UNIVERSITY OF WATERLOO

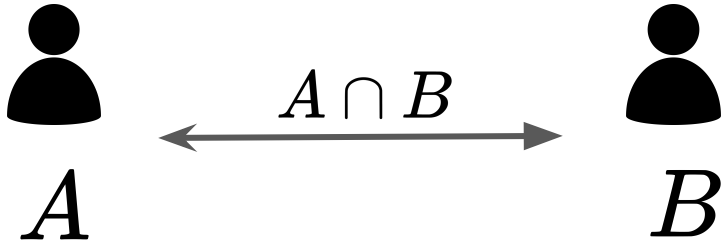
# Practical Over-Threshold Multi-Party Private Set Intersection

## Annual Computer Security Applications Conference (ACSAC 2020)

**Rasoul Akhavan Mahdavi**, Thomas Humphries, Bailey Kacsmar, Simeon Krastnikov,
Nils Lukas, John A. Premkumar, Masoumeh Shafieinejad
Simon Oya, Florian Kerschbaum, Erik-Oliver Blass

# Motivation

- Private Set Intersection (PSI)
  - Problem Description
  - Well studied problem
  - A variety of solutions
    - Multiple parties
    - Unbalanced set sizes

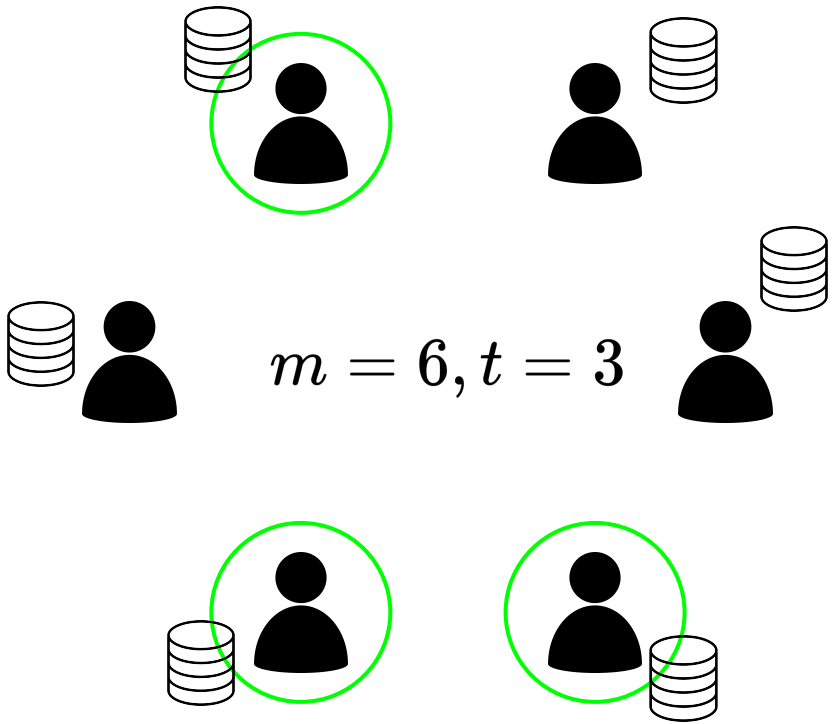



- Over-threshold Private Set Intersection
  - Problem Description
  - Parameterized by m, n and t
  - No Practical Deployments

# Motivation

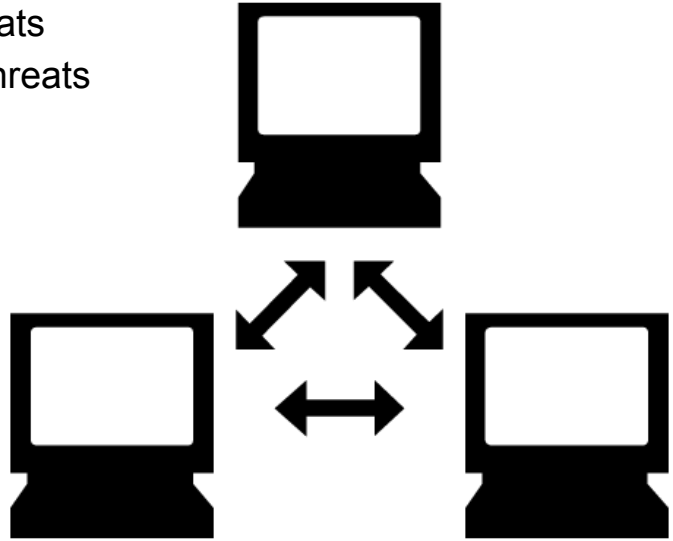
- Private Set Intersection (PSI)
  - Problem Description
  - Well studied problem
  - A variety of solutions
    - Multiple parties
    - Unbalanced set sizes

- Over-threshold Private Set Intersection
  - Problem Description
  - Parameterized by m, n and t
  - No Practical Deployments



# Motivation

- Use cases for Over Threshold PSI
  - Network operation centers collaborate to identify threats
  - Compare indicators of compromise to find common threats
  - Can’t release the indicators (privacy)
  - Intersection isn’t enough (over-threshold)



# Related Work

$m$: number of parties
$n$ : maximum set size
$t$ : threshold

| | Communication Rounds | Communication Complexity |
|---|---|---|
| Kissner & Song | $O(m)$ | $O(nm^3)$ |
| Secret Share MPC | $O(log^2(nm))$ | $O(nm^3 log^2(nm) + nm^3 t)$ |
| t-PSI | | |

# Related Work

$m$: number of parties
$n$ : maximum set size
$t$ : threshold
$k$ : number of keyholders/collusion threshold

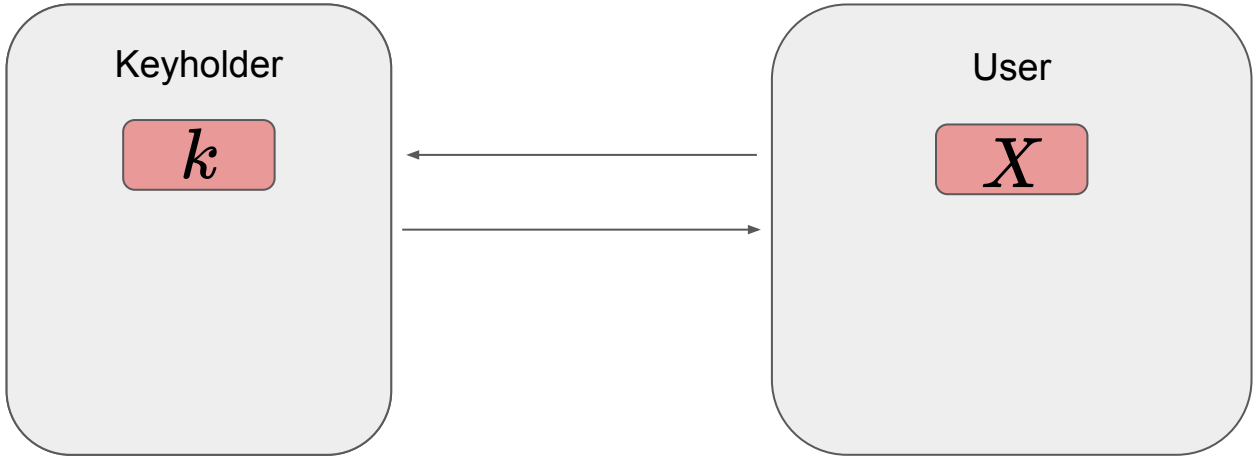
| | Communication Rounds | Communication Complexity |
|---|---|---|
| Kissner & Song | $O(m)$ | $O(nm^3)$ |
| Secret Share MPC | $O(log^2(nm))$ | $O(nm^3 log^2(nm) + nm^3 t)$ |
| t-PSI | $O(1)$ | $O(nmtk)$ |

# Our Work

- A new primitive: Oblivious Pseudo-Random Secret Sharing
- Our Over-Threshold PSI protocol
  - First attempt: $\mathrm{t\text{-}PSI}_0$
    - Less communication
    - Impractical Computation
  - More practical approach: $\mathrm{t\text{-}PSI}$
    - More communication
    - Practical Computation
- Experiments and Evaluation
  - Scalability for number of parties and set size

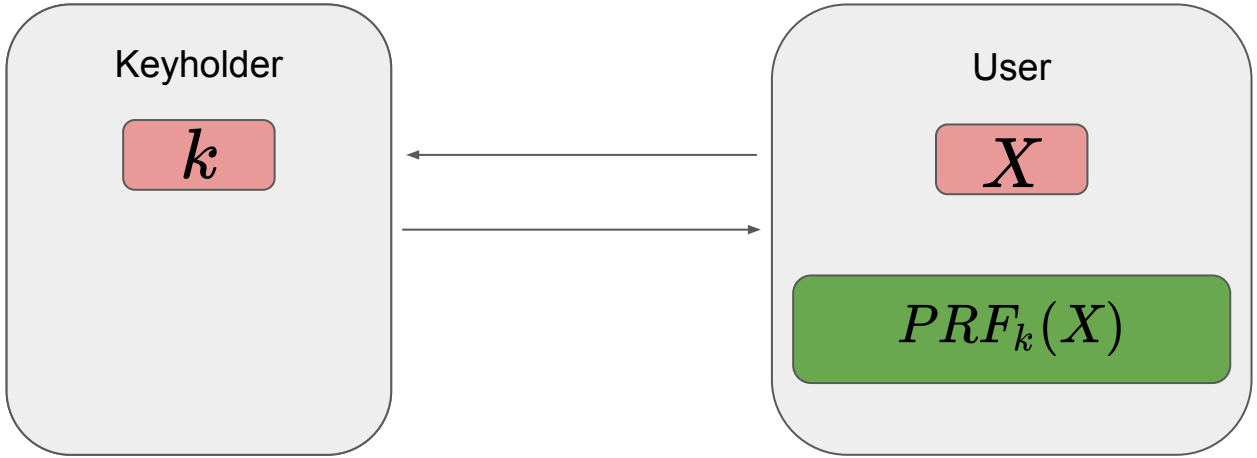
# Oblivious Pseudorandom Secret Sharing

Oblivious Pseudorandom Function



# Oblivious Pseudorandom Secret Sharing

## Oblivious Pseudorandom Function

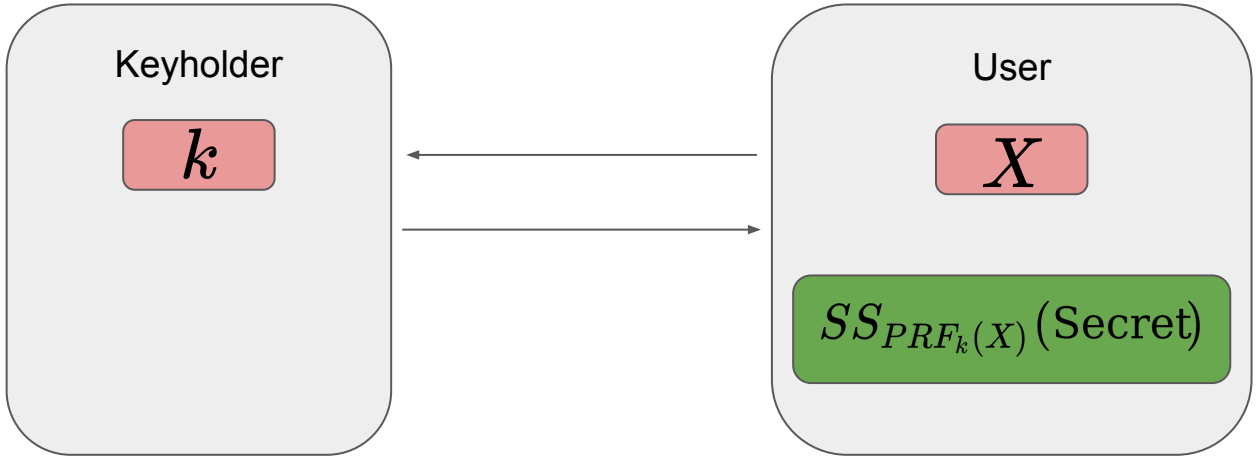



Keyholder doesn’t learn X

User doesn’t learn k

# Oblivious Pseudorandom Secret Sharing

## Oblivious Pseudorandom Secret Sharing Protocol

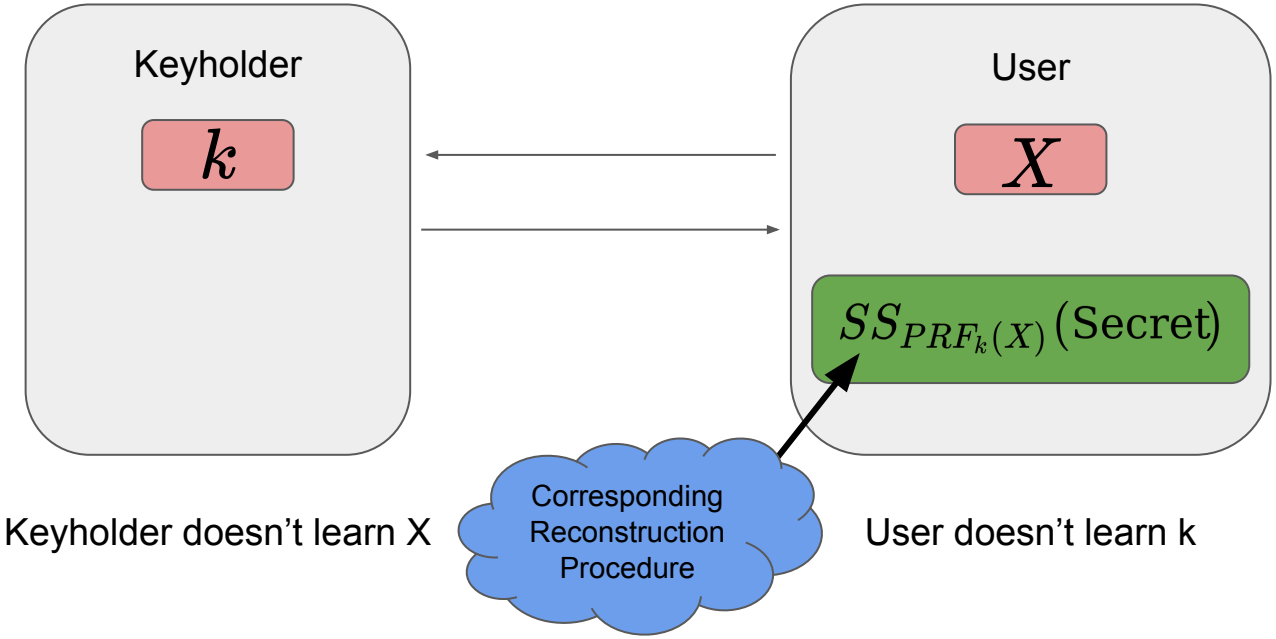



Keyholder doesn't learn X

User doesn't learn k

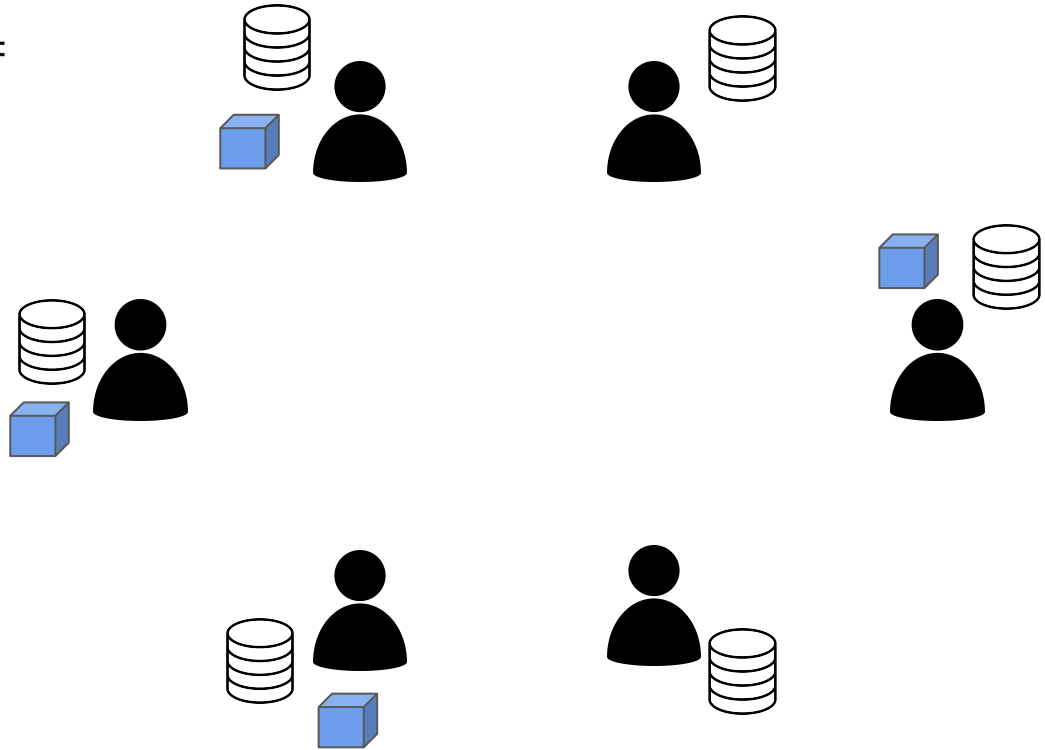
# Oblivious Pseudorandom Secret Sharing

## Oblivious Pseudorandom Secret Sharing Protocol



# Oblivious Pseudorandom Secret Sharing

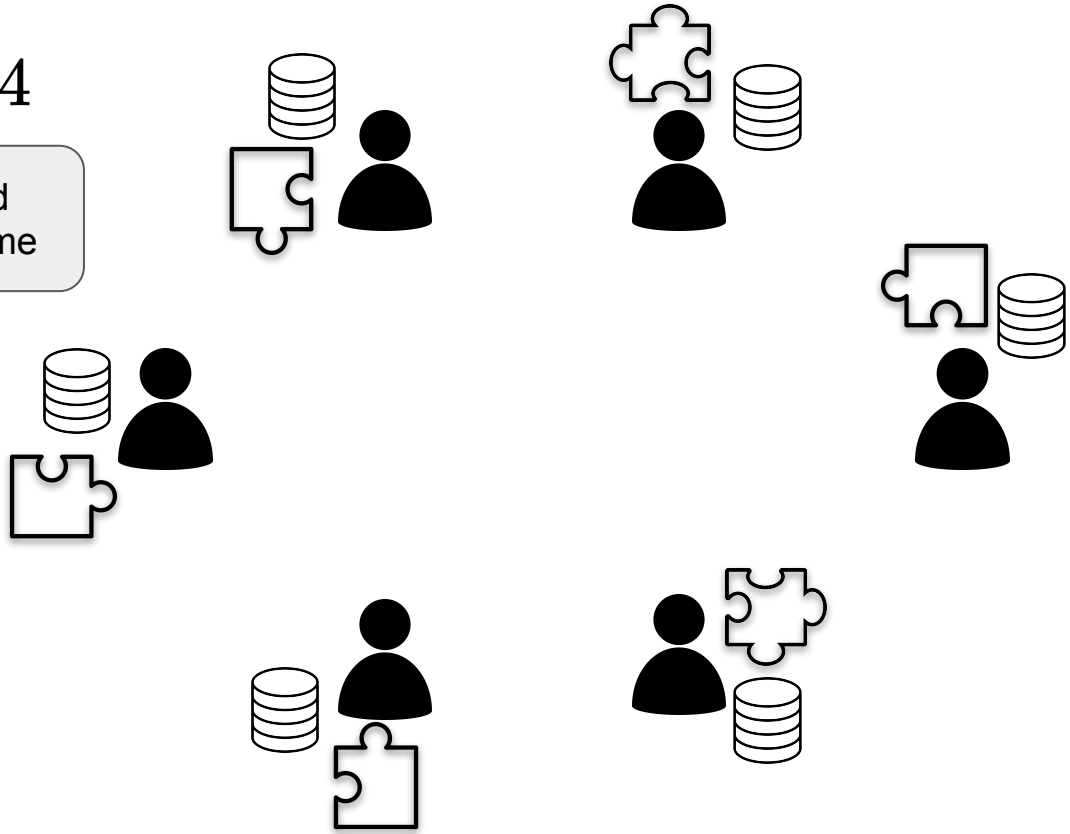
$m = 6, t = 4$



# Oblivious Pseudorandom Secret Sharing

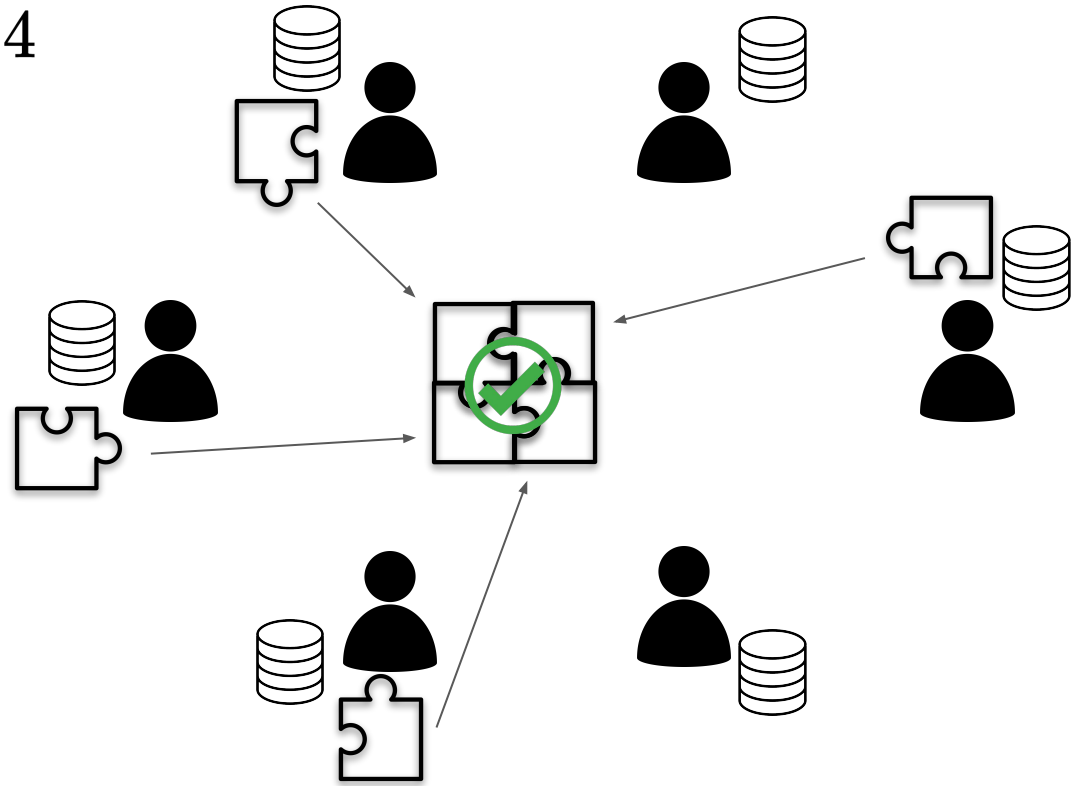
$m = 6, t = 4$

t out of m threshold secret sharing scheme



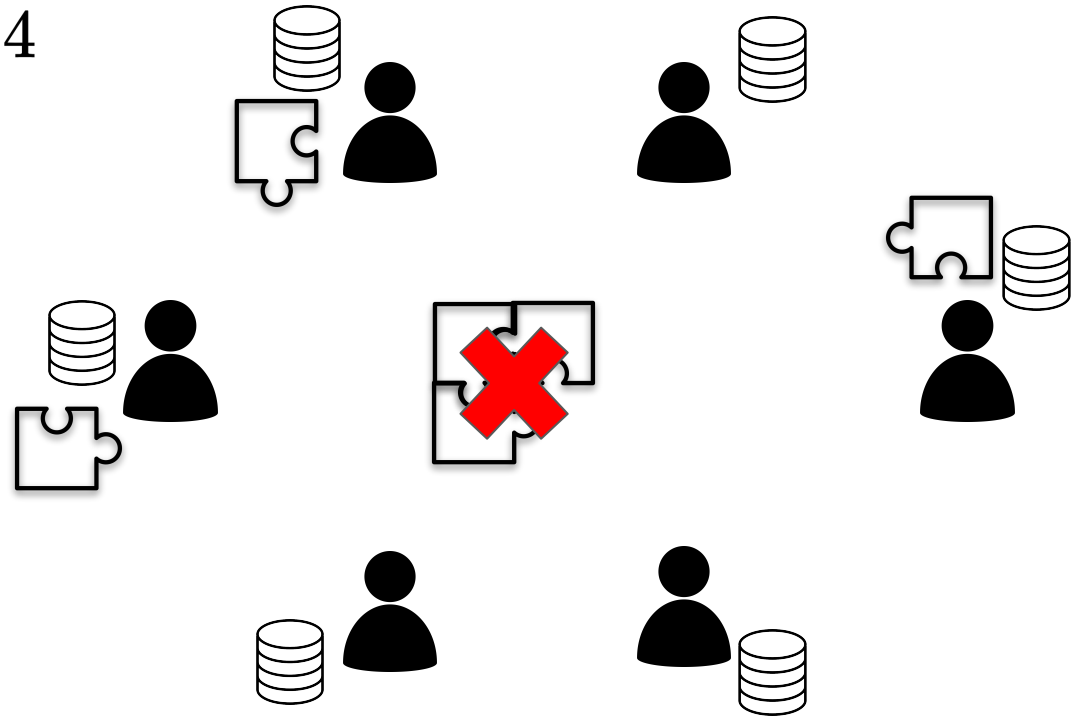
# Oblivious Pseudorandom Secret Sharing

$m = 6, t = 4$



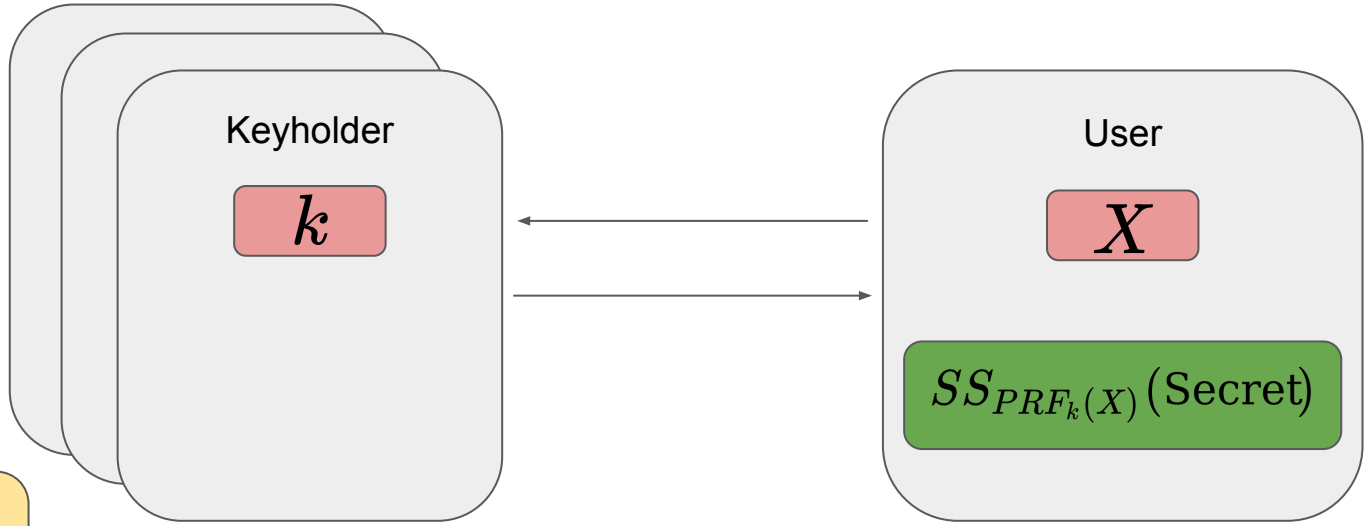
# Oblivious Pseudorandom Secret Sharing

$m = 6, t = 4$



# Oblivious Pseudorandom Secret Sharing

## Oblivious Pseudorandom Secret Sharing Protocol




Multiple Keyholders Allowed!

Keyholder doesn't learn X

User doesn't learn k

# OPR-SS Variants

$t$ : threshold
$k$ : number of keyholders

## First Variant

- Simpler share generation process
- 1 round of communication
- $O(k)$ communication complexity

- A very costly reconstruction

- Used in $\text{t-PSI}_0$

## Second Variant

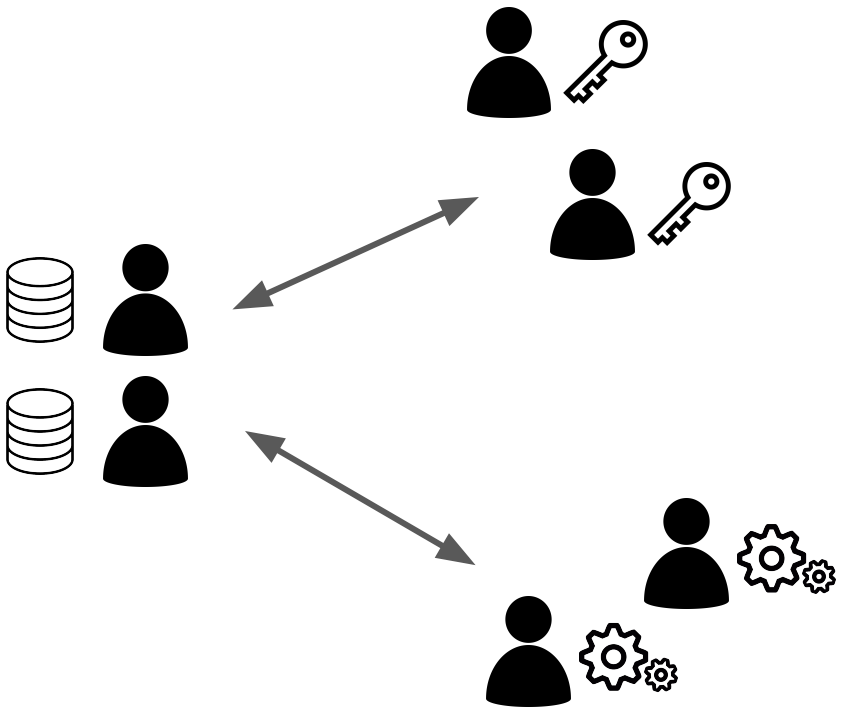
- More complex share generation process
- 2 round of communication
- $O(kt)$ communication complexity

- **Much more efficient reconstruction**
  (up to 1000 fold speedup)

- Used in $\text{t-PSI}$

# OT-MP-PSI Setup

- Participants (m)
  - Owners of data
- Keyholders (k)
  - Owners of a joint key
- Reconstructors
  - With computation power



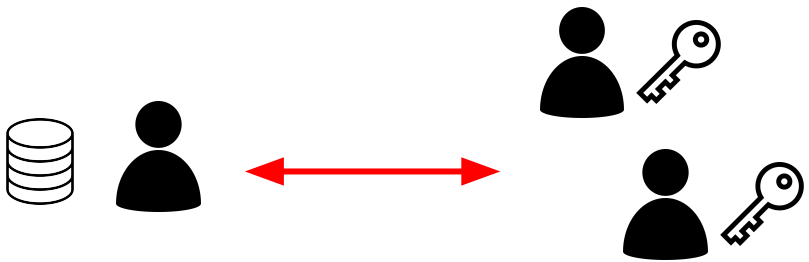
- Overlaps are allowed
- All entities are semi-honest

# OT-MP-PSI Protocol

- Share Generation
- Hashing-to-bins
- Reconstruction

# OT-MP-PSI Protocol

- Share Generation



- Hashing-to-bins

- Reconstruction

For each participant:
    For each element they own:
        Generate an OPR-SS for that element

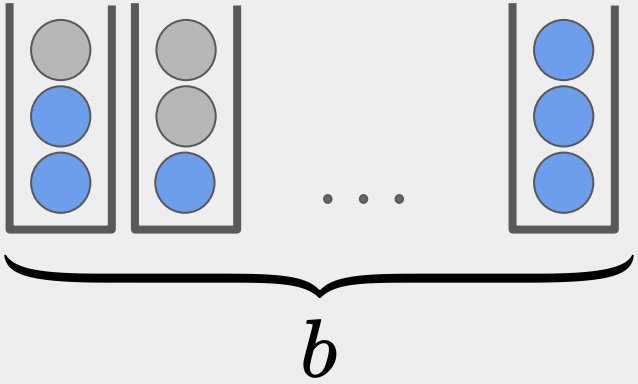
$\text{t-PSI}_0$ : use first variant
$\text{t-PSI}$ : use second variant

# OT-MP-PSI Protocol

- Share Generation

- Hashing-to-bins
  - Common technique
  - Done by each user separately
  - Number of bins is a tradeoff
- Reconstruction

- Predefined # of bins (b)
- Assign each element to bin (using hash)
- Pad the bins to max_size



# OT-MP-PSI Protocol

- Share Generation

- Hashing-to-bins

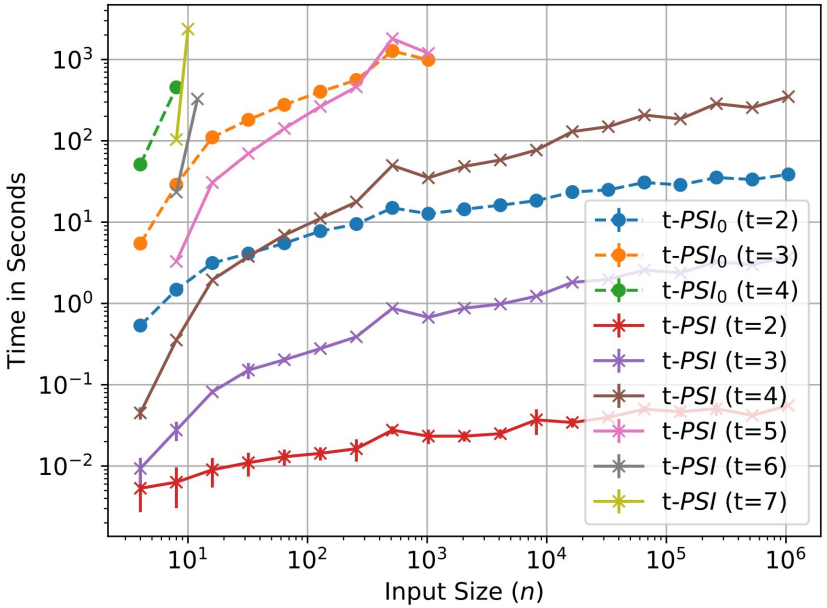
- Reconstruction
  - Can be parallelized

For each bin number:
For each combination of t participants:
For each combination of shares:
**Run reconstruction procedure**
if (success): report to participants

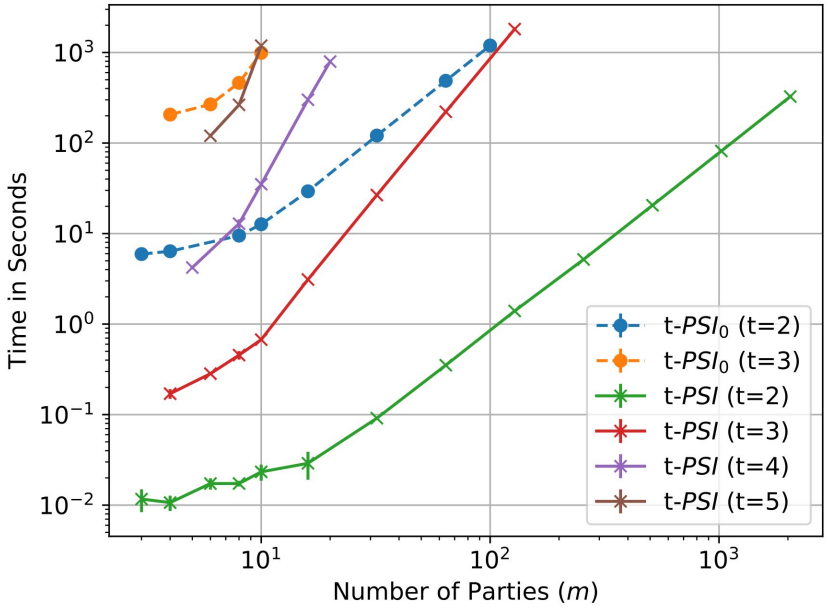
$\text{t-PSI}_0$: reconstruction for first variant (slow)
$\text{t-PSI}$ : reconstruction for second variant (fast)

# Experiments

Scalability for set sizes

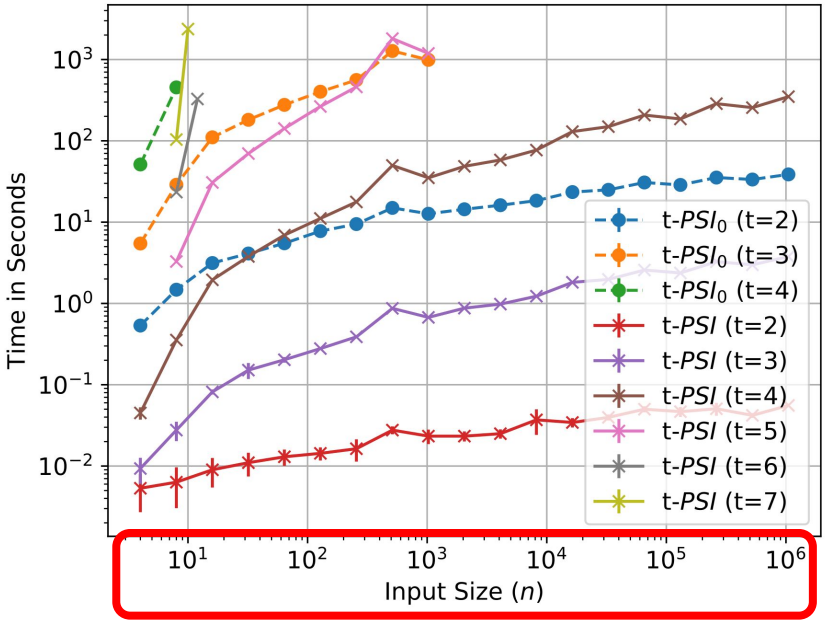



Scalability for number of participants

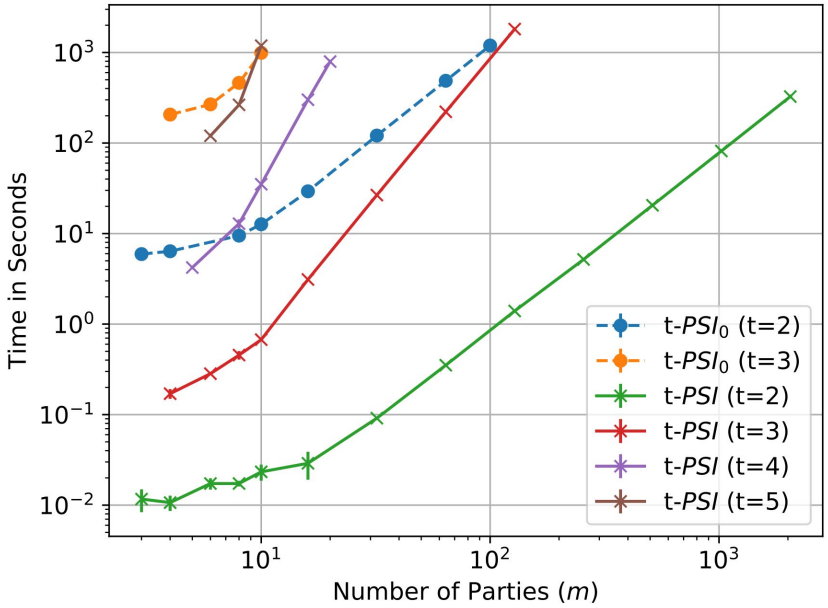


# Experiments

Scalability for set sizes

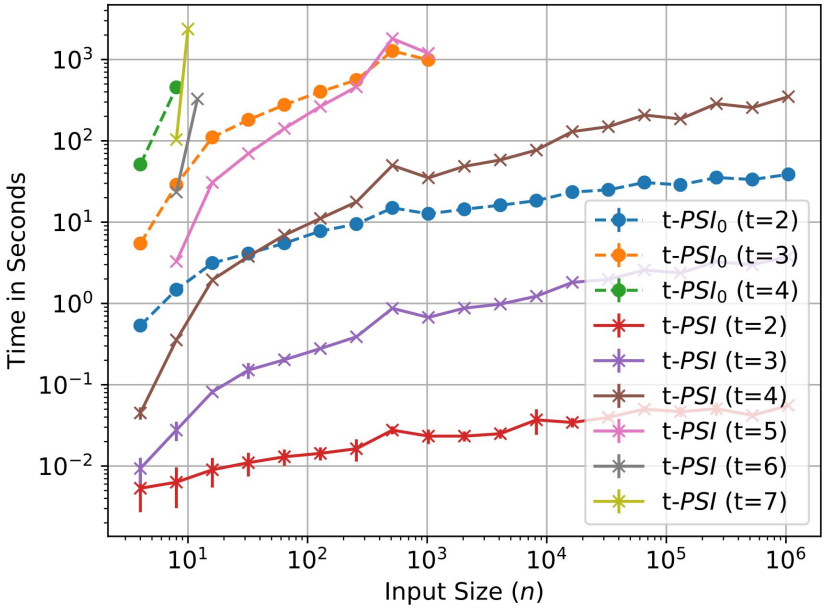



Scalability for number of participants

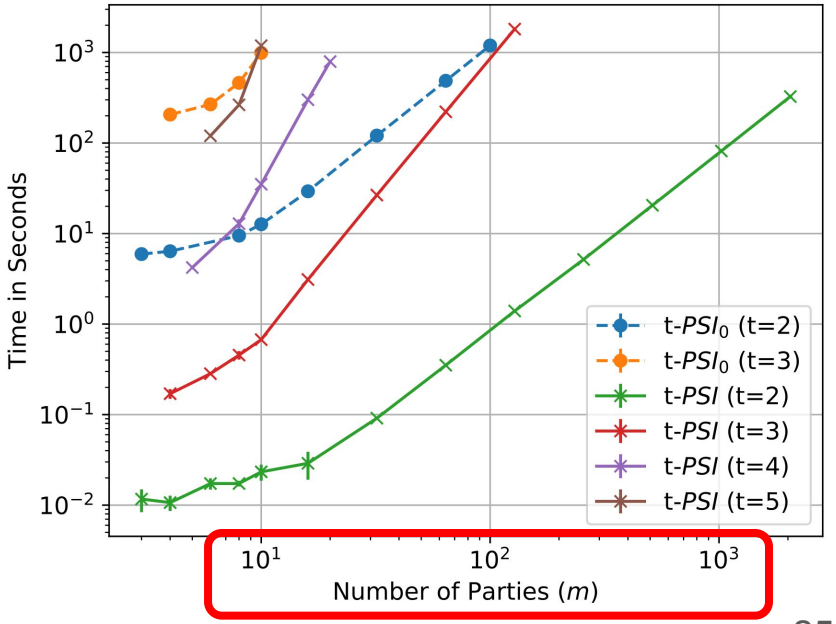


# Experiments

Scalability for set sizes




Scalability for number of participants



# Conclusion

- A new primitive, OPR-SS, for generating pseudorandom shares of a secret
- $\text{t-PSI}_0$
  - 1 round communication and $O(nmk)$ communication complexity
  - Expensive Reconstruction Stage
- $\text{t-PSI}$
  - 2 round communication and $O(nmkt)$ communication complexity
  - Cheaper Reconstruction Stage
- Experiments and Evaluation
  - Scalability up to millions of elements and thousands of clients
  - Public code on github (www.github.com/cryspuwaterloo/OT-MP-PSI)

# Thank you!

Questions?

UNIVERSITY OF WATERLOO

CrySP