Extensive and Secure Personal Data Management Systems

Nicolas Anciaux, Luc Bouganim

Extensive and Secure Personal Data Management Systems

by *Nicolas Anciaux* and *Luc Bouganim* (*Inria, UVSQ and Univ. Paris-Saclay*)

Personal Data Management Systems (PDMS) are emerging to provide individuals with appropriate tools to collect, manage and make accessible their personal data to applications. At the same time, the deployment of Trusted Execution Environments (TEE) both on the cloud (Intel SGX processors) and on smartphones (ARM TrustZone) opens new architectural perspectives for these PDMS. The PETRUS project [L1] proposes new solutions for PDMS, allowing third-party applications to exploit the individual's personal data while guaranteeing individuals the legitimacy of uses and privacy. A first proof of concept implementation of this new architecture has just been developed on Intel SGX processor.

The PDMS is a database system that manages the personal data of a user under their control, based on a paradigm inversion for the management of personal data: instead of externalizing the personal data of the user to the services performing the processing, it is the processing code that comes to execute on the side of the PDMS close to the data in a controlled execution environment. To exemplify this trend, let’s consider a scenario where companies incentivize their employees for environmentally friendly behavior by offering a green bonus based on the number of commutes made by bike. GPS traces are collected from a reliable service and processed locally by the user's PDMS. The result is then delivered to the employer with proof of compliance. Several similar scenarios are realistic, e.g., patients providing statistics to hospitals, power meters, etc. This scenario is challenging since it calls for extensiveness to develop and deploy ad hoc code, and security to assure PDMS users that their detailed personal data is not disclosed to third parties outside the sphere of control of their PDMS.

To solve this tension between extensiveness and security, we proposed in [1] a three-layer logical architecture where a minimal Secure Core implementing basic operations on personal data is extended with Isolated Data Tasks themselves accessed by Applications on which no security assumption is made (see Figure 1, left). The objective is to control the flow of raw personal data from the Core to the outside, such that only expected results are declassified to untrusted applications or third parties. The right part of Figure 1 gives details on the concrete implementation of an ES-PDMS, with some examples of data tasks to answer the above scenario. Properties 1 to 5 are generic security properties that must be properly orchestrated to strengthen the security of our architecture. Security properties 6

to 8 were designed for the specific ES-PDMS context and were introduced in [2] and demonstrated in [L2].

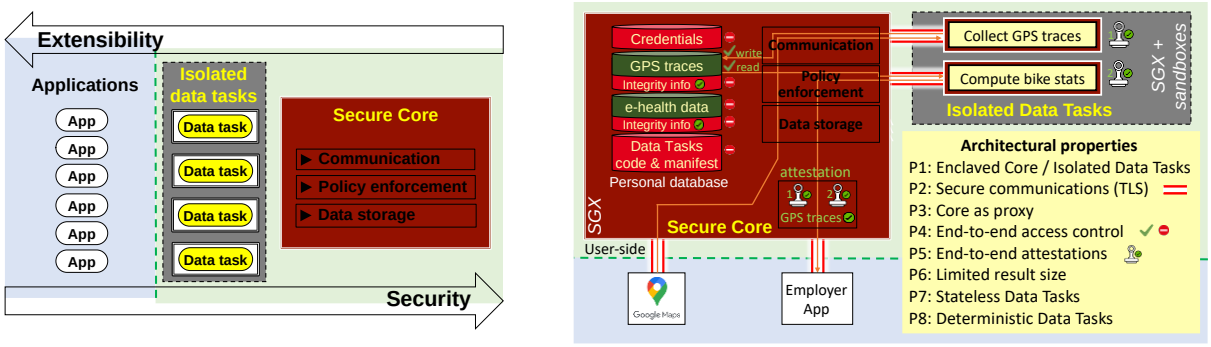

Figure 1: ES-PDMS logical architecture and a concrete implementation [L2].

This is a first step, and many challenges remain, depending on the type of processing and the manipulated data that must be properly protected:

- ***Personal data of a single PDMS user***. To properly address this use case, presented above, it is necessary to consider processing functions that handle large volumes of personal data, such as aggregation functions. These functions are defined by a third-party application called App, and their code is evaluated in the PDMS environment. However, the PDMS user cannot fully trust the App. Therefore, it is essential to focus on execution mechanisms that mitigate information leakage through the successive processing of these functions. As preliminary results, we introduce in [2] new execution strategies based on partitioning and replay of processing in different stateless Intel SGX enclaves, in order to guarantee an upper bound on data leakage through legitimate results, for certain categories of aggregation functions typical of PDMS use cases, and with a reasonable performance overhead.
- ***Personal data of a community of PDMS users***. Individuals can cross data within large communities of users, e.g., to compute statistics for epidemiological studies or to train a machine learning model for recommendation systems or automatic classification of user data. While, in the previous case, we can reasonably assume that the user will not attempt to attack the confidentiality of their own data, this is no longer the case as soon as we assume decentralized computations on PDMSs that will manipulate the data of other users. It is therefore necessary to consider implementation strategies that limit data leakage in case of a breach of the (hardware) security of the PDMS. As a first possible scenario, we consider in [3] a threat model that we call the "Wolf in the sheepfold", in which (a small number of) PDMSs have been instrumented by their owner and thus operate in the so-called "sealed glass proof" mode, without confidentiality (but still as expected, with integrity guarantees). Other

more difficult models are to be considered, where the communications between PDMS can be analyzed and where the nodes could be corrupted also in integrity or in greater number.

- ***Personal data of third parties***. Some cases of computation on PDMS require the exploitation of potentially sensitive third-party data (e.g., a machine learning model derived from a personal dataset or an IP-protected dataset). Integrating or storing third-party data in the application during deployment would, compared to the two cases above, raise additional security issues (the same set of sensitive data must be protected in all PDMSs running the application) with update and performance difficulties after deployment. Therefore, it is necessary to allow for the application to dynamically retrieve third-party data when running on the PDMS side, which represents a new potential leakage channel for PDMS data and requires new solutions to regulate PDMS data flows at runtime.

Since the initial paper [1] on the extensive and secure PDMS architecture, the PETRUS project has obtained interesting research results and realized some demonstrators [L2, L3] which underscores the potential of the PDMS approach, fostered by the development of TEEs.

Links:

[L1] PETRUS, PErsonal & TRUSted cloud: https://team.inria.fr/petrus/

[L2] PDMS demonstration: https://project.inria.fr/espdms/

[L3] Edgelet Computing demonstration: https://project.inria.fr/edgeletdemo/

References:

[1] N. Anciaux et al., "Personal Data Management Systems: The security and functionality standpoint", Information Systems, 2019, 80

[2] R. Carpentier, I. Sandu Popa, N. Anciaux, "Data Leakage Mitigation of User-Defined Functions on Secure Personal Data Management Systems", in SSDBM, 2022.

[3] L. Javet et al., "Edgelet Computing: Pushing Query Processing and Liability at the Extreme Edge of the Network", in CCGrid, 2022.

Please contact:

Nicolas Anciaux
Inria, UVSQ and Univ. Paris-Saclay, France
Nicolas.Anciaux@inria.fr