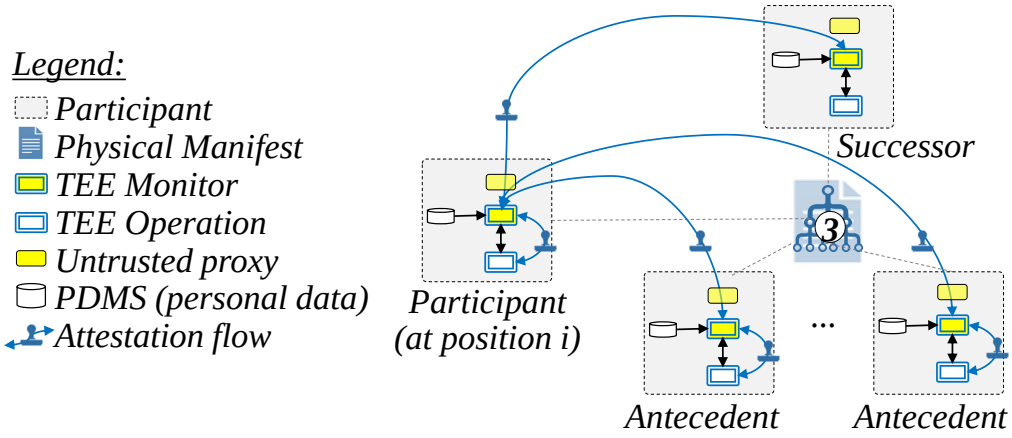


Figure 3.2: Attestation flow for position i

the remote TEE monitor. This double attestation by the antecedents and by the successors is mandatory to guarantee, for each participant, the validity of the inputs it receives and the authenticity of the recipients for its own outputs. This transitive attestation principle is depicted in Figure 3.2.

Following this strategy, local checkability is guaranteed. Intuitively, if a specific participant does not execute the genuine TEE monitor, it will be unable to provide a valid attestation to its partners (antecedents/successors) which will stop the execution and return an error. Then, if all participants run the correct TEE monitor and execute the same manifest, the execution is necessarily correct, since the TEE monitor only executes its dedicated code, and attestation prevents attacks from the OS on the result of the TEE computation code. If, however, one participant does not execute the correct manifest, its antecedents/successors will fail during the manifest verification. Finally, for any execution plan represented by a connected graph, the validity of the global execution is obtained by propagating errors through the execution graph, if an error occurs at any point during the computation. In order to prevent an attacker from running a large number of instances of a computation code in enclaves, each enclave must be tied to an identity, certified by a *citizen identity provider*.

3.3.3 Algorithm

The pseudo code of the TEE monitor is provided in Algorithm 1. For the sake of conciseness, we restrict this algorithm to the management of tree-based execution plans, however extending it to any graph is just a matter of allowing multiple successors. Note that the scheduling of the execution and errors propagation can be handled by untrusted code. Indeed, if a participant encounters an error, it would typically propagate it upstream so as not to let successor's enclaves hanging. However, it is by no means security critical as successor's enclave would simply never execute if they fail to receive their antecedents' inputs.

While hidden in the pseudo code, we assume that all communications between participants and the different enclaves are performed on secure channels. This is crucial to ensure that the endpoints of channels lie in real TEE enclaves and to prevent an adversary capable of observing the communications from getting access to user data. A primitive reaching this goal is called attested key exchange [19]. It allows to exchange a key with an enclave executing a

specific program, and hence ensures (using the attestation mechanism) that the endpoint of the channel lies within an enclave and that the enclave is executing the expected program, even if the administrator of the machine running the enclave is corrupted. We abstract this creation of a secure channel as $channel(remote, expected\_code)$ where $remote$ is the remote enclave and $expected\_code$ is the code expected to be running in the remote enclave. The cost is essentially 1 remote attestation and 2 communications. Once established, all communications are assumed to be done on this channel. For simplicity's sake we abstract away who is the initiator of the secure channel and view this process as symmetric.

Algorithm 1: TEE monitor

Input: LM the logical manifest, $id = (sk_i, pk_i, cert_i)$ the participants cryptographic identity and the corresponding certificate
Output: *boolean* indicating success

1 **if** $verify(LM) = false$ **then** `// verify manifest`
2 | **return** *error*
3 **end**
4 $PM \leftarrow Build\_phys\_manifest(LM, id)$ `// build phys. manifest`
5 $i \leftarrow get\_my\_position(PM, sk_i)$
6 $PM_i \leftarrow extract(PM, i)$
7 $Q_i, P_i, C_i, op_i \leftarrow Parse(PM_i)$
8 **foreach** $antecedent \in C_i$ **do** `// get antecedents' outputs`
9 | **if** $not(channel(antecedent, self.code))$ **then return** *error*
10 | **if** $not(id\_check(antecedent))$ **then return** *error*
11 | **if** $child.PM \neq PM$ **then return** *error*
12 | $input\_tuples+ \leftarrow accept\_input(antecedent)$
13 **end**
14 $input\_tuples+ \leftarrow out\_call(Q_i)$ `// query PDMS`
15 $E_{OP_i} \leftarrow create\_enclave(op_i)$ `// create` op_i `enclave`
16 **if** $not(channel(E_{OP_i}, op_i)$ **then return** error
17 $send\_tuples(input\_tuples, E_{op_i})$ `// produce output`
18 $res\_op_i \leftarrow accept\_input(E_{op_i})$ `// execute` op_i
19 $successor \leftarrow get\_successor(PM, res\_op_i)$
20 **if** $successor = querier$ **then**
21 | $a \leftarrow attest(res\_op_i, PM)$
22 | $send(a, res\_op_i)$
23 | **return** *success*
24 **end**
25 **if** $not(channel(successor), self.code)$ **then return** *error*
26 **if** $not(id\_check(successor))$ **then return** *error*
27 **if** $successor_i.PM \neq PM$ **then return** *error*
28 $send\_tuples(res\_op_i, P_i)$
29 **return** *success*

Algorithm description

In lines 1 to 7, the integrity of the logical manifest is verified by checking its signature, the physical manifest is built in collaboration with the other participating TEE monitors (cf. Section 3.4, which also covers the explanation of line 4, not required in this section) and the part of the manifest related to this participant is extracted (i.e., the set of its antecedents/successors, the data collection query used to retrieve data from the local PDMS and the code of the operator to be evaluated locally).

Then, in lines 8 to 13, the attestation of each antecedent is verified, by comparing the hash value of the code it is running to the hash value of the TEE monitor code (common to each participant). Once the antecedent TEE monitor is known to be correct, we check that it runs the correct manifest. We also check its identity by requiring its enclave to send it. This provides enough assurance because once we know the code of its enclave we know that it will honestly send its identity. Finally, the input tuples of the local operator are retrieved from its antecedents and/or the local PDMS of this participant.

In lines 15 to 17, the TEE monitor creates an additional enclave for the operator to be run (its code is part of the manifest) and requests an attestation from this enclave (the hash of the operator is compared to the hash of the code computed by the TEE monitor) to make sure that the host did not compromise or impersonate the operator code. Then the monitor establishes a secure channel with the operator enclave, using an attested key exchange as in [19] and TEE monitor calls the operator using the appropriate inputs.

Finally, in lines 18 to 29, the TEE monitor, either sends the result to the querier if its result is the final result, together with an attestation guaranteeing the result was indeed produced by the correct computation of the specified data; or sends its result to the next participants as planned by the DEP.

3.3.4 Assessment of the Local Assurance of Validity

Proposition 1. *Algorithm 1 satisfies the locally checkable execution property for the physical manifest* PM *derived from the logical manifest* LM *by the* build_phys_manifest *function.*

Proof. We sketch a game based proof that our protocol satisfies the locally checkable property. The goal of this proof is to show that performing an error free computation which is not a valid execution of DEP is equivalent to a game where, by construction, the execution is valid (up to the negligible probability of breaking either a cryptographic hypothesis or security of TEEs). Assume that at least one participant p performs an error free execution. We perform the proof in five game hops, successively removing bad events until the game is secure by construction:

- ***First game hop.*** We bound the probability that any participant does not execute the correct TEE monitor. This reduces to breaking the remote attestation property for the offending party as it necessarily is accepted by the participants it communicates with or an error would be produced.
- ***Second game hop.*** We forbid the event that a participant does not execute the same DEP as p. As all TEE monitors are honest at this step, and check agreement on the executed DEP, this reduces to a participant being able to inject a fake message into the secure channel between TEE monitors, i.e., breaking integrity of the secure channel between TEE monitors.

- ***Third game hop.*** We limit the probability that a participant is not executing the code allocated by DEP. This reduces to breaking the local attestation of the offending party's machine (if the code executed is forged) or security of the identity binding (if the code is supposed to be executed by another participant), i.e., security of signature schemes typically.
- ***Fourth game hop.*** We bound the probability that inputs/outputs to the computation codes are not the correct ones or are leaked. This reduces to breaking integrity or secrecy of the secure channel between the computation enclave and the monitor enclave.
- ***Fifth game hop.*** We bound the probability that messages exchanged between participants are the correct ones or are leaked, which again reduces to breaking the integrity/secrecy property of the secure channel between the TEE monitors.

Finally, we have an execution where all exchanged messages are correct and the whole code is executed as expected, which, by construction performs a valid computation. Note that if the execution is not error free, line 28 is never executed, and no data is sent, ensuring point (2) of the definition of locally checkable execution. Hence, the protocol proposed for executing a manifest achieves the locally checkable execution property: any participant, including the Querier, is guaranteed that any other participant runs the manifest as expected. Note that this sketch of proof holds for any connected graph and not simply for n-ary trees.

□

3.4 Resilience to Attack

According to our trust model, a small fraction of TEEs can be instrumented by malicious (colluding) participants owning them to conduct side-channel attacks compromising the TEE *confidentiality* property. This issue is paramount in our Manifest-based approach which draws its genericity and scalability from the fact that computing nodes manipulates cleartext genuine data, putting them at risk.

The *resilience to side-channel attacks* property introduced in Section 3.1, states first that the leakage generated by an attack must be circumscribed to the data manipulated by the sole corrupted TEEs. This is intrinsically achieved in our proposal by never sharing any encryption key among different nodes. A second requirement is to prevent any attacker from targeting specific personal data. *Randomness* and *Sampling* are introduced next to achieve this goal. Finally, DEP *reshaping* is proposed to tackle the third requirement, i.e., maximizing the average Cost/Benefit ratio of an attack.

3.4.1 Randomness

In a physical manifest, we distinguish participants assigned to a collection task (which contribute to the query with their own personal raw data) from participants assigned to a computation task (which process personal data produced by other participants). Attacking any TEE running a collection task has no interest since the attacker only gains access to his own personal data. Hence, the primary objective of an attacker is to tamper with the building phase of a physical manifest such that his TEE be assigned a computation task to leak the data it manipulates. The goal of our *randomness* counter-measure is to assign a random position in

the DEP to each participant to prevent any potential attackers (Querier or any participants) colluding with corrupted TEE from being assigned a computation task.

Definition 6 (Provably random execution plan)**.** *A distributed execution plan $PM.DEP$ is said to be provably random if any participant $p_j \in PM.P$ can verify that its position and the position of any other participants in $PM.P$ has been obtained randomly.*

If this condition is not met, the execution of the manifest must be aborted. We propose a solution to collectively construct $PM.DEP$ and demonstrate that it complies with the *provably random execution plan* property.

While existing solutions have been proposed to ensure that a random number is chosen and attested in distributed settings, e.g., [15], none can be applied to reach this specific goal as they assume the list of participants is known in advance, as opposed to our case where the participant list is chosen based on collected users' consents.

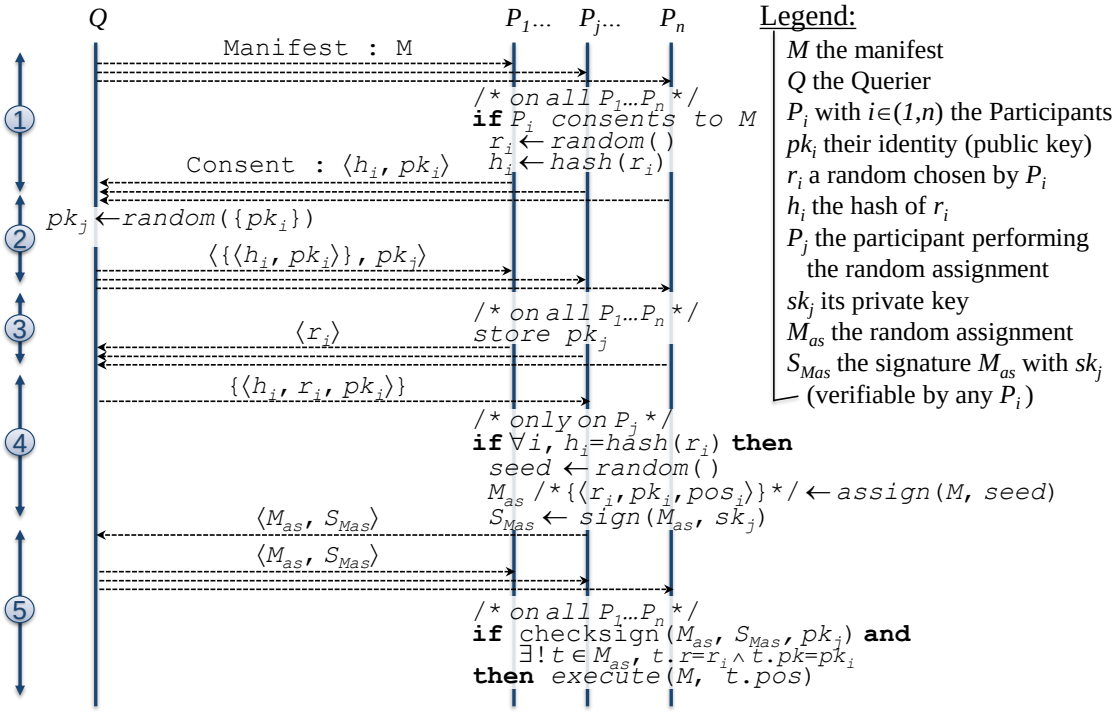


Figure 3.3: Sketch of the randomness protocol.

We propose a solution, sketched in Figure 3.3, to produce a provably random assignment. As we consider TEEs as trusted, the random assignment can be delegated to any TEE. The main difficulty relies on the fact that, although TEEs can choose a random number honestly and attest it, the challenge is avoiding any malicious Participant or Querier aborting and replaying the assignment process a large number of times, picking the best one for a potential attack. To avoid such attacks, we make sure, in a first step of our protocol, that the Querier commits to an assigning participant among the consenting participants. The process of assigning randomly position follows five steps:

- First, each consenting participant generates a random number and declares himself by

publishing his identity and the hash of the generated random number used later to prove reception of the list of participants.

- Second, once enough participants have committed to this unique identifier, the querier chooses an enclave randomly which will be designated as the randomness generator. As the owner of this enclave already has committed to a specific enclave identifier, the aforementioned enclave replay attack cannot be performed as a replayed enclave would generate a new unique identifier. All participants are informed of all commitments and the identity of the randomness generator.

- Third, all participants open their commitments. These opened commitments ensure that all participants are on the same page in terms of who is participating with which enclave identifier and who is the randomness generator.

- Fourth, the designated assigning participant is sent the full list of participants together with all the acknowledgements. He then checks that all acknowledgements are valid, and performs a random assignment of operators to participants.

- Finally, the randomness generator signs the assignment and sends it back to the Querier who broadcasts to all participant enclaves which can each deduce from it their own position in the execution plan. This final assignment being performed by *TEE monitor* (line 4 in Algorithm 1), the code of which is certified and common to all participants, the honesty of the assignment is ensured and can be verified by all participants.

Thus, the protocol ensures that when an individual consents to the execution of a manifest, the assignment can only be made once, at random. Any attempt to replay the assignment would be visible to the participants and a restart require to obtain their consents again.

Randomness Algorithm

The pseudo code of the randomness algorithm is provided in Algorithm 2.
Algorithm description. We describe the main phases of Algorithm 2 in more details. During the commitment step, each participant starts a fresh TEE monitor enclave, and feeds it with the logical manifest of the computation to be performed together with the cryptographic material corresponding to its identity. Upon initialization, the TEE monitor enclave generates a random number (which will act as a unique identifier for this specific enclave). Note that at this stage any (malicious) participant may start as many TEE monitor enclaves as he wishes and therefore has (to some extent) choice over the generated random identifier, thus, identifiers should be large enough to ensure that collision are highly improbable (typically 128 bits). Also, at this stage, identifiers are not a valid source of randomness as participants have some control over them.

Once a participant's TEE monitor has chosen its identifier, a cryptographic hash of said identifier (acting as a cryptographic commitment to the identifier value) is then transmitted to the querier, together with the identity of the participant. The querier then selects a participant (ideally at random) who will be choosing the randomness used in generating the physical manifest. The querier then transmits this choice to all participating enclaves, together with the list of hashes produced at the previous step.

Upon reception, all the participants' TEE monitor enclaves send their identifier to the querier. This is a way of ensuring that the querier has indeed sent the identity of the choosing

Algorithm 2: build physical manifest

Input: M the logical manifest, $id = (sk_i, pk_i, cert_i)$ the participants cryptographic identity and the corresponding certificate

Output: physical manifest with provably random assignment of participants

```
1  if not(check_certificate(pk_i, cert_i)) then return error
2  if not(check_key(pk_i, sk_i)) then return error
3  rid ← random()                          // get random identifier
4  hrid ← hash(rid)
5  send(hrid, pk_i)                        // send to Querier
6  l_hrid, l_pk, pk_c ← receive()  // receive enclave identifiers and identities from
     Querier
7  send(rid)                               // send to Querier
8  if pk_i = pk_c then                     // enclave selected as randomness generator
9  |  lrid ← receive()
10 |  foreach x, y ∈ l_rid, l_hrid do                          // check id list
11 |  |  if hash(x) ≠ y then return error
12 |  end
13 |  seed ← random()                      // generate honest seed
14 |  M_as ← assign(M, seed)
15 |  S_Mas ← sign(M_as)
16 |  send(M_as, S_Mas)
17 end
18 M_as, S_Mas ← receive()                 // (everyone) gets M_as and S_Mas
19 if not(check_sign(M_as, S_Mas, pk_c)) then
20 |  return error                                          // check generation
21 end
22 PM ← M_as, S_Mas
23 return PM
```

enclave to all the participants. Otherwise, a malicious querier could designate a choosing enclave, and ignore its answer and try another if it does not return the expected random assignment of positions. This step is also mandatory because the solution must support (a small number of) participants disconnecting for unpredictable reasons. If the querier has not proven that the identity of the choosing enclave is known to a majority of participants before getting the random assignment, it can simply act as if the choosing enclave had disconnected. Note that at this stage some enclaves may not respond, they are simply removed from the list of participants.

After that, the querier sends all the hash values and identifiers to the choosing TEE monitor enclave. The choosing enclave checks that the hashes indeed correspond to the identifiers and then generates a random seed that will be used to build the random assignment.

The randomness generator signs the assignment and sends it back to the Querier together with the signature, which subsequently broadcasts them to all participants. From this point on, the random assignment and its signature are added to the physical manifest. All TEE monitor enclaves check that the signature is indeed valid, and was indeed produced by the enclave designated at the previous step.

Proof. The result can be obtained using a game-based proof in order to show equivalence with an idealized version of the game, where the events where the participant list is not agreed upon by every participant is excluded and the random assignment is magically chosen by an oracle.

- The first game hop consists in bounding the probability that not all participants agree on the choice of a choosing enclave, even if the adversary is allowed to fake failure of a small number of potential choosing enclaves. As the participant list is transmitted to all enclaves before they open their commitment, all potential participating enclaves must have received a list or the adversary has broken security of hash functions. Note that in the very next phase of the computation, in order for the computation to execute, all participants must agree on the physical manifest, therefore the transmitted list and choice of choosing enclave must be the same for all participants. Note that without forcing the adversary to transmit the list to all participants before sending it to the choosing enclave, it could try several choosing enclaves in order to obtain a "bad" random assignment, and then remove them from the computation if the random assignment they provided does not suit its purpose.

- The second game hop simply consists in remarking that the isolation property makes sure that the assignment chosen by the (unique) choosing enclave is truly random, and confidentiality ensures that no information on this random assignment can be obtained before it is output by the choosing enclave. It can therefore be substituted with the random assignment provided by the oracle without the adversary being able to notice this.

Finally, we conclude by noticing that the target game satisfies provably random execution by construction. □

3.4.2 Sampling

The second counter-measure we propose to force an attacker to increase the number of corrupted TEEs, and then the Cost of the attack, simply consists in adding a *sampling* phase in

step 2 (physical manifest construction) by selecting a given rate σ of individuals accepting to contribute to the computation. The less this rate σ the more TEEs need be corrupted to keep the same probability of success for an attack.

More precisely, for a computation where n participants are needed, $\frac{n}{\sigma}$ consents are collected during the consenting phase instead of n. During the random assignment process, the assigning enclave selects n random participants among the $\frac{n}{\sigma}$ to be part of the computation. The remaining ones are just discarded.

It is trivial to see how the sampling indeed increases the cost of an attack. As the probability for any enclave to be effectively selected among the participating ones depends on σ (i.e. the probability to be selected is σ). An attacker needs to corrupt more TEEs to have the same probability of being selected and thus it is more costly to perform an attack.

3.4.3 DEP Reshaping

In our context, the *Cost* factor of the *Cost-to-Benefit* ratio is expressed in terms of number of TEEs to corrupt while the *Benefit* is measured by the amount of personal data leaked by the attack. While *randomness* and *sampling* contribute to exacerbate the *Cost* factor, our third countermeasure aims at reducing the amount of raw data exposed at a single TEE. To introduce the idea, let us consider a DEP with n participants, among which m computation nodes computing a function f and $n-m$ collection nodes contributing with their own data. According to the *randomness* countermeasure, the probability to corrupt exactly t computation nodes among m with c corrupted participants (i.e., conducting side-channel attacks), follows a hypergeometric distribution (i.e. $P_{n,m,c}(x=t) = \frac{\binom{m}{t}\binom{n-m}{c-t}}{\binom{n}{c}}$). The probability of corrupting t or more computation tasks over m is then:

$$P_{n,m,c}(t \leq x \leq m) = \sum_{t \leq i \leq m} \frac{\binom{m}{i}\binom{n-m}{c-i}}{\binom{n}{c}}$$

With $n = 10000$, $m = 10$ and $c = 100$ (which is a high number of corrupted participants, for illustrative purpose), the probability of corrupting at least one computation node is $P_{10000,10,100}(1 \leq x \leq 10) = 0.095$, while with $m = 100$ this probability falls to $P_{10000,100,100}(10 \leq x \leq 100) = 5.2 \times 10^{-8}$.

For simplicity, we assume that each participant contributes with exactly one tuple mapped to a single computation node, hence each computation node processes (and may endanger) on the average N/m tuples (1000 tuples in the considered settings).

Figure 3.4 (left side) plots the privacy benefit of increasing the number of computation nodes by reshaping the DEP so that each initial computation node m_i is split in rf new computation nodes sharing m_i's initial computing load, with rf denoting the *reshaping factor*. More precisely, this curve plots the probability to leak the same amount of data as with the original settings in function of the number c of corrupted nodes with different reshaping factor rf (e.g., $rf = 1$ is the initial settings with $m = 10$ computation nodes, $rf = 2$ means $m = 20$, etc.). Unsurprisingly, increasing rf dramatically decreases the probability of attack leaking the same amount of tuples since rf different computation nodes (among $rf \times m$) must now be corrupted with the same number c of corrupted participants.

Figure 4 (right side) goes further and shows the expected number of leaked tuples (i.e., sum of the probability of a successful attack on some computation nodes times the number of tuples leaked) in the case of successful side-channel attacks with $c = 100$ corrupted nodes.

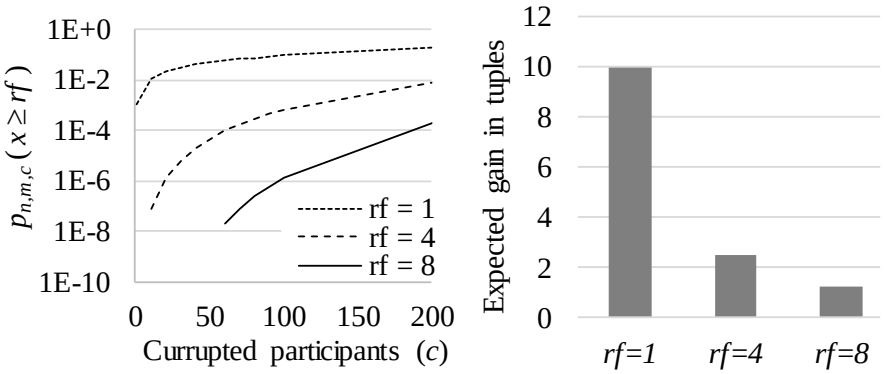


Figure 3.4: Left: probability of successful side-channel attacks for massive leaks (10% of the input tuples); Right: expected values in tuples of a successful side channel attack on 100 corrupted and colluding participants.

The expected gain is always small, although the number c of corrupted TEE is relatively high, and reduces linearly with rf, which is deterrent for attackers. Indeed, the probability to successfully break two computation nodes is close to zero, hence the expected gain is nearly given by the probability of breaking a single computation node times the number of leaked tuples processed in that node, which linearly decreases with rf. These results remain true as long as m and c are small compared to n, which is typically the case in our context.

The conclusion is that, while maximizing the distribution of a computation has recognized virtues in terms of performance and scalability (explaining the success of MapReduce or Spark models), this strategy leads as well to a better resilience against side-channel attacks. Maximizing the distribution can be done by exploiting some properties of the functions to be evaluated by *DEP* computation nodes:

Definition 7 (Distributive function)**.** *Let f be a function to be computed over a dataset $\mathcal{D}$, f is said distributive if it exists a function g such that $f(\mathcal{D}) = g(f(\mathcal{D}_1), f(\mathcal{D}_2), \ldots, f(\mathcal{D}_N))$ where $\mathcal{D}_i$'s form a partition of $\mathcal{D}$ (e.g., $\mathcal{D} = \cup_i(\sigma(i, \mathcal{D}_i))$ with σ a selection function).*

Definition 8 (Algebraic function)**.** *A function f is said algebraic if f can be computed by a combination of a fixed number of distributive functions (e.g., $mean(\mathcal{D}) = \frac{sum(\mathcal{D})}{count(\mathcal{D})}$).*

For any DEP node computing a distributive or algebraic function, the number of D input tuples exposed to that node can be linearly reduced by augmenting the number of D_i partitions in the same proportion. This general principle, called DEP *reshaping*, splits distributive/algebraic tasks allocated to a single participant into several tasks allocated to different nodes, each working on a partition of the initial input.

Definition 9 (rf-reshaping)**.** *Given an attribution function $at : V \rightarrow \{1, \ldots, \mathrm{rf}\}$ associating vertices to integers uniformly, a distributed execution plan $DEP'(V', E')$ is obtained by* rf-reshaping *from $DEP(V, E)$ such that: $V' \supseteq V$ and $\forall v_i = (a_i, op_i) \in V / distrib\_algebra(op_i) = true \Rightarrow v'_{i,j} = (a_{i,j}, op_i) \in V'$ with $j : 1..rf$, and $v'_{i,j} \in Ant(v_i)$ in E' and $\forall v \in Ant(v_i)$ in E, $v \in Ant(v'_{i,})$ with $j = at(v)$ in E'.*

Definition 9 is illustrated on Figure 3.5, showing a DEP with 6 additional computation nodes obtained by *3-reshaping* from an initial DEP with only 2 computation nodes. Note that

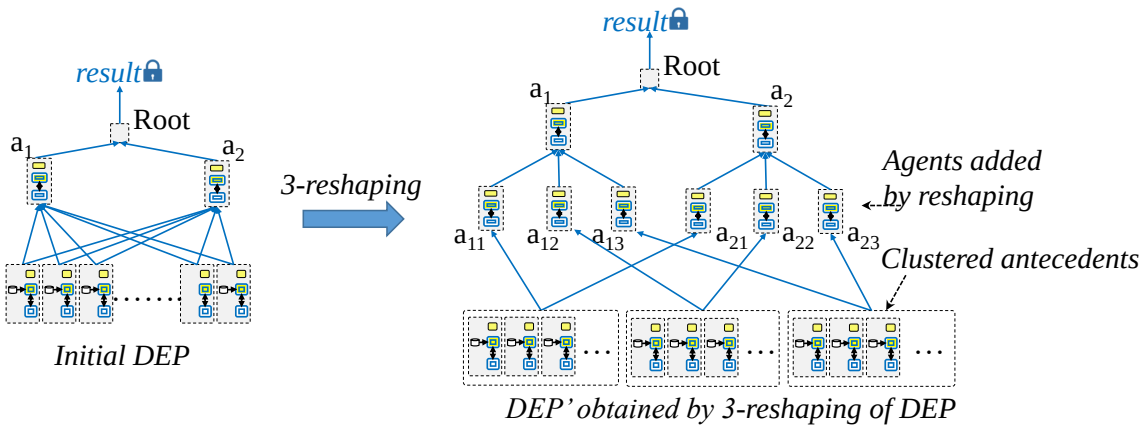


Figure 3.5: 3-reshaping of DEP with m=2 distributive computation nodes

the communication overhead is very small, as only one additional message from each added reshaped node to the original node is produced compared to the original execution. In the remainder, we call a cluster all vertices attributed to the same reshaped node (i.e. $at^{-1}(j)$).

All other things being equal, *rf-reshaping* drastically reduces the data exposure at each computing node. Indeed, for any distributive or algebraic vertex v_i in DEP, *rf-reshaping* divides the probability of gaining access to the entire input D of v_i by a factor $\left(\frac{n-rf}{c-rf}\right)\prod_{i=1..rf}\frac{(n-i)}{(c-i)}$.

The last issue is showing that *rf-reshaping* does not hurt the independence between the processed data and the dataflow as specified in the initial DEP. Recall that a communication flow E is said *data independent* if the DEP is such that personal data cannot be inferred from observing the communication pattern among participants. E can be *data independent* by construction (e.g., broadcast-based algorithm) or be deliberately made *data independent* for privacy concern (e.g., sending fake data among participants to normalize the communications). It is thus mandatory to preserve this independence.

Lemma 2. *If the communication flow E of a distributed execution plan $DEP(V,E)$ is* data independent*, the communication flow E' of any $DEP'(V',E')$ obtained by* rf-reshaping *of $DEP(V,E)$ is also* data independent.

The result is ensured by the fact that the communication flow in the DEP' only depends on the communication pattern in DEP and the at function in Definition 9, which in turn only depends on vertices identifiers and not on data. Hence, the communication flow E' reveals nothing more about the transmitted data compared to E.

The *rf-reshaping* principle can be applied in many practical examples of computations over distributed PDMSs, ranging from simple statistical queries to big data analysis, as illustrated in the Section 3.5. The *rf-reshaping* process can be automatically performed by a precompiler taking as input a logical manifest LM and producing a transformed logical manifest LM_{minExp} minimizing data exposure for the participants for each node computing distributive or algebraic functions. The degree of distribution impacts the performance and the protection of raw data in case of successful attacks (see Section 3.5), but selecting the optimal strategy and integrating it into a precompiler is let for future work.

3.5 Validation

This section validates the effectiveness of the approach in terms of security and performance and assesses whether it is practical by considering two different use-cases representative of distributed processing over personal data. We first describe our experimental setting (Sections 3.5.1), then evaluate the impact of our approach in terms of privacy preservation (Section 3.5.2), and finally study the performance of the solution (Section 3.5.3).

3.5.1 Experimental Setting

Platform and implementation

The platform for the experiments is composed of 8 SGX capable machines with Intel I5-7200U 64 bit clocked at 2.5GHz and with 2 cores, 4 threads and 16GB RAM, equipped with Intel SGX SDK 2.3.101 over Ubuntu 16.04. The manifest framework can be launched at a small scale (up to 100 mappers and corresponding reducers) using these SGX enabled machines with a guarantee that the mappers and reducers are running on different machines in order to get realistic results for remote attestations and data exchanges. A dedicated machine plays the role of the querier and implements a server storing the public manifest store. We also used the implementation to calibrate an analytical model. The accuracy of the model was checked on small scale experiments and used to conduct simulation for large scale experiments (thousands of participants).

Implemented use-cases

We implemented the manifest framework and run it for the two use-cases :

Group-by aggregation We consider a MapReduce-like implementation of an aggregate with a group by query run over distributed PDMSs acting as mappers. The processing is as follows:

- Each mapper sends a couple $(h(group\_key), value)$ to a reducer where h is a hash function which projects the group key on a given reducer.
- Each reducer computes the aggregate function over the values received for the group keys it manages.

If the aggregate function is distributive (e.g., count, min, max, sum, rank, etc.) or algebraic (e.g., avg, var, etc.), *rf-reshaping* is applied to all reducer nodes. In this case, sub-reducer nodes contribute to the computation of the function for a subset of a grouping value and the initial reducers combine their work.

K-means clustering k-means clustering can be similarly computed over distributed PDMSs:

- k initial means representing the centroid of k clusters are randomly generated by the querier and sent to all participants to initialize the processing.
- Each participant playing a mapper role computes its distance with these k means and sends back its data to the reducer node managing the closest cluster.

- Each reducer recomputes the new centroid of the cluster it manages based on the data received from the mappers and sends it back to all participants.

Steps 2 and 3 are repeated a given number of times or until convergence. The function computed by step 3 is algebraic since the centroid of a cluster c_i can be computed thanks to sums and counts computed over all sub-clusters of c_i. Hence, the number of reducers in step 3 can also be arbitrarily augmented by rf-reshaping, such that each of the k initial reducers is preceded in DEP by a set of sub-reducers computing a partial centroid.

Datasets

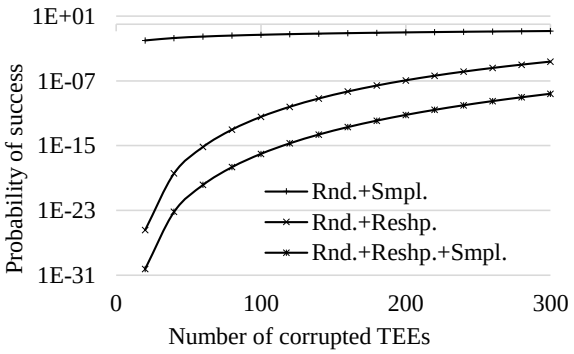
We used synthetic datasets with uniform or zipfian distributions. Note that the challenge of the manifest framework is not on studying the peak performance of the protocol for very specific data distributions, saving seconds or minutes when performing a study over thousands of participants being of little interest (manual surveys usually take weeks). The objective is more on assessing the pertinence of the approach, i.e., guaranteeing that the building and execution phases of a manifest can be run in a time minimizing the risk of failure of participants, this time being primarily linked to the number of participants and to the shape of the execution plan, and assessing the security counter-measures on both uniform and biased (zipfian) dataset distributions.

3.5.2 Security Evaluation

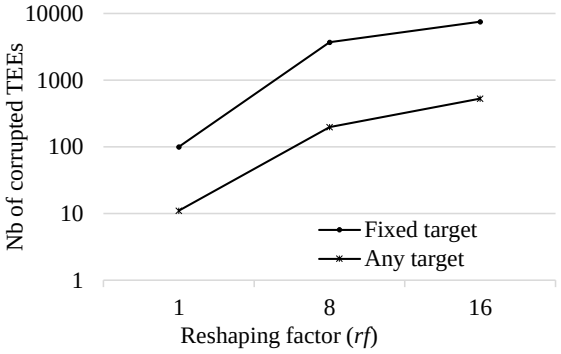
This section evaluates the effectiveness of the three security counter-measures introduced in Section 3.4, namely *randomness*, *sampling* and *rf-reshaping*. The results are obtained by simulation, with and without *randomness*, by varying the sampling rate σ and the reshaping factor *rf* (when *rf-reshaping* is applied) and assuming a number c of corrupted participants (i.e., instrumented TEEs) involved in the computation. We consider two types of attackers. An attacker with a fixed target which aims at gaining access to the tuples corresponding to a given reducer (or to the related sub-reducers) who processes a given grouping value of interest in the group by case, or to the tuples close to a given mean value for k-means. An attacker with any target, who considers that any tuple (processed by any reducer or sub-reducer) is of interest. Each execution is repeated 100 times to obtain significant average values. Note that the results obtained for *k-means* are similar to those obtained for the *group-by*.

Figure 3.6a shows the probability of a successful attack, i.e., resulting in the leakage of at least 10% of the input tuples (i.e., the volume of tuples processed in the average by one initial reducer), in function of the number of corrupted TEEs and combining randomness, sampling and rf-reshaping. Without counter-measure, the attacker impersonates any initial reducer with a probability of 1. With randomness, this probability decreases to 10% with 100 corrupted nodes, as the attacker now impersonates a random participant. Sampling is of little help (probability in 1-10%) unless a very large set of additional participants is used. With reshaping, the probability falls very close to zero.

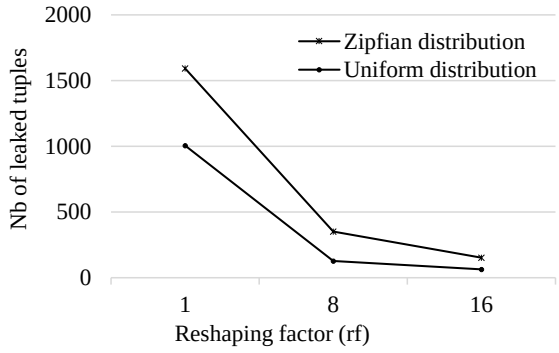
Figure 3.6b plots the impact of using *rf-reshaping* on the Cost of an attack, expressed in terms of number of corrupted TEEs to be injected to have a probability of successful attack of *1%*. The results show that the impact of *reshaping on Cost* is considerable: without reshaping, only *11* corrupted nodes are needed to reach a 1% probability of leaking *10%* random tuples (*any target*) and *100* corrupted nodes to leak all the tuples managed by a specific reducer (*fixed target*). With *16*-reshaping, this number increases to more than *500* corrupted TEEs



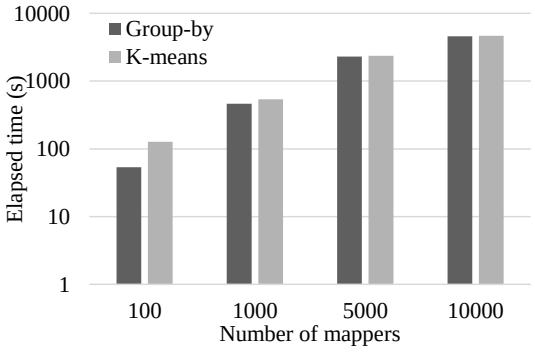
(a) Combining counter-measures



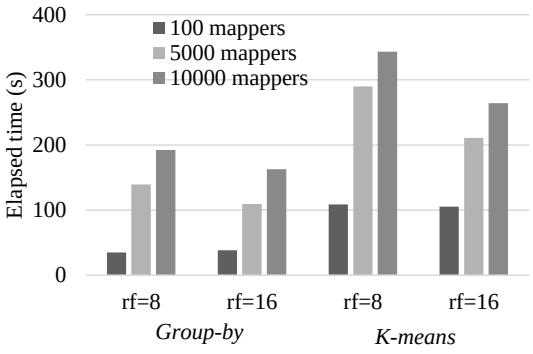
(b) Impact of rf-reshaping on Cost



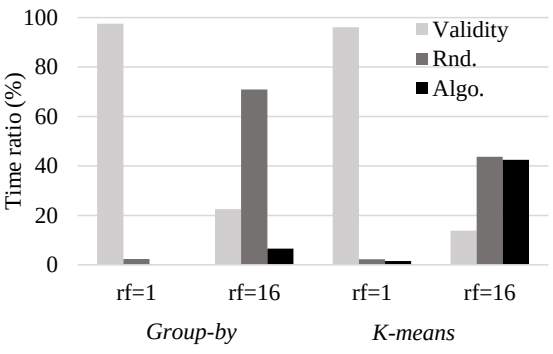
(c) Impact of rf-reshaping on Benefit



(d) Performance without rf-reshaping



(e) Performance with rf-reshaping



(f) Time ratio with rf-reshaping

Figure 3.6: Security and performance evaluation.

(*any* target) and more than *5000* (*fixed* target), far beyond our assumption of a small number of corrupted TEEs.

Figure 3.6c shows that the impact of reshaping on the Benefit, expressed in number of leaked tuples by a successful attack, is also tremendous: the average number of tuples leaked with 100 corrupted nodes is about 1000 tuples without reshaping down to 69 tuples with 16-reshaping and a uniform data distribution (any initial reducer processes about 10% of the input tuples). Gains are even better considering a biased distribution (Zipf, z=1.5). The conclusion is that *rf-reshaping* has a very deterrent impact on an attacker by acting on both factors of the Benefit/Cost ratio, even for low values of *rf*.

3.5.3 Performance evaluation

We introduce an analytical cost model for the *group-by* and *k-means* use-cases and calibrate this model with real measurements using our manifest framework implementation on SGX-enabled machines. We then provide the results of the experiments which combine real measurements (for small scale setting, up to *100* mappers) and simulation (for larger scale setting).

The elapsed time, that is the time spent between the query submission and the production of its result, highly depends on the shape of the *DEP* to be evaluated, which itself depends on the number N of involved participants, the *reshaping* factor *rf* counter-measure, and the sampling rate σ to be applied. The time for *k-means* differs from *group-by* in the number of iterations.

Figures 3.6d present the total execution time for *k-means* and *group-by* for different values of contributors/mappers (up to $N = 10000$) and $k = 50$ reducers. These first measurements do not consider *sampling* (i.e., $\sigma = 1$) nor *rf-reshaping*. Figures 3.6e show the execution times for the same settings, but applying different values of *rf-reshaping*. Figures 3.6f shows the ratio of the counter-measures compared to the total execution time.

Several conclusions can be drawn from these measures. First, the overall time can be considered rather high (tens of minutes for large numbers of participants). But this is not a critical issue in our context from a querier perspective, although it may increase the risk of failures or disconnections of participants. Second, most of the cost for the configuration without counter-measures is incurred by the local checks ensuring the *validity* of the execution, because remote attestations must be performed between each reducer and all the mappers. This incurs a linear increase of the total time with the number of participants, an unusual behavior for a MapReduce process. The main learning is that our manifest protocol adapts badly to processing scenarios which privilege a high data exposure, a sort of natural self-defense. Note finally that *k-means* and *group-by* show a similar global behavior, while *k-means* iterates on the execution plan. This is explained by the dominant cost of establishing all remote attestations, which only occurs at initialization. Once this done, remote attestations are not renewed at each iteration, the new centroid values generated by top reducers being broadcasted to the mappers by the initial way back (i.e. each reducer sends the new centroid to the mappers, the DEP being cyclic). Third, the impact of introducing *rf-reshaping* is tremendous with a global time reduced by more than one order of magnitude. This is due to the decrease in the number of remote attestations to be performed between mappers and reducers (see Section 3.4.3). This confirms that virtuous executions in terms of data exposure are also the less costly to execute in our context.

Section 3.5.3 has highlighted the strong effectiveness of the *randomness*, *sampling* and *rf-reshaping* counter-measures in terms of security (notably considering $\sigma = 0.5$ for *sampling* and

$rf = 16$ for *rf-reshaping* for the use-case of interest). This section shows that the overhead incurred by these counter-measures remains highly tractable in terms of computation time, considering the same values of σ and rf. Moreover, *rf-reshaping* has a dramatic positive impact on the performance, reducing the total execution time by an order similar to the factor rf.

3.6 Conclusion

While everyone was considering the battle for privacy as lost, smart disclosure initiatives and new regulations (e.g., GDPR) generated a new hope and pushed for the adoption of Personal Data Management Systems (PDMS) managed under individual's control. However, a strict personal usage of PDMS is of little interest, the richness of this new paradigm being on the capability to cross personal data of multiple individuals (e.g., perform economic, epidemiological or sociological studies, optimize global resources management...). However, without appropriate security measures, the risk is high to see individuals refuse their contribution, or worse accept it by negligence or ignorance with unprecedented privacy risks considering the broadness of the PDMS content.

This issue is particularly difficult to tackle and only fragmented and highly specific solutions have emerged so far. However, the current generalization of Trusted Execution Environment (TEE) at the edge of the network changes the game. This work capitalizes on this trend and proposes a generic secure decentralized computing framework where each participant gains the assurance that his data is used for the purpose he consents to and that only the final result is disclosed. Conversely, it provides the querier with the guarantee that this result has been honestly computed, by the expected code, on the expected data. We have shown the practicality of the solution both in terms of privacy protection and performance. We hope that this work will lay the groundwork for thinking differently about decentralized computing on personal data and will contribute to a wider usage of the PDMS paradigm.

Chapter 4

Communication Anonymization

Contents

The distributed query execution plans defined in our Manifest-based framework (see Definition 2 in Chapter 3) involve communications between nodes. Communications may depend on the values of the personal data being processed (for example, a given computing node aggregates personal data corresponding to a range of sensitive values). An attacker observing communications, even encrypted, between sources (consenting participants) and computing nodes, can potentially deduce personal data.

Data dependent communications must therefore be protected to avoid at runtime any leakage of personal data resulting from network monitoring. Resorting to traditional solutions for anonymizing communications would lead to a significant penalties in terms of performance, with privacy gains difficult to quantify formally.

This chapter proposes a preliminary solution to control Data dependent communications in distributed execution plans, adapted primarily to the manifest context. This solution offers formal guarantees while balancing data protection and performance. This ongoing work is conducted in collaboration with the Inria-Magnet team (Aurélien Bellet) since September

2020, with the objective to propose a solution adapted not only to the manifest context, but also to distributed queries in private database federations [21] and to secure Big Data processing (Map Reduce like) on a cloud infrastructure based on Intel SGX nodes [100].

4.1 Context

The authors of [100] show that by observing the communications between nodes occurring during the execution of a distributed map-reduce query, an attacker is able to deduce precise personal information. For example, in the computation of a group by query where each reducer node is in charge of a specific grouping key, by observing the network, an attacker can deduce that each mapper communicating with that reducer has this specific group key. More generally, distributed computations involving intensive data exchanges between computing nodes induce communication flows depending on data values (e.g. distributed computations of k-means, group by, join, etc.) for an attacker monitoring the network.

To overcome this issue in the context of the Manifest framework (see previous chapter), we need to control the dependency of communication patterns to data values in distributed query execution plans (DEP, see Definition 2).

A simple way to avoid such dependency would be to "cover" data-dependent communications with data-independent communications (broadcast-based solution).

In broadcast-based solutions, each time a set $\{m\}$ of messages is to be transmitted from a source node s to a given target node t_i, a batch of dummy messages is also transmitted to the set of potential target nodes $\{t\}$, containing t_i, so that any node of $\{t\}$ receives a batch of messages of equal size and the batch of messages received by t_i contains $\{m\}$ (see for example [100]). These solutions perfectly conceal the dependence of communications to data for an attacker monitoring network traffic, but they significantly increase network load and the number of secure communication channels to be established between nodes, making them impractical in our context.

Another solution is to use anonymous communication techniques, such as onion routing (e.g., TOR) and mixed networks [49]. However, these techniques cause significant network overheads [129] and make assumptions about network traffic with unclear privacy guarantees [44].

This chapter proposes solutions to meet a four-fold objective: (i) to provide formal privacy guarantees (differential privacy [51]) for the communication patterns in a distributed query execution plan (DEP), (ii) to make a generic proposal without any assumptions about the type of computation (for example, not limiting oneself to map-reduce like DEPs [100]), (iii) to work with real data, which means not adding noise to the data, and (iv) to preserve representative data samples, which means never discriminating messages according to the source nodes or targets chosen (discarded contributions should only be selected randomly and uniformly) to avoid producing biased results. Our goal is therefore to design a distributed algorithm to hide the communication patterns achieving these objectives, and to integrate this algorithm in the DEPs defined in the manifests.

4.2 Problem formulation and notations

We consider a set of source nodes $\mathcal{S} = \{s_i\}$ communicating with a set of target nodes $\mathcal{T} = \{t_i\}$. The messages sent between the sources and the targets are encrypted. The attackers observing

the network are able to observe which specific source is communicating with which specific target (hence deducing personal data), but cannot access the content of the communication (the transmitted message) nor observe the final result of the computation. Any message is considered indistinguishable from any other message to the attackers.

Our goal is to design a distributed algorithm $\mathcal{A}$ taking as input a communication data set $\mathcal{D} = \{< s, m, t >\}$[1] where a node $s \in \mathcal{S}$ must send a given message m to a target $t \in \mathcal{T}$, and producing in output a communication data set $\mathcal{O}$. The algorithm $\mathcal{A}$ provides ϵ-differential privacy guarantees on the communication patterns, if and only if for two neighboring input communication data sets $\mathcal{D}$ and $\mathcal{D}'$ which differ on a single target node $< s, m, t > \in \mathcal{D}$ and $< s, m, t' > \in \mathcal{D}'$ with $t \neq t'$, the probability ratio for $\mathcal{A}$ to produce any output communication data set $\mathcal{O}$, with a $\mathcal{D}$ versus $\mathcal{D}'$ as input, is lower than e^{ϵ}, i.e.:

$$\frac{P[\mathcal{A}(\mathcal{D}) \in \mathcal{O}]}{P[\mathcal{A}(\mathcal{D}') \in \mathcal{O})]} \leq e^{\epsilon}$$

In practice, we accept that in very rare cases (for extreme input and output sets), which occur with a very low probability, $\mathcal{A}$ will not meet the ϵ-differential privacy guarantee. In this case, $\mathcal{A}$ offers a (ϵ, δ) differential privacy guarantee defined as follows:

$$P\left(\frac{P[\mathcal{A}(\mathcal{D}) \in \mathcal{O}]}{P[\mathcal{A}(\mathcal{D}') \in \mathcal{O})]} \geq e^{\epsilon}\right) \leq \delta$$

The proposed solution must be generic, i.e. applicable to any DEP defined in the Manifest. Furthermore, it must not affect the integrity or accuracy of the final result. Therefore, we consider that (i) a representative (uniformly random) subset of σD messages should be transmitted to the target nodes, with σ a maximum sampling rate defined in the Manifest and (ii) the values of the messages exchanged should not be modified.

The proposed solution should enable to efficiently balance privacy, security and performance. We evaluate these three dimensions as follows:

- *Privacy.* It will be quantified by ϵ and δ. Smaller are ϵ and δ, better is the privacy of the communications.
- *Performance.* The performance (and feasibility) of the proposal is linked (i) to the number of secure channels (and therefore remote attestations, see previous chapters) to be established to guarantee the integrity of the execution and (ii) to the number and volume of message exchanges generated by $\mathcal{O}$.
- *Security.* The security of $\mathcal{A}$, is related to the security of the nodes used for the evaluation of $\mathcal{A}$. Any execution that would rely on a single trusted (central) node to run $\mathcal{A}$ can be considered less secure, since corruption of that single node would breach the privacy of all communications. On the contrary, relying on a larger set of independent secure nodes executing $\mathcal{A}$ would decentralize its evaluation, so that the corruption of a subset of the nodes would reveal only a subset of communications.

[1] In the remaining of the chapter, the source s may be omitted when the communication algorithm only takes into account messages sent from the same node. Similarly, since messages m are indistinguishable to attackers, they may also be omitted.

4.3 Local differential privacy to protect communications

Two main approaches coexist to design algorithms ensuring differential privacy when executing database queries. First, the central model [51], in which a trusted third party collects all (sensitive) input tuples $\{t\}$, evaluates the Q query on this set of tuples and adds noise to produce the (approximated) result $R = noise(Q(\{t\}))$ with differential confidentiality guarantees (i.e. neighboring input data sets cannot be distinguished from the observed results) and preserving utility (i.e. R is close to $Q(\{t\})$. In our context, this approach would lead to the introduction of an additional node in the DEP responsible for collecting all messages that would harm our security objective (see previous section).

Second, the local model [50] is well suited to highly distributed settings where the assumption of a central trusted third party does not hold. In the local model, the data sources hold each a given data value v_i, they locally add noise to that data value to obtain a non sensitive value $noise(v_i)$, before transmitting it to a central (untrusted) server which evaluates the (approximate) query result $R = Q(\{noise(v_i)\})$ on the set of collected (noisy) values which should be close to $Q(\{v_i\})$. Given the security objective to be achieved in our context, we focus in this section on the adaptation of the local model.

Differential privacy applies to (numerical) data values, but has not yet (to our knowledge) been applied to protect communications. The addition of noise to protect data values in the local model (see for example the implementation of "randomized-response" [132] adopted by Google and Apple) is based on two main principles: (i) real data values are exposed with probability of $\bar{\sigma} = 1 - \sigma$, and (ii) false data values are forged with probability of σ. In our context, where the objective is to protect communications rather than numerical data values, the first principle consists in introducing *sampling*, i.e. allowing the communication algorithm $\mathcal{A}$ to only transmit certain messages $< s, t >$ (other being removed) with a sampling rate σ as defined in the Manifest (see 3.4.2) so that a sufficient number of contributions are taken into account at query execution time. The second principle corresponds to *flooding* the communication flow, allowing $\mathcal{A}$ to produce dummy messages in output, indistinguishable from the real messages to the attackers, but which will then be discarded by the target nodes at execution time to avoid altering the final result.

4.3.1 Impact of Sampling on Privacy and Performance

We consider a communication algorithm $\mathcal{A}_\sigma$ based on the sampling principle with a sampling rate σ, which for each message $< s, t >$ in input, adds with probability of $\bar{\sigma} = 1 - \sigma$ the message to its output $\mathcal{O} = \mathcal{O} \cup < s, t >$ or otherwise (with probability of σ) adds a dummy message to its output $\mathcal{O} = \mathcal{O} \cup < s, t' >$, with $t' \in \mathcal{T}$ a potential target node chosen at random which will discard the message at execution, when received.

Privacy analysis

Let us now analyze to which extent this algorithm satisfies local differential privacy.

Theorem 2. *The sampling algorithm $\mathcal{A}_\sigma$ is ϵ-locally differentially private with $\epsilon = \ln\left(\frac{\bar{\sigma}}{\sigma/T}\right)$*

Proof. Let $< s, t >$ and $< s, t' >$ be two arbitrary inputs of $\mathcal{A}_\sigma$, with $t, t' \in \mathcal{T}$ the set of potential target nodes and $t \neq t'$. Let $\mathcal{O}$ be the output of an actual execution of $\mathcal{A}_\sigma$. We

can distinguish three cases (i) $\mathcal{O} = < s, t >$, (ii) $\mathcal{O} = < s, t' >$ and (iii) $\mathcal{O} = < s, t_i >$ where $t_i \in \mathcal{T} \setminus \{t, t'\}$.

The probability that the last output comes from one of the two inputs is the same, their ratio is then 1. The two first outputs are symmetric, let us consider, without loss of generality, that $\mathcal{O} = < s, t >$. The probability that the output comes from the first input is $\bar{\sigma}$ while the probability that it comes from the second input is $\frac{\sigma}{T}$, with $T = |\mathcal{T}|$. We have then :

$$\frac{P[\mathcal{A}_\sigma((s,t)) = \mathcal{O}]}{P[\mathcal{A}_\sigma((s,t')) = \mathcal{O}]} = \frac{\bar{\sigma}}{\sigma/T} = e^\epsilon$$

□

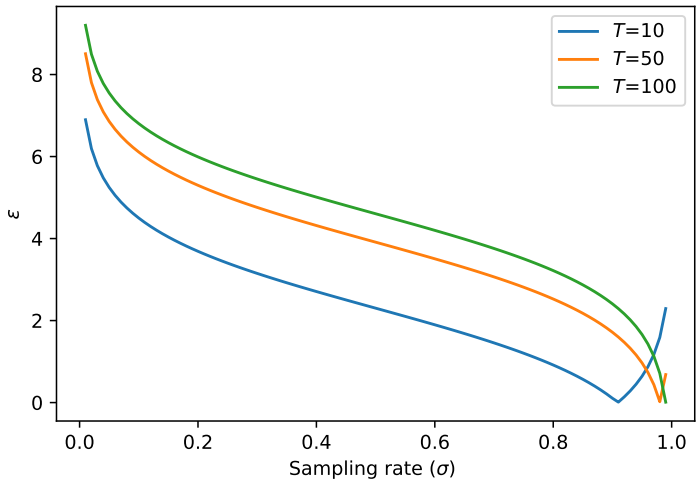


Figure 4.1: Privacy of $\mathcal{A}_\sigma$ depending on the sampling rate for different numbers of targets.

Figure 4.1 plots the evolution of ϵ for different values of T as σ increases. The figure shows that to obtain reasonable values for ϵ[2] a high value of σ is needed (≥ 0.9 in general). This implies that $\mathcal{A}_\sigma$, only based on the sampling principle, cannot be applied in practice (where a lower sampling rate is considered).

Performance analysis

We assume a simple DEP with $S = |\mathcal{S}|$ source nodes delivering a single message to one of the $T = |\mathcal{T}|$ potential target nodes. In order to keep the same number of real contributions, the total number of sources is $S_t = \frac{S}{\bar{\sigma}}$. If $\mathcal{A}_\sigma$ is applied in each source node, the total number of secure communication channels to be initiated is hence S_t and the total volume of exchanged messages is $S_t \times mu$, with mu the size of a single message.

4.3.2 Impact of Flooding Combined with Sampling

In this sub-section we consider an algorithm $\mathcal{A}_\sigma^d$ combining sampling with flooding principles. Flooding leads to adding d extra dummy messages to the output of $\mathcal{A}_\sigma$ described above. For

[2]Usually ϵ is recommended to be smaller then $ln(3)$ [52]

a given input tuple $< s, t >$, the result of $\mathcal{A}_\sigma^d$ is $\mathcal{O} = \mathcal{A}_\sigma(< s, t >) \cup \{< s, t_1 >, \ldots, < s, t_d >\}$ with t_i different targets randomly selected in $\mathcal{T}$.

Privacy analysis

Let us now analyze to which extent this algorithm satisfies local differential privacy.

Theorem 3. *Algorithm $\mathcal{A}_\sigma^d$ is ϵ-locally differentially private with* $\epsilon = \ln\left(\frac{\bar{\sigma}(T-d)(T-1)}{\sigma(d+1)(T-d-1)} + 1\right)$

Proof. Let $< s, t >$ and $< s, t' >$ be two arbitrary inputs of $\mathcal{A}_\sigma^d$, with $t \neq t'$. We can distinguish four cases for the output $\mathcal{O}$ produced by $\mathcal{A}_\sigma^d$ (i) $< s, t >$ and $< s, t' > \in \mathcal{O}$, (ii) $< s, t >$ and $< s, t' > \notin \mathcal{O}$, (iii) $< s, t > \in \mathcal{O}$ and $< s, t' > \notin \mathcal{O}$ and (iv) $< s, t > \notin \mathcal{O}$ and $< s, t' > \in \mathcal{O}$.

The probability that the first two outputs are produced by one of the two inputs are the same, their ratio is then 1. The two last outputs are symmetric, let us consider without loss of generality that $< s, t > \in \mathcal{O}$ and $< s, t' > \notin \mathcal{O}$. We have:

$$\frac{P[\mathcal{A}_\sigma^d(< s, t >) = \mathcal{O}]}{P[\mathcal{A}_\sigma^d(< s, t' >) = \mathcal{O}]} = \frac{\bar{\sigma}\left(\frac{T-d}{T}\right) + \sigma(d+1)\left(\frac{T-(d+1)}{T(T-1)}\right)}{\sigma(d+1)\left(\frac{T-(d+1)}{T(T-1)}\right)}$$

Hence for any $s \in \mathcal{S}$, $t, t' \in \mathcal{T}$ and subset of outputs $\mathcal{O}$:

$$\frac{P[\mathcal{A}_\sigma^d(< s, t >) = \mathcal{O}]}{P[\mathcal{A}_\sigma^d(< s, t' >) = \mathcal{O}]} = \frac{\bar{\sigma}(T-d)(T-1)}{\sigma(d+1)(T-d-1)} + 1 \leq e^\epsilon$$

□

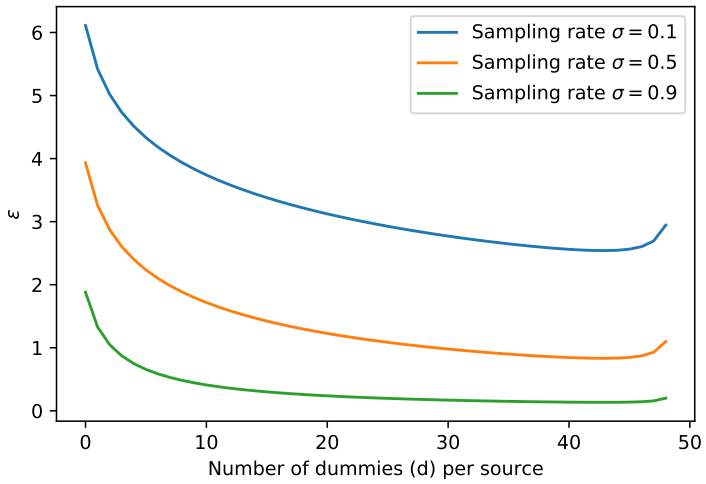


Figure 4.2: Privacy of $\mathcal{A}_\sigma^d$ when increasing dummy messages for different sampling rates.

Fig. 4.2 shows the evolution of ϵ for different number of dummies, and fixed parameters $S = 10000$, $T = 50$. The bigger σ is, the faster the ϵ decreases.

Performances analysis

We assume the same setting as above. If $\mathcal{A}_\sigma^d$ is applied in each source node, $d+1$ messages are sent by each source to $d+1$ targets. The total number of secure communication channels to be initiated is hence $S_t \times (d+1)$ and the total volume of exchanged messages is $S_t \times (d+1) \times mu$, with mu the size of a single message. $\mathcal{A}_\sigma^d$ introduces an overhead of $(d+1)$ times more than a regular execution.

Summary of conclusions The results show that the sampling and flooding algorithm provides sufficient confidentiality for high sampling rates. However, $\mathcal{A}_\sigma^d$ introduces many dummy messages when the expected sampling rate is low, with significant performance penalties.

4.4 Privacy amplification through scrambling

In the above solution, the $\mathcal{A}_\sigma^d$ algorithm is implemented in each source node of the DEP. Communication privacy is ensured by adding dummy messages by sampling and flooding, but any extra dummy message increases communication privacy for only one source node. The idea we develop in this section is to extend the protection offered by dummy messages to a set of source nodes. Therefore, we introduce in the DEP an additional node called *scrambler*, running a new algorithm $\mathcal{A}_{n,d}$, whose role is to collect a set of n messages (from n source nodes), select (a sample of) the messages and add d extra dummy messages, to be transmitted to the target nodes.

The scrambling nodes operate in trusted enclaves, offering the same security properties as the other nodes. If a scrambler is corrupted, confidentiality is no longer ensured (the node operates in sealed glass protection mode, see our specific threat model in chapter 3). Providing a single scrambling node for an entire DEP, if attacked, would result in the disclosure of all DEP communications, with massive leaks of personal data. We therefore consider a set of SF scrambling nodes, where SF, the scrambling factor, is defined as the ratio between the number n of messages processed by each scrambler and the total number of source nodes. Therefore, the corruption of a subset of scramblers never gives access to the full pattern of communications. In addition, messages processed by the scrambler are encrypted by the source node and can only be decrypted by a single target node.

4.4.1 Proposed algorithm

The pseudo-code of the algorithm $\mathcal{A}_{n,d}$ is given in Algorithm 3. The noise addition is done as follows. First, the scrambler collects n inputs from the sources. Then, each individual input is either added to the output batch with probability $\bar{\sigma}$, or replaced with a probability σ by a dummy message ω sent to a randomly selected target. Second, the scrambler adds to the output batch d dummy messages to be sent to random potential targets. Finally, the scrambler shuffles the output batch and sends the messages to the corresponding targets.

4.4.2 Performances analysis

We assume a simple DEP with S source nodes, T target nodes and $SF = S_t/n$ scramblers. Each scrambler opens a secure communication channel with n sources and T targets and

Algorithm 3: Noise addition (run by the scrambler)

Input: σ probability of lying, $\mathcal{T}$ list of possible targets, $\mathcal{B} = \{(m_i, t_i)\}$ a set of n tuples to send, d number of dummies.

Output: $\mathcal{O} = \{(m_j, t_j)\}$ where m_j can be either a real message or a dummy message and $t_j \in \mathcal{T}$

```
foreach (m, t) in B do
    rnd ← random(0, 1)
    if rnd < σ then
        t_tmp ← choose_random(T)
        O.add((ω, t_tmp))
    else
        O.add((m, t))
    end
end
for i = 0 to d do
    t_tmp ← choose_random(T)
    O.add((ω, t_tmp))
end
O.shuffle()
return O
```

exchanges $2 \times n + d$ messages. Hence the total number of secure channels is $S_t + SF \times T$ and the volume of exchanged messages is $S_t + SF \times (n + d) \times mu$.

4.4.3 Privacy analysis

In this section we evaluate the level of privacy provided by the algorithm $\mathcal{A}_{n,d}$.

Additional notations: To facilitate the reading of the remaining of the chapter, we introduce additional notations. The dataset $\mathcal{D} = \{< s_i, t_i >\}$ can be abstracted as $\mathcal{D} = \{x, y, z\}$ where x is the number of messages targeting t, z is the number of messages targeting t' and y is the number of messages targeting $t_i \in \mathcal{T} \setminus \{t, t'\}$ with $t \neq t'$. Following the same principle we denote the output as $\mathcal{O} = \{\alpha, \beta, \gamma\}$ where α is the number of messages the scrambler sends to the target t, γ is the number of messages are sent to the target t' and β is the number of messages are sent to $t_i \in \mathcal{T} \setminus \{t, t'\}$. The other used notations are summarized below:

- $\mathbb{P}_{n,d}\binom{\alpha}{\beta}{\gamma}\binom{x}{y}{z}$ is the probability to get an output $\mathcal{O} = \{\alpha, \beta, \gamma\}$ with the algorithm $\mathcal{A}_{n,d}$ given an input dataset $\mathcal{D} = \{x, y, z\}$
- $R_n^d\binom{\alpha}{\beta}{\gamma}\binom{x}{y}{z}$ is the ratio $\dfrac{\mathbb{P}_{n,d}\left(\begin{smallmatrix}\alpha\\ \beta\\ \gamma\end{smallmatrix}\middle|\begin{smallmatrix}x+1\\ y\\ z\end{smallmatrix}\right)}{\mathbb{P}_{n,d}\left(\begin{smallmatrix}\alpha\\ \beta\\ \gamma\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z+1\end{smallmatrix}\right)}$ and it is equal to e^ϵ
- $\mathbb{P}^{\text{dum}}_{n,d}\left(\begin{smallmatrix}k_1\\ k_2\\ k_3\end{smallmatrix}\middle|\begin{smallmatrix}\alpha - k_1\\ \beta - k2\\ \gamma - k_3\end{smallmatrix}\right)$ is the probability to send k_1 dummies (resp. k_2, k_3) to the target t (resp. $t_i \in \mathcal{T} \setminus \{t, t'\}$, t').

- $\Phi_{u,v}$ = is the probability to draw u times α, v times γ and $x+z-u-v$ times β (i.e $\mathbb{P}_{n,0}\binom{x+z-u-v}{z}^{u}_{v}$).
- $L(conditions)$ is a function equal to 1 when the conditions are met, 0 otherwise.

Useful formulas: based on the notations above, we provide below some useful formulas used in the remaining of the chapter.

$$\mathbb{P}_{n,d}\binom{\alpha}{\beta}^{x}_{y}_{\gamma}_{z} = \sum_{k_1+k_2=d} \mathbb{P}^{\text{dum}}_{n,d}\binom{k_1}{k_2}^{\alpha-k_1}_{\beta-(d-k_1-k_2)}_{k_2}_{\gamma-k_2} \cdot \mathbb{P}_{n,0}\binom{\alpha-k_1}{\beta-(d-k_1-k_2)}^{x}_{y}_{\gamma-k_2}_{z} \tag{4.1}$$

$$\mathbb{P}_{n,0}\binom{\alpha}{\beta}^{0}_{y}_{\gamma}_{0} = \frac{y!\bar{\sigma}^{\beta}\left(\frac{\sigma}{T-1}\right)^{y-\beta}}{\alpha!\beta!\gamma!} L\begin{pmatrix}\alpha\geqslant 0\\ \beta\geqslant 0\\ \gamma\geqslant 0\end{pmatrix} \tag{4.2}$$

$$\mathbb{P}_{n,0}\binom{\alpha}{\beta}^{x}_{y}_{\gamma}_{z} = \sum_{u+v\leq x+z} \Phi_{u,v}\mathbb{P}_{n,0}\binom{\alpha-u}{\beta-(x+z-u-v)}^{0}_{y}_{\gamma-v}_{0} \tag{4.3}$$

By replacing formula (4.2) in formula (4.3) we obtain:

$$\mathbb{P}_{n,0}\binom{\alpha}{\beta}^{x}_{y}_{\gamma}_{z} = \sum_{u+v\leq x+z} \Phi_{u,v}\frac{y!\bar{\sigma}^{\beta-v}\left(\frac{\sigma}{T-1}\right)^{y-\beta+v}}{(\alpha-u)!(\beta-(x+z-u-v))!(\gamma-v)!} L\begin{pmatrix}u\leq\alpha\\ x+z-u-v\leq\beta\\ v\leq\gamma\end{pmatrix} \tag{4.4}$$

Theorem 4. *The algorithm $\mathcal{A}_{n,d}$ is ϵ-differentially private with*

$$\epsilon = \ln\left(\frac{\sum_{k=0}^{d} C_d^k C_{n-1}^k \bar{\sigma}^k \left(\frac{\sigma}{T-1}\right)^{n-k-1}\left(\bar{\sigma}+k\cdot\frac{\left(\frac{\sigma}{T-1}\right)^2}{\bar{\sigma}}\right)}{\sum_{k=0}^{d} C_d^k C_{n-1}^k \bar{\sigma}^k \left(\frac{\sigma}{T-1}\right)^{n-k-1}\left(\left(\frac{\sigma}{T-1}\right)+k\cdot\frac{\left(\frac{\sigma}{T-1}\right)^2}{\bar{\sigma}}\right)}\right)$$

Proof. We first need to determine the input and the output producing the higher ratio[3] for two neighboring datasets. In other words we need to find $\mathcal{D}=\{x+1,y,z\}$, $\mathcal{D}'=\{x,y,z+1\}$ and $\mathcal{O}=\{\alpha,\beta,\gamma\}$ such that $R_n^d\binom{\alpha}{\beta}^{x}_{y}_{\gamma}_{z} = \frac{P[\mathcal{A}(\mathcal{D})\in\mathcal{O}]}{P[\mathcal{A}(\mathcal{D}')\in\mathcal{O})]}$ is maximum.

We have:

$$R_n^d\binom{\alpha}{\beta}^{x}_{y}_{\gamma}_{z} = \frac{\mathbb{P}_{n,d}\binom{\alpha}{\beta}^{x+1}_{y}_{\gamma}_{z}}{\mathbb{P}_{n,d}\binom{\alpha}{\beta}^{x}_{y}_{\gamma}_{z+1}}$$

We start by developing the numerator:

$$\mathbb{P}_{n,d}\binom{\alpha}{\beta}^{x+1}_{y}_{\gamma}_{z} = \sum_{k_1+k_2=d} \mathbb{P}^{\text{dum}}_{n,d}\binom{k_1}{k_2}^{\alpha-k_1}_{\beta-(d-k_1-k_2)}_{k_2}_{\gamma-k_2} \cdot \mathbb{P}_{n,0}\binom{\alpha-k_1}{\beta-(d-k_1-k_2)}^{x+1}_{y}_{\gamma-k_2}_{z}$$

$$\begin{aligned}\mathbb{P}_{n,d}\binom{\alpha}{\beta}^{x+1}_{y}_{\gamma}_{z} = \sum_{k_1+k_2=d} \mathbb{P}^{\text{dum}}_{n,d}\binom{k_1}{k_2}^{\alpha-k_1}_{\beta-(d-k_1-k_2)}_{k_2}_{\gamma-k_2} \cdot \Bigg(&\bar{\sigma}\mathbb{P}_{n,0}\binom{\alpha-k_1-1}{\beta-(d-k_1-k_2)}^{x}_{y}_{\gamma-k_2}_{z} \\ &+ \left(\frac{\sigma}{T-1}\right)\mathbb{P}_{n,0}\binom{\alpha-k_1}{\beta-(d-k_1-k_2)-1}^{x}_{y}_{\gamma-k_2}_{z} + \left(\frac{\sigma}{T-1}\right)\mathbb{P}_{n,0}\binom{\alpha-k_1}{\beta-(d-k_1-k_2)}^{x}_{y}_{\gamma-k_2-1}_{z}\Bigg)\end{aligned}$$

[3]Ignoring the symmetric case where the ratio is minimum.

We then apply the formula 4.4 to each $\mathbb{P}_{n,0}$:

$$\mathbb{P}_{n,0}\left(\begin{smallmatrix}\alpha-k_1-1\\ \beta-(d-k_1-k_2)\\ \gamma-k_2\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z\end{smallmatrix}\right) = \sum_{u+v\leq x+z} \Phi_{u,v} \frac{y!\bar{\sigma}^{\beta-(d-k_1-k_2)-v}\left(\frac{\sigma}{T-1}\right)^{y-\beta-(d-k_1-k_2)+v}}{(\alpha-k_1-1-u)!(\beta-(d-k_1-k_2)-(x+z-u-v))!(\gamma-k_2-v)!}$$
$$\cdot L\left(\begin{smallmatrix}u\leq \alpha-k_1-1\\ x+z-u-v\leq \beta-(d-k_1-k_2)\\ v\leq \gamma-k_2\end{smallmatrix}\right)$$
$$\mathbb{P}_{n,0}\left(\begin{smallmatrix}\alpha-k_1-1\\ \beta-(d-k_1-k_2)\\ \gamma-k_2\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z\end{smallmatrix}\right) = \sum_{u+v\leq x+z} f\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right)(\alpha-k_1-u)L\left(\begin{smallmatrix}\alpha-k_1-u\geq 1\\ 1\\ 1\end{smallmatrix}\right)$$

Where $f\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right) = \Phi_{u,v}\frac{y!\bar{\sigma}^{\beta-(d-k_1-k_2)-v}\left(\frac{\sigma}{T-1}\right)^{y-\beta-(d-k_1-k_2)+v}}{(\alpha-k_1-u)!(\beta-(d-k_1-k_2)-(x+z-u-v))!(\gamma-k_2-v)!}L\left(\begin{smallmatrix}u\leq \alpha-k_1\\ x+z-u-v\leq \beta-(d-k_1-k_2)\\ v\leq \gamma-k_2\end{smallmatrix}\right)$

In the same way we obtain for the two other $\mathbb{P}_{n,0}$:

$$\mathbb{P}_{n,0}\left(\begin{smallmatrix}\alpha-k_1\\ \beta-(d-k_1-k_2)-1\\ \gamma-k_2\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z\end{smallmatrix}\right) = \sum_{u+v\leq x+z} f\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right)\frac{\left(\frac{\sigma}{T-1}\right)(\beta-(d-k_1-k_2)-(x+z-u-v))}{\bar{\sigma}}$$
$$\cdot L\left(\begin{smallmatrix}(\beta-(d-k_1-k_2)-(x+z-u-v))-1\geq 0\\ 1\end{smallmatrix}\right)$$

$$\mathbb{P}_{n,0}\left(\begin{smallmatrix}\alpha-k_1\\ \beta-(d-k_1-k_2)\\ \gamma-k_2-1\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z\end{smallmatrix}\right) = \sum_{u+v\leq x+z} f\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right)(\gamma-k_2-v)L\left(\begin{smallmatrix}1\\ 1\\ \gamma-k_2-v-1\end{smallmatrix}\right)$$

By replacing in the numerator we obtain:

$$\mathbb{P}_{n,d}\left(\begin{smallmatrix}\alpha\\ \beta\\ \gamma\end{smallmatrix}\middle|\begin{smallmatrix}x+1\\ y\\ z\end{smallmatrix}\right) = \sum_{\substack{k_1+k_2\leq d\\ u+v\leq x+z}} \Omega\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right)\cdot\Bigg(\bar{\sigma}(\alpha-k_1-u)L\left(\begin{smallmatrix}\alpha-k_1-u\geq 1\\ 1\\ 1\end{smallmatrix}\right)$$
$$+\frac{\left(\frac{\sigma}{T-1}\right)^2(\beta-(d-k_1-k_2)-(x+z-u-v))}{\bar{\sigma}}\cdot L\left(\begin{smallmatrix}(\beta-(d-k_1-k_2)-(x+z-u-v))\geq 1\\ 1\end{smallmatrix}\right)$$
$$+\left(\frac{\sigma}{T-1}\right)(\gamma-k_2-v)L\left(\begin{smallmatrix}1\\ 1\\ \gamma-k_2-v\geq 1\end{smallmatrix}\right)\Bigg)$$

Where $\Omega\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right) = \mathbb{P}^{\text{dum}}_{n,d}\left(\begin{smallmatrix}k_1\\ d-k_1-k_2\\ k_2\end{smallmatrix}\middle|\begin{smallmatrix}\alpha-k_1\\ \beta-(d-k_1-k_2)\\ \gamma-k_2\end{smallmatrix}\right) f\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right)$
With the same reasoning for the denominator, we obtain:

$$\mathbb{P}_{n,d}\left(\begin{smallmatrix}\alpha\\ \beta\\ \gamma\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z+1\end{smallmatrix}\right) = \sum_{\substack{k_1+k_2\leq d\\ u+v\leq x+z}} \Omega\left(\begin{smallmatrix}\alpha,k_1\\ \beta,k_2\\ \gamma,k_3\end{smallmatrix}\right)\cdot\Bigg(\left(\frac{\sigma}{T-1}\right)(\alpha-k_1-u)L\left(\begin{smallmatrix}\alpha-k_1-u\geq 1\\ 1\\ 1\end{smallmatrix}\right)$$
$$+\frac{\left(\frac{\sigma}{T-1}\right)^2(\beta-(d-k_1-k_2)-(x+z-u-v))}{\bar{\sigma}}\cdot L\left(\begin{smallmatrix}(\beta-(d-k_1-k_2)-(x+z-u-v))\geq 1\\ 1\end{smallmatrix}\right)$$
$$+\bar{\sigma}(\gamma-k_2-v)L\left(\begin{smallmatrix}1\\ 1\\ \gamma-k_2-v\geq 1\end{smallmatrix}\right)\Bigg)$$

The ratio can then be written as:

$$R_n^d\left(\begin{smallmatrix}\alpha\\ \beta\\ \gamma\end{smallmatrix}\middle|\begin{smallmatrix}x\\ y\\ z\end{smallmatrix}\right) = \frac{\sum\limits_{\substack{k_1+k_2\leq d\\ u+v\leq x+z}} \Omega\left(\begin{smallmatrix}\alpha, k_1\\ \beta, k_2\\ \gamma, k_3\end{smallmatrix}\right)\cdot\left(\bar{\sigma}\chi_1+\chi_2+\left(\frac{\sigma}{T-1}\right)\chi_3\right)}{\sum\limits_{\substack{k_1+k_2\leq d\\ u+v\leq x+z}} \Omega\left(\begin{smallmatrix}\alpha, k_1\\ \beta, k_2\\ \gamma, k_3\end{smallmatrix}\right)\cdot\left(\left(\frac{\sigma}{T-1}\right)\chi_1+\chi_2+\bar{\sigma}\chi_3\right)}$$

Where:

$$\chi_1 = (\alpha - k_1 - u)L\left(\begin{smallmatrix}\alpha-k_1-u\geq 1\\ 1\\ 1\end{smallmatrix}\right)$$

$$\chi_2 = \frac{\left(\frac{\sigma}{T-1}\right)^2(\beta-(d-k_1-k_2)-(x+z-u-v))}{\bar{\sigma}}$$

$$\chi_3 = (\gamma - k_2 - v)L\left(\begin{smallmatrix}1\\ 1\\ \gamma-k_2-v\geq 1\end{smallmatrix}\right)$$

As $\bar{\sigma} >> \left(\frac{\sigma}{T-1}\right)$, to maximize the ratio, one need to maximize χ_1 and minimize χ_3. On the one hand χ_1 is maximal when α is maximal (i.e. $\alpha = n$)[4]. On the other hand, χ_3 is minimal when $\gamma = 0$. Thus, the output producing the higher ratio is $\mathcal{O} = \{n, d, 0\}$.

To further increase the ratio, we need to minimize k_1 and u. The first one depends on d, a fixed parameter of the algorithm we cannot change. The later one, u, is varying from 0 to $x + z$ and takes its minimal value when $x + z = 0$. We deduce from this that the two inputs producing the higher ratio are $\mathcal{D} = \{1, n-1, 0\}$ and $\mathcal{D}' = \{0, n-1, 1\}$.

By replacing the new indices in the ratio we obtain:

$$R_n^d\left(\begin{smallmatrix}n\\ d\\ 0\end{smallmatrix}\middle|\begin{smallmatrix}0\\ n\\ 0\end{smallmatrix}-1\right) = \frac{\sum_{k=0}^{d} C_d^k C_{n-1}^k \bar{\sigma}^k \left(\frac{\sigma}{T-1}\right)^{n-k-1}\left(\bar{\sigma}+k\cdot\frac{\left(\frac{\sigma}{T-1}\right)^2}{\bar{\sigma}}\right)}{\sum_{k=0}^{d} C_d^k C_{n-1}^k \bar{\sigma}^k \left(\frac{\sigma}{T-1}\right)^{n-k-1}\left(\left(\frac{\sigma}{T-1}\right)+k\cdot\frac{\left(\frac{\sigma}{T-1}\right)^2}{\bar{\sigma}}\right)} = e^{\epsilon}$$

□

This result proves that the privacy provided by the scrambler with $\mathcal{A}_{n,d}$ increases with the number of collected values n and with the number of extra dummies d. It clearly shows that this approach benefits from extra dummies introduced by both sampling and flooding for all the underlying source nodes.

Privacy in practice with (ϵ, δ)-differential privacy However, the formula above does not fully reflects privacy benefits, as it only gives theoretical values for ϵ in very specific (worst cases) setting for input and output sets, unlikely to happen in practice. As specified in the problem formulation, in our context, an (ϵ, δ)-differential privacy analysis of our algorithm would avoid this pitfall and provide a much better metric for privacy in practice.

However, it is difficult to provide analytically a value of (ϵ, δ)-differential privacy for our algorithm. Nevertheless, we can provide an upper bound. To do so, we rely on recent work [18]

[4]Note that the total number of messages that a scrambler sends to the same target is capped to the number of messages that it received from the sources (i.e. n), as sending more would not improve the privacy.

which studies differential privacy in another context. The objective of the authors of [18] is to demonstrate an amplification of the (ϵ, δ)-differential privacy bounds provided by any existing local differential privacy algorithm (such as the "randomized response" algorithm [132]), by re-centralizing n values produced locally by such algorithm and shuffling these n values in a secure shuffler node before aggregating them (on an untrusted third party).

The bounds obtained in [18] are directly applicable to our $\mathcal{A}_{n,d}$ algorithm, but only when $d = 0$ (i.e. without additional dummies). The adaptation of the bounds obtained for the case $d > 0$ is a work in progress, carried out in collaboration with the Inria-Magnet team. We indicate below why the bounds provided [18] can be adapted to our context.

In their setting, the authors of [18] consider the view of the attacker as $View_{\mathcal{A}}(\overrightarrow{x}) = (Y, \overrightarrow{x}_{\cap}, \overrightarrow{b})$:

- $Y = \{y_1, ..., y_n\}$ is a multiset containing the output of each local randomizer.
- $\overrightarrow{x}_{\cap} = (x_1, ..., x_{n-1})$ is a tuple containing the inputs of the first $n-1$ sources.
- $\overrightarrow{b} = (b_1, ..., b_n)$ a tuple containing binary value indicating which users submitted their true values.

They then show that $View_{\mathcal{M}}$ satisfies (ϵ, δ)-differential privacy by proving that:

$$P\left(\frac{P[View_{\mathcal{M}}(\overrightarrow{x}) = V]}{P[View_{\mathcal{M}}(\overrightarrow{x'}) = V]} \geq e^{\epsilon}\right) \leq \delta$$

.

In our setting, the view of the attacker can be modeled as :

- The output $\mathcal{O}$ (i.e., a histogram with T bins whose counts sum to $n + d$).
- The true targets $t_1, \ldots, t_{n-1}$ of the first $n-1$ users.
- The set $\mathcal{L}$ of users who submit random values, or equivalently, the binary vector $b = (b_1, \ldots, b_n)$ indicating which users submitted their true values.

One can see that the differences between their considered view and ours are small. Their results can thus be considered to give a privacy bound for our solution. The actual adaptation of the proof to our context is an ongoing work. However, we have already adapted their original code[5] to cover the case of dummies, and we are able to show preliminary results of expected privacy gains in the performance section below.

4.5 Evaluation

In this Section we assess the effectiveness of our solution by evaluating its privacy and performances on a representative example of distributed computation, a group-by computation using a MapReduce-like framework. The goal here is to present some preliminary results, a more detailed ongoing study shows the generality of the approaches for a broader scope. The details of the parameters used in our experiments are summarized in Table 4.1. Note that the first figure is obtained with the implementation of Theorem 4 while the other figures are obtained using the adapted code of [18].

Parameters	Values	Description and remarks
S	10000	Total number of sources in the *DEP*
T	10	Total number of targets in the *DEP*
σ	$\in [0.1 - 0.9]$	Sampling rate
δ	10^{-3}	The same everywhere
d	0-200	Number of dummies added by each scrambler
n	100-500	Size of the scramblers' input

Table 4.1: Experiments parameters

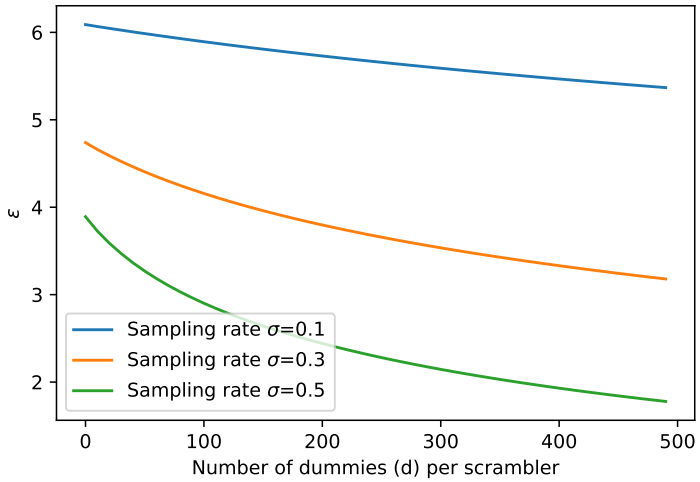


Figure 4.3: Theoretical value for ϵ as d increases (the results are independent from n).

Figure 4.3 plots the results obtained with the Theorem 4. Our experiments have shown that the value of ϵ does not depend on the size of the buffer n which points the need to compute a tighter bound. Indeed, in practice, sending a message with a batch of 1000 messages should provide better privacy than sending it with a batch of 100 messages.

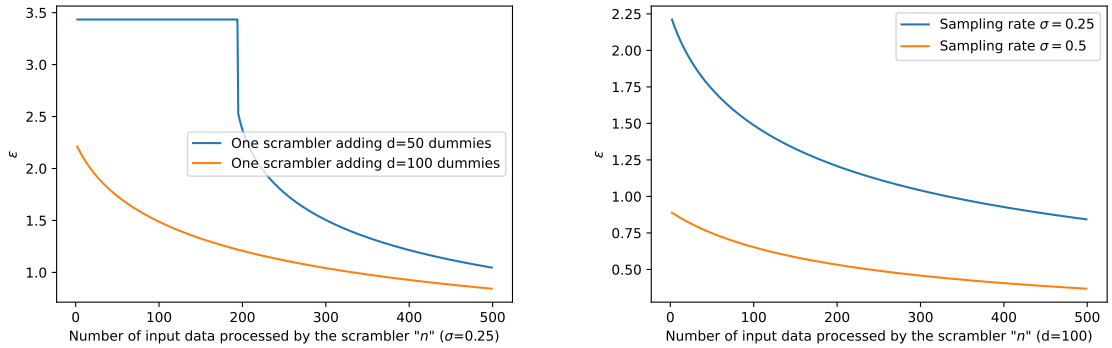


Figure 4.4: Value of ϵ as the number of input data processed by the scrambler "n" increases.

[5]The code is made available at https://gitlab.inria.fr/rladjel/amplification-by-shuffling

Figure 4.4 shows the positive impact of increasing the number of input data processed by the scrambler "n" on the privacy. The results obtained with the tighter bound are unsurprisingly better (see the comparison with the Figure 4.3). The left curve plots the evolution of ϵ when n increases for a fixed sampling rate $\sigma = 0.25$ while the right one plots the evolution of ϵ when n increases for a fixed amount of added dummies by each scrambler ($d = 100$). We can see that higher are the sampling rate σ and the number of additional dummies d, better is the impact of the amplification on the privacy.

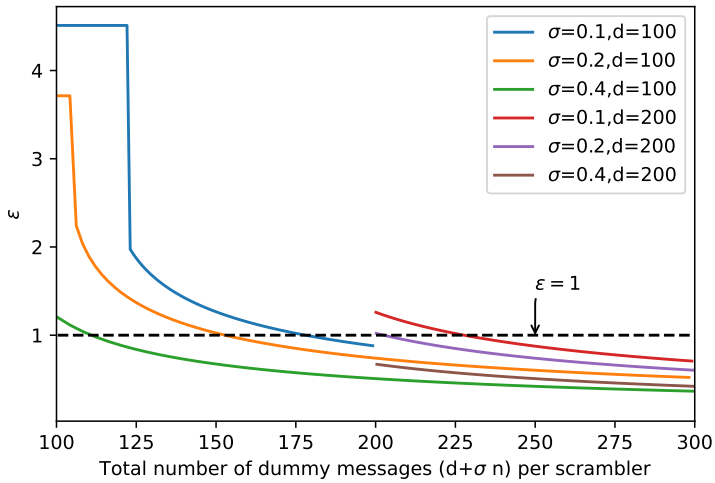


Figure 4.5: Value of ϵ for different values of d as the total number of dummy messages increases.

Figure 4.5 plots the evolution of ϵ as the total number of dummy messages increases. Note that the total number of dummies here represents the dummies added by the scrambler d and the one obtained while sampling the inputs $\sigma \times n$. The figure shows that different combinations of the parameters lead to a good level of privacy (around $\epsilon = 1$)[6]. The parameters can be tuned depending on the context. To lower the number of consenting users, σ need to be low (the blue or red curves). To reduce the network load (i.e. the volume of transmitted message) one need to reduce the number of additional dummies d and the sampling rate σ (i.e. the blue or yellow curves) and finally, to reduce the risk of exposition in case of a security failure of any scrambler, one need to reduce the number of input data processed by each scrambler (the red, purple and brown curves).

Figure 4.6 shows the performances gain versus broadcast in term of number of secure channels and volume of exchanged data. The x-axis corresponds to the ratio versus the broadcast while the y-axis is the value of ϵ given for a fixed ratio. The figure shows that we can obtain good privacy guarantees $\epsilon = 1$ while being 6 times (resp. 3) better than the broadcast in terms of the number of secure channels (volume of messages).

[6]Usually ϵ is recommended to be smaller then $ln(3)$ [52]

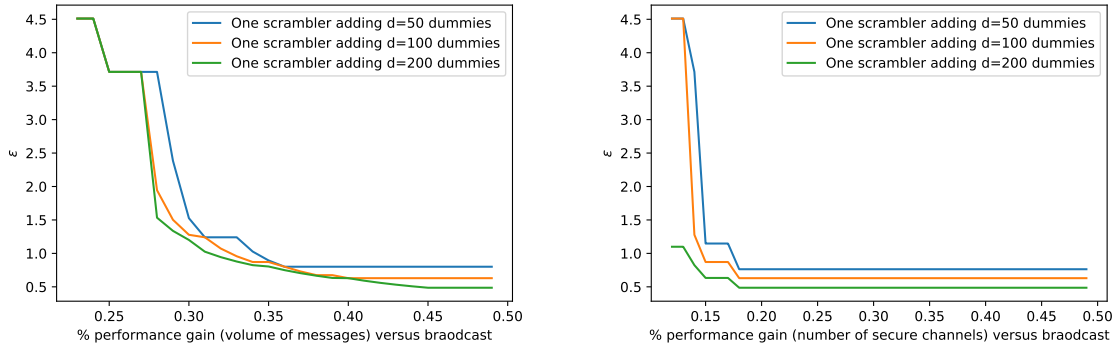


Figure 4.6: Performances analysis versus broadcast

4.6 Related work

4.6.1 Differential Privacy

Differential privacy [51] has become a gold standard metric of privacy in many scientific fields such as databases, data mining, machine learning. It is also starting to see real-world deployments, see for instance its recent adoption by the US Census Bureau [85] and its use in telemetry by several big tech companies [56, 46, 48]. We refer to [51] for a general reference on differential privacy and review below the literature most closely related to our work.

4.6.2 Differentially private histograms

Our problem is related to the task of computing histograms over a discrete domain, i.e., counting the number of elements of each "type" in a set of n elements. Indeed, from the perspective of the adversary, a given source (or scrambler) reveals a histogram of messages over targets (i.e., how many messages go to each target). In the centralized setting where one has access to the n elements in clear, the standard approach for differentially private histograms is to compute the true histogram and then to add independent noise to each count [51]. For instance, the truncated geometric mechanism [63] adds noise sampled from the geometric distribution and then truncates the resulting counts to the interval $[0..n]$ to avoid negative or unreasonably high counts. While such approaches could in principle be applied to our setting at the level of each scrambler (or source), this would result in non-uniform probabilities for messages to be dropped. In particular, messages aimed for targets with lower counts would have a higher probability of being dropped, leading to representativeness issues for the downstream calculation and potential unfair treatment of data sources. Instead, we build upon randomized response [56], which is the standard technique for differentially private histograms in the local model (where each data source holds a single input). In our approach, each message is treated independently, ensuring that it has the same probability of reaching its destination regardless of its value or target.

4.6.3 Privacy amplification by shuffling

Our work leverages recent results on privacy amplification by shuffling [37, 55, 18], which have shown that differential privacy guarantees are amplified when each data source sends

its messages to a secure shuffler (scrambler) after applying a local randomization. While these works considered privacy for the content of messages with data-independent communications, an original aspect of our work is to leverage these results to guarantee the privacy of data-dependent communications. In this context, we propose to add dummy messages to complement the randomization of the original messages, and we extend the proof of [18] to analyze this extension.

4.6.4 Communication anonymization techniques

M2R [49] and *Observing and preventing leakage in MapReduce* [100] are the closest works to ours. They both propose a solution to hide the communication patterns between mappers and reducers in a MapReduce setting. *M2R* uses a cascaded mix network to securely shuffle the output of the mappers in order to ensure unlinkability. However, mixnets do not offers formal guarantees [44]. The authors in [100] propose the addition of a secure shuffle step that permutes then pads the outputs of the mappers to a maximum set beforehand. The overhead in term of exchanged messages can be high in case of biased distribution. Moreover, both solutions assume a unique controller that leads the computations and provides the data unlike our setting where no unique entity should control the computation.

4.7 Conclusion and perspective

In this chapter, we propose a new solution to control data dependency in communications generated by a distributed database query execution plan, with (ϵ, δ)-differential privacy guarantees. Unlike differential privacy protection techniques applied to data content, our proposal supports the processing of real (accurate) data and preserves full accuracy for the end result. We provide preliminary results on the possible trade-offs between privacy, security and performance. Our current work consists in formalizing the (ϵ, δ) boundaries that could be reached (in collaboration with the Inria-Magnet project team) and showing precisely how to adjust the parameters and integrate the solution into the DEP formulated in our Manifest-based framework. We also plan to extend this technique to other contexts, such as private database federations [21] and secure Big Data processing (Map-Reduce like) on a cloud infrastructure based on Intel SGX nodes [100].

Chapter 5

MBF Application in the Medical-Social Field

Contents

In this chapter we present an application of the manifest based framework introduced in Chapter 3 in the medical-social field using an on-going deployment architecture of a PDMSs and assesses the practicality of the Manifest framework. We introduce the concept of Trusted PDMS (TPDMS), that is the combination of a TEE and a PDMS in a same dedicated box where only trusted code can be installed. We assume that each individual is equipped with a TPDMS embedded in a dedicated hardware device. The proposed solution hence falls in the *Tamper Resistant Personal Server* family seen in Chapter 2.1, with the salient difference that the box has more computing power. More precisely, the box embeds a Trusted Computing Base, i.e. a certified software composed of: (1) a personal data manager managing and protecting the individual's data (storing, updating, querying data and enforcing access control rules) and (2) a code loader ensuring the confidentiality and integrity of the code (in the TEE sense) executed in the box. Thus, only the trusted data manager and code loader, and additional external code certified and verified by the code loader (through a signature of that code) can run in the box. Persistent personal data are stored outside the security sphere, in a stable memory attached to the box (e.g., a SD card or a disk), but encrypted by the TPDMS

to protect them in confidentiality and integrity. A TPDMS provides means to securely execute external code in the box, opening the door to the design of secure distributed computation protocols.

This chapter is based on a work [76] published and presented in the 28th International Conference on Information Systems Development (ISD2019)[1]. And [78] which is an extention of [76] published in Transactions on Large-Scale Data and Knowledge-Centered Systems journal volume XLIV (Special Issue on "Data Management - Principles, Technologies and Applications")[2].

5.1 Overview

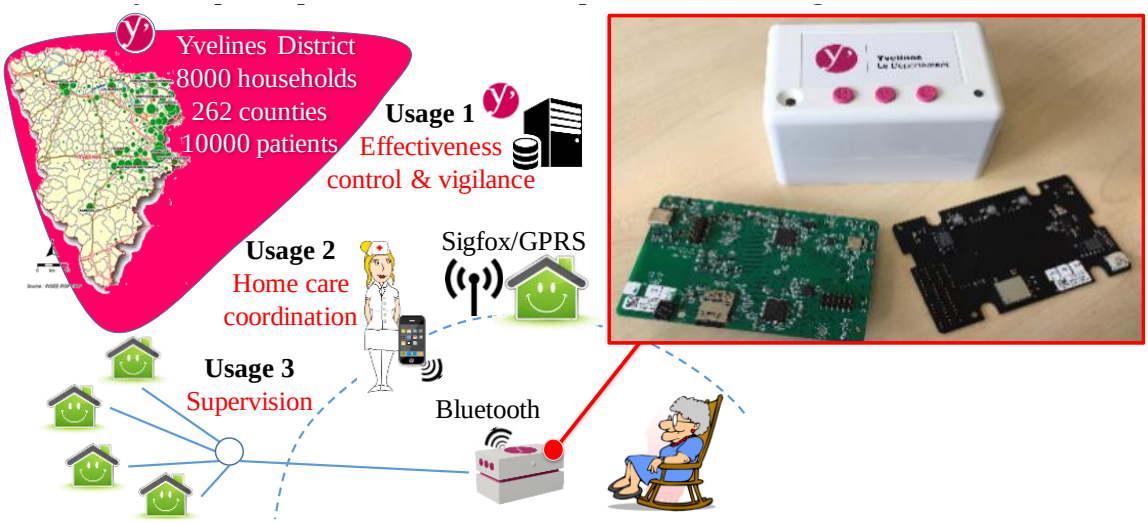


Figure 5.1: Architecture of the THPC solution.

End of 2017, the Yvelines district in France launched a public call for tender to deploy an Electronic Healthcare Record (EHR) infrastructure facilitating the medical and social care at home for elderly people. 10.000 patients are primarily targeted by this initiative, with the objective to use it as a testbed for a larger medium-term national/international deployment. The question raised by this initiative is threefold:

- How to make patients, caregivers and professionals trust the EHR security despite the recent and massive privacy leakages due to piracy, scrutinization and opaque business practices inherent to any data centralization process?
- How to combine privacy expectations with the collective benefits of data analysis tools to rationalize care, improve business practices and predict disease evolution?
- How to make patient's healthcare folder available even in a disconnected mode considering the low adoption of Internet by elderly people?

[1] https://isd2019.isen.fr/
[2] https://www.irit.fr/tldks/volumes/

The Hippocad company, in partnership with the Inria research institute and the University of Versailles (UVSQ), won this call for tender with a solution called hereafter THPC (Trusted Health Personal Cloud). THPC is based on a home box, pictured in Figure 5.1, combining 3 usages: (1) effectiveness control and vigilance, (2) home care coordination and (3) supervision. The hardware incorporates a number of sensors and communication modules (in particular SigFox) managed by a first microcontroller (called MCU1) devoted to the communication and sensing tasks. The data delivered by the box are used by the Yvelines district to cover usage (1), that is adjusting the care payment to their duration and performing a remote vigilance of the patient home. A second microcontroller (MCU2: STM32F417, 168 MHz, 192 KB RAM, 1 MB of NOR storage) is devoted to the PDMS managing the patient folder, a μ-SD card hosting the raw patient data (encrypted by the PDMS) and a tamper-resistant TPM (Trusted Platform Module) securing the cryptographic keys and the boot phase of the PDMS in MCU2. As detailed next, the combination of a TPM with MCU2 forms a TPDMS. Care professionals interact with the PDMS (i.e., query and update the patient's folder) through Bluetooth connected smartphone apps, covering usage (2). Finally, volunteer patients accepting to contribute to distributed computations (usage (3)), are equipped with a box variant where the SigFox module is replaced by a GPRS module.

The PDMS engine itself has been specifically designed by Inria research institute and the University of Versailles (UVSQ) to accommodate the constraints of MCU2. This embedded PDMS is a full-fledged personal database server capable of storing data in tables, querying them in SQL, and provides access control policies. Hence, care professionals can each interact with the patient's folder according to the privileges associated to their role (e.g., a physician and a nurse will not get access to the same data). Finally, the patient's data is replicated in an encrypted archive on a remote server to be able to recover it in case of crash. A specific master key recovery process (based on Shamir's secret sharing [117]) has been designed to guarantee that no one but the patient can recover this master key.

5.2 THPC as an instance of Trusted PDMS

The THPC platform described above is an illustrative example of TPDMS. As introduced above, a TPDMS is a combination of a TEE and a PDMS software embedded in a same dedicated hardware device, providing confidentiality and integrity guarantees for the code running in this device. The presence of two separate MCUs answers security concerns, indeed the Trusted Computing Base (TCB) is limited to the code located in MCU2 and does not include drivers and sensors (managed by MCU1) and is thus minimalistic. Additionally, the TCB is cryptographically signed. The TPM protecting the box is used at boot time (and NOR flash time) to check the genuineness of the PDMS code by checking the signature. The PDMS code in turn can download external pieces of code corresponding to the operators extracted from a Manifest, check their integrity thanks to the code signature provided by the Regulatory body, and run it. Hence, no code other than the TCB and signed operators can run in the box. The TPM also protects the cryptographic certificate that securely identifies the box and the master key protecting the personal database footprint on the μ-SD card. Note however that, while the TPM is tamper-resistant, the MCU2 is not. Hence, a motivated attacker could physically instrument his box to spy the content of the RAM at run time.

5.3 Distributed computations of interest

Example 5. *(Group-by' Manifest expressed by health organization.)*

```
Purpose:
    Compute the avg number days of hospitalization prescribed
    group by patient's age and dependency-level(Iso-Resource Group, GIR)
Operators:
    mapper source code
    reducer source code
Distributed execution plan and dataflow:
    number of mappers: 10.000
    number of reducers: 10
    any mapper linked to all reducers
Collection rules:
    SELECT GIR, to_year(sysdate-birthdate) FROM Patient;
    SELECT avg(qty)FROM Prescription
    WHERE prescType = 'hospitalization';
Querier: ARS-Health-Agency, Public key: Rex2%ÃźHj6k7âĂ
Manifest signature : dF$3s1£
```

The next critical step of the project is to integrate usage (3) (supervision). GPRS variant of the boxes are under development to establish a communication network via a central server settled by the Hippocad company, which plays the role of a communication gateway between the THPC boxes (it relays encrypted data bunches between THPC boxes but cannot access to the underlying data). Two essential distributed computations have already been selected, namely the *Group-by* and *K-means* computations. *Group-by* allows computing simple statistics by evaluating aggregate functions (sum, average, minimum, maximum, etc.) on population subsets grouped by various dimensions (the level of dependence or GIR, age, gender, income, etc.). Such statistics are of general interest in their own and are also often used to calibrate more sophisticated data analysis techniques (e.g., accurately calibrate the K parameter of a *K-means* computation). *K-means* is one of the most popular clustering technique and is broadly used to analyze health data [81]. To date however, few studies was conducted on home care [73] because data management techniques for this type of care are still emerging. Yet, *K-means* techniques already delivered significant results to predict the evolution of patient dependency level after a hip fracture [53] or Alzheimer symptoms [3], and derive from that the required evolution of the home cares to be provided and their cost. The first two Manifest-based computations considered in the project cover these use cases as follows:

- The *Group-by* manifest is the one presented in Example 5, using the usual map-reduce implementation of a group-by computation, where operators executed by participants are the map and reduce task respectively. It computes the sum and average duration of home visits by professionals grouped by professional category and level of dependence (GIR) of the patient. Such statistics are expected to help adjusting the duration of interventions and the level of assistance according to the patients' situation.

- The *K-means* manifest is inspired by a previous study conducted in Canada with elderly people in home care. This study analyses 37 variables, and provides 7 centroids [14] that

finely characterize the people cared for. Exactly as for the experiments in Chapter 3. The *K-means* manifest is computed over distributed PDMSs in three steps: (1) k initial means representing the centroid of k clusters are randomly generated by the Querier and sent to all participants to initialize the processing, (2) each participant playing a mapper role computes its distance with these k means and sends back its data to the reducer node managing the closest cluster, (3) each reducer recomputes the new centroid of the cluster it manages based on the data received from the mappers and sends it back to all participants. Steps 2 and 3 are repeated a given number of times or until convergence.

In Section 5.8.2, we give preliminary measures obtained by a combination of real measures and simulations for these two manifests since they are not yet deployed. Running manifests in the THPC context has required an adaptation of the random assignation protocol presented in Chapter 3, Section 3.4.1 to cope with the intrinsic limitation of GPRS in terms of communication bandwidth.

5.4 Adaptation of the Random Assignment Protocol to the THPC context

Given the low bandwidth of the THPC boxes (GPRS communications), a critical problem is limiting the amount of data transmitted to all participants, as hundreds of KBs broadcasted to all thousands of participants would not be compatible with acceptable performance. In order to reach this goal, we optimize the two main parts of the random assignation protocol (Chapter 3, Section 3.4.1) that lead to transmission of large amounts of data. The main optimization is making sure that we do not need to transmit neither the whole assignment nor the whole manifest to all participants as they only need their part of the assignment and the manifest related to their part of the computation. However, we need to make sure that the integrity of the whole manifest and assignment is ensured. In order to achieve these two seemingly antagonistic goals, we make use of Merkle hash trees [88] over the corresponding data structures.

The properties of the Merkle hash tree ensures that given the root of the hash tree, it is possible to provide a small checksum proving (in the cryptographic sense) that an element belongs to the corresponding hash tree, and it is computationally infeasible to forge such a proof. Note that the checksum is a logarithmic (in the number of values in the tree) number of hashed values and thus stays manageable (small size). Additionally, we avoid broadcasting the whole list of participants as only the assigning participant needs to perform checks on this list. We only broadcast a cryptographic hash of this list, and only send it in full to the assigning participant who actually needs to check it. The assigning participant however does not need to send back the full assignment, only a Merkle hash tree signed with his private key, and the random seed used to generate the assignment (so that the Querier can reconstruct it) is sent back. Finally, in order to perform his task in the manifest, any participant only needs his position together with the corresponding operator, collection queries and data flow and proof of membership to the logical manifest. Additionally, the participant needs to receive proof that the assignment is correct.

Summing up, we reduce the communication load during assignment building phase from a few broadcasts of a few hundreds of KB (for tens of thousands of participants) to only one large download for the assigning participant (again a few hundreds of KB), and small

downloads/uploads (a few tens of Bytes) for all other participants, drastically reducing the overall communication load, and making it manageable in constrained setting.

5.5 Fault tolerance protocol

Any distributed solution involving end-users computing resources must consider the case of participants' failures, i.e., becoming unreachable due to unexpected disconnections, shuts down, low communication throughput, etc. This statement is particularly true in our medical-social context involving battery-powered devices connected to the network by a GPRS module.

With the Manifest-based framework presented in Chapter 3, any participant failure conducts to stop the execution (fortunately without exposing any result), forcing the querier to restart processing from its beginning. The objective is thus to support a ratio of participant failures while enabling the execution to be completed. However, failures may impact the security of the solution: a malicious participant may deliberately attempt to weaken other participants' connectivity (e.g., denial of service attack) to harm the confidentiality or integrity of the computation. We consider here both the security and the performance aspects of handling failures.

Participants failures in our context may occur either during step 2 (random assignment of operators) or step 3 (secure distributed evaluation). Failures at step 2 are easily tackled by removing faulty participants from the protocol. Failures at step 3 are more difficult to address. Traditional fault-tolerant solutions rely either on redundant execution methods (e.g., perform k independent executions of a same operator and select a result) or on check-pointing mechanisms (e.g., store intermediate results of operators and restart computation from these points). Both solutions unfortunately increase data exposure, either increasing the number of participants processing the same data or introducing additional persistent copies of such data. We select the first solution anyway (redundant execution) because its negative effects are largely alleviated by the randomness measure integrated in our protocol, proscribing attackers targeting specific nodes.

We explain below how to integrate redundant execution in a manifest, for the case of a n-ary tree-based execution plan. Let assume a redundancy factor of k (with $k = 2$ or 3 in practice), a failure-resilient solution can be formed as follows:

1. For a manifest M requesting N participants, $k \times N$ participants are actually selected.

2. When assigning operators to participants, k participants are assigned the same operator in M. They inherit the same position in the execution plan and the same operator to run, to form a so-called bundle of redundant participants. Hence, participants $p_i, p_j, \ldots, p_k \in bundle_l$ all execute the same operator op_l.

3. The assignment function from participant to role is de facto no longer injective. However, the position of each participant in the execution plan is still determined at random. Consequently, bundles are also populated randomly.

4. Each participant in a successor bundle is connected to all participants in an antecedent bundle by edges in the execution plan (antecedent/successor refer to the position of participants/bundles in the execution plan/tree).

5. Instead of iterating for each antecedent, a participant iterates for each bundle, and the participants in this bundle are considered one after the other at random. If one does not answer after a certain delay, it is considered as 'failed' and a next participant in the same bundle is contacted. If all participants of the same bundle fails, the whole processing is abandoned.

Correctness. Any participant consuming an input data (resp. sending an output data) checks beforehand the integrity and identity of its antecedents (resp. successor) in the execution plan, as in a standard (i.e., non-fault-tolerant) execution. If a complete bundle fails, the error is propagated along the execution plan such that the execution ends with an error and no result is published. Finally, what if participants of a bundle play the role of data collectors, each being connected to its local PDMS? The local data are not the same at each participant and the output delivered by the bundle hence depends on which active participant is finally selected in the bundle. This randomness in the result of the computation is actually not different from the one incurred by selecting N among potentially P consenting participants in the protocol and does not hurt the consistency of the execution.

In terms of performance, this strategy does not impact the response time since participants in the same bundle work in parallel (it can even be better considering that the input of an antecedent bundle may arrive faster than the one of a single antecedent). However, the overall computation cost (sum of all computation costs) is increased by a factor of k and the number of communications by a factor of k^k. While this grows extremely rapidly, note that we only need to consider very small values of k (typically 2) as we allow for one failure per bundle, and the probability of a whole bundle failing is extremely small even if bundles are small.

5.6 Anonymous communication protocol

Anonymizing the communications is mandatory in the medical field. In our distributed computation framework, sensitive data can be inferred by observing the information flow between computation nodes. Typically, in canonical map-reduce computations encoded with our manifest-based framework, as shown in Example 5, each participant acting as a mapper node sends its $< GIR, [age, \#days - of - hospitalization] >$ to the given reducer in charge of aggregating the information for that GIR. Observing the communications would reveal the recipient reducer for any participant and hence disclose its GIR value (i.e., its level of dependence).

Distributed execution plans often exhibit such data dependent data flows, as directing tuples to be processed together to a same computation node is necessary to evaluate certain statistics (e.g., computing a median) and/or improve performance. In the general case, two main strategies can be adopted to hide data dependent communications: (1) use anonymous communication networks (e.g., use TOR) to hide any link between data recipient nodes and source nodes, or (2) "cover" data dependent communications within fixed, data independent patterns. Resorting to the first approach requires tackling the issue of adapting onion routing protocols to our resource constrained THPC platform. This is still considered an open issue in the general case of constrained IoT devices (see, e.g. [67]), making such approach infeasible in practice in the short term.

Using the second approach would simply mean replacing sensitive communication patterns with data independent ones (e.g., broadcasts) at the price of extra communications. In the previous example, this could be achieved by enforcing (as part of the manifest) that any

message sent from a mapper node to a reducer node also triggers sending one extra 'empty' messages of same size to all other reducers, thus forming a 'broadcast-like' communication pattern (and hence hiding the value of GIR). The expected performance penalty is high, especially in the context of our THPC solution, considering the limitation of GPRS in terms of communication bandwidth.

We present below a simple way to adopt data independent communication patterns in a manifest, while limiting the communication overhead to a minimum acceptable in our context.
Adopting data independent communication patterns. Let consider for simplicity a n-ary tree-based execution plan in the manifest, where at a given level l in the tree, the data exchanges issued from the child nodes to the parent nodes (at level $l+1$) reveal information on data values processed at the child nodes. For the sake of simplicity, we consider that each child node transmits a unique message to a given parent node, selected on the basis of a sensitive information hold at the child node. The naive solution to avoid exposing sensitive information is to cover such child-to-parent message by broadcasting an empty message of same size to all other potential parent nodes at level $l+1$. In terms of extra communications, with n_l nodes at level l each with a single message of size $|t|$ bytes to be transmitted to a parent node at level $l+1$, this causes $n_l \times (n_{l+1}-1)$ additional messages, with extra size $|t| \times n_l \times (n_{l+1}-1)$ bytes in total. In practice, considering, e.g. 10.000 mappers and 10 reducers as in Example 5 and a tuple size of 100 bytes, this leads to 9.000 extra communications with 900 KB data exchanged (mostly composed of 'empty' tuples), with unacceptable performance in the THPC setting.
Minimizing communication overhead. To reduce the communication overhead, we modify the distributed execution plan in the manifest as follows (see Figure 5.2): for each level l in the tree where data exchanges issued from child nodes to parent nodes (at level $l+1$) reveal information on data values (1) we form a k-equipartition[3] of the set of child nodes, we allocate one scrambler node per k-partition (with n_l/k scrambler nodes allocated in total) and connect each child node to the scrambler node responsible for its k-partition; and (2) we connect each scrambler node to all the parent nodes and fix at exactly $k' \leq k$ the number of messages each scrambler sends to each parent node.

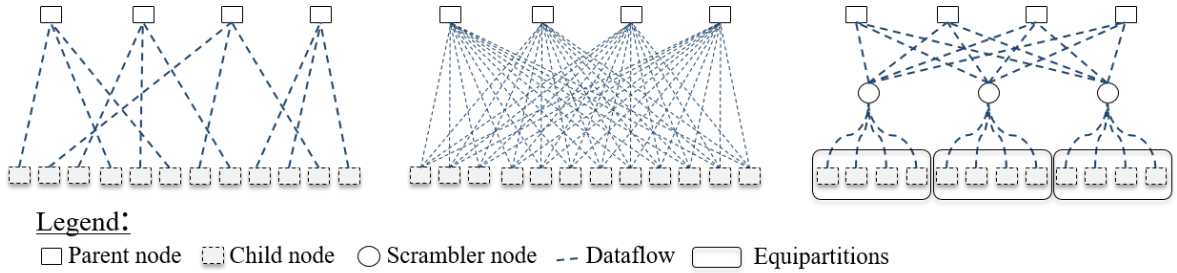


Figure 5.2: Covering sensitive data exchanges with data independent communication patterns (Left: data dependent; Middle: naive, Right: scrambler-based with k=4).

Each scrambler node acts in two phases. First, it collects one tuple $< P_i, E_{K_{P_i}}(M_{C_j}) >$ per child node C_j of the k-partition it takes in charge, with P_i the parent node the message has to be transmitted and $E_{K_{P_i}}(M_{C_j})$ the message M produced by C_j encrypted with the public key[4] K of P_i. Second, the scrambler prepares k' messages packages (each of same size)

[3] A k-equipartition of a set is the partitioning of this set in partitions of cardinality k.
[4] We assume that each node is endowed with a public/private key pair.

destined to each parent node, it places the encrypted messages collected from the child nodes in the appropriate package for the expected parent nodes and fills in the remaining packages with 'empty' messages (indistinguishable from others, as being same size and encrypted). In terms of extra communications, this causes $n_{l+1} \times n_l/k$ additional messages, each of size $k^{'} \times |t|$ bytes.

Resilience to attacks. The communication pattern introduced by scrambler nodes is fully deterministic and prevents from disclosing sensitive information (the only information disclosed is the size $k^{'} \times |t|$ of data exchanged from scramblers to parent nodes). The deterrence of side-channel attacks property must (1) guarantee that the leakage remains circumscribed to the data manipulated by the sole corrupted TPDMS and (2) prevent the attackers from targeting a specific intermediate result (e.g., sensitive data or data of some targeted participants). If a given scrambler is corrupted, it only reveals information regarding the data flow of the partition it has in charge, which ensures that the leakage remains circumscribed (first part of the property). Remark also that only the local communication pattern is exposed to corrupted scramblers, but not the content (payload) of the routed messages (as being encrypted with the recipient node's public key). Hence, lower k leads thus to more k-partitions and more scramblers, with a better resilience to side channel attacks. In addition, the random assignment of the (scrambler) operators to participants prevents potential attackers from targeting specific scramblers (enforcing the second part of the property). Note also that the impact on mutual trust is null, as the addition of scramblers does not change the deterministic nature of the query execution plan (nodes can check the integrity of predecessors, enforcing the global integrity of the query tree).

Performance analysis. In terms of performance, the value of k determines the size of the k-partitions and the number of scramblers $\lceil n_l/k \rceil$, but has no effect on the total volume of data exchanged. Indeed, the addition of scramblers lets unchanged the number of messages issued from child nodes (in our setting, n_l messages, one per child node, transmitted to the scrambler responsible for its partition), but it introduces $\lceil n_l/k \rceil \times n_{l+1} \times k^{'}$ (with $1 \leq k^{'} \leq k$ additional messages from scramblers to parent nodes, each of size $|t|$ bytes. Hence, the lower $k^{'}$ leads to the better efficiency. The worst case in terms of communications is $k = k^{'}$. At one extreme, with $k = k^{'} = n_l$ a single scrambler is introduced which transmits n_{l+1} groups of $k^{'} = n_l$ messages with in total the same overhead as that of the naive solution in number of transmitted bytes. At the other extreme, $k = k^{'} = 1$ leads to introduce n_l scramblers each sending n_{l+1} messages (one message per parent node) with the same global overhead. The performance with scramblers hence becomes better than the naive solution when $\frac{k^{'}}{k} < \frac{n_{l+1}-1}{n_{l+1}}$.

Calibration of the parameters k and $k^{'}$. Reducing the value of k increases the resilience to attacks (with lower k, more scramblers, and a better resilience to side channel attacks). It also improves the degree of parallelism (more scramblers run in parallel) and plays on the overall resource/energy consumption (due to fewer messages processed at each scrambler, with less memory consumed and less energy). This is of importance in the THPC context where the memory, processing and lifetime of each node is limited. Assuming k has been reduced appropriately to match (privacy and) resource constraints, the second step is to minimize the value of $k^{'}$ to reduce the communication overhead. At the same time, a too small value $k^{'}$ increases the risk of execution failure, if more than $k^{'}$ messages have to be transmitted at execution from a given scrambler to a given parent node. To enable fine tuning the value of $k^{'}$ at runtime, the strategy we adopt is to ask each scrambler, once all input tuples have been collected from the k-partition they have in charge, to first transmit to all parent nodes the

maximum number of tuples each has to transmit to this parent nodes, and ask the parent nodes to send back to scramblers the maximal received value, such that $k^{'}$ is fixed in all as the maximal value received from all parent nodes[5] . Note that during this phase, the only additional data transmitted from scrambler to parent is the number of intended messages for this specific parent node. As this data is known to the parent node regardless of the protocol used to fix k, it does not negatively impact security. In practice, well calibrated query execution tree lead to process in all parent nodes a roughly similar amount of tuples (for good load balancing and efficiency), leading to select $k^{'}$ bigger but close to $\lceil n_l/k \rceil$ to minimize the number of empty tuples to be send. Typically, considering Example 5, where $n_{l+1} = 10$ and $n_l = 10.000$, with 10 scramblers (i.e., $k = 1000$), most executions end up with $k^{'} \leq 200$, which means 5 times less data transmitted than using the naive strategy (equivalent to $k' = 900$).

In conclusion, the principle described here can be implemented to protect sensitive (data dependent) communication patterns with acceptable overhead in many practical examples of distributed PDMS calculations, ranging from simple statistical queries to big-data (map-reduce style) processing, as illustrated in the section on validation. The process of adding scramblers can be performed automatically by a precompiler taking as input a logical manifest and producing a transformed logical manifest covering the communications identified as sensitive. The appropriate value of the k' parameter does not need to be established at pre-compilation, but can be adjusted at runtime (as described above). The selection of the value of k to form the k-equipartitions is dictated by resource constraints in our context and must be provided for at pre-compilation. Tuning of the value of k and study of optimal strategies, as well as their integration in a precompiler are left for future work.

5.7 Lessons learned for the Deployment of THPC Solution

While the THPC platform is still under deployment over the 10.000 targeted patients, we can already draw interesting lessons learned and present preliminary performance and security results of the Manifest framework applied to the *Group-by* and *K-means* cases.

An important criterion for Yvelines district when selecting the THPC solution was its compliance with the new GDPR regulations and its ability to foster its adoption by patients and professionals.

5.7.1 Adoption by patients

From the patients' perspective, a crucial point was to provide differentiated views of their medical-social folder (e.g., a nurse is not supposed to see the income of the elderly person). To this end, Yvelines district has defined a RBAC matrix (role-based access control) with the help of French G29 members so that a professional owning a badge with a certificate attesting role R can play this role with the appropriate privileges on all patients' boxes. Each patient can explicitly - and physically - express his consent (or not) to the access of a given professional by allowing access to his box during the consultation, as he would do with a paper folder. The patient can also express his consent, with the help of his referent, for each manifest. A

[5]Note that this formally makes the communication flow data dependent as the chosen $k^{'}$ depends on the data sent to each scrambler. This, however, only leaks information on the distribution of data, not on any individual data. We do not view this as a significant threat.

notable effect of our proposal is to consent to a specific use of the data and to disclose only the computed result rather than all raw data as usual (e.g., consenting to an Android application manifest provides an unconditional access to the raw data).

5.7.2 Adoption by professionals

Professionals were, at first, reluctant to use an EHR solution which could disclose their contact details, their planning and statistical information that may reveal their professional practice (e.g., quantity of drugs prescribed or duration and frequency of home visits). In this respect, a decentralized and secured solution has been a great vector of adoption compared to any central server solution. Similarly, professionals were reluctant to see the data related to their practice involved in statistical studies unless strict anonymization guarantees can be provided. While the consent of the professionals is not requested for distributed computations, a desirable effect of our proposal is to never disclose individual data referring to a given professional, and submit all computation to regulatory approval.

5.8 Validation

We validate the effectiveness of our approach on the *Group-by* and *K-means* use-cases.

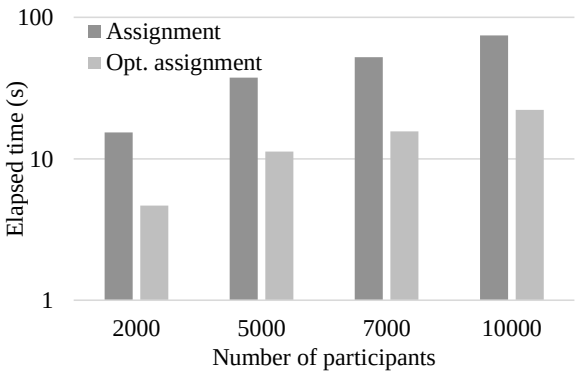
5.8.1 Experimental setting

For both examples, we implemented the corresponding mapper and reducer code in the THPC box with a server used to route (encrypted) messages between participants, as described in Section 5.3. We computed the execution time while considering different numbers of participating users and corresponding amount of data transferred during the computations. We used a simulation to derive execution times with large numbers of participants. The results are shown in Figures 5.3a,5.3b,5.3c,5.3d and 5.3e (all the curves are in log. scale). For the *Group-by* case we consider an implementation with *10* reducers and *50* different group keys, while for the *K-means* we consider *7* different clusters with *1* cluster per reducer as in [14] using a traditional distance metric [68]. Finally, we used synthetic datasets, as the objective is not to choose the most efficient implementation of a given computation, but rather to assess the effectiveness of the manifest based protocol on real use-cases. As cryptographic tools we used *ECC* 256 bits for asymmetric encryption, *ECDSA* signature scheme, *AES-GCM 128* bits for symmetric encryption and *SHA-2* as a hash function, leveraging the hardware acceleration provided by MCU2 for *AES, SHA-2* and *ECC*.

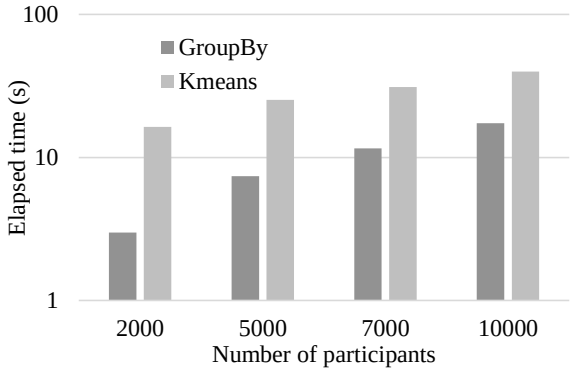
5.8.2 Performance evaluation

Figure 5.3a shows the execution time of the random assignment protocol without optimization (up to 75 seconds for 10.000 participants) and with the optimization presented in Section 5.4 (22 seconds for 10.000 participants). This time only depends on the number of participants rather than on the computation performed.

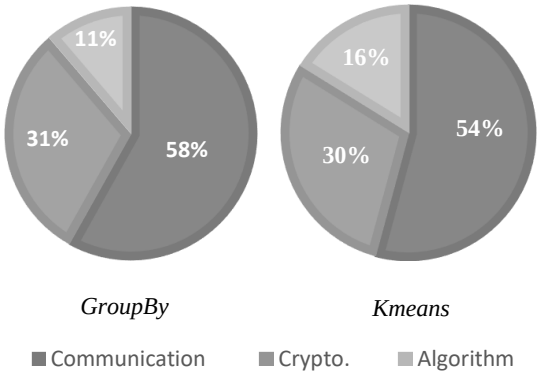
Figure 5.3b shows the time needed to execute, without randomness, the two examples mentioned in Section 5.3, namely 18 seconds (resp. 40 seconds) for a group-by (resp. k-means) query over 10000 participants. Summing these two figures in order to obtain the total execution time shows that the overhead incurred by the randomness protocol is not that high



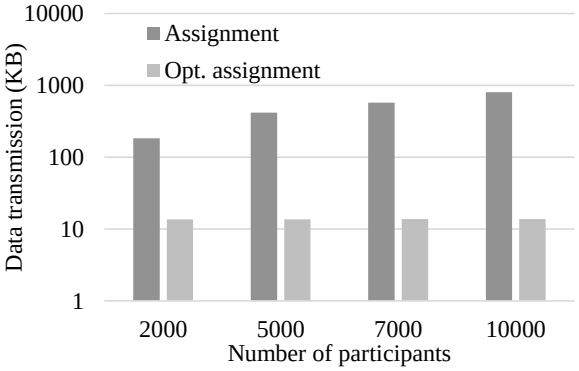
(a) Execution time (assignment).



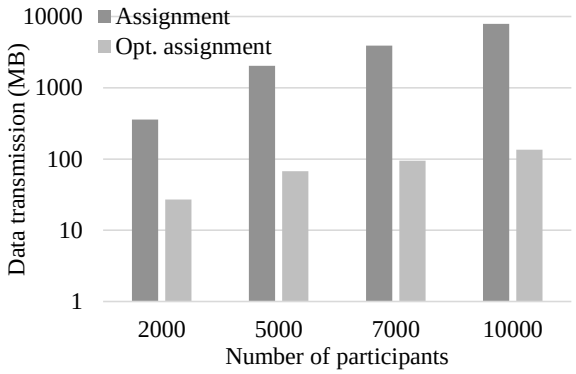
(b) Execution time (computation).



(c) Overall performance ratios.



(d) Exchanged data (by participant).



(e) Exchanged data (in total).

Figure 5.3: Security and performance evaluation.

(e.g. less than 20% of the k-means global time) and mostly due to the communication cost (cf. Figure 5.3c).

Figure 5.3c plots the ratio of the communications time (data transfer) and the cryptographic operations together with the algorithm time. It shows that the majority of time is due to the data transfer, which is not surprising in our context (low bandwidth in GPRS networks) and grows linearly with the number of participants.

Figure 5.3d shows the amount of exchanged data for one participant and Figure 5.3e the total amount of exchanged data during the whole computation. We computed the amount of data before and after the optimization, highlighting its dramatic impact (e.g., from 800KB per

participant without optimization down to 13KB and from more than 7GB to barely 130MB in total).

The main lessons drawn from these experiments are: First, even with the hardware limitations of the box in terms of computing power and communication throughput, the global time is rather low and acceptable for this kind of study (less than a minute for *10000* participants in comparison with manual epidemiological surveys which may last weeks). Second, the optimization of the assignment protocol has a decisive impact on both execution times and data volumes exchanged, with a significant financial benefit in the context of pay-per-use communication services (such as GPRS network).

5.9 Conclusion

The concept of Trusted PDMS (TPDMS) combined with a Manifest-based framework leverages the security properties provided by TEE to build a comprehensive personal data management solution. This solution reconciles the privacy preservation expected by individuals with the ability to perform collective computations of prime societal interest. The on-going deployment of the solution over *10.000* patients demonstrates the practicality of the approach and we expect that it could pave the way to new research works and industrial initiatives tackling the privacy-preserving distributed computing challenge with a new spirit and vision.

Chapter 6

Personal Agency Through the Manifest-Based Framework

Contents

6.1 Introduction

Empowerment, portability and PIMS.

While the world is being turned upside down by Artificial intelligence and the use of personal data, the place of individuals and control over their data have become central issues in the new European Data Protection Regulation that came into force in May 2018. The European Union's intention with this regulation was to empower individuals[1], which notably involved recognising a new prerogative: the right to personal data portability. Portability gives individuals the ability to extricate themselves from a captive ecosystem, and provides them with enhanced control over their personal data. According to the Article 29 Working Party, it should "re-balance the relationship between data subjects and data controllers"[2], and

[1] Communication from the Commission to the European Parliament and the Council, *Data protection as a pillar of citizens' empowerment and the EU's approach to the digital transition – two years of application of the General Data Protection Regulation*, COM(2020) 254 final, p. 2

[2] Guidelines of 13 April 2017 of Article 29 Working Party on the right to data portability, WP242 rev.01, `https://ec.europa.eu/newsroom/article29/item-detail.cfm?item_id=611233`

represents a new medium in the development of innovative and virtuous European economics around personal data. The corollary to this new right is the conception and implementation of technical platforms to "empower individuals by improving their right to self-determination regarding their personal data"[3], commonly known as PIMS[4]. These provide individuals with personal data management systems[5] to collate all their data in a single system – to be managed under their control. This gives rise to commercial structures such as Digi.me and Cozy Cloud, as well as governmental initiatives like Mydata.org[6] and Self-Data[7], supported by personal data protection agencies.

Empowerment goals partially achieved.

However, most analysts agree that the objectives of empowerment are only partially achieved today, with many barriers still to overcome. A recent CERRE Report[8] underlines that the way data portability rights can be exercised remains "minimal and far from ideal", due for instance to delays in processing data portability requests and a lack of standard models for retrieved data. The implementation of data portability still requires new mechanisms to "allow user's trust and controls on the procedures of right of data sharing"[9]. Needless to say, the obstacles are not merely technological, but also of a legal and economic nature. Recent publications consider that the portability right needs to be clarified to enable its most ambitious promises[10], to better adapt to the business model of the collaborative economy[11] and to quantify the expected gain for citizens' privacy (e.g., when using PIMS) whereas all one's personal data would be delegated to a provider anyway[12]. The European Commission is also conducting discussions along these lines: as part of its Data Strategy, it supports the creation of personal data spaces, which implies strengthening the right to portability enshrined in the General Data Protection Regulation[13].

[3]The motto of the MyData movement, which unifies PIMS editors and organisations, see `https://mydata.org/about/`

[4]S. Abiteboul, B. André, D. Kaplan, "Managing your digital life with a Personal information management system", *Communications of the ACM* 2015, 58 (5), pp. 32-35

[5]Anciaux, N., Bonnet, P., Bouganim, L., Nguyen, B., Pucheral, P., Popa, I. S., & Scerri, G. (2019). Personal data management systems: The security and functionality standpoint. *Information Systems*, *80*, 13-35

[6]See `https://mydata.org/about/`

[7]See `http://mesinfos.fing.org/english/`

[8]Centre for regulation in Europe (CERRE). June 2020. Krämer, J., Senellart, P., de Streel, A. (2020). Making data portability more effective for the digital economy: economic implications and regulatory challenges. `https://cerre.eu/sites/cerre/files/cerre_making_data_portability_more_effective_for_the_digital_economy_june2020.pdf`

[9]Martinelli, S. (2019). Sharing data and privacy in the platform economy: the right to data portability and "porting rights". In Regulating New Technologies in Uncertain Times (pp. 133-152). TMC Asser Press, The Hague.

[10]See in particular the so-called "Fusing scenario", where data portability fosters the creation of platforms of interoperable services, in: De Hert, P., Papakonstantinou, V., Malgieri, G., Beslay, L., & Sanchez, I. (2018). The right to data portability in the GDPR: Towards user-centric interoperability of digital services. *Computer Law & Security Review*, *34*(2), 193-203.

[11]Drechsler, L. (2018, June). Practical Challenges to the Right to Data Portability in the Collaborative Economy. In *Collaborative Economy: Challenges and Opportunities, Proceedings of the 14th International Conference on Internet, Law & Politics. Universitat Oberta de Catalunya, Barcelona* (pp. 21-22).

[12]Urquhart, L., Sailaja, N., & McAuley, D. (2018). Realising the right to data portability for the domestic Internet of things. *Personal and Ubiquitous Computing*, *22*(2), 317-332.

[13]*Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, "A European strategy for data"*, COM(2020) 66 Final

A new angle to study empowerment: personal agency.

Considering the actual state of the regulation, one dimension underpinning the notion of empowerment appears insufficiently explored – that of "personal agency".

The concept of personal agency is a product of social sciences and forms the basis of individual empowerment. It was coined by Martha Nussbaum and Amartya Sen[14], then further developed by Ruth Alsop[15] and Deepa Narayan[16]. Personal agency characterises individual empowerment through two key components: the individual's "ability to envisage options and make a choice"[17] and "the capacity to transform choices into desired actions"[18]. According to these authors, the concept of personal agency can be used to characterise various vectors of empowerment related to human development, poverty reduction and women's status improvement.

We argue in this chapter that this concept can be transposed to personal data management, to offer a new reading of empowerment measures in this context. Hence, our point is to extend the statements made by Tim Berners-Lee, founder of the Internet and winner of the Turing Award, criticising the current situation of the Web and the monopolies it engenders in personal data management. Now working on a new PIMS solution, he uses the concept of *personal agency* as the key to empowering individuals[19] ("you will have far more personal agency over data").[20]

Definition of personal agency in the personal data management context.

Based on its initial meaning, personal agency – transposed to the context of personal data management by individuals – could be said to rest on two pillars.

(1) The first aim is to be clarified to enable all individuals to "**make decisions**" about their own data. On one hand, this requires a range of different options to be open to the individual, as promoted by portability which allows migration from one service to another. On the other hand, individuals should not only be able to give their consent and hence to access necessary information (e.g. in the general terms of use for services that process their data), but also to understand it (e.g. by adequately designing information to ensure educated consent). In other words, they should be capable of measuring the effects – and the risks – entailed by their decisions around the use of their data, especially by considering each party's responsibilities. Such are the required conditions to ensure informed decision-making.

[14]Nobel-Prize winning economist A. Sen defines personal agency as a dimension of his capability approach ("Well-being, Personal Agency and Freedom: The Dewey Lectures 1984", *The Journal of Philosophy* 1985, vol. 82, p. 206).

[15]R. Alsop *et al.*, "Measuring Empowerment in Practice: Structuring Analysis and Framing Indicators".

[16]D. Narayan *et al.*, "Measuring Empowerment: Cross-disciplinary Perspectives", 2005, p. 6: "*personal agency is defined by the capacity of actors to take purposeful action, a function of both individual and collective assets and capabilities* [...]".

[17]R. Alsop *et al.*, esp. p. 6.

[18]R. Alsop *et al.*, esp. p. 3.

[19]`https://inrupt.com/blog/one-small-step-for-the-web` Open letter by Tim Berners-Lee, 23 Oct. 2018.

[20]Other writers propounding control by individuals over their personal data also recommend exploring this avenue: C. Lazaro and D. Le Métayer, "Control over personal data: true remedy or fairy tale?", SCRIPTed, 12:1, 2015, INRIA Research Report vol. 8681. Available at: `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2689223`

(2) Each individual should be in a position to "**become an agent**" of the way their decisions are implemented, *i.e.* to be able to orchestrate how their data is processed and ensure that this complies with their decisions. Varying scales of control and safeguards can be argued for, bearing in mind that delegation of control is not always sufficient to secure personal agency[21]. In essence, the digital ecosystem where individuals with personal agency evolve would enable them to ensure that processing carried out on their data abides by their decisions, in full compliance and confidentiality. The underlying IT architecture and the levels of protection offered to individuals and third-party services with whom they interact should therefore be carefully considered to uphold personal agency.

Rationale and chapter outline.

This definition being established, our goal is to further investigate the meaning of personal agency and identify key conditions to empower individuals regarding Big Data features. Therefore, personal agency should be broken down according to the particular types of decision and use at stake. These may be divided into two main groups: exclusively individual uses related to a single person's data, referred to as "Personal Big Data", and collective uses by a community of individuals, called "Big Personal Data".

The first part of the chapter is thus dedicated to personal agency in the "Personal Big Data" context. We first review data portability and its current implementation through PIMS. Then, we underline the strongest form of empowerment to be currently suggested. Finally, we introduce a new "*bilateral trust*" condition, which should be met for the individual to interact with personal agency with third parties.

In the second part of the Chapter, we investigate the case of empowerment in a collective context, where Big Data functionalities would be provided to a community of citizens, called "Big Personal Data". Firstly, we describe the collective uses of personal data in the field of AI and review the current dominant scenarios, in order to conclude that they disregard personal agency. Secondly, we review alternative suggestions to provide for personal agency. Finally, we introduce a second necessary condition of trust driven by personal agency –*mutual trust*– and outline a preliminary proposal for a legal and technical construction on this basis. The last section summarises our main findings and concludes the chapter.

6.2 Empowerment with personal agency for "Personal Big Data"

This part follows a three-phase development. In the first stage, we shall briefly review the history of data portability and its implementation using PIMS, and identify two different levels of empowerment currently set out: weak and strong empowerment. In a second phase, we examine the strongest form of empowerment as implemented by PIMS, highlight its functionalities and underline their current shortcomings. In a third section, we propose a new

[21]The notion of personal agency introduced by A. Sen, Ruth Alsop or Deepa Narayan includes the effective power of a person (or group) and direct control of the procedure through the means at its disposal (see: Alkire, S, 2008, Concepts and measures of agency). Other works (see: Bandura, 1989, 2000; Crocker & Robeyns, 2009) introduce the related concept of "proxy agency" to consider delegation to another individual or a third party system. For the sake of simplicity, we focus here on personal agency exercised directly by the individual, using the available legal and technical tools.

condition based on personal agency, to enable empowerment in the context of "Personal Big Data". This new condition allows the individual to establish a *bilateral trust* relationship with third parties wishing to exploit results of their data processing. We illustrate this by an example and discuss two important questions to make it applicable, centred on architectural choices and the individual's liability.

6.2.1 Asserting individual empowerment: an overview

Emergence of the idea of portability.

Data portability, as a salient new right in the GDPR[22], opens new legal and technological opportunities. Emerging from joint projects such as Blue Button (medical data) and Green Button (electricity consumption data) in the United States, MiData (related to energy, financial, telecommunications and retail data) in Great Britain or MesInfos in France, driven by FING (a non-profit French think tank) and supported by CNIL (the French personal data protection agency), this allows citizens to download all or some of their data in a structured, commonly used and machine-readable[23] digital format.[24] Those projects might pave the way for empowering individuals at different scales. We propose to highlight existing PIMS in order to ascertain to what extent they empower individuals, and confront them to our vision of a developed empowerment based on personal agency.

A brief history of PIMS: development of new functionalities and terminology.

One of the earliest systems allowing individuals to manage their personal data under their exclusive control was introduced in the United States in 2008 by Eben Moglen, Professor at Columbia University. Called "FreedomBox"[25] this system uses personal servers like plug-computer (a low-cost mini-PC such as RaspberryPI) and free software to help individuals elude State control and keep social exchanges private.

The concept of a personal data server then emerged in academia, with proposals from INRIA[26] and MIT[27], possibly benefiting individuals (cross-analysis of personal data hosted in different data silos, quantified self-tracking), third parties (sharing results of personal data computations) or society as a whole (collaboration between groups of individuals).

[22]Not only as an extension of the right of access.

[23]On limits of the terms used: S. Elfering, *Unlocking the Right to Data Portability: An Analysis of the Interface with the Sui Generis Database Right*, MILPC Studies vol. 38, spec. p. 21 and f.

[24]Provision 68 encourages data controllers « *to develop interoperable formats* » enabling data portability, without creating an obligation to adopt or maintain systems that are technically compatible. The final version of the GDPR went back on Amendment 111 of the Parliament; European Parliament, legislative resolution of 12 March 2014, COM(2012)0011, 2012/0011(COD), art. 15(2a). For the extension of this right to standardised format, see: European Commission, op. cit., COM(2020) 254 final, p. 8, and P. Jyrcys, C. Donewald, J. Globocnik, M. Lampinen, "My Data, My Terms: A Proposal for Personal Data Use Licences", *Harvard Journal of Law & Technology*, vol. 33, Digest Spring 2020, p. 9. See, also: CNIL, *Le corps, nouvel objet connecté*, *Cahiers IP* 2014, vol. 2, p. 23 et seq.

[25]Initiated by E. Moglen and G. Bdale. Presented at FOSDEM 2013. Foundation website: https://freedomboxfoundation.org/.

[26]T. Allard, N. Anciaux, L. Bouganim, Y. Guo, L. Le Folgoc, B. Nguyen, P. Pucheral, I. Ray, I. Ray et S. Yin, "Secure Personal Data Servers: a Vision Paper", proceedings of the international conference on Very Large Data Bases (VLDB), vol. 3, p. 25-35, 2010.

[27]Y.-A. de Montjoye, E. Shmueli, S. S. Wang, A. S. Pentland, "OpenPDS: Protecting the Privacy of Metadata through SafeAnswers", *PLOS ONE* 2014, vol. 9(7).

Commercial proposals appeared from 2012 onwards (Meeco, Cozy Cloud, etc.) and the terminology shifted towards the concept of "personal cloud". These solutions include online offerings ("cloud") and are exclusively aimed at individuals ("personal"), who are given a "digital home" ("Welcome to your new digital home"[28] is the Cozy Cloud motto) with advanced capacities for quantified self-analysis ("Meeco's manifesto reads: "[...] What if you and I had the same power?").[29]

FING uses the term PIMS, or "personal information management systems", as a technical solution that integrates and applies big data processing to an individual's data for self-tracking purposes.[30] FING has also introduced the concept of "self data", defined as the exploitation of personal data by individuals for their own purposes.[31]

Lastly, Tim Berners-Lee, founder of the Web, published an open letter[32] in September 2018 criticising the current state of the Web and the monopolies it engenders in personal data management. With his own roadmap including personal data management techniques, the Turing Award winner has in turn launched a personal data management solution.[33] Tim Berners-Lee invokes the principle of "personal empowerment through data" ("data should empower you") but also uses the concept of personal agency ("you will have far more personal agency over data"),[34] which he believes is fundamental to the success of the next era of the Web.

Enshrining the right to portability

As a follow-up to these projects, reforms have been made in European and French law enshrining the right to personal data portability for data subjects,[35] with the intent to turn this new prerogative into an empowerment tool to adjust the balance of power between major service suppliers and their users[36]. Individuals can retrieve their data free of charge in an open-access, machine-readable format and can thus move from one operator to another without losing their track record. They can also take control and manage their data and its use themselves[37]. Portability consequently has become an instrument for "privacy by using"[38], a tool to learn about privacy protection mechanisms, encouraging individuals to reclaim their

[28] See the Cozy Cloud website: https://support.cozy.io/article/280-etape-3-bienvenue-dans-votre-domicile-numerique.

[29] See the Meeco manifesto: https://www.meeco.me/manifesto.

[30] "La gestion de votre 'vie numérique' avec un système de gestion des informations personnelles", by S. Abiteboul (INRIA and ENS Cachan), B. André (Cozy Cloud) and D. Kaplan (FING and MesInfos), on the Le Monde *Blog Binaire*, 2014. http://binaire.blog.lemonde.fr/files/2014/07/personalInfoSystem.short_.fr_.3.pdf.

[31] Wikipedia French: https://fr.wikipedia.org/wiki/Self_Data.

[32] Open letter by Tim Berners-Lee, dated 23 Oct. 2018: https://inrupt.com/blog/one-small-step-for-the-web.

[33] MIT Solid project: https://solid.mit.edu/.

[34] Open letter by Tim Berners-Lee, op. cit.

[35] Art. 20 GDPR and Art. 39 of the French Data Protection Act [LIL] (as amended by Act no. 2018-493 of 20 June 2018) – Art. L. 224-42-3 et seq. of the Consumer Code, established by Act no. 2016-1321 of 7 Oct. 2016 on the digital Republic, subsequently repealed by the Act of 20 June 2018.

[36] On the scope of regulation through data as a counterpower given to final users on a market, see: Autorité de la concurrence, AMF, Arafer, Arcep, CNIL, CRE, CSA, *Nouvelles modalités de régulation. La régulation par la donnée*, 8 juill. 2019, esp. p. 3, for further references, see ref. 54

[37] See C. Zolynski and M. Leroy, "La portabilité des données personnelles et non personnelles, ou comment penser une stratégie européenne de la donnée", *Légicom* 2018, p. 105, esp. p. 108 – See also, C. Berthet and C. Zolynski, "L'empouvoirement des citoyens de la République numérique : regards sur une réforme en construction", *RLDI* 2018, p. 60, esp. no. 9 et seq.

[38] A. Rallet, F. Rochelandet, C. Zolynski, "De la Privacy by Design à la Privacy by Using: Regards croisés droit/économie", *Réseaux* 2015, vol. 189(1), p. 15-46.

informational autonomy[39], as an essential part of digital literacy. In fact, this new right enables individuals to take back some control over their personal data, in two different ways: by both "*receiv[ing] the personal data concerning him or her*"[40] and having "*the personal data transmitted directly from one controller to another*"[41]. Yet, the strict scope of application of this right counteracts the idea of empowered individuals[42], fully able to control their personal data, even more in the Big Data era. This right is subject to various conditions narrowing its field of application[43]. Indeed, it can only be exercised for data that (i) the individual "provided to a controller"[44] on the basis of consent or contractual performance[45] and (ii) only if the processing is carried out by automated means. Therefore, portability appears considerably limited, concerning its legal scope[46] and material scope[47].

Proposal for a strengthened data portability right.

As this right has not yet reached its full potential, academics and institutions call for an enhanced regulation, some through competition law[48], others through data protection law[49]. For instance, the Commission underlines the "*absence of technical tools and standards that make the exercise of their rights simple and not only burdensome*" and acknowledges that true empowerment should not be limited to mere portability[50] as "*switching of service providers*", but also aim at "*enabling data reuse in digital ecosystems*".[51] The Commission therefore highlights the need to create a supportive environment for the development of these solutions,

[39]Article 29 Data Protection Working Party, *Guidelines on the right to data portability*, Adopted on 13 December 2016, as last revised and adopted on 5 April 2017, WP 242 rev. 01, footnote 1, p. 4

[40]Art. 20(1) of the GDPR

[41]Art. 20(2) of the GDPR

[42]H. Ursic, "The Failure of Control Rights in the Big Data Era – Does a Holistic Approach Offer a Solution ?", in M. Bakhoum, B. Gallogo Conde, M.-O. Mackernordt & G. Surblyte (Eds.), *Personal Data in Competition, Consumer Protection and IP Law – Towards a Holistic Approach ?*, Berlin Heidelberg, Springer, 2017, Available at SSRN: `https://ssrn.com/abstract=3134745`

[43]See : J. Belo, P. Macedo Alves, "The right to data portability: an in-depth look", *Journal of Data Protection & Privacy* 2018, vol. 2, 1, pp. 53-61

[44]For a suggested interpretation, see: P. De Hert, V. Papakonstantinou, G. Malgieri, L.Beslay, I. Sanchez, « The right to data portability in the GDPR. Towards user-centric interoperability of digital services », *Computer Law & Security Review* (2018), pp. 193-203, spec. p. 199

[45]Art. 20(1)(a) of the GDPR

[46]O. Tambou, *Manuel de droit européen de la protection des données à caractère personnel*, Bruylant, 2020, spec. 203, n.255 and further

[47]I. Graef, M. Husovec, N. Purtova, "Data portability and Data Control: Lessons for an Emerging Concept in EU Law", *German Law Journal*, 2018, Vol. 19, No. 06, spec. p. 1370 and f., J. Drexl, *Data Access and Control in the Era of Connected Devices, Study on Behalf of the European Consumer Organisation BEUC*, BEUC Study, Brussels, 2018, n.43, 110

[48]O. Lynksey, *The Foundations of EU Data Protection Law,* Oxford University Press, 2015, 265, as quoted in H. Ursic, 2017, EDPS, *Preliminary Opinion of the European Data Protection Supervisor, Privacy and competittiveness in the age of big data : The interplay between data protection, competition law and consumer protection in the Digital* Economy, March 2014, §26, p. 15

[49]COM(2020) 66 final, p. 21: "*Explore enhancing the portability right for individuals under Article 20 of the GDPR (. . .) (possibly as part of the Data Act in 2021)*", Centre for regulation in Europe (CERRE), June 2020. Krämer, J., Senellart, P., & de Streel, A. (2020). *Making data portability more effective for the digital economy: economic implications and regulatory challenges*, spec. n.6.2.1, p. 78

[50]H. Ursic, *op. cit.*

[51]*Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, "A European strategy for data"*, COM(2020)66 Final, p.10

requiring a more progressive interpretation of the article 20 provisions. Recital 68 of the GDPR, which focuses on the sole transmission to another data processor[52], might, according to the European Commission, require to enhance data portability to give individuals "*more control over who can access and use machine-generated data*" such as "*real-time data access and making machine-readable formats compulsory*".[53] Consequently, the EU Commission issues its will to possibly include this extension as part of the 2021 Data Act.

Current status: weak empowerment and strong empowerment.

In its actual state, empowerment gained from exercising the right to portability seems to range from "weak empowerment", where individuals merely switch from one commercial service to another, to "strong empowerment", where individuals migrate to a personal data management service and thus regain sovereignty over their data. Legislators plainly conceived this prerogative as competition-focused[54], in the manner of the phone number portability advocated to force telecommunications operators to open up the market by lowering the exit barriers[55]. The right to data portability is therefore an instrument for extricating oneself from a captive ecosystem: it allows individuals to migrate from one operator to another without losing their data and without the drudgery of retrieving data from different systems.[56] It imparts service users with more freedom of choice, and could stimulate competition through innovation. Empowerment limited to this choice can be referred to as "weak".

Empowerment may be characterised as "strong" when data recovery gives individuals an active role in the data lifecycle. It was in this context that personal cloud solutions were developed as a corollary to these new portability rights. They form the technical lever to

[52]Art. 20§3 of the GDPR

[53]COM(2020) 66 final, p. 20

[54]Commission Staff Working Paper, *Impact Assessment, Accompanying the document Regulation of the European Parliament and of the Council on the protection of individuals to the processing of personal data and on the free movement of such data (General Data Protection Regulation) and Directive of the European Parliament and of the Council on the protection of individuals with regard to the processing of personal data by competent authorities for the purposes of prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, and the free movement of such data* (*COM(2012) 10 final, COM(2012)) 11 final,* Jan., 2012, SEC(2012) 72 final, *e.g.* at p. 28 stating that "*Portability is a key factor for effective competition, as evidenced in other market sectors, e.g. number portability in the telecom sector.*", Commissioner Joaquin Almunia, Speech, *Competition and personal data protection*, 26th, November 2012, available at: `https://ec.europa.eu/commission/presscorner/detail/en/SPEECH_12_860`, European Commission, Staff Working Document on the free flow of data and emerging issues of the European data economy Accompanying the document Communication of the Commission, *Building a European Data Economy* (COM(2017) 9 final), SWD(2017) 2 final, spec. p. 47, Article 29 Data Protection Working Party, WP 242 rev. 01, p. 4, De Hert P., Papakonstantinou V., Malgieri G., Beslay L., Sanchez I., " The right to data portability in the GDPR. Towards user-centric interoperability of digital services ", *Computer Law & Security Review* (2018), pp. 193-203, I. Graef, J. Verschakelen, P. Valcke, "Putting the right to data portability into a competition law perspective", *Law: The Journal of the Higher School of Economics*, *Annual Review* 2013, p. 53-63, Y. Poullet, "Is the general data protection regulation the solution?", *Computer Law &Security Review* 34 (2018), pp. 773-778, spec. p. 777, For a broader view on data over competition aspects: Autorité de la Concurrence, Bunderskartellamt, *Competition Law and Data*, 10th May, 2016, p. 11 and f.

[55]For an overview of mobile number portability and its effects on competition, see: B. Usero Sánchez, G. Asimakopoulos, "Regulation and competition in the European mobile communications industry: an examination of the implementation of mobile number portability", *Telecommunications Policy* 36 (2012), pp. 187-196, on cross-border portability of online content services, see: European Commission, DG for Communications Networks, Content and Technology, *Annual Activity Report. 2019*, 31st March, 2020, Ares(2020)1859706, p. 5 and further

[56]C. Zolynski and M. Leroy, op. cit.

exercise the right to portability. An individual's personal cloud has a range of connectors (to his bank, his employer and any external service that possesses his personal data) which lets them retrieve his personal data automatically.[57] With these offerings, he can combine all his data in a single system and adjust access in favour of innovative services.

6.2.2 Current meaning of strong empowerment: "Personal Big Data"

"Strong empowerment" being the most advanced and promising version of data portability, we shall clarify its meaning in terms of features, in the context of PIMS.

From current promises and proposals to Personal Big Data.

Recent reviews of personal cloud solutions, whether conducted from a social sciences perspective,[58] a technical one[59] or in experimental form,[60] unanimously agree on the intended purposes. The key point here is to re-establish the individuals' control over the lifecycle of their personal data, from collection to destruction, while enhancing the use by individuals of their own data.

In terms of features[61,62], the first and foremost promise is to automatically reconstitute full personal records, which were originally stored in different data silos[63] (banking, medical history, internet search history, geolocation, social exchanges, etc.). The second key promise made to individuals is the cross-analysis of personal information, allowing them to benefit from the interconnection of personal records from different sources. For instance, a medical examination and its corresponding prescription can be automatically retrieved from bank records of the reimbursement for medication. As a third promise, cross-exploitation of individuals' data also allows them to derive statistical information and complex computed results from their records, in a quantified self-tracking perspective. For example, comparing medical data such as weight or cholesterol levels with physical activity or step counts allows them to monitor their health.

[57] PIMS plateforms propose a set of connectors to easily retrieve personal data for the individual from many sources. Existing connectors can be used in personal cloud applications, and additional ones can be implemented by developers at will. See for instance Cozy Cloud documentation: `https://docs.cozy.io/en/tutorials/konnector/` and Digi.Me presentation: `https://digi.me/sources/`

[58] T. Lehtiniemi, "Personal Data Spaces: An Intervention in Surveillance Capitalism?" *Surveillance & Society* 2017, vol. 15(5), p. 626-639.

[59] N. Anciaux *et al.*, "Personal Data Management Systems: The security and functionality standpoint", *Information Systems* 2019, vol. 80, p. 13-35.

[60] Pilote MesInfos 2016-2018. Synthèse/Enseignements/Actions. G. Jacquart, S. Medjek, M. Molins. Available at: mesinfos.fing.org/wp-content/uploads/2018/06/LivrableA5_Synthese-Enseignements-Actions_VF_Web.pdf.

[61] For more details on the three promises of personal cloud systems summarized here, we refer the interested reader to section 2 "*Existing personal cloud solutions*" of the above mentioned paper (N. Anciaux *et al.*).

[62] Note that while the concept presupposes exclusive control by individuals over their data, Personal Big Data does not assume that individuals "own" their data (see p. 23 of the FING paper, op. cit.).

[63] For instance: CNIL, *Vie privée à l'horizon 2020*, Cahiers 2012, vol. 1, p. 55, CNIL, *Le corps, nouvel objet connecté*, *Cahiers IP* 2014, vol. 2, p. 23 et seq, A. Poikola, K. Kuikkaniemi, H. Honko, *MyData – A Nordic Model for human-centered personal data management and processing*, available on: http://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/78439/MyData-nordic-model.pdf

6.2.3 Personal agency as a determining condition of individual empowerment

While certain legal and technical conditions regarding data portability rights are met, such as personal cloud tools, the ability for individuals to perform Personal Big Data, thus achieving empowerment, raises a key question: what kind of personal agency do individuals hold to *implement* their decision? Ensuring that users have the capacity to act entails checking whether they are able to assume the new power granted to them, i.e. in this case, responsibility for making decisions relating to how their own data is managed (knowing who to share it with, hence making informed decisions), and the ability to orchestrate the implementation and effectiveness of their decisions (being able to contribute to implementation and to assume control over it). We shall introduce here a new trust condition that individuals must be able to establish to exercise their personal agency, that we call *bilateral trust*. It will be illustrated by Example 6, and open questions regarding the technical processing architecture and the legal liabilities of the individual in this context shall be discussed.

Personal agency in the context of "Personal Big Data": a new trust condition.

Assuming management of one's own data in terms of Personal Big Data with personal agency would presuppose a capacity to (i) administer and secure one's data, (ii) stipulate and apprehend permissions to different applications and third parties, and (iii) define which processing is authorised and set up safeguards to ensure one's decisions are effective. We thus argue that transposing personal agency to the Personal Big Data context would lead individuals to secure bilateral trust whilst the personal data underlying their decisions is processed. On one hand, individuals must be assured that their data is handled in line with the decisions made and that the Personal Big Data computation will indeed be implemented faithfully and confidentially (i.e. the expected code is executed, and the personal data provided in inputs is not exposed). On the other hand, third parties and external applications need a reciprocal guarantee from the individual, that the Personal Big Data processing results are indeed computed over the right datasets and are run as expected. This means being able to settle a two-sided trust guarantee, which we call *bilateral trust*. In Example 6, we illustrate this condition in the simple case of computing an energy bill based on a customer's energy consumption traces.

In light of this necessary *bilateral trust*, the technical and legal conditions in which the PIMS solutions are offered shall be analysed in order to determine which are likely to ensure personal agency. In other terms, attempting to assess the personal agency of service users implies ascertaining which party enjoys actual agency. From a technical perspective, one must assess who is trusting whom and thus who the administrator is, therefore questioning the processing architecture. From a legal standpoint, liability issues are raised, for example in the case of error or dispute. An additional concern is to make the proposal appropriate (and acceptable) in practice, while avoiding to overburden the individual. In the following, we discuss these open questions.

Example 6. *("Personal Big Data" to enable citizen to compute energy bills.)*

More and more citizens are concerned about feeding personal data to external services. When calculating an energy bill, the energy consumption traces generated by a smart metre, which reveal details of the individual's activity, are sent to the energy supplier who then calculates the bill and charges the customer.

Big Data for the citizen: an ability to interact with third parties without revealing personal data. The PIMS alternative ensures that "services move to the data" rather than personal data being sent to services. Citizens can thus exercise data portability to retrieve traces of consumption from their smart metre, and a computation code/app provided by their energy supplier is downloaded on their PIMS to produce the bill. The issue related to personal agency is that the energy supplier must trust the individual.

Bilateral trust as a necessary condition for "Personal agency". Personal agency aims to empower citizens to perform such Personal Big Data computations on their own. This requires making individuals capable of bilateral trust, by means of two main new capabilities:

1. First, the ability to guarantee to the individual that their raw personal data remains confidential. To the extent that detailed energy consumption trails may reveal the individual's activity, this is a prerequisite to trigger adoption.

2. Second, the capacity for the individual to undertake that the final result was indeed computed on the expected dataset (the provider must be sure that the data subject has not truncated their data to lower the bill) and used the expected code (the one furnished by the energy provider). This is an essential issue if the client is to be charged according to the result.

The provider may only have access to the aggregated energy consumption result (needed to charge the client), or be allowed to hold a finer degree of data (for instance, in case of billing error or dispute).

Architectural choices – a measure of personal agency

One notable feature of the personal cloud is that the processing and applications of Personal Big Data "are moving" to the relevant data, as opposed to personal data which migrates toward remote services, as it occurs with most existing cloud services[64]. An individual's personal agency can be measured by its capacity to implement this type of application under the exclusive control of the individual in a digital ecosystem that allows them to build the desired reciprocal trust. Personal agency would therefore depend on architectural choices for personal clouds, i.e. the technical solutions implemented.

With centralised approaches (for example, MyDex.org or MyData.org), data administration and security is based on the personal cloud platform provider. This type of centralised management built on delegation technically allows secondary uses (beyond the individual's

[64]D. Mula, "The Right to Data Portability and Cloud Computing Consumer Laws", *in Personal Data in Competition, Consumer Protection and Intellectual Property Law, Towards a Holistic Approach?*, Springer, MPI Studies on Intellectual Property and Competition Law, 28, 2018 pp. 397-409, spec. p. 398 and 399

control) and exacerbates the risk of large-scale attacks (affecting millions of individuals). This requires strong trust[65] from individuals to the platform provider and all the personal applications running on the system.

Self-hosting is a solution based on decentralised architecture, where each individual manages their own personal data on domestic hardware (for example, Di.Me,[66] CloudLocker, Cozy Cloud, Databox or Tim Berners-Lee's Solid). This gives individuals physical control over the platform which, if properly implemented, gives very high overall security (the cost-benefit ratio of an attack is dissuasive because any one attack only reveals a single individual's data). But responsibility for administering this system might befall individuals, with the attendant risk of error, loss or theft of personal data[67]. The DynDNS attack in late 2016, which infected non-secure embedded systems like printers and internet boxes, points out the vulnerability of self-hosted solutions.

Intermediate architectural solutions to these two extreme approaches can pave the way for different compromises according to the level of personal agency sought.

Personal agency and risk of "boomerang effect".

Are individuals with personal agency therefore called upon to bear all responsibility when it comes to processing their own data with these Personal Big Data solutions? This raises concerns about excessive responsibility, leading to a potential "boomerang effect". The regulators also stress that users must be informed of the risks they run in taking over management of their own data, in that they lose access to the data security solutions offered by data controllers and take responsibility for the data.[68] This is all the more true as the liability regime established by the GDPR does not seem designed[69] to take into account the shift in perspective caused by these new individual data management solutions. Some are bound to criticise a potential elusion of liability by operators offering these new individual data management services, who might claim that their individual users should be qualified as data controllers[70]. Yet, the latter might benefit from the purely personal or household activity exemption, excluding the

[65]Anciaux, N., Bonnet, P., Bouganim, L., Nguyen, B., Pucheral, P., Popa, I. S., & Scerri, G. (2019). Personal data management systems: The security and functionality standpoint. *Information Systems*, *80*, 13-35. See in Section 2.1: "*Typically, data leakage resulting from attacks conducted against the personal cloud provider or the applications (which could be granted access to large subsets of raw personal data), or resulting from human errors, negligence or corruption of personal cloud employees and application developers, cannot be avoided in practice. This is critical because such solutions rely on a centralized cloud infrastructure settings which exacerbate the risk of exposing a large number of personal cloud owners, and hence may be subject to many sophisticated attacks.*"

[66]M. Sjöberg *et al.*, "Digital me: Controlling and making sense of my digital footprint", 5th International Workshop "Symbiotic Interaction", p. 155-167, 2017. http://hiit.github.io/dime-server/.

[67]S. Abiteboul, B. André, D. Kaplan, "Managing your digital life with a Personal information management system", *Communications of the ACM* 2015, 58 (5), pp. 32-35

[68]See the G 29 Guidelines on the right to data portability, WP 242 rev. 01, 5 Apr. 2017, p. 22 et seq. `https://ec.europa.eu/newsroom/article29/item-detail.cfm?item_id=611233`

[69]For instance, liability for the conception and determination of means of processing cannot weight on the shoulders of a user processing data outside the scope of the domestic exemption through a medium furnished by an intermediary

[70]See WP29, Opinion 5/2009 on online social networking, Adopted on 12 June 2009, WP 163, p. 5, A. Debet, « La Commission des clauses abusives et la protection des données personnelles sur les réseaux sociaux : une incursion hésitante dans un territoire inconnu », *Revue des contrats* 2015, n.3, p. 496, N. Metallinos, *in* A. Debet, J. Massot et N. Metallinos, *Informatique et libertés, La protection des données à caractère personnel en droit français et européen*, Lextenso, coll. Les Intégrales, 2015, spec. n° 557, p. 271

application of the GDPR[71]. In this case, the GDPR applies to the provider of the means for such processing[72]. The issue is that those means must be essential[73], and not only technical and organisational, to trigger the qualification of data controller[74]. In the case of personal data management systems, the provider might be qualified as data controller[75]. However, as he merely supplies the means for individuals to "compute" their personal data, he cannot be liable for all the obligations under the GDPR[76]. This qualification must reflect the exact involvement of the provider so as not to exceed his personal responsibility[77]. Since users of technical means might be qualified as joint controllers alongside the provider, as set out by the European Court of Justice [78], the issue is still pending for individual users of PIMS.

Conditions and framework for personal agency.

In this context, how can the individual's ability to make informed decisions relating to the use of their personal data be ensured? On this point, the legal framework needs clarification, particularly regarding the expression of consent by the agent (the first condition of personal agency). One must ensure that users have the technical and cognitive capacities to make informed choices (second condition of personal agency). In addition, the entire liability chain between individuals, providers, suppliers and third-party services should be clarified in relation to the relevant architecture. In this respect, the new model enshrined by the GDPR is based on the accountability of all intervening actors throughout the data lifecycle. Beyond the agent, the liability of third parties involved in this ecosystem should be thought over, whether they provide tools or control applications, once it is agreed that technical liability cannot be laid exclusively at the door of the individual, albeit one who has sovereign control over their own data. It must be shared between actors in varying degrees, according to the architectures and each party's level of intervention in the use of the data. These are the conditions to ensure that "strong empowerment" does not ultimately lead to the exclusion of all safeguards provided for by the GDPR to protect individuals in data processing.

[71]GDPR, art. 2(2)(c)

[72]GDPR, recital 18

[73]Such as the data to be processed, the duration of processing, who might have the right to access this data... *See:* WP29, 1/2010, p. 14, EDPS, *Guidelines on the concepts of controller, processor and joint controllership under Regulation (EU) 2018/1725*, Nov. 2019, p. 10, EDPB, *Guidelines 07/2020 on the concepts of controller and processor in the GDPR,* version 1.0, adopted on 2 Sept. 2020, p. 14, §38

[74]WP 29, 1/2010, p. 14: "*determining the means would imply control only when the determination concerns the essential elements of the means.*" See, N. Metallinos *in* A. Debet, J. Massot et N. Metallinos, *op. cit.* spec. n.468, p. 234 stating that services providers generally qualified as data processors might be qualified as data controllers when offering their services to individuals

[75]S. Abiteboul et al., (2015), where the PIMS provider acts on behalf of end-users, while providing security and informing them on options for security and privacy, *see also* EDPB, 07/2020, p. 20, §62 and f. on joint controllership for the provision and use of a tool, *otherwise*, see GDPR, recital 18

[76]EDPB, 07/2020, p. 9, §12

[77]EDPS, *Guidelines on the concepts of controller, processor and joint controllership under Regulation (EU) 2018/1725*, Nov. 2019, p. 9, determination of a sole operation of the processing is enough to be qualified as data controller

[78]*See for instance*: CJEU, Gr. Ch., June 5th, 2018, C-210/16, Wirtschaftsakademie Schleswig-Holstein GmbH, CJEU, July 29th, 2019, C-40/17, Fashion ID GmbH & Co. KG. Summing up these cases: EDPB, 07/2020, p. 19, §§61-66

Limits to individual empowerment: a call for a new vision.

Whether weak (ability to migrate from one service to another) or strong (ability to conduct third-party processing of one's own data with bilateral guarantees), empowerment as promoted heretofore is undeniably a prerequisite in building a new approach ensuring individuals a degree of personal agency over their own data. However, it should be noted that the current purposes of PIMS solutions, as propounded by their editors and considered by academics or associations, essentially entail strictly individual benefits[79]. Moreover, the uses and technologies to create bilateral trust between individuals and a third party remain poorly implemented. Lastly, self-hosting solutions are limited to individuals who are sufficiently concerned about protecting their privacy to acquire the necessary expertise (to the point of playing sorcerer's apprentice) to install and administer their own system. Thus, Personal Big Data nowadays is aimed mainly at users interested in self-tracking and in cross-checking data for their sole benefit. These advantages remain insufficient to trigger widespread adoption. Some proposals have attempted to provide stronger incentives, although they are still based on individual interests. In many cases, these entail enabling individuals to "monetise" their personal data. For example, the start-up Embleema offers a personal cloud system based on blockchain, where individuals can collate their healthcare data from hospital and laboratory sites, from DMPs or connected objects (like Fitbit watches) to monetise access.[80] Over the medium term, the aim is to set up a marketplace for healthcare data, giving stakeholders access to "real-time" data on patients in return for payment. This reopens the debate on whether individuals "own their data" which opposes proponents of liberal analysis[81] and defenders of the right to informational autonomy as the ultimate guarantee of individual freedom in the digital age.[82]

For data empowerment to genuinely take off, we need to broaden the ambitions. Indeed, the power derived by the individual from their personal data remains limited as long as the ability to establish bilateral trust is not provided. This is a necessary (unsatisfied) condition of agency as demonstrated by the first part of this chapter. Moreover, full power over personal data requires the exploitation of the data of many individuals (and not just one), following the example of the personal data exploitation deployed by the major web players which is based on using personal data of millions of individuals. Individual empowerment could be enhanced through a form of collective empowerment[83], to secure social or societal progress surpassing purely self-centred benefits. The following part considers the conditions for collective agency.

[79]Anciaux, N., Bonnet, P., Bouganim, L., Nguyen, B., Pucheral, P., Popa, I. S., & Scerri, G. (2019). Personal data management systems: The security and functionality standpoint. *Information Systems*, *80*, 13-35. See the conclusion of their survey of PIMS solutions in Section 2, where the authors explicitly state that the distributed computations step (i.e., what we call "Big Personal Data" in this chapter) is "currently poorly covered".

[80]See Embleema's "PatientTruth" solution: "Store Data, Share Records, Earn Tokens". (https://www.embleema.com/fr/patienttruth/).

[81]G. Noto la Diega, "Data as Digital Assets. The Case of Targeted Advertising", *in Personal Data in Competition, Consumer Protection and Intellectual Property Law, Towards a Holistic Approach?*, Springer, MPI Studies on Intellectual Property and Competition Law, 28, 2018, pp. 445-495, spec. p. 452

[82]On this debate, see the CNIL activity report for 2017, p. 52.

[83]On this, see also Peugeot, "Brève histoire de l"empowerment : à la reconquête du sens politique", 3 Nov. 2015, http://www.internetactu.net/2015/11/13/breve-histoire-de-lempowerment-a-la-reconquete-du-sens-politique/.

6.3 Drafting collective empowerment based on personal agency

This part is organised in three sections. First, we will appraise the current schemes to access and process vast amounts of personal data (called "Big Personal Data"), in order to conclude that these tend to disregard personal agency. In a second section, we shall analyse potential alternatives, whose aim is to facilitate the collective exercise of portability rights in a regulatory framework, with a nascent sense of personal agency. Finally, in a last section exploring the concept of collective personal agency for "Big Personal Data", we shall introduce a new necessary trust condition, referred to as *mutual trust*, then illustrate our proposal with an example and investigate the basis for a legal and technical framework.

6.3.1 A global race for collective uses: approaches devoid of personal agency

Cross-checking personal data among vast populations has both individual and social advantages in many areas such as healthcare, banking, smart cities, social assistance, etc. This collective use of personal data is based on computation methods often referred to as "Big Personal Data"[84], in that they involve Big Data processing on the personal data of thousands or even millions of individuals. The processes underlying Big Personal Data combine techniques ranging from simple statistical analysis (grouping, aggregation) through automatic information search (automatic classification, rule discovery) to learning (based for instance on neural networks). As noted by the task force "AI for Humanity" in France, led by Cédric Villani, a combination of these techniques and their rapid growth have given rise to fierce competition in the global race for Artificial Intelligence.[85] With data now seen as a "major competitive advantage", "data sharing between private stakeholders has been identified as one of the main levers to catching up with American and Chinese stakeholders, who have the advantage of having access to massive amounts of data".[86] This explains the new ambition at the European level to access huge amounts of data "particularly from major stakeholders, who have a de facto monopoly on the collection of certain categories of data".[87] However, compared to Personal Big Data (see the first part above), Big Personal Data introduces a new difficulty: that of gathering data from large sets of individuals and carrying out the required processing. What are the contemplated scenarios and what is the situation regarding the data subjects' personal agency?

Identified models: B to G and B to B.

Firstly, some advocate an "open model", which involves enshrining the wider concept of "data of general interest", a category of data established in France since the Law for a Digital

[84] In this part of the chapter, we use the term "Big Personal Data" (a Big Data processing using the personal data of a large number of individuals) as opposed to the term "Big Personal Data" (where data of only one individual is involved, see note 58) used in the first part. For a detailed definition of "Big Personal Data", we refer the reader to Section 2 of the paper: McDonagh, M. P. Data Protection in the Age of Big Data: The Challenges Posed by Big Personal Data.

[85] C. Villani, "For a meaningfull Artificial Intelligence. Toward a French and European Strategy", March 2018, p. 25

[86] Villani report, op. cit. p. 27.

[87] COM(2020) 66 final, p. 26 and f.

Republic in 2016.[88] This type of model allows to move forward in opening up private sector data; concurrently, the European Commission has also envisaged arrangements to facilitate access to data held by private companies.[89] From this perspective, the Villani task force recommended gradually opening up datasets from private operators "on a case-by-case basis" and according to the sector "for motives of general interest".[90] This could take place in two different ways:[91]

– the opening up of private data for general interest purposes in favour of public authorities (Business to Government, or "B to G") to help the development of public policies. For instance, mobility data inferred from flows of people or vehicles could be obtained from operators such as Orange, Waze and Uber and processed by the government, in particular to conduct research into reducing road traffic accidents;

– data sharing in favour of other economic stakeholders (Business to Business, or "B to B") for economic purposes such as innovation, research, the development of new services or AI or to boost competition. The banking sector is cited as an example, where Directive PSDP2 requires banking institutions to provide access to their clients' data to encourage the development of innovative businesses ("Fintech").

However, data sharing should be subject to certain conditions: in addition to compliance to the GDPR, the principle of proportionality needs to be respected and the relevant companies' interests must not be adversely affected – which presupposes protecting business secrecy and the possibility to monetise data – and this in turn prohibits to subject such access to compulsory gratuity "for trade between companies for which there would normally be a charge". The principle of transparency must also be respected.

Models under discussion: G to B.

A third form of opening up consists in giving the economic sector access to data currently held and managed by state actors (Government to Business, or "G to B"). For instance, the "Health Data Hub"[92] task force was set up in France to investigate the provision of healthcare datasets held by the State to economic stakeholders. The task force concluded that "healthcare data financed through social welfare is a communal heritage and recommended that "this data should be fully exploited for the benefit of the largest number of people" once they have been matched and documented with metadata to facilitate exploitation.[93] Respect for privacy is based on personal data pseudonymity (where data is stripped of any directly identifying information). To ensure overall economic viability, the proposal also suggests that access to the "hub" could be "charged for private stakeholders in the form of a fixed subscription fee

[88] L. Cytermann, "Le partage des données, un enjeu d'intérêt général à l'ère de l'Intelligence artificielle", *Rev. aff. eur.* 2018, no. 1, p. 65.

[89] "Building a European Data Economy", COM(2017) 09 final and "Towards a common European data space", COM(2018) 232 final. While the recently amended PSI Directive does not impose this opening up, the text nevertheless acknowledges that Member States remain responsible in their decision to apply the requirements of the directive to private companies, in particular those that provide services of general interest (Directive 2019/1024 of 20 June 2019).

[90] Villani report, op. cit. p. 34.

[91] États généraux des nouvelles régulations du numérique, consultation document, 2018, p. 16, https://cnnumerique.fr/files/users/user192/Synthese_EGNUM.pdf.

[92] M. Cuggia, D. Polton, G. Wainrib, S. Combes, "Health-data-hub, Mission de préfiguration", Oct. 2018, https://solidarites-sante.gouv.fr/IMG/pdf/181012_-_rapport_health_data_hub.pdf.

[93] More broadly on data and value created by public sector, see: Commission, COM(2020) 66 final, p. 6

and a variable charge depending on usage".[94]

Limitations: models devoid of personal agency.

Fears could be raised about these three data sharing models (B to G, B to B, G to B). These differ mainly in terms of the public or private nature of the recipient entity, which is in charge of managing the vast amounts of collected personal data. However, most agree on a personal data management model that is centralised and administered by a single entity – beyond the data subjects' control.[95] These models are comparable to the current cloud solutions, criticised for the issues they raise in terms of security, privacy and informational asymmetry as regards the individuals involved. Thus, the sophistication and frequency of cyberattacks increases alongside the rising volume of data that could potentially be disclosed. There is a further risk of re-identification if the data anonymisation techniques are too weak to provide appropriate protection; this risk is proven as regards pseudonymity, which is no longer considered an adequate anonymisation technique.[96] Moreover, the potential for secondary uses which are inconsistent with the initial purposes depends exclusively on the trust placed in the centralising entity and on all its providers. None of these approaches however contemplates obtaining the data subjects' consent. Access to Big Personal Data therefore seems focused on simply transposing the existing controversial method of managing personal data, to the detriment of any form of personal agency for individuals, even in the case of sensitive healthcare data.

6.3.2 Alternatives ensuring a form of personal agency

Since the aforementioned scenarios are devoid of personal agency, we shall examine some promising alternative proposals under discussion.

Collective portability as a condition of personal agency.

Alternative proposals providing individuals with the means to collectively control the use of their personal data are being encouraged, in line with the development of "civic portability". The Villani task force has indeed suggested extending portability rights from an individual to a collective prerogative, particularly as regards AI.[97] Thus, groups of citizens sharing common values and willing to act collectively (on the model of class actions), could exercise their portability rights and share their data with a public authority for a specific purpose, related to a public service mission. In the field of healthcare for instance, patients could make their medical data available to a research institute to improve the detection or treatment of a pathology. The objective here would be to enable the creation of new databases in favour

[94] M. Cuggia et al, *op. cit.* p. 4

[95] Cuggia, M., Combes, S. (2019). The French health data hub and the German medical informatics initiatives: two national projects to promote data sharing in healthcare. Yearbook of medical informatics, 28(1), 195.

[96] For instance, a recent article shows that a handful of attributes containing demographic information is sufficient to unambiguously identify almost 100% of (American) individuals in any dataset: L. Rocher et al., "Estimating the success of re-identifications in incomplete datasets using generative models", Nature Communications, 10: 3069 (2019).

[97] Villani report, op. cit. p. 37.

of public services by allowing the free movement of data "under the exclusive control of citizens".[98]

There again, portability could act as the cornerstone of this initiative, giving each individual the capacity to consent to processing and even to secure collective processing, thereby upholding another form of agency. This invites reflection on possible collective portability and empowerment. Meanwhile, some authors focus on the definition of group privacy and data protection[99], notably by the expansion of data groups[100]. These emerging theories highlight the need for data protection law to broaden its scope of application, taking into account its collective aspect[101].

Employment law or data trusts as insufficient attempts to ensure personal agency.

Insofar as combining individual portability initiatives is insufficient to enable individuals to jointly orchestrate the uses resulting from the collection of their personal data, it seems that portability on its own cannot guarantee collective agency. One must therefore analyse how data subjects may be empowered to conduct collective processing under their control.

A first solution would be to incorporate collective "civic" portability within the existing legal framework. Thus, since the relevant personal data may result from an individual's labour (for example, data from Uber drivers), one could foresee overlapping analogies between employment law and data protection law. This gives rise to new ideas:[102] terms of use negotiated along the lines of collective agreements, a collective portability exercised within associations or trade unions[103]. Another solution could induce reconsidering the personal data governance model, given that "consent-based models of data governance fail to protect the public against privacy violations and the unethical collection and use of personal data".[104] Some authors have explored the implementation of new governance based on data trusts, and investigated other ways of regulating data usage. Such work however reached mixed conclusions, in that such control on data processing is ultimately based on the trust individuals place in their fiduciaries.[105]

Despite their shortcomings, these studies reinforce the argument that the individuals' personal agency in data processing is closely linked to the ensuing empowerment prospects. The next section addresses the ways in which a group of individuals may be enabled to

[98] *Ibid.* – See also CNIL, "La plateforme d'une ville. Les données personnelles au cœur de la fabrique de la smart city", *Cahiers IP* 2018, no. 5, p. 48.

[99] B. Mittelstadt, "From Individual to Group Privacy in Big Data Analytics", *Philos. Technol.* 2017, 30, p. 475-494.

[100] U. Pagallo, « The Group, the Private, and the Individual : a New Level of Data Protection ? », *in* L. Taylor, L. Floridi, B. van der Sloot (*Eds.*), *Group Privacy: New Challenges of Data Technologies*, Dordrecht, Springer, 2017

[101] See: N. Purtova, "Do Property Rights in Personal Data Make Sense after the Big Data Turn? Individual control and Transparency", *Tilburg Law School, Legal Studies Research Paper Series*, n.21/2017,, spec. p. 17 "personal data cannot be considered as concerning just an individual anymore; data processing resulting from a decision of one person will inevitably have spill-over effects on others (. . .)".

[102] L. Maurel and L. Aufrère, "Pour une protection sociale des données personnelles", 5 Feb. 2018, https://scinfolex.com/2018/02/05/pour-une-protection-sociale-des-donnees-personnelles.

[103] See L. Taylor, L. Floridi, B. van der Sloot (*Eds.*), *Group Privacy: New Challenges of Data Technologies*, Dordrecht, Springer, 2017.

[104] *Data Trusts. A new tool for data governance*, ElementAI and Nesta, 2018, p. 30.

[105] See also: T. Hardjono, A. Pentland, "Data Cooperatives: Towards a Foundation for Decentralized Personal Data Management", available at: https://arxiv.org/pdf/1905.08819.pdf

implement and control all the effects of a Big Personal Data processing, and which legal and technical framework is to be promoted.

6.3.3 Towards strong empowerment safeguarding personal agency for Big Personal Data

In this section, we shall analyse two (extreme) scenarios and their perspectives on personal agency. We will then introduce a new notion of *mutual trust* as a component of personal agency in the context of Big Personal Data, as illustrated by an example. We shall then discuss the underlying issues, in order to establish a new technical and legal framework for personal agency in this context.

The issue of personal agency in collective computations.

Big Personal Data processing by a large set of individuals can be led according to various technical scenarios, with different perspectives in terms of personal agency. There are two different methods: on one hand, the centralised approach, which consists in bringing all the data to one entity for processing; on the other hand, the decentralised approach, where each contributing individual is treated as an autonomous entity, capable of interacting with all the others to operate the processing together. Although many technical solutions exist halfway between these two extremes, analysis of the latter allows to identify different prospects regarding personal agency.

The first approach requires a centralised controller, governed by a third party entity, to administer the digital environment in which the computations are to be performed. The effect on the appropriate security measures is colossal since the benefits of an attack on this centralised entity are very high (access to the personal data of millions of individuals). In addition, the trustworthiness of the central entity is key to avoid secondary use of the data. Personal agency thus resides solely in the trust individuals consent to place in a third party entity.

The second approach does not introduce a centralised control point. Instead, each individual can be seen as a computation node that bears responsibility for part of the processing. Accordingly, their control of each node gives individuals a role as an agent of the computation agent. This approach however poses distinct risks to individuals. Firstly, by its very nature, computation implies an exchange of personal data between participants, thus transforming each of them into a potential attacker. Secondly, the external infrastructure supporting the data exchanges (for example, internet gateways) can observe some of these exchanges. Lastly, the data processed at each node and data exchanges between nodes can neither be defined nor even understood or administered by a non-expert individual (without a specific framework). Thus, this approach offers a new perspective on personal agency, but also presupposes the definition of a legal and technical framework that allows individuals to exercise their rights freely.

Personal agency for Big Personal Data: mutual trust.

Elevating individuals to agents in terms of Big Personal Data consists in enabling them to decide (for example through consenting) whether to contribute to such a processing with their own personal data. It also means providing assurances to all data subjects involved that the processing is conducted in line with the stated purpose, with integrity and confidentiality.

While personal agency relating to Personal Big Data establishes bilateral trust between an individual and a third party, each individual agent in Big Personal Data processings must be able to establish ***mutual trust*** between all participating individuals and the third party entity to which the results are sent. On one hand, each individual must have a guarantee that their own data cannot be disclosed during processing and that all the other participants will act in a trustworthy manner and implement the processing as expected, in accordance with what each has been consented to. If one of the agents noticed a failure or violation, no result should ever be produced. Conversely, the recipient of the final result must be assured that it complies with the expected processing, based on the right dataset from the stated number of participants. We illustrate this mutual trust condition through Example 7, which stages a collective of parents computing statistics based on the online gaming data of their children.

Example 7. ***(Big Personal Data to help parents reduce their children's addiction to video games.)***

More and more parents are worried about their children's addiction to online video games, such as free-to-play cooperative multiplayer "Battle Royale" games. These concerns are justified when video games companies have at their disposal petabytes of data used to feed Big Data algorithms to make the games as addictive as possible. Indeed, the main source of income for this category of games (free-to-play) are in-game purchases and events, which explains the publishers' willingness to maximise the time that millions of users spend playing. Confronted to this issue, parents may feel powerless. They can either prevent their children from playing games at the risk of isolating them, or do nothing and let them sink into addiction.

Big Data for the citizen: an ability to explore 'anti-toxic' conditions. A reasonable solution would be to analyse the playing habits of children populations, to help determine the attitude parents could adopt when their child seems to develop an addiction. Thus, just as games editors use Big Data to quantify the impact of new game features on increasing children's playtime, parents should be empowered with Big Data means to collectively help defining better conditions to prevent children from being addicted.

Mutual trust as a necessary condition for "Personal agency". The notion of personal agency introduced here aims to empower willing parents to jointly define and perform Big Data computations for their collective benefit. Making parents "agents" of such collective computations requires providing them with *mutual trust*, by means of three new capabilities:

1. First, the ability to ensure that the children's personal data will remain confidential. This is a prerequisite to convince parents to supply children's data in the computation and trigger broad adoption.

2. Second, the capacity to attest that the final result was indeed computed on the expected data, with the agreed code and the appropriate number of participants. This is a necessary condition if the result is to serve as a basis for future decisions and recommendations.

3. Third, the capability to ensure compliance with the legal basis for processing, the legitimate obtaining of the relevant personal data through the exercise of

the data portability right, informed consent, with clear statements concerning purpose, minimal personal data collection and no further use of personal data.

The Manifest-based framework : a new legal-technical solution operated in three phases.

Step 1: Formulate a hypothesis to be checked. Consider a collective (or association) of parents of young players aiming to reduce their children's playtime. For example, one could allow children to play more frequently but for a shorter time, limit the amount of games instead of the playtime, provide a fixed amount of money to be spent in the game (rather than let the children "win" it in the game), organise collective sessions (e.g., with remote classmates) rather than playing alone, block videos related to these games with parental control on Youtube, etc. Would some strategies overcome others in terms of reducing playtime in the long run?

Step 2: Express a Manifest for the collective computation.. To test some of these options, the parents may express a Manifest, which is both a set of rules describing the computation and a formulation of the legal basis for the considered processing. The manifest can act as a contract, drafted between all the participants, giving their consent to the collection and processing of personal data, and to the random attribution of an 'agent' role in the computation, such as data collector or data aggregator. The obligations can be deferred until the realisation of a future condition (e.g., reaching the required threshold for the processing to start). This manifest must be validated by a regulatory body (e.g. CNIL in France) which certifies its compliance with privacy laws. This certified manifest is then published so that parents who wish to participate can download it and give their consent.

Step 3: Execution of the Manifest. This phase starts when the number of consenting parents reaches the threshold specified in the manifest. Any participating parent is endowed with the aforementioned three capacities. From a technical standpoint, the condition to enforce these abilities in recent proposals is that the participating parents' personal computer is equipped with a processor implementing 'trusted execution environments' in hardware (which is the case for recent computers endowed with Intel or AMD processors).

Rethinking a legal and technical framework to secure collective agency.

To achieve a generalisation of Big Personal Data, a framework needs to be defined, firstly to support the essential elements of personal agency and secondly to avert the risks of privacy breach as well as damage to the integrity of the computation. Among the relevant issues, the first one is whether the regulation forbids individuals to collectively use the personal data they have recovered pursuant to the exercise of their portability right. If the GDPR allows such Big Personal Data processing[106], the second issue is on its relevant legal basis. In our view, the choice of consent as a legal basis for inter-individual processing is the best option, insofar as consent is the only legal basis empowering and providing agency to individual. The conditions for consent to be lawful in this context supports this point: the data subject must be "*offered*

[106] J. Belo, P. Macedo Alves, "The right to data portability: an in-depth look", *Journal of Data Protection & Privacy* 2018, vol. 2, 1, 53-61, spec. p. 55

control and (be) *offered a genuine choice with regard to accepting or declining the terms offered or declining them without detriment*"[107]. Consent is the only legal basis allowing individuals to have a granular control over which data is being processed; it may also be retracted at any point, along with the data processed under such legal basis. The regulation requires consent[108] to be freely given, specific, informed and unambiguous, by a clear affirmative act[109]. It could be expressed for example by means of a manifest stipulating the type of Big Personal Data processing to be performed, where the purpose, the collected data, the computation code distributed to the participants, the result produced and the recipient entity are laid out, as well as the minimum number of participants required to achieve a useful result. Next, the manifest would need to be verified and validated by a regulatory authority (such as the CNIL, the French data protection authority, or ANSSI, the French national cybersecurity agency). In addition to validating each case, the issue for the regulator is whether to draw up data collection clauses clarifying the types of algorithms implemented and the permissible output. Once approved, the manifest would be published and made available to adequate groups of people, who could then decide whether or not to sign up. The regulator's endorsement would provide various guarantees ensuring respect for their personal data, securing their personal agency and, in time, could give rise to the drafting of a sectoral code of conduct to delineate the responsibilities and undertakings of stakeholders in Big Personal Data. Finally, a secure mechanism to ensure the agency of participants should be able to allocate the processing across all participants and execute it, without deviating from the manifest or revealing any data other than the final result, thus safeguarding the mutual trust outlined above.

A realistic objective in the current state of technology.

The conventional computation techniques used in business systems cannot be applied here due to the unusual scale of distribution (the computation could in theory encompass a fraction of the population of a country). Some secure distributed computation protocols based on cryptographic techniques (called "secure multiparty computation") could be used in some cases but cannot yet perform satisfactorily if extended to support generic computations for a large number of participants.

However, new technologies are currently being developed and use trusted computing hardware – which one is usually already equipped with – to set up generic secure distributed processing on a large scale. Most smartphones and PCs belonging to individuals now have secure processors such as Intel SGX, ARM Trustzone, AMD PSP, etc. A recent study[110] shows that the concept of mutual trust as defined above is compatible with these types of hardware.

6.4 Conclusion

Summary of the chapter.

In this chapter, we have showed that the notion of personal agency, as set out by social sciences, can be transposed to the case of personal data processing in the Big Data context.

[107] WP29, *Guidelines on consent under Regulation 2016/679*, As last revised and Adopted on 10 April 2018, WP 259 rev.01

[108] GDPR, art. 6(1)(a)

[109] GDPR, recital. 32

[110] Ladjel, R., Anciaux, N., Pucheral, P., & Scerri, G. (2019). Trustworthy distributed computations on personal data using trusted execution environments. *TrustCom/BigDataSE* 2019, pp. 381-388.

We provided a general definition of personal agency, which offers a new angle to analyse the proposed approaches for personal data processing operations. Existing approaches have been explored in the light of this definition in the cases of "Personal Big Data" and "Big Personal Data". In both situations, this leads to the formulation of new necessary conditions related to the degree of trust that individuals must be able to provide to each other to be considered as "agents" of the processing. In the case of "Personal Big Data", where a single individual provides results of data processing to third parties, a *bilateral trust* condition must be established. In the case of "Big personal data", where each individual becomes part of a collective in order to extract aggregate results from their personal data (pooling), a condition of *mutual trust* is required. We therefore outlined a preliminary proposal for a legal-technical co-construction illustrated by examples, which reflect the feasibility of these conditions in the current state of technology, and discuss related challenges which remain to be addressed.

Conditions remaining to be resolved.

Of course, many details will still need to be ironed out. Some technical building blocks demonstrating the feasibility of such a solution remain to be established. Many other open-ended questions may be raised: should user consent be set up for each processing operation or for a group of processing operations? How can it be formally demonstrated that one can withstand a small number of users who tamper with their hardware to attack security features – though the hardware security technologies are difficult to attack, vulnerabilities can always arise in an environment where security amounts to a race between hackers and manufacturers? How should the mechanisms described be integrated in a real operating system, and more particularly within existing PIMS products? Combining the circulation of huge amounts of data with informational sovereignty for each individual means that they should be seen as agents of the ecosystem currently being set up. A new structure therefore needs to be built to ensure full personal agency, with underlying mutual guarantees for individuals and for the entire ecosystem in which they operate. The terms still need to be adjusted but this new way must be explored to avoid individuals to be seen as mere datafied objects in the future.[111]

[111]This contribution is the fruit of interdisciplinary discussions conducted as part of the *GDP-ERE* project financed by *Data IA* Convergence Institute of the University of Paris Saclay and the ANR *Perso Cloud*.

Chapter 7

Conclusion

The growth of data leakage scandals and massive surveillance revelations over the last decade have highlighted the limits of the current web model. Despite data protection laws (e.g. GDPR) and the development in parallel of PDMS solutions that allow individuals to retrieve, to store and to manage all their digital content in the same place and under their control, the completely decentralized model struggles to establish itself. The main reasons are the limitations of these solutions, which, despite the addition of new value-added services (crossing data of multiple sources), sacrifice the collective usage of personal data (crossing data of multiple individuals). Our solution is a first attempt to provide a framework to compute over a set of decentralized PDMSs while guaranteeing the integrity of computations and the confidentiality of data. The remaining of this chapter is organized as follows: first we will summarize our contributions and then give some perspectives for future work.

7.1 Summary of the Contributions

First, we proposed a manifest-based framework leveraging the properties and the omnipresence of TEEs to compute generic functions over a large number of possibly untrusted users, holding their data in a PDMS. Our framework establishes a *mutual trust* between users and the Querier, and ensures the honesty of the computation even in the presence of malicious participants. It provides, for each participant and the Querier, a solution to *locally* check that the protocol has indeed been honestly executed, without resorting to any trusted third party. Our solution includes accurate counter-measures against malicious participants who managed to break the confidentiality of their TEE enabled device through side-channels attacks. We finally showed the effectiveness and the security of the solution through a qualitative and quantitative evaluation on two practical use-cases.

Second, we proposed a solution to control data-dependant communications with (ϵ, δ)-Differential privacy guarantees. Our solution preserves the accuracy of the final result with a low overhead compared to traditional solutions.

Third, we introduced the concept of *Trusted Personal Data Management Systems (TPDMS)* and demonstrated the practicality of our solution through an ongoing deployment of the technology in the medical-social field over 10k patients receiving care at home. We showed that our solution is agnostic to the platform and can be run with acceptable overhead even in constrained environments. We then evaluated it in terms of security, performance and societal impact.

Finally, we formalized the concept of *Personal Agency* in the personal cloud context in collaboration with a team of lawyers. We studied to which extent the *Personal Agency* is achieved in both centralized and decentralized settings with current solutions. We finally demonstrated that our solution provides strong *Personal Agency* with a concrete example.

7.2 Perspectives

In an era where the world is becoming entirely connected, where data are produced and processed massively, giving birth to new value-added big data services based on crossing data of millions of users, we believe that our work may initiate new ways to think about such services. Indeed, the current web model is moving away from a completely centralised setting where users have the choice between taking advantages of the services provided by big companies or protecting their privacy, to a model where the Personal Cloud empowers users and gives them control over their personal data but sacrifices the utility of cross-user computations. In this context, providing solutions to compute over completely decentralized Personal Clouds without resorting to a third party while giving strong guarantees of integrity and confidentiality is of utmost interest. Our solution is a first step in this direction and can be extended in the following aspects.

- ***Big data for citizens:*** an important prerequisite to help individuals emancipate themselves from captive ecosystems proposed by big companies is to give them the means to run the same algorithms used by those companies over their personal data. Our solution is a starting point in this direction. However, it is utopian to expect non-expert users to write specific manifests that implement complex queries or computations. The solution to overcome this problem is to build libraries that implement the basic operations used in big data algorithms. Such libraries will contain certified and formally proven building blocks that may be used to express complex queries. The main difficulties are (i) to identify the basic operations needed to implement more complex algorithms, (ii) to implement and formally prove these basic operations and (iii) to ensure that writing queries using a composition of these basic operations will not compromise the integrity and the confidentiality properties.
- ***Hiding the communication patterns:*** in Chapter 4 we proposed a mechanism to hide the communication patterns, the perspectives for this contribution are twofold:
  - *Apply this mechanism to our solution.* Indeed, the next step is to find the best way to integrate this mechanism to our Manifest-Based framework. The main difficulties are (i) to study the impact of this addition on the manifest itself (i.e what needs to be modified in the manifest) and (ii) to propose a protocol that automatically identifies the data dependant patterns and adds the correct amount of scramblers to protect the computation.
  - *Increase the privacy.* The other perspective is to optimize the amplification. Two directions may be investigated (i) pushing the addition of dummies to the local randomizer to protect the sources against malicious/compromised scramblers and (ii) studying the impact on privacy, when having an attacker with limited prior knowledge.

.

- ***Deployment of an operational platform:*** we showed in Chapter 5 that our solution can be adapted to run over the THPC platform. The next step is to deploy the solution for real world use. However, as discussed before, the hardware boxes in the THPC platform are connected through SigFox or GPRS. Moreover, the boxes are weakly connected (i.e. there is no direct connection between the nodes of the network). To communicate, the nodes need to send their encrypted messages to a central entity that relays them to the recipient. To move toward an effective solution over completely decentralized PDMSs, the solution has to be adapted to be DTN[1] compliant, where no central entity is implied in the computations. The main difficulties are (i) to adapt the random assignment protocol such that the collection and computation tasks are distributed randomly over the participants, (ii) to propose a fault tolerant protocol suitable for the DTN context that reduces the communication overhead and (iii) to organize the computations such that the guarantees provided by the solution still hold in the DTN context. A thesis on this subject started in January 2020 and is conducted by Ludovic Javet.

[1] Delay-tolerant network

Annexe A

Résumé en Français du manuscrit

Qu'elles proviennent de smartphones, d'appareils connectés, de capteurs ou de compteurs intelligents, la quantité des données générées et échangées quotidiennement croît de manière exponentielle. En 2018, 33 Zettaoctets de données ont été produites dans le monde entier. Ce volume de données extrêmement important continue de croître chaque jour. L'International Data Corporation (IDC) prévoit que ce nombre devrait atteindre 175 Zettaoctets en 2025 [111]. Avec un revenu généré estimé à 203 milliards d'euros en 2020, cette grande quantité de données économiquement précieuses est, sans surprise, une mine d'or pour les personnes qui les détiennent. Le Forum économique mondial les compare au "nouveau pétrole" [59].

Traditionnellement, ces données sont collectées et stockées dans des serveurs centralisés détenus par de grandes entreprises (Google, Amazon, Facebook, compagnies d'assurance, etc.). Cette collecte et cette centralisation massives de données permettent le croisement des données de millions d'utilisateurs. Grâce aux algorithmes très efficaces développés au cours des dernières décennies, allant de la simple analyse statistique (regroupements, agrégation) et de la recherche automatique d'informations (classification automatique, découverte de règles) à l'apprentissage (basé par exemple sur les réseaux de neurones), les entreprises sont désormais en mesure de proposer des services sur mesure directement inspirés du comportement des utilisateurs, ce qui augmente la productivité, l'ergonomie et l'utilité. Ainsi, le croisement de données provenant de plusieurs individus est d'un intérêt personnel et sociétal majeur.

Malheureusement, ces derniers temps, ce modèle traditionnel a montré ses limites. En effet, la centralisation souffre de nombreux inconvénients. La sensibilisation du public aux dangers que représente le monopole des données orchestré par les géants du Web a commencé en 2013 lorsque le lanceur d'alerte Edward Snowden a révélé l'un des plus grands scandales du 21e siècle [107]. Snowden a révélé que le gouvernement américain, par le biais de ses agences de renseignement, effectuait une surveillance massive des individus avec la complicité des détenteurs des données. Cependant, ce n'est pas le seul problème dont souffre la centralisation. En 2017, un rapport publié par Cracked Labs [39] révèle comment les différentes sociétés du web partagent et mettent en commun les données personnelles de leurs utilisateurs collectées directement ou indirectement et comment cette quantité astronomique de données est utilisée pour créer des profils extrêmement précis contenant des informations personnelles sensibles et intrusives de millions d'individus. Le résultat de ce profilage massif est la manipulation des individus qui peut aller de la simple influence sur leurs habitudes d'achat, à des questions plus préoccupantes comme la manipulation de l'opinion publique en allant même jusqu'à influencer les résultats d'une élection. C'est typiquement le cas des élections américaines de 2016, où

le scandale de Cambridge Analytica [34] a révélé comment les élections ont été influencées après avoir analysé les profils de millions d'utilisateurs de Facebook et utilisé les informations apprises pour influencer le vote des individus ciblés.

De plus, les failles dans la protection des données sont un autre élément qui mine le modèle centralisé. Qu'elles soient intentionnelles (utilisation abusive, attaque malveillante) ou simplement dues à la négligence (fuite de données, mauvaise gestion), ces failles entraînent la fuite d'une grande quantité de données. Et leur nombre augmente de plus en plus. En effet, une attaque contre un serveur contenant des millions de données représente une grande victoire pour les attaquants car le rapport coût-bénéfice est très élevé. Parmi les milliers de fuites annuelles, on peut citer celle de Facebook, qui en 2019 a exposé 540 millions d'enregistrements d'utilisateurs sur les serveurs cloud d'Amazon en raison d'une sécurité insuffisante [119]. La même année, Microsoft a accidentellement exposé 250 millions d'enregistrements [30]. Le record absolu est détenu par Yahoo qui a subi une attaque, à partir de 2013, qui a exposé 3 milliards de comptes d'utilisateurss [105].

Le résultat de cette situation est que les utilisateurs perdent le contrôle de leurs propres données. Ces menaces soulignent la nécessité de disposer de plateformes personnelles qui permettent à leurs utilisateurs de collecter, de gérer et de partager leurs propres données. Pour toutes ces raisons, de nombreuses voix s'élèvent pour demander une révision de l'architecture actuelle du Web, y compris celle du fondateur du Web lui-même. En 2018, Tim Berners Lee a publié une lettre ouverte [24] dénonçant le monopole de quelques grandes sociétés sur la collecte de données personnelles, il dit notamment que « le Web a évolué en un moteur d'iniquité et de division, influencé par des forces puissantes qui l'utilisent pour leurs propres objectifs ». Grâce à des initiatives de divulgation intelligente, le nouveau web qu'il décrit dans sa lettre ouverte n'est plus un rêve ou une utopie impossible à atteindre.

Le programme de divulgation intelligente a débuté en 2010 avec l'initiative « Blue Button » qui permet aux patients de télécharger leurs données de santé personnelles en cliquant simplement sur un « bouton bleu ». L'ancien président des États-Unis, Barack Obama, a déclaré en septembre 2011, lors de l'ouverture d'un partenariat public-privé à New York : « Nous avons développé de nouveaux outils appelés "smart disclosures" afin que les données que nous rendons publiques puissent aider les gens à faire des choix en matière de santé, aider les petites entreprises à innover et aider les scientifiques à réaliser de nouvelles avancées » [98]. L'initiative du « Blue Button » a connu un tel succès qu'elle a ouvert la voie à d'autres initiatives comme le « Green Button » pour les données personnelles sur la consommation d'énergie et le « Red Button » pour les données personnelles sur l'éducation. Des initiatives semblables ont été proposées en Europe, d'abord au niveau national pour chaque pays, comme MiData [90] (données sur l'énergie, les finances, les télécommunications et le commerce de détail) en Grande-Bretagne ou MesInfos [89] en France, puis à un niveau plus large, au sein de l'Union européenne, avec le règlement général sur la protection des données (RGDP) [99] et en particulier sa prérogative en matière de portabilité des données. La portabilité des données permet aux utilisateurs d'accéder à leurs données personnelles auprès des entreprises ou des agences gouvernementales qui les ont collectées. Dans le journal officiel français, la portabilité des données est définie comme « la personne concernée a le droit de recevoir les données à caractère personnel la concernant qu'elle a fournies à un responsable du traitement, dans un format structuré, communément utilisé et lisible par machine et a le droit de transmettre ces données à un autre responsable du traitement sans entrave de la part du responsable du traitement auquel les données à caractère personnel ont été fournies ». Il s'agit clairement d'un grand pas en avant pour redonner aux utilisateurs le contrôle de leurs données à caractère

personnel et leur permettre de s'exprimer. Mais cela ne suffit pas pour aider les utilisateurs à s'échapper d'un écosystème captif. En effet, les utilisateurs ont besoin d'une solution technique qui leur permette de stocker, de gérer, de partager et d'exploiter ces données récupérées. C'est exactement ce que proposent les systèmes de gestion des données personnelles, également appelés « Cloud personnels ».

Les solutions de systèmes de gestion des données personnelles sont en plein essor. Leur objectif est de permettre aux utilisateurs de tirer parti de leurs données personnelles pour leur propre bien. Elles permettent aux individus de stocker tout leur environnement numérique au même endroit. Cela ouvre la voie à de nouveaux services à valeur ajoutée qui n'étaient pas possibles avec le modèle centralisé. En effet, les utilisateurs sont désormais en mesure de croiser leurs données collectées à partir de différentes sources (par exemple, croiser les relevés bancaires avec l'historique des achats ou les dossiers médicaux avec les données des montres connectées, etc.). Les différentes solutions de systèmes de gestion des données personnelles seront examinées en détail dans le chapitre 2.

Alors que le stockage de données, auparavant dispersées dans différents silos, dans des systèmes de gestion des données personnelles augmente le contrôle de l'utilisateur sur celles-ci, les utilisations collaboratives des données sont souvent négligées dans ce contexte. Cependant, comme indiqué ci-dessus, les avantages tirés du croisement de données appartenant à plusieurs personnes sont considérables et présentent des avantages à la fois personnels et sociaux dans de nombreux domaines (santé, banque, villes intelligentes, etc.). Par exemple, le calcul de statistiques pour une étude épidémiologique ou sociologique, l'entrainement d'un réseau de neurones pour organiser les écritures bancaires en catégories. Un utilisateur peut vouloir partager sa position GPS pour avoir une prévision précise du trafic [84], ou son dossier médical pour entrainer un réseau de neurones partagé afin qu'il puisse détecter plusieurs maladies [42, 103]. Il peut également vouloir adapter son contrat d'énergie en fonction de sa consommation réelle sans compromettre sa vie privée [92]. Une approche naïve de ce problème consiste à envoyer des données personnelles à une tierce partie de confiance qui effectuera lesdits calculs collaboratifs. Mais comme indiqué ci-dessus, l'hypothèse d'un « tiers de confiance » est forte et irréaliste compte tenu de toutes les menaces qui pèsent sur le modèle centralisé. De plus, envoyer des données personnelles à un tiers signifie perdre le contrôle sur celles-ci et donc renoncer à l'un des principaux avantages du modèle décentralisé.

L'objectif de cette thèse est de lever cette hypothèse de confiance irréaliste et de proposer un Framework qui permet le croisement des données personnelles de plusieurs individus et leur assure la souveraineté sur leurs données et la capacité de faire des choix informés et indépendants. Cela soulève deux questions importantes mais difficile à répondre dans un contexte décentralisé :

1. *Comment convaincre les utilisateurs d'engager leurs données dans un calcul distribué qu'ils ne peuvent pas contrôler ?*

2. *Comment garantir l'intégrité d'un calcul effectué par un très grand nombre de participants potentiellement malveillants ?*

Pour répondre à ces questions, il est nécessaire d'établir une confiance mutuelle entre toutes les parties impliquées dans un calcul distribué. D'une part, tout participant doit obtenir la garantie que seules les données requises par le calcul sont collectées et que seul le résultat final du calcul auquel il consent à contribuer est divulgué (c'est-à-dire qu'aucune donnée brute n'est divulguée). D'autre part, l'initiateur du calcul doit obtenir la garantie que le résultat final a

été honnêtement calculé, avec le bon code et sur des données réelles. De plus, pour avoir un intérêt pratique, le Framework doit :

- être **générique**, ce qui signifie être capable de calculer des fonctions arbitraires, allant de simples statistiques à des algorithmes complexes d'apprentissage automatique.
- **passer à l'échelle** c'est-à-dire qu'il doit pouvoir être appliqué à un grand nombre de participants.

Les contributions de cette thèse sont les suivantes :

1. Nous proposons un Framework qui permet de faire tout type de calcul de manière sécurisée sur un ensemble de systèmes de gestion des données personnelles décentralisés, même pour un très grand nombre d'utilisateurs tout en répondant aux deux questions initiales. Le participant obtient l'assurance que ses données sont utilisées pour la finalité à laquelle il consent et que seul le résultat final est divulgué et que ce résultat a été honnêtement calculé.
2. Dans les calculs distribués, le flux des communications dépend souvent des données pour des raisons d'efficacité (distribution des données sur une valeur de hachage, calcul de barycentres dans les algorithmes de clustering...). Mais ce flux peut révéler des informations sensibles [100]. Pour résoudre ce problème, nous proposons une solution pour contrôler la dépendance des communications aux données sans nuire à l'intégrité et l'exactitude du résultat final. Nous quantifions formellement le niveau de confidentialité fourni par notre solution.
3. Nous proposons une adaptation de notre solution dans le domaine médico-social pour une architecture existante en prenant en compte les contraintes liées à l'architecture. Nous démontrons la praticabilité de notre solution à travers un mélange entre simulation et mesures réelles. Nous évaluons notre solution en termes de sécurité, de performance et d'impact sociétal.
4. Nous définissons et formalisons l'agentivité personnelle un produit des sciences sociales qui constitue la base de l'autonomisation individuelle, dans le contexte du cloud personnel et nous analysons dans quelle mesure l'agentivité personnelle est réalisée dans les modèles actuels. Enfin, nous montrons en quoi notre Framework satisfait certaines conditions de l'agentivité.

Ce manuscrit est divisé en sept chapitres :
L'introduction présente notre travail et donne le contexte général.
Le chapitre 2 introduit les concepts nécessaires pour comprendre les apports de la thèse et les positionner par rapport à l'état de l'art. Dans un premier temps, nous dresserons un panorama des différentes familles de systèmes de gestion des données personnelles et montrerons pourquoi les solutions actuelles ne peuvent pas répondre à nos objectifs, notamment la capacité à effectuer des calculs croisant les données de plusieurs individus. Nous étudierons ensuite les différentes techniques existantes qui sont utilisées dans la littérature pour effectuer des calculs distribués et évaluerons la possibilité de les appliquer à notre contexte. Enfin, nous présenterons le troisième sujet lié à notre travail, l'utilisation de matériel sécurisé pour effectuer des calculs dans un contexte de base de données.

Dans le chapitre 3 nous définissons et formalisons le problème que nous traitons. Nous proposons ensuite un Framework qui répond à tous les objectifs ci-dessus, en faisant l'hypothèse que le flux de communication est indépendant des données. Enfin, nous évaluerons l'efficacité du Framework et sa sécurité. Ce chapitre est basé sur un article [77] publié et présenté à TrustCom/BigDataSE[1] en 2019 et présenté à APVP'19[2] et BDA'19[3].

Dans le chapitre 4 nous proposons un algorithme qui permet de contrôler la dépendance du flux des communications aux données et ainsi lever l'hypothèse de l'anonymat des communications introduite dans chapitre précédent. Nous prouvons formellement la robustesse de la solution proposée contre des attaquants capables d'observer tous les flux de communication et de montrer que la fuite d'informations est négligeable, même si l'attaquant connaît les données de tous les participants sauf un, la probabilité qu'il puisse deviner la donnée du dernier participant est faible.

Dans le chapitre 5 nous présentons une adaptation de notre Framework dans le domaine médico-social pour une architecture réelle en cours de déploiement sur le territoire des Yvelines en France. Nous évaluons la praticabilité et l'adaptabilité du Framework même dans des environnements avec de fortes contraintes (bande passante limitée, participants faiblement connectés...). Ce chapitre est basé sur un article [76] publié et présenté à ISD[4] en 2019 et [78] qui a été publié dans TLDKS journal volume XLIV[5].

Le chapitre 6 présente un travail réalisé en collaboration avec des juristes. Nous posons les bases juridiques et techniques d'une portabilité collective et montrons comment notre Framework permet d'atteindre certaines de ces propriétés. Ce chapitre est basé sur un travail réalisé en collaboration avec des juristes et publié dans la Global Privacy Law Review[6] .

Enfin, le chapitre 7 conclut cette thèse en résumant les principales contributions et en donnant quelques orientations intéressantes pour les travaux futurs.

[1] https ://forumpoint2.eventsair.com/QuickEventWebsitePortal/trustcom19/tc19
[2] https ://project.inria.fr/apvp2019/programme/
[3] https ://bda.liris.cnrs.fr/
[4] https ://isd2019.isen.fr/
[5] https ://www.irit.fr/tldks/volumes/
[6] http ://www.kluwerlaw.com/journals/global-privacy-law-review/

Bibliographie

[1] Serge Abiteboul, Benjamin André, and Daniel Kaplan. Managing your digital life. *Commun. ACM*, 58(5) :32–35, 2015.

[2] Abbas Acar, Hidayet Aksu, A. Selcuk Uluagac, and Mauro Conti. A survey on homomorphic encryption schemes : Theory and implementation. *ACM Comput. Surv.*, 51(4) :79 :1–79 :35, 2018.

[3] Hany Alashwal, Mohamed El Halaby, Jacob J Crouse, Areeg Abdalla, and Ahmed A Moustafa. The application of unsupervised clustering methods to alzheimer's disease. *Frontiers in computational neuroscience*, 13, 2019.

[4] T. Allard, G. Hébrail, F. Masseglia, and E. Pacitti. Chiaroscuro : Transparency and privacy for massive personal time-series clustering. In *SIGMOD Conference*, 2015.

[5] Tristan Allard, Nicolas Anciaux, Luc Bouganim, Yanli Guo, Lionel Le Folgoc, Benjamin Nguyen, Philippe Pucheral, Indrajit Ray, Indrakshi Ray, and Shaoyi Yin. Secure personal data servers : a vision paper. *Proceedings of the VLDB Endowment*, 3(1-2) :25–35, 2010.

[6] Ittai Anati, Shay Gueron, Simon P Johnson, and Vincent R Scarlata. Innovative technology for cpu based attestation and sealing. In *Proceedings of the 2nd international workshop on hardware and architectural support for security and privacy*, 2013.

[7] Nicolas Anciaux, Philippe Bonnet, Luc Bouganim, Benjamin Nguyen, Iulian Sandu Popa, and Philippe Pucheral. Trusted cells : A sea change for personal data services. In *CIDR*. www.cidrdb.org, 2013.

[8] Nicolas Anciaux, Philippe Bonnet, Luc Bouganim, Benjamin Nguyen, Philippe Pucheral, Iulian Sandu Popa, and Guillaume Scerri. Personal data management systems : The security and functionality standpoint. *Information Systems*, 2018.

[9] Nicolas Anciaux, Luc Bouganim, Philippe Pucheral, Yanli Guo, Lionel Le Folgoc, and Shaoyi Yin. Milo-db : a personal, secure and portable database machine. *Distributed and Parallel Databases*, 32(1) :37–63, 2014.

[10] Nicolas Anciaux, Luc Bouganim, Philippe Pucheral, Iulian Sandu Popa, and Guillaume Scerri. Personal Database Security and Trusted Execution Environments : A Tutorial at the Crossroads. *Proceedings of the VLDB Endowment (PVLDB)*, August 2019.

[11] Nicolas Anciaux, Luc Bouganim, Philippe Pucheral, Shaoyi Yin, Quentin Lefebvre, Aydogan Ersoz, and Alexei Troussov. Logiciel plugdb-engine, 2015.

[12] Arvind Arasu, Spyros Blanas, Ken Eguro, Raghav Kaushik, Donald Kossmann, Ravi Ramamurthy, and Ramaratnam Venkatesan. Orthogonal security with cipherbase. In *Proceedings of the Sixth Biennial Conference on Innovative Data Systems Research (CIDR 2013), Asilomar, CA, USA, January 6-9, 2013*. Sixth Biennial Conference on Innovative Data Systems Research (CIDR 2013), 2013.

[13] Arvind Arasu and Raghav Kaushik. Oblivious query processing. *arXiv preprint arXiv :1312.4012*, 2013.

[14] Joshua J Armstrong, Mu Zhu, John P Hirdes, and Paul Stolee. K-means cluster analysis of rehabilitation service users in the home health care system of ontario : Examining the heterogeneity of a complex geriatric population. *Archives of physical medicine and rehabilitation*, 93(12) :2198–2205, 2012.

[15] Michael Backes, Peter Druschel, Andreas Haeberlen, and Dominique Unruh. Csar : A practical and provable technique to make randomized systems accountable. In *NDSS*, volume 9, pages 341–353, 2009.

[16] Maurice Bailleu, Jörg Thalheim, Pramod Bhatotia, Christof Fetzer, Michio Honda, and Kapil Vaswani. SPEICHER : Securing lsm-based key-value stores using shielded execution. In *17th USENIX Conference on File and Storage Technologies (FAST 19)*, pages 173–190, 2019.

[17] Sumeet Bajaj and Radu Sion. Trusteddb : A trusted hardware-based database with privacy and data confidentiality. *IEEE Transactions on Knowledge and Data Engineering*, 26(3) :752–765, 2013.

[18] Borja Balle, James Bell, Adria Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *CRYPTO*, pages 638–667. Springer, 2019.

[19] M. Barbosa, B. Portela, G. Scerri, and B. Warinschi. Foundations of hardware-based attested computation and application to SGX. In *EuroS&P*, 2016.

[20] J. Bater, G. Elliott, C. Eggen, S. Goel, A. N. Kho, and J. Rogers. SMCQL : secure query processing for private data networks. *PVLDB*, 10(6), 2017.

[21] Johes Bater, Yongjoo Park, Xi He, Xiao Wang, and Jennie Rogers. Saqe : practical privacy-preserving approximate query processing for data federations. *Proceedings of the VLDB Endowment*, 13(12) :2691–2705, 2020.

[22] Mohammad-Mahdi Bazm, Marc Lacoste, Mario Südholt, and Jean-Marc Menaud. Side channels in the cloud : Isolation challenges, attacks, and countermeasures. *hal-01591808*, 2017. `https://hal.inria.fr/hal-01591808`.

[23] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 1–10, New York, NY, USA, 1988. Association for Computing Machinery.

[24] Tim Berners-Lee. *One Small Step for the Web.*, 2018. `https://inrupt.com/one-small-step-for-the-web`.

[25] BitsAbout.me. *BitsaboutMe is a new service that empowers you to reclaim control over your personal data, in order to better protect your privacy and to get a fair deal when sharing your personal data profile with trustworthy companies and institutions*, 2012. `https://bitsabout.me`.

[26] Dan Bogdanov, Sven Laur, and Jan Willemson. Sharemind : A framework for fast privacy-preserving computations. In *European Symposium on Research in Computer Security*, pages 192–206. Springer, 2008.

[27] Peter Bogetoft, Dan Lund Christensen, Ivan Damgård, Martin Geisler, Thomas Jakobsen, Mikkel Krøigaard, Janus Dam Nielsen, Jesper Buus Nielsen, Kurt Nielsen, Jakob Pagter, Michael Schwartzbach, and Tomas Toft. Secure multiparty computation goes live. In Roger Dingledine and Philippe Golle, editors, *Financial Cryptography and Data Security*, pages 325–343, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[28] Dan Boneh, Craig Gentry, Shai Halevi, Frank Wang, and David J. Wu. Private database queries using somewhat homomorphic encryption. In *ACNS*, 2013.

[29] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-dnf formulas on ciphertexts. In *Theory of Cryptography Conference*, pages 325–341. Springer, 2005.

[30] Igor Bonifacic. *Microsoft accidently exposed 250 million customer service records*, 2020. `https://www.engadget.com/2020-01-22-microsoft-database-exposure.html?guccounter=1`.

[31] A. Boutet, D. Frey, R. Guerraoui, A. Jégou, and A. M. Kermarrec. Privacy-preserving distributed collaborative filtering. *Computing*, 98(8), 2016.

[32] Antoine Boutet, Davide Frey, Arnaud Jégou, Anne-Marie Kermarrec, and Heverson B Ribeiro. Freerec : An anonymous and distributed personalization architecture. *Computing*, 97(9) :961–980, 2015.

[33] Zvika Brakerski and Vinod Vaikuntanathan. Fully homomorphic encryption from ring-lwe and security for key dependent messages. In *CRYPTO*, pages 505–524. Springer, 2011.

[34] Carole Cadwalladr and Emma Graham-Harrison. Revealed : 50 million facebook profiles harvested for cambridge analytica in major data breach. *The guardian*, 17 :22, 2018.

[35] David Chaum, Claude Crépeau, and Ivan Damgard. Multiparty unconditionally secure protocols. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 11–19, 1988.

[36] Jung Hee Cheon, Jean-Sébastien Coron, Jinsu Kim, Moon Sung Lee, Tancrede Lepoint, Mehdi Tibouchi, and Aaram Yun. Batch fully homomorphic encryption over the integers. In *EUROCRYPT*, pages 315–335. Springer, 2013.

[37] Albert Cheu, Adam D. Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed Differential Privacy via Shuffling. In *EUROCRYPT*, pages 375–403. Springer, 2019.

[38] Ashish Choudhury, Jake Loftus, Emmanuela Orsini, Arpita Patra, and Nigel P Smart. Asiacrypt. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 221–240. Springer, 2013.

[39] Wolfie Christl. *Corporate Surveillance in Everyday Life*, 2017.

[40] Cozy Cloud. *A smart personal cloud to gather all your data*, 2012. `https://cozy.io/en`.

[41] CloudLocker. *CloudLocker : Your Private and Secure Personal Cloud Device*, 2013. `https://www.cloudlocker.eu/en/index.html`.

[42] Joseph A Cruz and David S Wishart. Applications of machine learning in cancer prediction and prognosis. *Cancer informatics*, 2 :117693510600200030, 2006.

[43] Ivan Damgård, Valerio Pastro, Nigel Smart, and Sarah Zakarias. Multiparty computation from somewhat homomorphic encryption. In *CRYPTO*, pages 643–662. Springer, 2012.

[44] George Danezis. Measuring anonymity : a few thoughts and a differentially private bound. In *Proceedings of the DIMACS Workshop on Measuring Anonymity*, 2013.

[45] Yves-Alexandre de Montjoye, Erez Shmueli, Samuel S Wang, and Alex Sandy Pentland. openPDS : Protecting the privacy of metadata through safeanswers. *PLoS One*, 9(7) :e98790, 2014.

[46] Differential Privacy Team, Apple. *Learning With Privacy At Scale*, 2017. `https://machinelearning.apple.com/docs/learning-with-privacy-at-scale/appledifferentialprivacysystem.pdf`.

[47] Digi.me. *See what your data can do for you with digi.me Private Sharing*, 2009. `https://digi.me`.

[48] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *NIPS*, 2017.

[49] T. T. Anh Dinh, P. Saxena, E. C. Chang, B. C. Ooi, and C. Zhang. M2R : enabling stronger privacy in mapreduce computation. In *USENIX Security Symposium*, 2015.

[50] John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 429–438. IEEE, 2013.

[51] Cynthia Dwork. Differential privacy. In *ICALP (2)*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2006.

[52] Cynthia Dwork. Differential privacy : A survey of results. In *International conference on theory and applications of models of computation*, pages 1–19. Springer, 2008.

[53] Mahmoud Elbattah and Owen Molloy. Clustering-aided approach for predicting patient outcomes with application to elderly healthcare in ireland. In *Workshops at the Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[54] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE transactions on Information Theory*, 31(4) :469–472, 1985.

[55] Ulfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, and Kunal Talwar. Amplification by Shuffling : From Local to Central Differential Privacy via Anonymity. In *SODA*, 2019.

[56] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor : Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067, 2014.

[57] S. Eskandarian and M. Zaharia. Oblidb : Oblivious query processing using hardware enclaves. *arXiv*, 2017.

[58] Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. *Communications of the ACM*, 28(6) :637–647, 1985.

[59] The World Economic Forum. *Rethinking Personal Data : Strengthening Trust.*, 2012. `https://iapp.org/media/pdf/knowledge_center/WEF_IT_RethinkingPersonalData_Report_2012.pdf`.

[60] Michael J Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In *EUROCRYPT*, pages 1–19. Springer, 2004.

[61] B. Fuhry, R. Bahmani, F. Brasser, F. Hahn, F. Kerschbaum, and A. Sadeghi. Hardidx : Practical and secure index with SGX in a malicious environment. *Journal of Computer Security*, 26(5), 2018.

[62] Craig Gentry. *A fully homomorphic encryption scheme*, volume 20. Stanford university Stanford, 2009.

[63] Arpita Ghosh, Tim Roughgarden, and Mukund Sundararajan. Universally utility-maximizing privacy mechanisms. *SIAM Journal on Computing*, 2012.

[64] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, page 218–229, New York, NY, USA, 1987. Association for Computing Machinery.

[65] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of computer and system sciences*, 28(2) :270–299, 1984.

[66] Hamed Haddadi, Heidi Howard, Amir Chaudhry, Jon Crowcroft, Anil Madhavapeddy, and Richard Mortier. Personal data : Thinking inside the box. *CoRR*, abs/1501.04737, 2015.

[67] Jens Hiller, Jan Pennekamp, Markus Dahlmanns, Martin Henze, Andriy Panchenko, and Klaus Wehrle. Tailoring onion routing to the internet of things : Security and privacy in untrusted environments. In *2019 IEEE 27th International Conference on Network Protocols (ICNP)*, pages 1–12. IEEE, 2019.

[68] Zhexue Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, 2(3) :283–304, 1998.

[69] T. Hunt, Z. Zhu, Y. Xu, S. Peter, and E. Witchel. Ryoan : A distributed sandbox for untrusted computation on secret data. In *OSDI*, 2016.

[70] I. Ioannidis, A. Grama, and M. Atallah. A secure protocol for computing dot-products in clustered and distributed environments. In *Proceedings International Conference on Parallel Processing*, pages 379–384, 2002.

[71] Márk Jelasity, Alberto Montresor, and Ozalp Babaoglu. Gossip-based aggregation in large dynamic networks. *ACM Transactions on Computer Systems (TOCS)*, 23(3) :219–252, 2005.

[72] Noah Johnson, Joseph P Near, and Dawn Song. Towards practical differential privacy for sql queries. *Proceedings of the VLDB Endowment*, 11(5) :526–539, 2018.

[73] Shanthi Johnson, Juanita Bacsu, Hasanthi Abeykoon, Tom McIntosh, Bonnie Jeffery, and Nuelle Novik. No place like home : A systematic review of home care for older adults in canada. *Canadian Journal on Aging/La Revue canadienne du vieillissement*, 37(4) :400–419, 2018.

[74] Taehoon Kim, Joongun Park, Jaewook Woo, Seungheun Jeon, and Jaehyuk Huh. Shieldstore : Shielded in-memory key-value storage with sgx. In *Proceedings of the Fourteenth EuroSys Conference 2019*, pages 1–15, 2019.

[75] Paul Kocher, Jann Horn, Anders Fogh, Daniel Genkin, Daniel Gruss, Werner Haas, Mike Hamburg, Moritz Lipp, Stefan Mangard, Thomas Prescher, et al. Spectre attacks : Exploiting speculative execution. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 1–19. IEEE, 2019.

[76] Riad Ladjel, Nicolas Anciaux, Philippe Pucheral, and Guillaume Scerri. A manifest-based framework for organizing the management of personal data at the edge of the network. In *ISD*, 2019.

[77] Riad Ladjel, Nicolas Anciaux, Philippe Pucheral, and Guillaume Scerri. Trustworthy distributed computations on personal data using trusted execution environments. In *TrustCom*, 2019.

[78] Riad Ladjel, Nicolas Anciaux, Philippe Pucheral, and Guillaume Scerri. Secure distributed queries over large sets of personal home boxes. In *Transactions on Large-Scale Data-and Knowledge-Centered Systems XLIV*, pages 108–131. Springer, 2020.

[79] Sven Laur, Riivo Talviste, and Jan Willemson. From oblivious aes to efficient and secure database join in the multiparty setting. In *International Conference on Applied Cryptography and Network Security*, pages 84–101. Springer, 2013.

[80] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness : Privacy beyond k-anonymity and l-diversity. In *ICDE*, pages 106–115. IEEE Computer Society, 2007.

[81] Minlei Liao, Yunfeng Li, Farid Kianifard, Engels Obi, and Stephen Arcona. Cluster analysis and its application to healthcare claims data : a study of end-stage renal disease patients who initiated hemodialysis. *BMC nephrology*, 17(1) :25, 2016.

[82] Julien Loudet, Iulian Sandu Popa, and Luc Bouganim. DISPERS : securing highly distributed queries on personal data management systems. *PVLDB*, 12(12) :1886–1889, 2019.

[83] Julien Loudet, Iulian Sandu Popa, and Luc Bouganim. SEP2P : secure and efficient P2P personal data processing. In *EDBT*, pages 145–156. OpenProceedings.org, 2019.

[84] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi Li, and Fei-Yue Wang. Traffic flow prediction with big data : a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2) :865–873, 2014.

[85] Ashwin Machanavajjhala, Daniel Kifer, John Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy : Theory meets practice on the map. In *2008 IEEE 24th international conference on data engineering*, pages 277–286. IEEE, 2008.

[86] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. *L*-diversity : Privacy beyond *k*-anonymity. *TKDD*, 1(1) :3, 2007.

[87] Meeco. *Meeco — the distributed technology powering consent and personal data*, 2012. `https://www.meeco.me`.

[88] Ralph C. Merkle. A digital signature based on a conventional encryption function. In *CRYPTO*, volume 293 of *Lecture Notes in Computer Science*, pages 369–378. Springer, 1987.

[89] MesInfos. *The MesInfos project explores and implements the self data concept in France*, 2012. `http://mesinfos.fing.org/english`.

[90] midata. *The midata vision of consumer empowerment*, 2011. `https://www.gov.uk/government/news/the-midata-vision-of-consumer-empowerment`.

[91] Pratyush Mishra, Rishabh Poddar, Jerry Chen, Alessandro Chiesa, and Raluca Ada Popa. Oblix : An efficient oblivious search index. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 279–296. IEEE, 2018.

[92] Andrés Molina-Markham, Prashant Shenoy, Kevin Fu, Emmanuel Cecchet, and David Irwin. Private memoirs of a smart meter. In *Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building*, pages 61–66, 2010.

[93] mycloud.com. *My Cloud Home is the perfect storage solution to easily keep all your photos, videos, music and files organized in one central place at home*, 2015. `https://www.mycloud.com`.

[94] MyData.org. *MyData Global's mission is to empower individuals by improving their reight to self-determination regarding their personal data*, 2014. `https://mydata.org`.

[95] Nextcloud. *The self-hosted productivity platform that keeps you in control*, 2016. `https://nextcloud.com`.

[96] Thông T Nguyên, Xiaokui Xiao, Yin Yang, Siu Cheung Hui, Hyejin Shin, and Junbum Shin. Collecting and analyzing data from smart device users with local differential privacy. *arXiv preprint arXiv :1606.05053*, 2016.

[97] Inc. Novathings. *helixee — The French cloud that respects your privacy.* `http://www.helixee.me/home/`.

[98] Barack Obama. *Public Papers of the Presidents of the United States : Barack Obama, 2009.* Government Printing Office, 2011.

[99] Official Journal of the European Union. *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*, 2016. `https://eur-lex.europa.eu/eli/reg/2016/679/oj`.

[100] Olga Ohrimenko, Manuel Costa, Cédric Fournet, Christos Gkantsidis, Markulf Kohlweiss, and Divya Sharma. Observing and preventing leakage in MapReduce. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1570–1581, 2015.

[101] Olga Ohrimenko, Felix Schuster, Cédric Fournet, Aastha Mehta, Sebastian Nowozin, Kapil Vaswani, and Manuel Costa. Oblivious multi-party machine learning on trusted processors. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 619–636, 2016.

[102] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *EUROCRYPT*, pages 223–238. Springer, 1999.

[103] Sellappan Palaniappan and Rafiah Awang. Intelligent heart disease prediction system using data mining techniques. In *2008 IEEE/ACS international conference on computer systems and applications*, pages 108–115. IEEE, 2008.

[104] Perkeep. *Perkeep (née Camlistore) is a set of open source formats, protocols, and software for modeling, storing, searching, sharing and synchronizing data in the post-PC era.*, 2013. `https://perkeep.org`.

[105] Nicole Perlroth. All 3 billion yahoo accounts were affected by 2013 attack. *New York Times*, 2017.

[106] R. Pires, D. Gavril, P. Felber, E. Onica, and M. Pasin. A lightweight mapreduce framework for secure processing with SGX. In *CCGrid*, 2017.

[107] Laura Poitras. Citizenfour. *Lectures, publications reçues*, 2015.

[108] C. Priebe, K. Vaswani, and M. Costa. Enclavedb : A secure database using SGX. In *2018 IEEE Symposium on Security and Privacy (SP)*, 2018.

[109] Philippe Pucheral, Luc Bouganim, Patrick Valduriez, and Christophe Bobineau. PicoDBMS : scaling down database techniques for the smartcard. *The VLDB Journal*, 10(2-3) :120–132, 2001.

[110] ArvindArasu SpyrosBlanas KenEguro ManasJoglekar RaghavKaushik and DonaldKossmann RaviRamamurthy PrasangUpadhyaya RamarathnamVenkatesan. Engineering security and performance with cipherbase. *Data Engineering*, page 65, 2012.

[111] David Reinsel, John Gantz, and John Rydning. The digitization of the world. *From Edge to Core, An IDC White Paper-# US44413318*, 2018.

[112] Ronald L Rivest, Len Adleman, and Michael L Dertouzos. On data banks and privacy homomorphisms. *Foundations of secure computation*, 4(11) :169–180, 1978.

[113] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2) :120–126, 1978.

[114] M. Sabt, M. Achemlal, and A. Bouabdallah. Trusted execution environment : What it is, and what it is not. In *TrustCom/BigDataSE/ISPA (1)*, 2015.

[115] Pierangela Samarati and Latanya Sweeney. Protecting privacy when disclosing information : k-anonymity and its enforcement through generalization and suppression. *technical report, SRI International*, 1998.

[116] F. Schuster, M. Costa, C. Fournet, C. Gkantsidis, M. Peinado, G. Mainar-Ruiz, and M. Russinovich. VC3 : trustworthy data analytics in the cloud using SGX. In *2015 IEEE Symposium on Security and Privacy (SP)*, 2015.

[117] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11) :612–613, 1979.

[118] Rashid Sheikh, Beerendra Kumar, and Durgesh Kumar Mishra. A distributed k-secure sum protocol for secure multi-party computations. *arXiv preprint arXiv :1003.4071*, 2010.

[119] Jason Silverstein. Hundreds of millions of facebook user records were exposed on amazon cloud server. *CBS News*, 4, 2019.

[120] SpiderOak. *SpiderOak Share provides a secure way to exchange and sync your files using No Knowledge Encryption.*, 2018. `https://spideroak.com/spideroak-share`.

[121] Sync. *Sync's end-to-end encrypted storage platform and apps ensure that only you can access your data in the cloud.*, 2011. `https://www.sync.com`.

[122] Jun Tang, Aleksandra Korolova, Xiaolong Bai, Xueqiang Wang, and Xiaofeng Wang. Privacy loss in apple's implementation of differential privacy on macos 10.12. *arXiv preprint arXiv :1709.02753*, 2017.

[123] Yuzhe Tang, Ju Chen, Kai Li, Jianliang Xu, and Qi Zhang. Authenticated key-value stores with hardware enclaves. *arXiv preprint arXiv :1904.12068*, 2019.

[124] Dai Hai Ton That, Iulian Sandu Popa, Karine Zeitouni, and Cristian Borcea. Pampas : Privacy-aware mobile participatory sensing using secure probes. In *Proceedings of the 28th International Conference on Scientific and Statistical Database Management*, page 4. ACM, 2016.

[125] Quoc-Cuong To, Benjamin Nguyen, and Philippe Pucheral. Privacy-preserving query execution using a decentralized architecture and tamper resistant hardware. In *EDBT*, pages 487–498. OpenProceedings.org, 2014.

[126] Quoc-Cuong To, Benjamin Nguyen, and Philippe Pucheral. Private and scalable execution of SQL aggregates on a secure decentralized architecture. *ACM Trans. Database Syst.*, 41(3) :16 :1–16 :43, 2016.

[127] F. Tramèr, F. Zhang, H. Lin, J.P. Hubaux, A. Juels, and E. Shi. Sealed-glass proofs : Using transparent enclaves to prove and sell knowledge. In *EuroS&P*, 2017.

[128] Jo Van Bulck, Marina Minkin, Ofir Weisse, Daniel Genkin, Baris Kasikci, Frank Piessens, Mark Silberstein, Thomas F Wenisch, Yuval Yarom, and Raoul Strackx. Foreshadow : Extracting the keys to the intel SGX kingdom with transient out-of-order execution. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 991–1008, 2018.

[129] Jelle van den Hooff, David Lazar, Matei Zaharia, and Nickolai Zeldovich. Vuvuzela : Scalable private messaging resistant to traffic analysis. In *Proceedings of the 25th Symposium on Operating Systems Principles*, SOSP '15, page 137–152, New York, NY, USA, 2015. Association for Computing Machinery.

[130] Marten Van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully homomorphic encryption over the integers. In *Annual EUROCRYPT*, pages 24–43. Springer, 2010.

[131] W. Wang, G. Chen, X. Pan, Y. Zhang, X. Wang, V. Bindschaedler, H. Tang, and C. A. Gunter. Leaky cauldron on the dark land : Understanding memory side-channel hazards in SGX. In *CCS*, 2017.

[132] Stanley L Warner. Randomized response : A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309) :63–69, 1965.

[133] Andrew C Yao. Protocols for secure computations. In *23rd annual symposium on foundations of computer science (FOCS 1982)*, pages 160–164. IEEE, 1982.

[134] Andrew Chi-Chih Yao. How to generate and exchange secrets. In *27th Annual Symposium on Foundations of Computer Science (FOCS 1986)*, pages 162–167. IEEE, 1986.

[135] Wenting Zheng, Ankur Dave, Jethro G Beekman, Raluca Ada Popa, Joseph E Gonzalez, and Ion Stoica. Opaque : An oblivious and encrypted distributed analytics platform. In *14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17)*, pages 283–298, 2017.

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Calculs Distribués et Sécurisés pour le Cloud Personnel

Mots clés : Cloud Personnel ; Protection de la Vie Privée ; Système Personnel de Gestion de Données ; Calculs Distribués

Résumé : Grâce aux "smart disclosure initiatives", traduit en français par « ouvertures intelligentes » et aux nouvelles réglementations comme le RGPD, les individus ont la possibilité de reprendre le contrôle sur leurs données en les stockant localement de manière décentralisée. En parallèle, les solutions dites de clouds personnels ou « système personnel de gestion de données » se multiplient, leur objectif étant de permettre aux utilisateurs d'exploiter leurs données personnelles pour leur propre bien.

Cette gestion décentralisée des données personnelles offre une protection naturelle contre les attaques massives sur les serveurs centralisés et ouvre de nouvelles opportunités en permettant aux utilisateurs de croiser leurs données collectées auprès de différentes sources. D'un autre côté, cette approche empêche le croisement de données provenant de plusieurs utilisateurs pour effectuer des calculs distribués. L'objectif de cette thèse est de concevoir un protocole de calcul distribué, générique, qui passe à l'échelle et qui permet de croiser les données personnelles de plusieurs utilisateurs en offrant de fortes garanties de sécurité et de protection de la vie privée. Le protocole répond également aux deux questions soulevées par cette approche : comment préserver la confiance des individus dans leur cloud personnel lorsqu'ils effectuent des calculs croisant des données provenant de plusieurs individus ? Et comment garantir l'intégrité du résultat final lorsqu'il a été calculé par une myriade de clouds personnels collaboratifs mais indépendants ?

Title : Secure Distributed Computations for the Personal Cloud

Keywords : Personal cloud ; Privacy-preserving ; Personal Data Management System ; Distributed computations

Abstract : Thanks to smart disclosure initiatives and new regulations like GDPR, individuals are able to get the control back on their data and store them locally in a decentralized way. In parallel, personal data management system (PDMS) solutions, also called personal clouds, are flourishing. Their goal is to empower users to leverage their personal data for their own good.

This decentralized way of managing personal data provides a de facto protection against massive attacks on central servers and opens new opportunities by allowing users to cross their data gathered from different sources. On the other side, this approach prevents the crossing of data from multiple users to perform distributed computations. The goal of this thesis is to design a generic and scalable secure decentralized computing framework which allows the crossing of personal data of multiple users while answering the following two questions raised by this approach. How to preserve individuals' trust on their PDMS when performing global computations crossing data from multiple individuals ? And how to guarantee the integrity of the final result when it has been computed by a myriad of collaborative but independent PDMSs ?

Maison du doctorat de l'Université Paris-Saclay
2ème étage aile ouest, Ecole normale supérieure Paris-Saclay
4 avenue des Sciences
91190 Gif sur Yvette, France