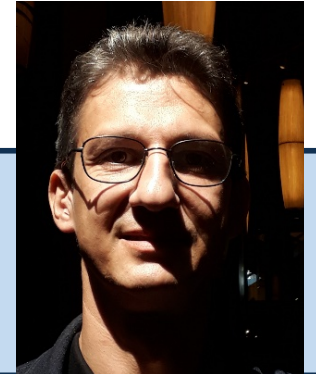


Tutorial Outline

PART I. Personal Data Management Systems (PDMS)

Review of functionalities & addressed privacy threats

Individual's PDMS vs (corporate) DBMS and main properties to achieve



PART II. TEE-based Data Management

The promises of Trusted Execution Environments (TEEs)

A review of privacy-preserving data management using TEEs



PART III. Bridging the Gap between PDMS and TEEs

How could the main properties be achieved?

A quick view of remaining challenges



10 years history of Personal Data Management Systems

Since 2008 – FreedomBox@Columbia (Eben Moglen)

Free individuals from state control

PDMS = Low-cost open HW + open SW



New HW since 3/19

Since 2010 – PDS@Inria [AAB+10], MiloDB [ABP+14], PDMS [ABB+19]

Manage (specific) personal folders at hand, enforce privacy policies

PDMS = Tamper resistant HW (smart card or TEEs) + embedded DBMS



2012 – OpenPDS@MIT [MSW+14], 2016 – DataBox-BBCBox@Nottingham [MZC+16]

Manage your data locally, externalize only safe answers

PDMS = SW running on user's device (smartphone, tablet)



Since 2013 – Gov. [MyDex, MesInfos] & commercial initiatives [NextCloud, Cozy, ...]

Collect personal data from different data silos & provide transversal Apps

PDMS = Online SW with Apps (terminology shift: PDS → personal cloud)



Since 2018 – Solid PODs and Inrupt (Tim Berner Lee)

To re-decentralize the Web of personal data, give agency to individuals

PDMS = Personal Online Data store (PODs)



10 years history of Personal Data Management Systems

Since 2008 – FreedomBox@Columbia (Eben Moglen)

Free individuals from state control

PDMS = Low-cost open HW + open SW



New HW since 3/19

Since 2010 – PDS@Inria [AAB+10], MiloDB [ABP+14], PDMS [ABB+19]

Manage (specific) personal folders at hand, enforce privacy policies

PDMS = Tamper resistant HW (smart card or TEEs) + emul.

2012 – OpenPDS@MIT [MSW+14], 2016 – PDS@MIT [MBC+16]

Manage your data locally, external services

PDMS = SW running on user's device



Higham [MZC+16]

Since 2013

Commercial initiatives [NextCloud, Cozy, ...]

Connect personal data silos & provide transversal Apps

Apps (terminology shift: PDS → personal cloud)



Since 2017 – Solid PODs and Inrupt (Tim Berner Lee)

To re-decentralize the Web of personal data, give agency to individuals

PDMS = Personal Online Data store (PODs)



What are their functionalities?
What are the privacy threats considered?

Main classes of architectures for a PDMS



Online personal cloud

E.g., Cozy, Digi.me, NextCloud, BitsAbout.Me, Perkeep

Functionality:

Data collectors for everything (banks, energy, health, geolocation, 'likes' graphs, ...)

Personal (cross-)computation (1 individual) features for App developers

Backup (full retention: Perkeep)

Trust model:

Personal cloud provider & Apps considered **fully honest**

Security standards, PEN tests (Cozy), code transparency (community checks)

No-knowledge personal cloud



E.g., MyDex, SpiderOak, Digi.me

Functionality:

Secure data store, personal data encrypted (encryption keys managed at client side)

Secure backup and point in time recovery

Trust model:

Personal cloud provider is **untrusted** (but the client device is not)

Considered attacks: **data snooping** and **secondary usages** (server), **ransomware** (client)

Main classes of architectures for a PDMS

Online personal cloud

→ Advanced functionality, strong trust assumptions

E.g., Cozy, Digi.me, NextCloud, BitsAbout.Me, Perkeep



Functionality:

Data collectors for everything (banks, energy, health, geolocation, 'likes' graphs, ...)

Personal (cross-)computation (1 individual) features for App developers

Backup (full retention: Perkeep)

Trust model:

Personal cloud provider & Apps considered **fully honest**

Security standards, PEN tests (Cozy), code transparency (community checks)

No-knowledge personal cloud

→ Increased security, minimalist functionality

E.g., MyDex, SpiderOak, Digi.me

Functionality:

Secure data store, personal data encrypted (encryption keys managed at client side)

Secure backup and point in time recovery

Trust model:

Personal cloud provider is **untrusted** (but the client device is not)

Considered attacks: **data snooping** and **secondary usages** (server), **ransomware** (client)

Main classes of architectures for a PDMS (cont.)

Home (or edge) cloud software

E.g., OpenPDS [MSW+14], Databox [MZC+16]

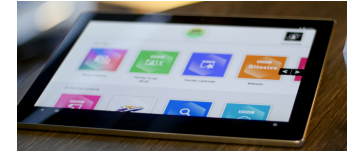
Functionality:

Trusted storage on end-user device or at the edge (1 store per IoT device)

Personal computation provided safe answers and aggregated views, never raw data

Data dissemination rules to share computed results

Trust model: user device and SW must be trusted



Home cloud plugs (dedicated)

E.g., FreedomBox, CloudLocker

Functionality: data store and backup in a dedicated hardware plug

Trust model: Plug code must be trusted (dedicated => limited attack surface)



Tamper-resistant home cloud

E.g., PDS [AAB+10], PlugDB [ANSP14, ALSP+15, LASP+17, ABB+19]

Functionality: (simple) store, share, compute (local/global) in a secure HW device

Trust model: secure HW + embedded SW are trusted



Main classes of architectures for a PDMS (cont.)

Home (or edge) cloud software

→ 'formal' security lost, more functionality

E.g., OpenPDS [MSW+14], Databox [MZC+16]

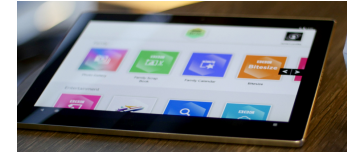
Functionality:

Trusted storage on end-user device or at the edge (1 store per IoT device)

Personal computation provided safe answers and aggregated views, never raw data

Data dissemination rules to share computed results

Trust model: user device and SW must be trusted



Home cloud plugs (dedicated)

E.g., FreedomBox, CloudLocker

Functionality: data store and back

Trust model: Plug code must be trusted

→ Security at the price of functionality, advanced processing on untrusted device



Tamper-resistant home cloud

E.g., PDS [AAB+10], PlugDB [ANSP14, ALSP+15, LASP+17, ABB+19]

Functionality: (simple) store, share, compute (local/global) in a secure HW device

Trust model: secure HW + embedded SW are trusted



Synthesis : functionalities

		Architecture				
		Online personal cloud	Zero-knowledge personal cloud	Home cloud software	Home cloud plug	Tamper resistant home cloud
Functionality	Storage	Regular DBMS	Files/KVS store	Files/KVS/DBMS at user-side	Files/KVS in the Plug	Embedded DBMS
	Backup	Regular DBMS	Encrypted archive, Pt-in-time recovery	Replication / offline store	Replication / offline store	Replication / offline store
	Data collection	Web scrapping	By users / Apps	By users / Apps	By users	By users / Apps
	Personal computations	Linked/transversal queries	Apps level	Safe answer, local data aggregation	Apps level	Simple transversal queries
	Distributed computations					Simple distributed SQL statistics at large scale
	Data dissemination	[synchro.]	At Apps level	Privileges for 3 rd parties and Apps	[synchro.]	Privileges for 3 rd parties and Apps, Secure AC

1- The whole personal cloud data life-cycle must be covered !

Synthesis : functionalities (cont.)

		Architecture				
		Online personal cloud	Zero-knowledge personal cloud	Home cloud software	Home cloud plug	Tamper resistant home cloud
Functionality	Storage	Regular DBMS	Files/KVS store	Files/KVS/DBMS at user-side	Files/KVS in the Plug	Embedded DBMS
	Backup	Regular DBMS	Encrypted archive, Pt-in-time recovery	Replication / offline store	Replication / offline store	Replication / offline store
	Data collection	Web scrapping	By users / Apps	By users / Apps	By users	By users / Apps
	Personal computations	Linked/transversal queries	Apps level	Safe answer, local data aggregation	Apps level	Simple transversal queries
	Distributed computations	✗	✗	✗	✗	Simple distributed SQL statistics at large scale
	Data dissemination	[synchro.]	At Apps level	Privileges for 3 rd parties and Apps	[synchro.]	Privileges for 3 rd parties and Apps, Secure AC

1- The whole personal cloud data life-cycle must be covered !

2- Distributed computations are poorly covered...

Less useful? No, Big-Data perspectives !

More difficult? Yes, efficient and secure (solutions in the tamper resistant context)

Synthesis : trust

		Representative Personal Cloud approaches				
		Online personal cloud	Zero-knowledge personal cloud	Home cloud software	Home cloud Plug	Tamper resistant personal server
Trust	Considered threats		Data snooping Data leakage 2 nd ary usages Client Failure Ransomware	Data snooping Data leakage 2 nd ary usages Over-priv. Apps	Data snooping Data leakage 2 nd ary usages Plug Failure	Data snooping Data leakage 2 nd ary usages Over-priv. Apps
	Trust model	Fully-honest personal cloud & Apps	Semi-honest or Malicious personal cloud Trusted Apps Trusted client	Trusted personal cloud Trusted client Untrusted Apps	Trusted personal cloud Trusted Plug Trusted Apps	Trusted personal cloud Semi-honest infra. Untrusted Apps
	Privacy and security measures	Security stds, Business model Open source	client-side encrypt ^o 'no-knowledge' store	Safe answers Separated stores Local audit	Closed platform (dedicated device), physical ownership	Secure HW small TCB secure distributed protocols

1- different privacy threats considered, all must be circumvented

Synthesis : trust (cont.)

		Representative Personal Cloud approaches				
		Online personal cloud	Zero-knowledge personal cloud	Home cloud software	Home cloud Plug	Tamper resistant personal server
Trust	Considered threats		Data snooping Data leakage 2 nd ary usages Client Failure Ransomware	Data snooping Data leakage 2 nd ary usages Over-priv. Apps	Data snooping Data leakage 2 nd ary usages Plug Failure	Data snooping Data leakage 2 nd ary usages Over-priv. Apps
	Trust model	Fully-honest personal cloud & Apps	Semi-honest or Malicious personal cloud Trusted Apps Trusted client	Trusted personal cloud Trusted client Untrusted Apps	Trusted personal cloud Trusted Plug Trusted Apps	Trusted personal cloud Semi-honest infra. Untrusted Apps
	Privacy and security measures	Security stds, Business model Open source	client-side encrypt ^o 'no-knowledge' store	Safe answers Separated stores Local audit	Closed platform (dedicated device), physical ownership	Secure HW small TCB secure distributed protocols

1- different privacy threats considered, all must be circumvented

2- unifying the solutions is not trivial (if not impossible)

Wide spectrum of architectural choices...

... but different – irreconcilable – trust models and security measures

Personal Data Management: anything new?

Objective:

- (1) provide the set of functionalities
- (2) address all threats

Decades of research in

.... secure data collection, storage, backup, queries!

Next:

Specificities of (individual's) PDMS vs (corporate) DBMS

.... and derived properties for an extensive and secure PDMS

In [ABB+19]: 5 properties are defined...

Expected PDMS functionalities & properties: Data Collection

Corporate DBMS

A basic operation using wrappers/APIs

Well-known & predefined wrappers/APIs
... audited and patched by the admins

Individual's PDMS

Primary data directly fed into user's PDMS

Secondary data needs data scrapping

Huge set of scrappers
...with untrusted code (e.g., Weboob)
...accessing sensitive data (credentials)
...in an untrusted environment !

Property: A PDMS enforces *piped data collection* iff:

- 1- the only PDMS data, accessible to the data collector, is the credentials;**
- 2- the credentials/collected data cannot be leaked outside the PDMS.**

The only external channel provided to the data collector is with a single data provider
... and the code is suitably isolated not to leak data elsewhere

Expected PDMS functions & properties: Personal computations

Corporate DBMS

Computations on corporate data

Set of (trusted) applications selected,
... audited and patched by admins

Individual's PDMS

Apps crossing several data from individual

For the PDMS owner or an external service
(e.g., Pay as you drive).
Apps 'move' to data but...
Apps are untrusted (user's viewpoint)
→ **local data must not leak**
Computations are untrusted (service viewpoint)
→ **results must be attested**

Property: A PDMS enforces *bilaterally trusted computations* iff:

- 1- the data computation can only access the expected data from the PDMS;
- 2- only the final result – not the raw data – can be exposed to a 3rd party;
- 3- it provides a proof that the result was produced by the expected code.

'Bilateral' → guarantees to the owner and the 3rd party willing to execute code

To owner : minimal collection principle is fulfilled, raw data cannot leak

To 3rd party: code remotely sent has been computed (it may include any verification on data)

Expected PDMS functions & properties: Collective computations

Corporate DBMS

Not common → practical solutions

e.g., few Hospitals run a collective query
A trusted party may be used (by contract)
SMC usable [BEE+17] (few participants)

Individual's PDMS

Common → new solutions are needed

e.g., Big Data and IA (recommendations,
participative studies, community learning...)
Mutual confidentiality & integrity are critical
At a very large scale
(no trusted party nor SMC)

Property: A PDMS enforces *mutually trusted collective computations* iff:

- 1- the data computation can only access the required participant data;**
- 2- only the final result – not the raw data – can be exposed to a 3rd party or any participant;**
- 3- it provides a proof that the result was produced by the expected code on the expected set of participants.**

'Mutual' → guarantees also hold between the participants

Definition of an Extensive and Secure PDMS (ES-PDMS)

An Extensive & Secure PDMS

provides the expected set of functionalities to cover the complete data life-cycle
data collection,
storage and recovery,
cross-computations,
collective computations,
data dissemination.

and is compliant with their respective security properties counterparts,
pipelined data collection,
mutual data at rest protection,
bilaterally trusted personal computation,
mutually trusted collective computation,
controlled data dissemination.

How do we get there?

The field of TEE-based secure data management is rapidly developing
→ let's take a closer look...

Thanks !

Questions ?



Inria

References (1)

- [AAB+10] T. Allard, N. AnCIAUX, L. BouganIM, Y. Guo, L. L. Folgoc, B. Nguyen, P. Pucheral, I. Ray, S. Yin. Secure personal data servers: a vision paper. PVLDB, 3(1), 25-35, 2010.
- [ABB+19] N. AnCIAUX, P. Bonnet, L. BouganIM, B. Nguyen, P. Pucheral, I. S. Popa, G. Scerri. Personal data management systems: The security and functionality standpoint. Inf. Syst., 80:13–35, 2019.
- [ABD+19] M. Acosta, T. Berners-Lee, A. Dimou, J. Domingue, L-D. Ibá, K. Janowicz, M-E. Vidal, A. Zaveri: The FAIR TRADE Framework for Assessing Decentralised Data Solutions. WWW 2019
- [ABP+14] N. AnCIAUX, L. BouganIM, P. Pucheral, Y. Guo, L. L. Folgoc, S. Yin. Milo-DB: a personal, secure and portable database machine. Distributed and Parallel Databases, 32(1):37–63, 2014.
- [AEJ+15] A. Arasu, K. Eguro, M. Joglekar, R. Kaushik, D. Kossmann, R. Ramamurthy: Transaction processing on confidential data using cipherbase. ICDE 2015: 435-446
- [AEK+14] A. Arasu, K. Eguro, R. Kaushik, R. Ramamurthy: Querying encrypted data. SIGMOD Conference 2014: 1259-1261
- [AK13] A. Arasu, R. Kaushik: Oblivious Query Processing. ICDT 2014.
- [ALS+15] N. AnCIAUX, S. Lallali, I. Sandu Popa, P. Pucheral: A Scalable Search Engine for Mass Storage Smart Objects. PVLDB 8(9): 910-921 (2015)
- [ANS13] N. AnCIAUX, B. Nguyen, I. Sandu Popa: Personal Data Management with Secure Hardware: How to Keep Your Data at Hand. MDM (2) 2013: 1-2
- [ANS14] N. AnCIAUX, B. Nguyen, I. Sandu Popa: Tutorial: Managing Personal Data with Strong Privacy Guarantees. EDBT 2014: 672-673

References (2)

- [BBB+17] R. Bahmani, M. Barbosa, F. Brasser, B. Portela, A.-R. Sadeghi, G. Scerri, B. Warinschi: Secure Multiparty Computation from SGX. *Financial Cryptography 2017*: 477-497
- [BEE+17] J. Bater, G. Elliott, C. Eggen, S. Goel, A. Kho, J. Rogers: SMCQL: secure querying for federated databases. *PVLDB 2017*
- [BGC+18] V. Bindschaedler, P. Grubbs, D. Cash, T. Ristenpart, V. Shmatikov: The tao of inference in privacy-protected databases. *PVLDB 2018*
- [BPS+16] M. Barbosa, B. Portela, G. Scerri, B. Warinschi: Foundations of Hardware-Based Attested Computation and Application to SGX. *EuroS&P 2016*: 245-260
- [BS11] S. Bajaj, R. Sion: TrustedDB: a trusted hardware-based database with privacy and data confidentiality. *SIGMOD Conference 2011*: 205-216
- [DSC+15] T. T. A. Dinh, P. Saxena, E. Chang, B. C. Ooi, C. Zhang: M2R: Enabling Stronger Privacy in MapReduce Computation. *USENIX Security 2015*
- [EZ17] S. Eskandarian, M. Zaharia: An oblivious general-purpose SQL database for the cloud. *CoRR*, abs/1710.00458, 2017
- [FBB+18] B. Fuhry, R. Bahmani, F. Brasser, F. Hahn, F. Kerschbaum, A.-R. Sadeghi: HardIDX: Practical and secure index with SGX in a malicious environment. *Journal of Computer Security* 26(5): 677-706 (2018)
- [HZX18] T. Hunt, Z. Zhu, Y. Xu, S. Peter, E. Witchel: Ryoan: A Distributed Sandbox for Untrusted Computation on Secret Data. *ACM Trans. Comput. Syst.* 35(4): 13:1-13:32 (2018)

References (3)

- [LAP+19] R. Ladjel, N. Ancaux, P. Pucheral, G. Scerri. Trustworthy Distributed Computations on Personal Data Using Trusted Execution Environments. TrustCom, 2019.
- [LAS+17] S. Lallali, N. Ancaux, I. Sandu Popa, P. Pucheral: Supporting secure keyword search in the personal cloud. Inf. Syst. 72: 1-26 (2017)
- [LSB19a] J. Loudet, I. Sandu Popa, L. Bouganim: SEP2P: Secure and Efficient P2P Personal Data Processing. EDBT 2019.
- [LSB19b] J. Loudet, I. Sandu-Popa, L. Bouganim. DISPERS: Securing Highly Distributed Queries on Personal Data Management Systems. PVLDB 2019
- [LWG+13] S. Lee, E.L. Wong, D. Goel, M. Dahlin, V. Shmatikov, πbox: A platform for privacy-preserving apps, in: NSDI, 2013.
- [MPC+18] P. Mishra, R. Poddar, J. Chen, A. Chiesa, R. A. Popa: Oblix: An Efficient Oblivious Search Index. S&P 2018.
- [MSW+14] Y-A. de Montjoye, E. Shmueli, SS. Wang, AS. Pentland: OpenPDS: Protecting the Privacy of Metadata through SafeAnswers. PLoS ONE 9(7) 2014
- [MZC+16] R. Mortier, J. Zhao, J. Crowcroft, L. Wang, Q. Li, H. Haddadi, Y. Amar, A. Crabtree, J. Colley, T. Lodge, T. Brown, D. McAuley, C. Greenhalgh: Personal Data Management with the Databox: What's Inside the Box? ACM CoNEXT Cloud-Assisted Networking workshop, 2016
- [OCF+15] O. Ohrimenko, M. Costa, C. Fournet, C. Gkantsidis, M. Kohlweiss, D.Sharma: Observing and Preventing Leakage in MapReduce. CCS 2015.

References (4)

- [OSF+16] O. Ohrimenko, F. Schuster, C. Fournet, A. Mehta, S. Nowozin, K. Vaswani, M. Costa: Oblivious Multi-Party Machine Learning on Trusted Processors. USENIX Security 2016.
- [PGF+17] R. Pires, D. Gavril, P. Felber, E. Onica, M. Pasin: A lightweight MapReduce framework for secure processing with SGX. CCGrid 2017
- [PVC18] C. Priebe, K. Vaswani, M. Costa: EnclaveDB: A Secure Database Using SGX. IEEE Symposium on Security and Privacy 2018: 264-278
- [RHM19] L. Roche, J. M. Hendrickx, Y-A. de Montjoye: Estimating the success of re-identifications in incomplete datasets using generative models. Nature Communications 2019
- [SCF+15] F. Schuster, M. Costa, C. Fournet, C. Gkantsidis, M. Peinado, G. Mainar-Ruiz, M. Russinovich: VC3: Trustworthy Data Analytics in the Cloud Using SGX. S&P 2015
- [TAP17] P. Tran-Van, N. AnCIAUX, P. Pucheral: SWYSWYK: A Privacy-by-Design Paradigm for Personal Information Management Systems. ISD 2017
- [TCL+19] Y. Tang, J. Chen, K. Li, J. Xu, Q. Zhang: Authenticated Key-Value Stores with Hardware Enclaves. CoRR abs/1904.12068 (2019)
- [WAK18] N. Weichbrodt, P.-L. Aublin, R. Kapitza: SGX-perf: A Performance Analysis Tool for Intel SGX Enclaves. Middleware 2018
- [ZDB+17] W. Zheng, A. Dave, J. G. Beekman, R. A. Popa, J. E. Gonzalez, I. Stoica. Opaque: An oblivious and encrypted distributed analytics platform. NSDI 2017