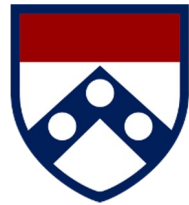
PReVer: Towards Private Regulated Verified Data

Mohammad Javad Amiri[1], Tristan Allard[2], Divyakant Agrawal[3], Amr El Abbadi[3]

[1]University of Pennsylvania, [2]Univ Rennes, CNRS, IRISA, [3]University of California Santa Barbara



Introduction

- A vision to support privacy-preserving regulated dynamic data
- Data privacy and secure query processing on data
  - Focus on supporting richer queries on a static outsourced DB (e.g., point queries, range queries, joins)
  - Minimal support for updates to the outsourced DB
  - Need to extend the support of updates to the functionalities traditionally provided by DBMS!
- Supporting constraints is especially challenging
  - Need to execute Boolean functions that query the outsourced DB.
  - Need to perform updates conditionally.
  - All this in a privacy-preserving and verifiable manner!

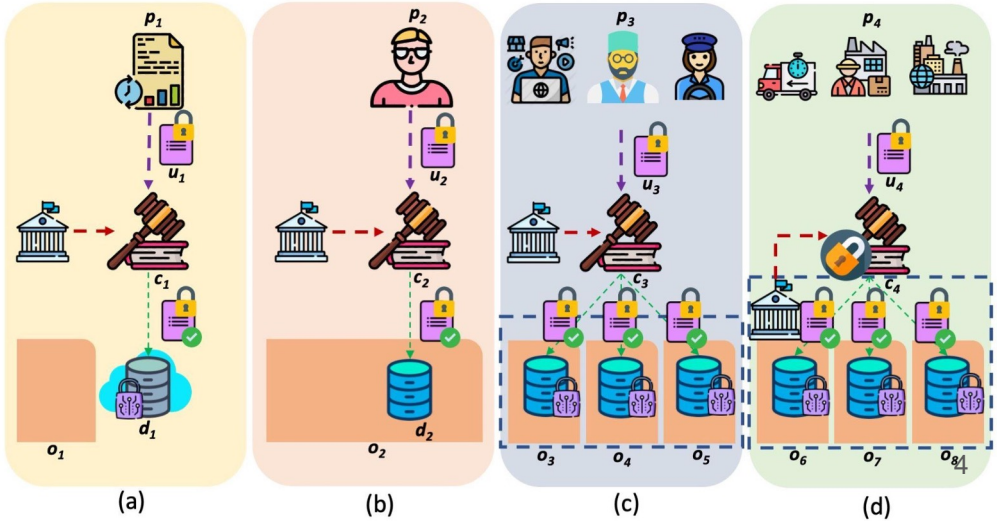
Regulated Dynamic Data

Problem: Verifying incoming updates with respect to constraints and incorporating the verified updates into data

- Constraints: internal constraints or global regulations
- Data: maintained by an untrusted or a set of mutually distrustful infrastructures
- Integrity
  - Updates on constraints
  - Updates on stored data
  - Stored data
- Privacy
  - Updates
  - Constraints
  - Data

Motivating Examples

- Data, updates and constraints may be private or public
- Single database or multiple (federated) databases
- Applications:
  - (a) Environmental Sustainability
  - (b) In-Person Conference Participation
  - (c) Multi-Platform Crowdworking
  - (d) Supply-chain Management

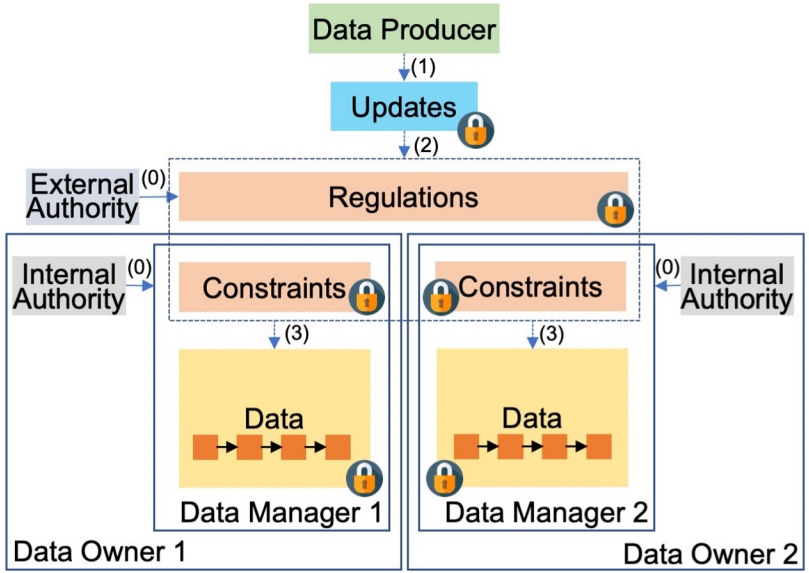


A Model for Regulated Dynamic Data

- Participants
  - Data Producers: produce updates
  - Data Owners: own the data, maintain data locally or outsource it to an external third party
  - Data Managers
    - Store and manage data on behalf of data owners
    - incorporate updates that are consistent with regulations and constraints into the data
  - Authority: defines constraints,
    - Internal constraints (i.e., data owner), External constraints (e.g., official institution)
  - A single participant might hold several roles (e.g., owner and authority)
- Threat Model
  - The choice of the threat model is not fixed but depends on each instantiation of the framework
  - Honest, honest-but-curious, covert, malicious
  - Participants might collude with each other

PReVer Framework

- A single framework that can be instantiated in many ways:
  - Participants setting
  - Supported constraints
  - Privacy-preserving techniques
- Flow of PReVer
  - Authorities define constraints
  - The data producer sends an update
  - The update is verified
  - The update is incorporated into data
- We identify illustrative challenges for specific participant settings (not exhaustive)
  - Single private outsourced DB
  - Multiple private DBs
  - Public Database



Research Challenge 1

Single private **outsourced** DB: a single data owner, a single data manager which is the cloud (untrusted), the data authority is the data owner.

Enable an untrusted data manager to verify updates against constraints and execute updates on private data in a privacy-preserving manner.

Example: Environmental Sustainability

- Fully homomorphic encryption
  - Support general computation over encrypted data but comes with considerable overhead.
- Secure search encryption and Oblivious RAM techniques
  - Support searches and updates but no execution capacities.
- Differentially private indexing
  - Support searches and updates. But no execution capacities on the non-indexed attributes
  - The privacy budget is limited.
- Trusted execution environment
  - Support secure computation but limited resources (bottleneck).

Research Challenge 2

Multiple private databases: multiple data managers, a centralized authority or multiple decentralized authorities

Enable a set of trusted and untrusted federated data managers to verify distributed constraints over distributed private data and to perform updates conditionally, all this with sound privacy and integrity guarantees.

Example: Multi-platform Crowdworking Environment

- Centralized approach: token-based systems
  - Unclear support of constraints other than simple `COUNT` aggregates.
- Decentralized approach: secure multi-party computation
  - Support the secure distributed computation of a large variety of building blocks (useful for executing constraints)
  - Performing updates and protecting the Boolean output of a constraint are not clear

Research Challenge 3

Public database(s)

Enable a data manager to verify updates against constraints over public data and execute the updates with sound privacy guarantees on the updates.

Example: In-Person Conference Participation

- Private Information Retrieval (PIR): allows users to retrieve items from a public database by specifying an index without revealing the items retrieved.
  - Restricted to retrieving a single item in a privacy-preserving manner.
  - Need to be extended to support updates.

Research Challenge 4

The integrity of stored data in both single and federated databases

Enable any participants to verify the integrity of stored data with privacy guarantees.

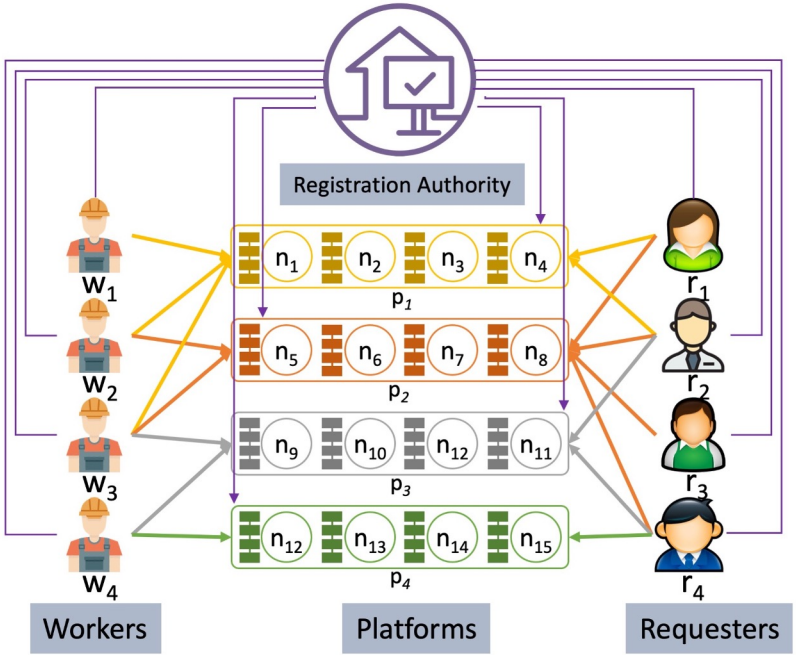
Example: Multi-platform Crowdworking Environment

- Data needs to be stored in an immutable and verifiable manner
  - Append-only ledgers
- Single database: centralized ledger databases
  - Fault-tolerance
- Federated databases: permissioned blockchains
  - Computational overhead is significant
  - Verifying the private data of each participant is challenging

An Application: Multi-platform Crowdworking

- Settings: private data, private updates, public regulations, multiple databases
- Needs to address challenges 2 and 4.
- A recent proposal: Separ [WWW'21]
  - Challenge 2: Token-based system
    - Centralized trusted authority
    - Single-use pseudonymous tokens
    - Distributed ledgers

    ⇒ Support lower/upper-bound constraints
  - Challenge 4: Permissioned blockchains
    - SharPer [SIGMOD'21]



Conclusion

- We lay out a vision to design PReVer, a universal framework for managing regulated dynamic data in a privacy-preserving manner.
- Context:
  - Private and public data, updates and constraints
  - Single and federated databases
- Any implementation of the PReVer framework needs to address several critical research questions.
- How to support scalable efficient consistent and verifiable execution of updates on data with constraints while preserving privacy?

➢ See Separ (WWW'21) for an illustrative instantiation of PReVer dedicated to multi-platform crowdworking.

Thank You!

Questions?