Alchemy: *Query Optimization for Private Data Federations*

Madhav Suresh
Jennie Rogers
Northwestern | ENGINEERING

Setting

- ➔ Multiple private data owners
- ➔ Query the union of their records
- ➔ Query results may be precise or differentially private

Problem

- ➔ Execution behavior leaks statistics
- ➔ Secure oblivious query execution is slow (SMCQL)
- ➔ Traditional query optimization utilizes private information
- ➔ Non-Oblivious privacy guarantees are hard

Goals

- ➔ Fast privacy-preserving query execution
- ➔ Leverage differentially private-statistics gathered with MPC to offer semi-oblivious, private query processing
- ➔ Reason symbolically about constraints from the schema to constrain intermediate results sizes
- ➔ End-to-end privacy guarantees

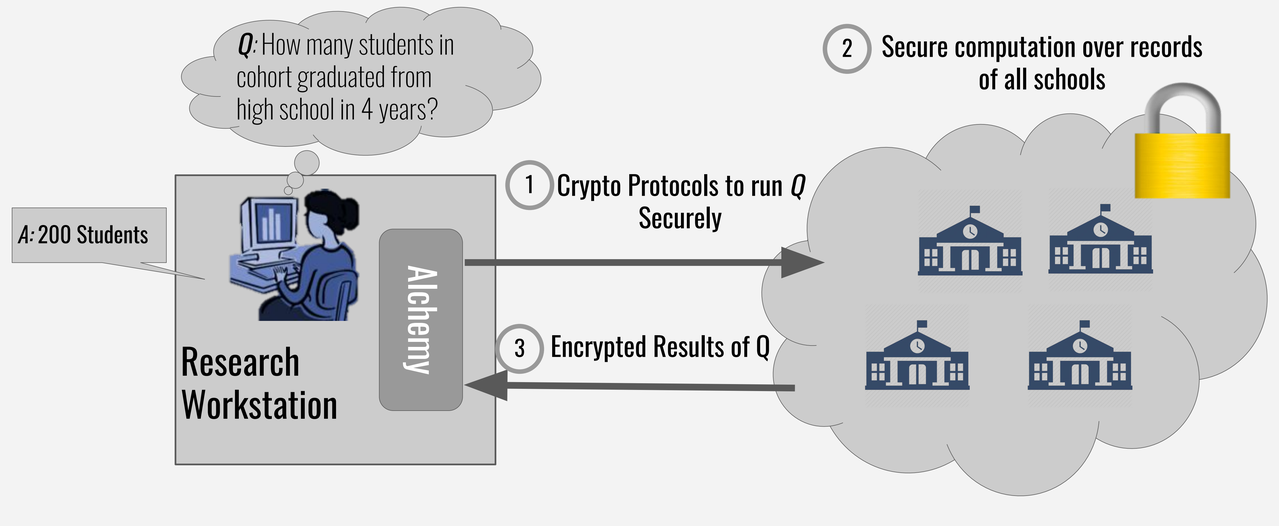
Solution

- ➔ Combine relational constraints with DP statistics
- ➔ Provide upper bounds for output cardinalties and runtimes
- ➔ Private query processing specific optimization rules

Challenges

- ➔ Histogram selection and parameterization
- ➔ End to End Privacy Guarantees in both results and stats
- ➔ Maximize result accuracy and query processing efficiency

Private Data Federations



- ➔ Traditional data federations require trust between data owners or a trusted third party
- ➔ In our setting data owners do not trust each other

Secure Multiparty Computation

- ➔ Group of *N* data owners want to securely compute function f on the union of their data
- ➔ Share only *encrypted data* for computation
- ➔ Can compute arbitrary functions