





FLOSS:
Fast Linear Online Secret-Shared Shuffle

Ian Chang
University of Washington

Sela Navot
University of Washington

Alex Ozdemir
Georgia Tech
MPI-SP

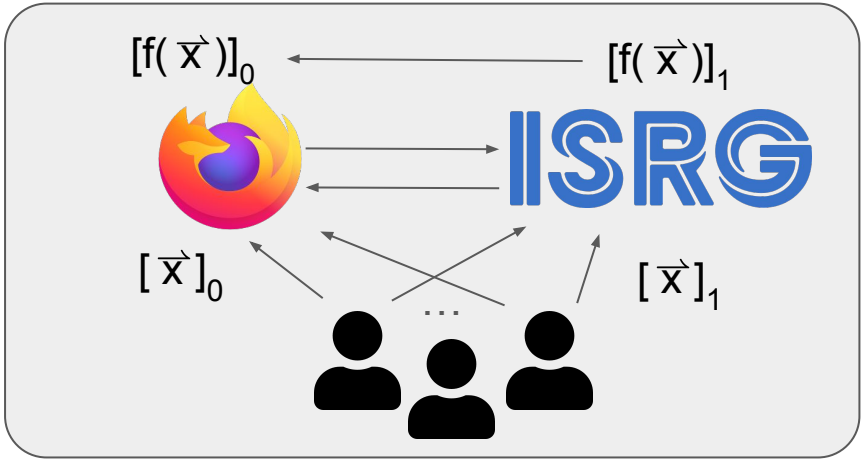
Nirvan Tyagi
University of Washington

Application: Private Analytics

- Goal: Aggregate statistics over users
- Ongoing standards:
  - Private-Preserving Measurement (IETF)
  - Private-Preserving Attribution (W3C)
- Idea: Multi-server multi-party computation

Application: Private Analytics

- Goal: Aggregate statistics over users
- Ongoing standards:
  - Private-Preserving Measurement (IETF)
  - Private-Preserving Attribution (W3C)
- Idea: Multi-server multi-party computation



Input: (Additive) sharing of $\vec{x}$: $[\vec{x}]_0 + [\vec{x}]_1 = \vec{x}$

Output: (Additive) sharing of $[f(\vec{x})]$: $[f(\vec{x})]_0 + [f(\vec{x})]_1 = f(\vec{x})$

The Problem: Secret-shared shuffling

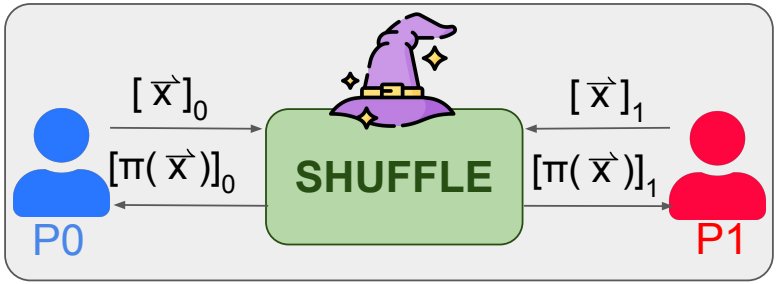
Applications of MPC analytics include:

- Ad attribution [CJKLMST23, etc.]
- Graph analytics [AFOPRT21, etc.]
- Contact discovery / PSI [KS08, etc.]
- Heavy hitters [AHIKNPT22, etc.]
- Database joins/filtering [MSS14, etc.]

The Problem: Secret-shared shuffling

Applications of MPC analytics include:

- Ad attribution [CJKLMST23, etc.]
- Graph analytics [AFOPRT21, etc.]
- Contact discovery / PSI [KS08, etc.]
- Heavy hitters [AHIKNPT22, etc.]
- Database joins/filtering [MSS14, etc.]



Input: $([\vec{x}]_0, [\vec{x}]_1)$

Output: for some **random** permutation π, $([\pi(\vec{x})]_0, [\pi(\vec{x})]_1)$

Property: $[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 = \pi(\vec{x})$

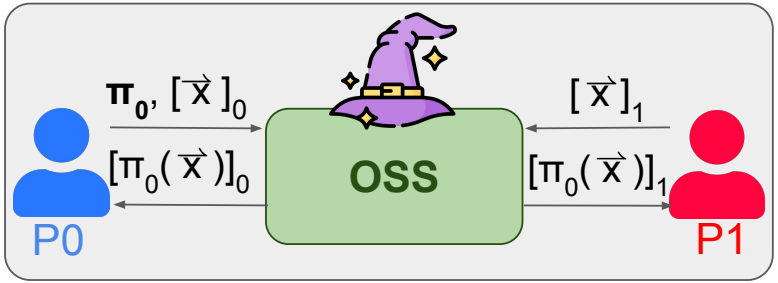
Observation: **Shuffling** is a key subprotocol!

This work: FLOSS

New 2PC Secret-Shared Shuffling Protocol

- Fast: Linear time online, 800x faster for shuffling 2^{20} entries
- Secure: Malicious-secure model

Relaxation: One-Sided Shuffle (OSS)

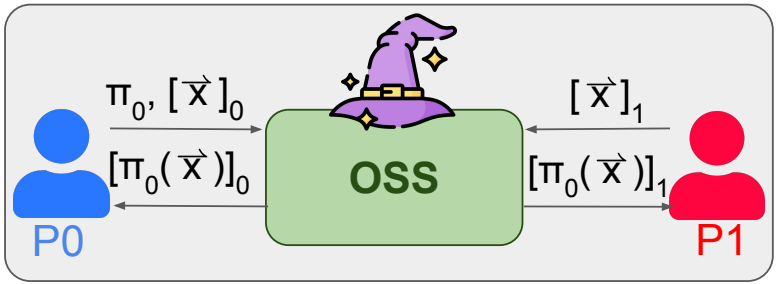


Input: $((\boldsymbol{\pi_0}, [\vec{x}]_0), [\vec{x}]_1)$

Output: $([\pi_0(\vec{x})]_0, [\pi_0(\vec{x})]_1)$

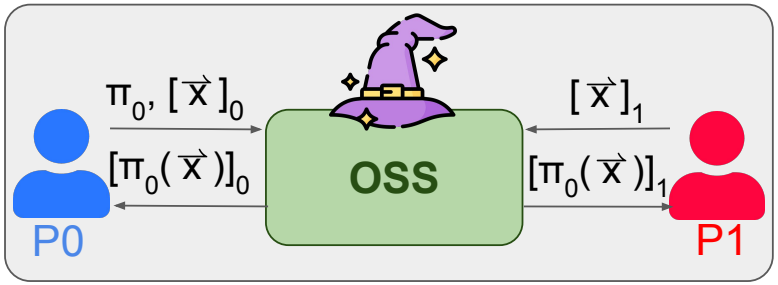
Property: $[\pi_0(\vec{x})]_0 + [\pi_0(\vec{x})]_1 = \boldsymbol{\pi_0}(\vec{x})$

Key Idea: Two-Sided Shuffle = Two Composed OSS

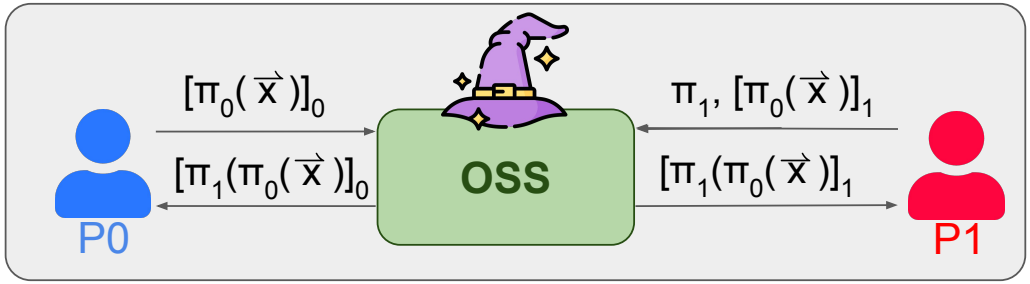


Output: $([\pi_0(\vec{x})]_0, [\pi_0(\vec{x})]_1)$

Key Idea: Two-Sided Shuffle = Two Composed OSS



Output: $([\pi_0(\vec{x})]_0, [\pi_0(\vec{x})]_1)$

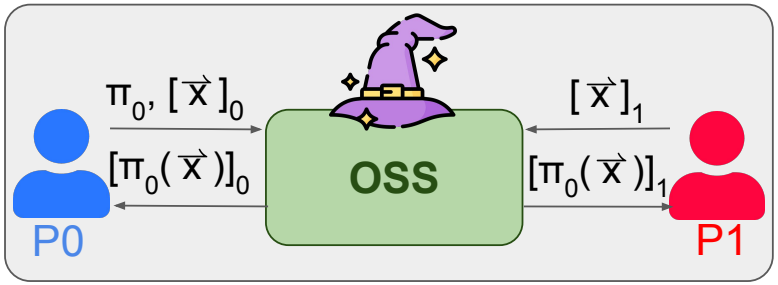


Input: $([\pi_0(\vec{x})]_0, (\pi_1, [\pi_0(\vec{x})]_1))$

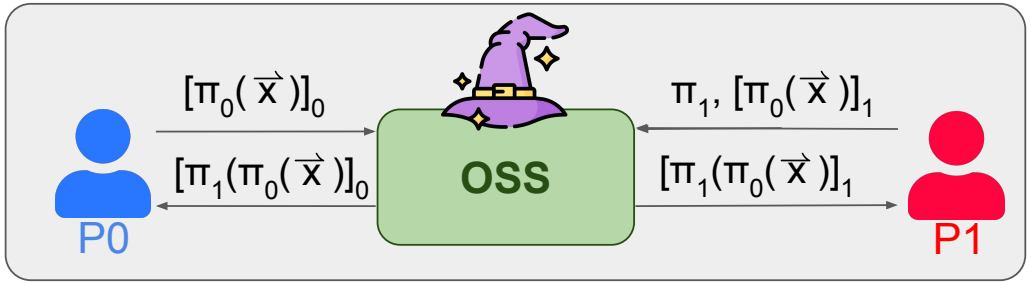
Output: $([\pi_1(\pi_0(\vec{x}))]_0, [\pi_1(\pi_0(\vec{x})]_1))$

Property: $[\pi_1(\pi_0(\vec{x}))]_0 + [\pi_1(\pi_0(\vec{x}))]_1 = (\pi_1 \circ \pi_0)(\vec{x})$

Key Idea: Two-Sided Shuffle = Two Composed OSS



Output: $([\pi_0(\vec{x})]_0, [\pi_0(\vec{x})]_1)$



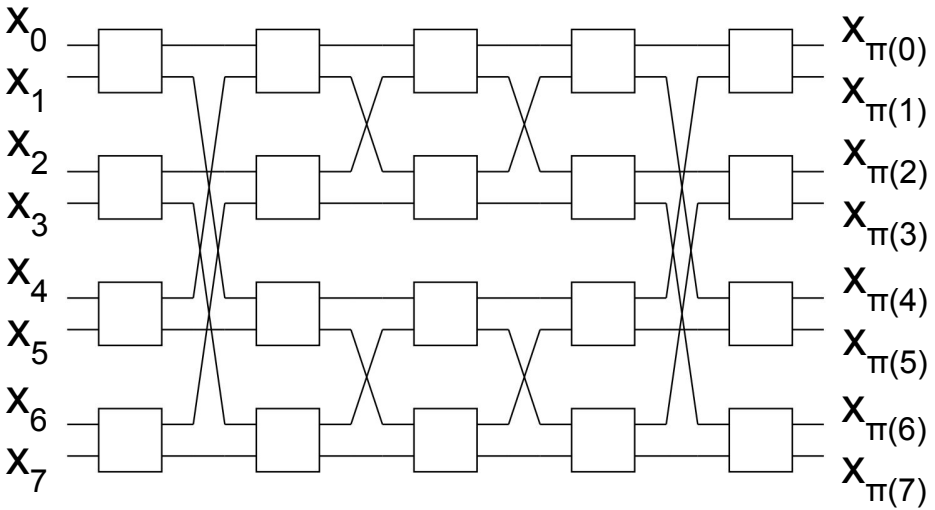
Input: $([\pi_0(\vec{x})]_0, (\pi_1, [\pi_0(\vec{x})]_1))$

Output: $([\pi_1(\pi_0(\vec{x}))]_0, [\pi_1(\pi_0(\vec{x})]_1))$

Property: $[\pi_1(\pi_0(\vec{x}))]_0 + [\pi_1(\pi_0(\vec{x}))]_1 = (\pi_1 \circ \pi_0)(\vec{x})$

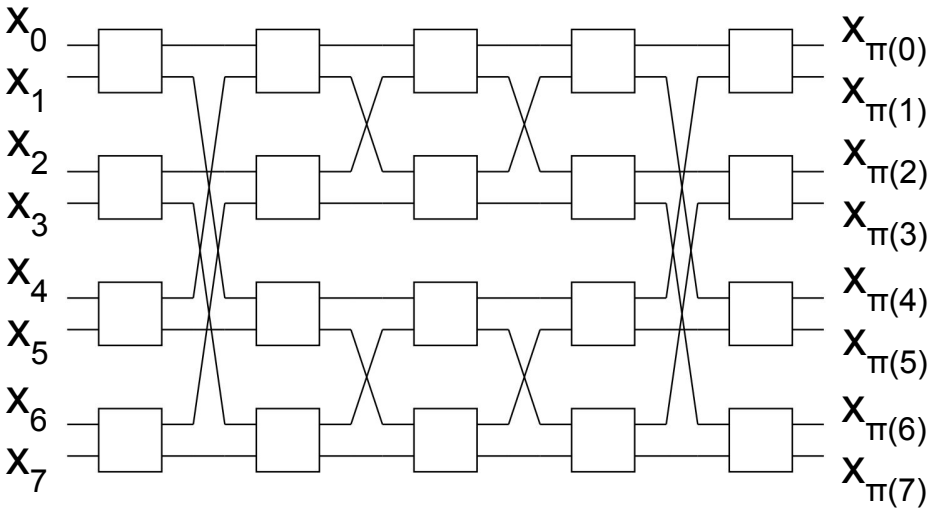
Observation: $\pi_1 \circ \pi_0$ appears **random** to P0 and P1!

Prior Work: Shuffling via Permutation Networks



- MPC of permutation network described as circuit [Waksman68, MSS14]
- **Quasilinear** in vector size: **O(n log(n))** MPC operations

Prior Work: Shuffling via Permutation Networks

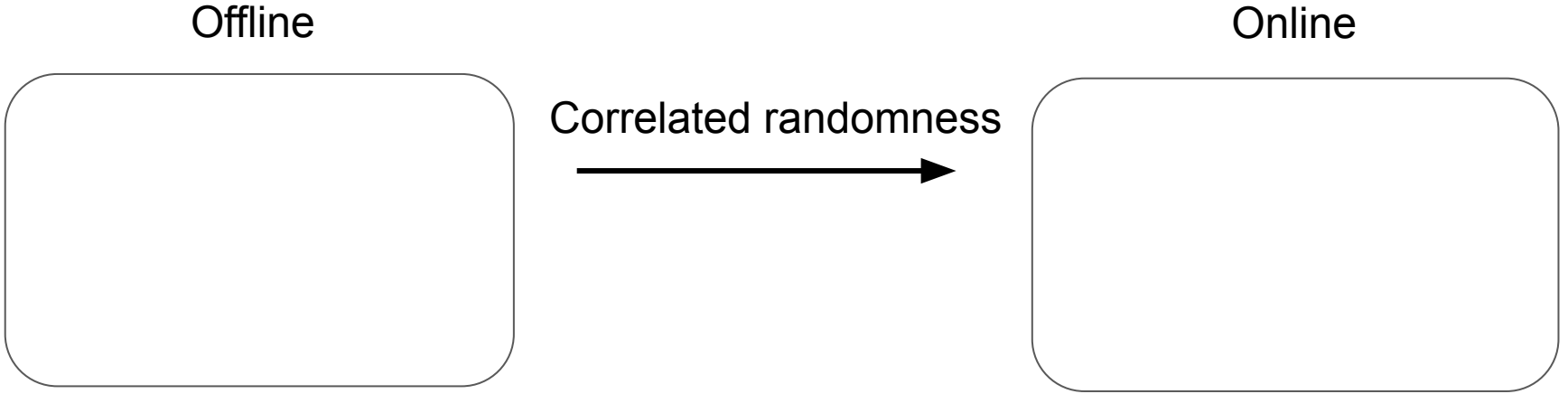


- MPC of permutation network described as circuit [Waksman68, MSS14]
- **Quasilinear** in vector size: **O(n log(n))** MPC operations

Can we achieve a **faster** secret-shared shuffle**?**

Key Idea: Move costs to offline preprocessing phase

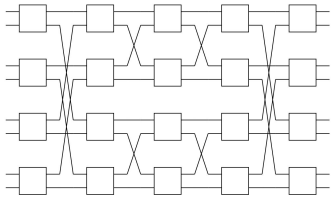
MPC Preprocessing model, e.g., Beaver triples for multiplication [DPSZ12]



Key Idea: Move costs to offline preprocessing phase

MPC Preprocessing model, e.g., Beaver triples for multiplication [DPSZ12]

Offline



Permutation Network

O(n log(n)) operations

Correlated randomness

Shuffle tuples

Online

FLOSS

O(n) operations

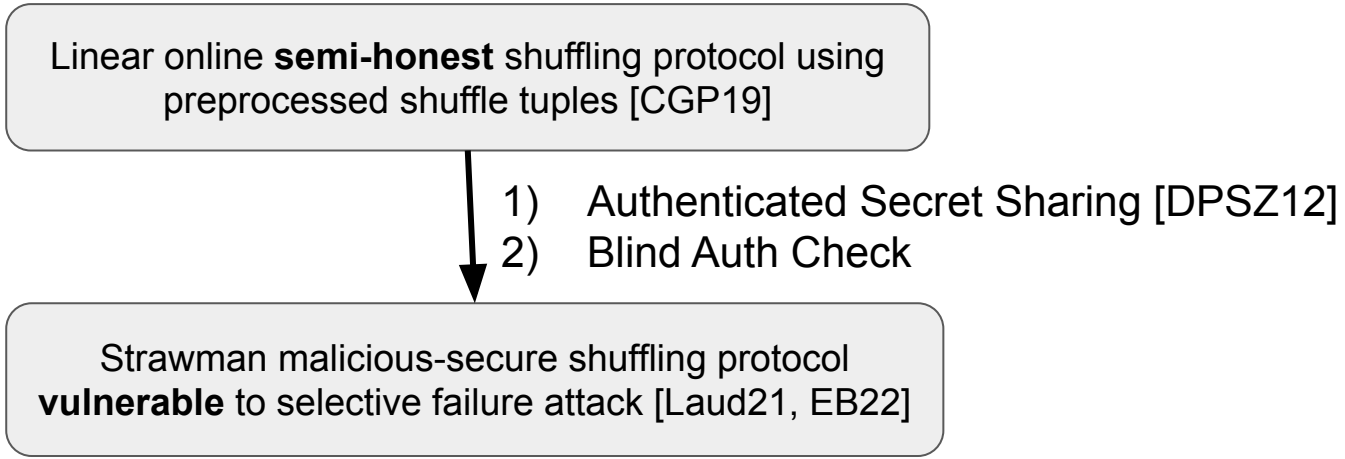
Preprocessing for shuffling:

- Honest-but-curious model: [CGP19]
- Malicious-secure model: Our work **FLOSS** [CNOT26]
- Concurrent: MOSAC [SLDBDDZC26]

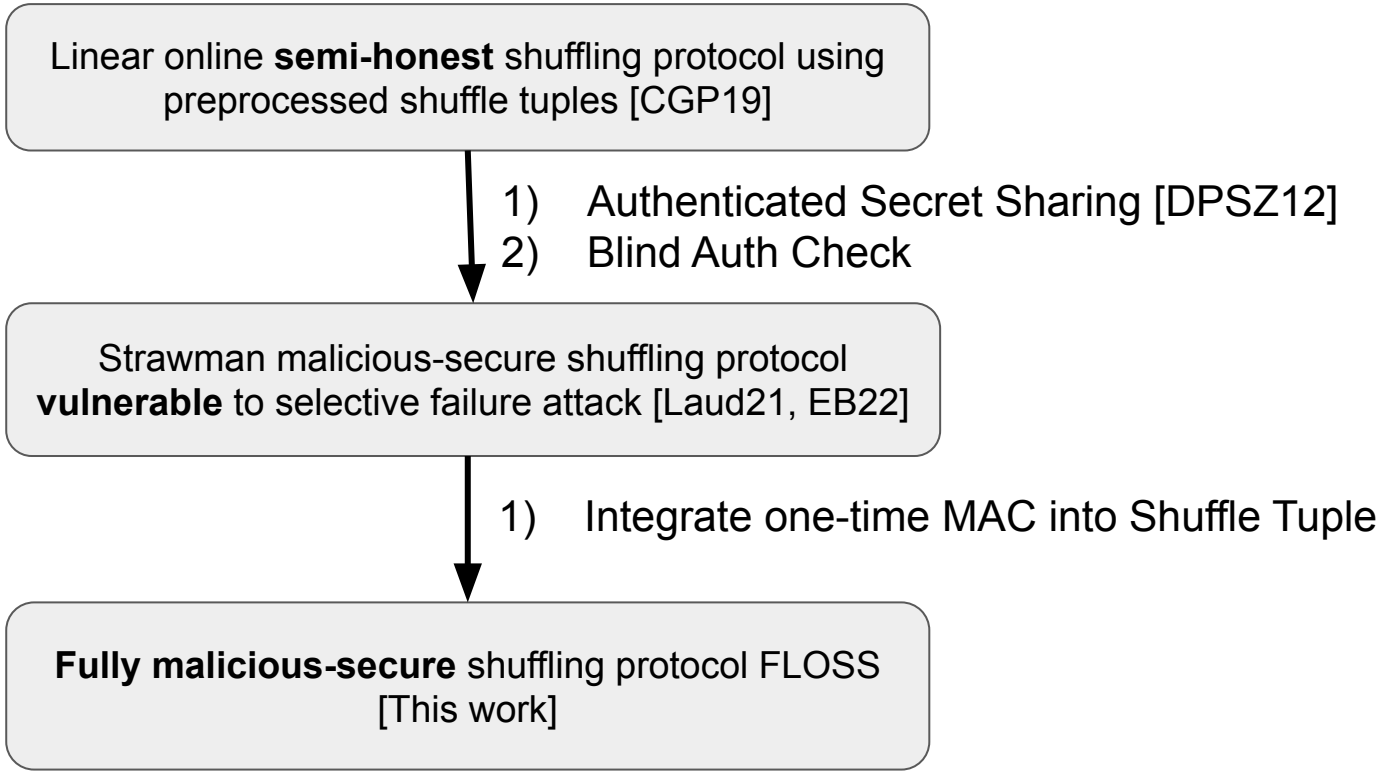
Our Construction: Roadmap

Linear online **semi-honest** shuffling protocol using preprocessed shuffle tuples [CGP19]

Our Construction: Roadmap



Our Construction: Roadmap



Semi-honest: CGP shuffle tuple

Party 0 selects permutation π

Party 1 selects random $\vec{a}$

Party 0

π, $[\pi(\vec{a})]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$

Semi-honest: CGP shuffle tuple

Party 0 selects permutation π

Party 1 selects random $\vec{a}$

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$



Semi-honest: CGP shuffle tuple

Party 0 selects permutation π

Party 1 selects random $\vec{a}$

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$

Semi-honest: CGP shuffle tuple

Party 0 selects permutation π

Party 1 selects random $\vec{a}$

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$

Semi-honest: CGP shuffle tuple

Party 0 selects permutation π **Party 1 selects random $\vec{a}$**

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$

$[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 =$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$

Semi-honest: CGP shuffle tuple

Party 0 selects permutation π **Party 1 selects random $\vec{a}$**

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$ ←

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$

$[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 =$

$= \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0 + [\pi(\vec{a})]_1$

Semi-honest: CGP shuffle tuple

Party 0 selects permutation π

Party 1 selects random $\vec{a}$

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$

$[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 =$

$= \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0 + [\pi(\vec{a})]_1$

$= \pi(\vec{x}) - \pi(\vec{a}) + \pi(\vec{a})$

Semi-honest: CGP shuffle tuple

Party 0 selects permutation π **Party 1 selects random $\vec{a}$**

Party 0

π, $[\pi(\vec{a})]_0$
$[\vec{x}]_0$

Party 1

$\vec{a}$, $[\pi(\vec{a})]_1$
$[\vec{x}]_1$

$[\vec{x}]_1 - \vec{a}$ ←

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$

$$[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 =$$
$$= \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0 + [\pi(\vec{a})]_1$$
$$= \pi(\vec{x}) - \pi(\vec{a}) + \pi(\vec{a})$$
$$= \pi(\vec{x})$$

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

Parties hold data share $[\vec{x}]$ and **authenticated share** $[\gamma\vec{x}]$

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π

Party 0

π, $[\pi(\vec{a})]_0$,$[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0$, $[\gamma\vec{x}]_0)$

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$,$[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1$, $[\gamma\vec{x}]_1)$

Parties hold data share $[\vec{x}]$ and **authenticated share** $[\gamma\vec{x}]$

Authenticated share $[\gamma\vec{x}]$ + Auth Check $\Rightarrow$ Prevents arbitrary tampering of shares [DPSZ12]

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π **Party 1 selects random $\vec{a}$, $\vec{b}$**

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π **Party 1 selects random $\vec{a}$, $\vec{b}$**

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0 + [\gamma]_1) \cdot ([\pi(\vec{x})]_0 + [\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0 + [\pi(\gamma\vec{x})]_1$

"Blind Auth Check" prevents arbitrary tampering of shares
without reconstructing shares

Strawman: Use Authenticated Shares [Laud21, EB22]

Party 0 selects permutation π | **Party 1 selects random $\vec{a}$, $\vec{b}$**

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0 + [\gamma]_1) \cdot ([\pi(\vec{x})]_0 + [\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0 + [\pi(\gamma\vec{x})]_1$

This protocol is **insecure**!

Strawman: Selective Failure Attack [Song et al.]

Party 0 selects permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$



Strawman: Selective Failure Attack [Song et al.]

Party 0 selects permutation π **Party 1 selects random $\vec{a}$, $\vec{b}$**

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1 - \pi'(\vec{e})$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0 + [\gamma]_1) \cdot ([\pi(\vec{x})]_0 + [\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0 + [\pi(\gamma\vec{x})]_1$

Strawman: Selective Failure Attack [Song et al.]

Party 0 selects permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 0

π, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1 - \pi'(\vec{e})$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0 + [\gamma]_1) \cdot ([\pi(\vec{x})]_0 + [\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0 + [\pi(\gamma\vec{x})]_1$

Strawman: Selective Failure Attack [Song et al.]

Party 0 selects permutation π **Party 1 selects random $\vec{a}$, $\vec{b}$**

Party 0

$\pi, [\pi(\vec{a})]_0, [\pi(\vec{b})]_0$
$[\gamma]_0, ([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}, \vec{b}, [\pi(\vec{a})]_1, [\pi(\vec{b})]_1$
$[\gamma]_1, ([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1 - \pi'(\vec{e})$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0 + [\gamma]_1) \cdot ([\pi(\vec{x})]_0 + [\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0 + [\pi(\gamma\vec{x})]_1$

$[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 = \pi(\vec{x}) + \pi(\vec{e}) - \pi'(\vec{e})$

Strawman: Selective Failure Attack [Song et al.]

Party 0 selects permutation π **Party 1 selects random $\vec{a}$, $\vec{b}$**

Party 0

$\pi, [\pi(\vec{a})]_0, [\pi(\vec{b})]_0$
$[\gamma]_0, ([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}, \vec{b}, [\pi(\vec{a})]_1, [\pi(\vec{b})]_1$
$[\gamma]_1, ([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1 - \pi'(\vec{e})$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0 + [\gamma]_1) \cdot ([\pi(\vec{x})]_0 + [\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0 + [\pi(\gamma\vec{x})]_1$

$[\pi(\vec{x})]_0 + [\pi(\vec{x})]_1 = \pi(\vec{x}) + \pi(\vec{e}) - \pi'(\vec{e})$

Selective Failure Attack: Blind Auth Check passes ⇔ P1 correctly guesses π' = π

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 0

π, **K**, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $\mathbf{[K\pi(\vec{a})]_0}$, $\mathbf{[K\pi(\vec{b})]_0}$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $\mathbf{[K\pi(\vec{a})]_1}$, $\mathbf{[K\pi(\vec{b})]_1}$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 0

π, $\mathbf{K}$, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $[\mathbf{K}\pi(\vec{a})]_0$, $[\mathbf{K}\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $[\mathbf{K}\pi(\vec{a})]_1$, $[\mathbf{K}\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

Idea: Permuter (Party 0) authenticates shuffle tuple with one-time MAC key **K**

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

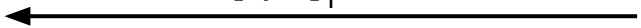
Party 0

$\pi, K, [\pi(\vec{a})]_0, [\pi(\vec{b})]_0, [K\pi(\vec{a})]_0, [K\pi(\vec{b})]_0$
$[\gamma]_0, ([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}, \vec{b}, [\pi(\vec{a})]_1, [\pi(\vec{b})]_1, [K\pi(\vec{a})]_1, [K\pi(\vec{b})]_1$
$[\gamma]_1, ([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$



Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

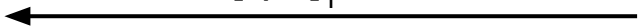
Party 0

π, K, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $[K\pi(\vec{a})]_0$, $[K\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $[K\pi(\vec{a})]_1$, $[K\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$
Set $[K\pi(\vec{x})]_0 = K \cdot \pi([\vec{x}]_0) + K \cdot \pi([\vec{x}]_1 - \vec{a}) + [K\pi(\vec{a})]_0$
Set $[K\pi(\gamma\vec{x})]_0 = K \cdot \pi([\gamma\vec{x}]_0) + K \cdot \pi([\gamma\vec{x}]_1 - \vec{b}) + [K\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$
Set $[K\pi(\vec{x})]_1 = [K\pi(\vec{a})]_1$
Set $[K\pi(\gamma\vec{x})]_1 = [K\pi(\vec{b})]_1$

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

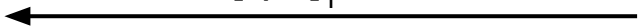
Party 0

π, K, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $[K\pi(\vec{a})]_0$, $[K\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $[K\pi(\vec{a})]_1$, $[K\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$



Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$
Set $[K\pi(\vec{x})]_0 = K \cdot \pi([\vec{x}]_0) + K \cdot \pi([\vec{x}]_1 - \vec{a}) + [K\pi(\vec{a})]_0$
Set $[K\pi(\gamma\vec{x})]_0 = K \cdot \pi([\gamma\vec{x}]_0) + K \cdot \pi([\gamma\vec{x}]_1 - \vec{b}) + [K\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$
Set $[K\pi(\vec{x})]_1 = [K\pi(\vec{a})]_1$
Set $[K\pi(\gamma\vec{x})]_1 = [K\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0+[\gamma]_1) \cdot ([\pi(\vec{x})]_0+[\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0+[\pi(\gamma\vec{x})]_1$
(in parallel) $([\gamma]_0+[\gamma]_1) \cdot ([K\pi(\vec{x})]_0+[K\pi(\vec{x})]_1) \stackrel{?}{=} [K\pi(\gamma\vec{x})]_0+[K\pi(\gamma\vec{x})]_1$

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 0

π, K, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $[K\pi(\vec{a})]_0$, $[K\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $[K\pi(\vec{a})]_1$, $[K\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$

←

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$
Set $[K\pi(\vec{x})]_0 = K \cdot \pi([\vec{x}]_0) + K \cdot \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [K\pi(\vec{a})]_0$
Set $[K\pi(\gamma\vec{x})]_0 = K \cdot \pi([\gamma\vec{x}]_0) + K \cdot \pi([\gamma\vec{x}]_1 - \vec{b}) + [K\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1 - \pi'(\vec{e})$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$
Set $[K\pi(\vec{x})]_1 = [K\pi(\vec{a})]_1 - K' \cdot \pi'(\vec{e})$
Set $[K\pi(\gamma\vec{x})]_1 = [K\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0+[\gamma]_1) \cdot ([\pi(\vec{x})]_0+[\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0+[\pi(\gamma\vec{x})]_1$
(in parallel) $([\gamma]_0+[\gamma]_1) \cdot ([K\pi(\vec{x})]_0+[K\pi(\vec{x})]_1) \stackrel{?}{=} [K\pi(\gamma\vec{x})]_0+[K\pi(\gamma\vec{x})]_1$

$[\pi(\vec{x})]_0+[\pi(\vec{x})]_1 = \pi(\vec{x}) + \pi(\vec{e}) - \pi'(\vec{e})$
$[K\pi(\vec{x})]_0+[K\pi(\vec{x})]_1 = K \cdot \pi(\vec{x}) + \mathbf{K \cdot \pi(\vec{e}) - K' \cdot \pi'(\vec{e})}$

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

Party 0

π, K, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $[K\pi(\vec{a})]_0$, $[K\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $[K\pi(\vec{a})]_1$, $[K\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a} + \vec{e}$
$[\gamma\vec{x}]_1 - \vec{b}$

Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$
Set $[K\pi(\vec{x})]_0 = K \cdot \pi([\vec{x}]_0) + K \cdot \pi([\vec{x}]_1 - \vec{a} + \vec{e}) + [K\pi(\vec{a})]_0$
Set $[K\pi(\gamma\vec{x})]_0 = K \cdot \pi([\gamma\vec{x}]_0) + K \cdot \pi([\gamma\vec{x}]_1 - \vec{b}) + [K\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1 - \pi'(\vec{e})$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$
Set $[K\pi(\vec{x})]_1 = [K\pi(\vec{a})]_1 - K' \cdot \pi'(\vec{e})$
Set $[K\pi(\gamma\vec{x})]_1 = [K\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0+[\gamma]_1) \cdot ([\pi(\vec{x})]_0+[\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0+[\pi(\gamma\vec{x})]_1$
(in parallel) $([\gamma]_0+[\gamma]_1) \cdot ([K\pi(\vec{x})]_0+[K\pi(\vec{x})]_1) \stackrel{?}{=} [K\pi(\gamma\vec{x})]_0+[K\pi(\gamma\vec{x})]_1$

Blind Auth Check passes ⇔ P1 correctly guesses π' = π **and** K' = K

$[\pi(\vec{x})]_0+[\pi(\vec{x})]_1 = \pi(\vec{x}) + \pi(\vec{e}) - \pi'(\vec{e})$
$[K\pi(\vec{x})]_0+[K\pi(\vec{x})]_1 = K \cdot \pi(\vec{x}) + K \cdot \pi(\vec{e}) - K' \cdot \pi'(\vec{e})$

Malicious-secure: FLOSS shuffle tuple

Party 0 selects random K, permutation π

Party 1 selects random $\vec{a}$, $\vec{b}$

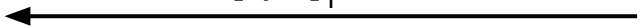
Party 0

π, K, $[\pi(\vec{a})]_0$, $[\pi(\vec{b})]_0$, $[K\pi(\vec{a})]_0$, $[K\pi(\vec{b})]_0$
$[\gamma]_0$, $([\vec{x}]_0, [\gamma\vec{x}]_0)$

Party 1

$\vec{a}$, $\vec{b}$, $[\pi(\vec{a})]_1$, $[\pi(\vec{b})]_1$, $[K\pi(\vec{a})]_1$, $[K\pi(\vec{b})]_1$
$[\gamma]_1$, $([\vec{x}]_1, [\gamma\vec{x}]_1)$

$[\vec{x}]_1 - \vec{a}$
$[\gamma\vec{x}]_1 - \vec{b}$



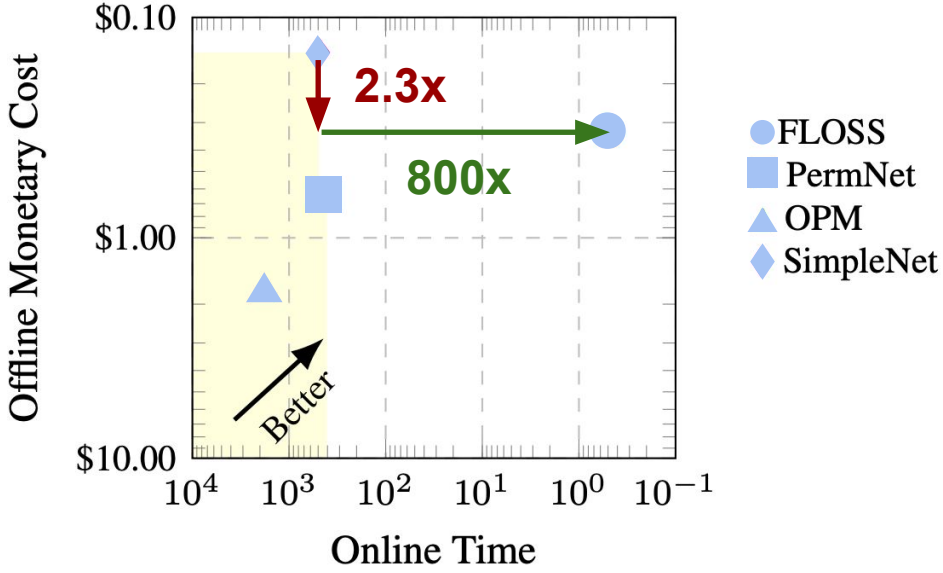
Set $[\pi(\vec{x})]_0 = \pi([\vec{x}]_0) + \pi([\vec{x}]_1 - \vec{a}) + [\pi(\vec{a})]_0$
Set $[\pi(\gamma\vec{x})]_0 = \pi([\gamma\vec{x}]_0) + \pi([\gamma\vec{x}]_1 - \vec{b}) + [\pi(\vec{b})]_0$
Set $[K\pi(\vec{x})]_0 = K \cdot \pi([\vec{x}]_0) + K \cdot \pi([\vec{x}]_1 - \vec{a}) + [K\pi(\vec{a})]_0$
Set $[K\pi(\gamma\vec{x})]_0 = K \cdot \pi([\gamma\vec{x}]_0) + K \cdot \pi([\gamma\vec{x}]_1 - \vec{b}) + [K\pi(\vec{b})]_0$

Set $[\pi(\vec{x})]_1 = [\pi(\vec{a})]_1$
Set $[\pi(\gamma\vec{x})]_1 = [\pi(\vec{b})]_1$
Set $[K\pi(\vec{x})]_1 = [K\pi(\vec{a})]_1$
Set $[K\pi(\gamma\vec{x})]_1 = [K\pi(\vec{b})]_1$

Blind Auth Check: $([\gamma]_0+[\gamma]_1) \cdot ([\pi(\vec{x})]_0+[\pi(\vec{x})]_1) \stackrel{?}{=} [\pi(\gamma\vec{x})]_0+[\pi(\gamma\vec{x})]_1$
(in parallel) $([\gamma]_0+[\gamma]_1) \cdot ([K\pi(\vec{x})]_0+[K\pi(\vec{x})]_1) \stackrel{?}{=} [K\pi(\gamma\vec{x})]_0+[K\pi(\gamma\vec{x})]_1$

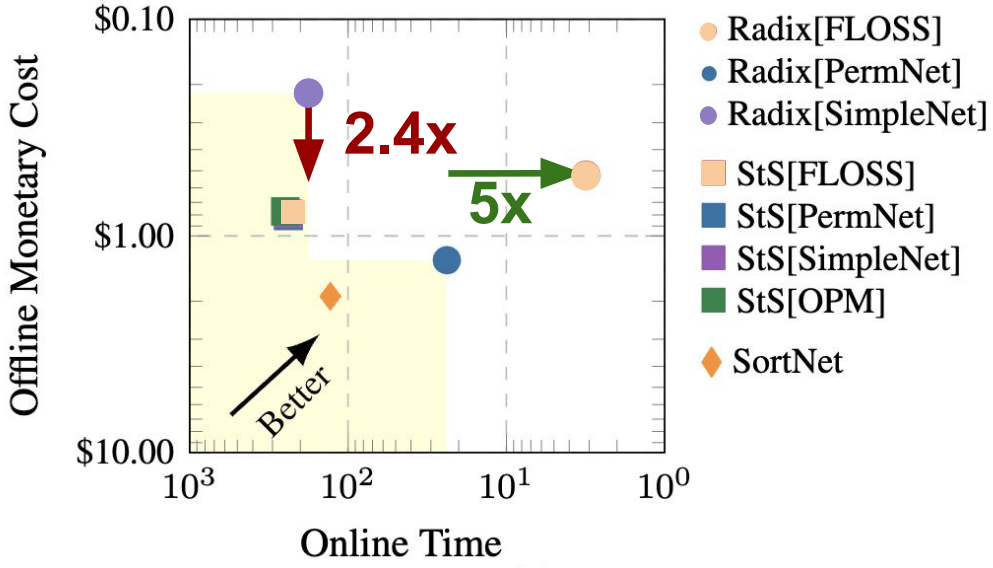
- **γ** prevents tampering of $[\vec{x}]$
- **K** prevents tampering the **shuffling** of $[\vec{x}]$ and $[\gamma\vec{x}]$

FLOSS: Shuffling Evaluation



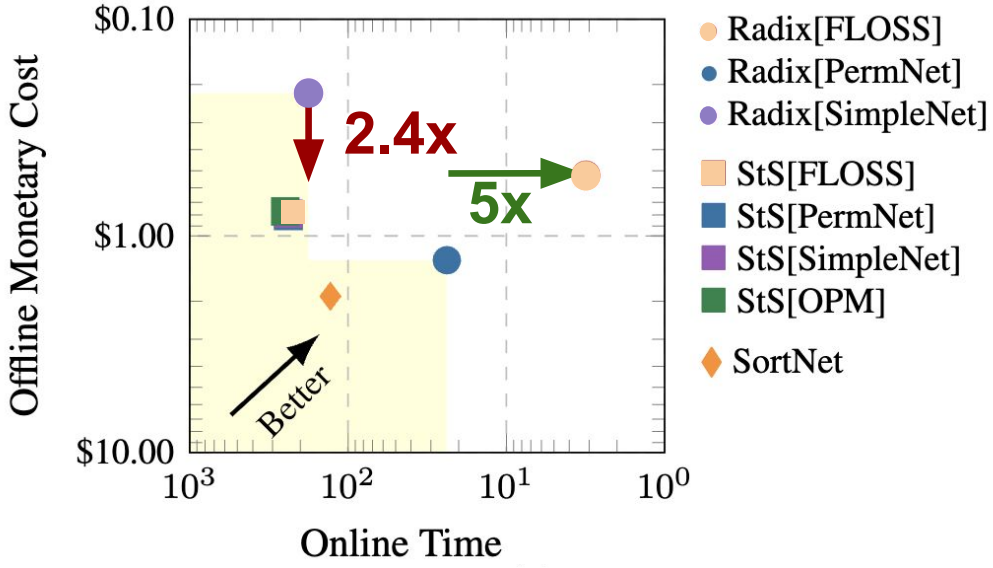
800x online speedup and **2x more expensive offline monetary cost** for shuffling vectors of size 2^{20}

FLOSS: Sorting Evaluation



5x online speedup and **2.4x more expensive offline monetary cost** for sorting vectors of size 2^{13}

FLOSS: Sorting Evaluation



5x online speedup and **2.4x more expensive offline monetary cost** for sorting vectors of size 2^{13}

ePrint: https://eprint.iacr.org/2026/672
Code: https://github.com/iachang/floss