# Efficient Set Membership Proofs using MPC-in-the-Head

https://eprint.iacr.org/2021/1656.pdf

Aarushi Goel (JHU), Mathias Hall-Andersen (Aarhus), **Gabriel Kaptchuk (BU)**, and Matthew Green (JHU)

BOSTON UNIVERSITY

# Set Membership Statements

$$x_1 \in \mathbf{L} \text{ or } x_2 \in \mathbf{L} \text{ or } \ldots \text{ or } x_\ell \in \mathbf{L}$$

# Set Membership Statements

$$x_1 \in \mathbf{L} \text{ or } x_2 \in \mathbf{L} \text{ or } \ldots \text{ or } x_\ell \in \mathbf{L}$$

$$R(x_1,w)=1 \text{ or } R(x_2,w)=1 \text{ or } \ldots \text{ or } R(x_\ell,w)=1$$

# Set Membership Statements

$$x_1 \in \mathbf{L} \text{ or } x_2 \in \mathbf{L} \text{ or } \ldots \text{ or } x_\ell \in \mathbf{L}$$

$$R(x_1,w)=1 \text{ or } R(x_2,w)=1 \text{ or } \ldots \text{ or } R(x_\ell,w)=1$$

$\alpha \in [\ell]$ is the "active branch"

# Set Membership Statements

- Hiding in a crowd

BOSTON UNIVERSITY

# Set Membership Statements

- Hiding in a crowd
- Ring Signatures

$$\mathrm{Verify}_m(pk_1,\sigma)=1 \text{ or } \mathrm{Verify}_m(pk_2,\sigma)=1 \text{ or } \dots \text{ or } \mathrm{Verify}_m(pk_\ell,\sigma)=1$$

# Set Membership Statements

- Hiding in a crowd

- Ring Signatures

$$\mathrm{Verify}_m(pk_1,\sigma)=1 \text{ or } \mathrm{Verify}_m(pk_2,\sigma)=1 \text{ or } \ldots \text{ or } \mathrm{Verify}_m(pk_\ell,\sigma)=1$$

- Confidential Transactions (ala. Monero or ZCash)

$$\mathrm{SpendVerify}(coin_1,\sigma)=1 \text{ or } \ldots \text{ or } \mathrm{SpendVerify}(coin_\ell,\sigma)=1$$

BOSTON UNIVERSITY

# Our Contributions

- Framework for Efficient Set Membership in MPC-in-the-Head
- Integration into known MPC-in-the-Head
- Applications:
    - Smallest Symmetric PQ ring signatures
    - Extremely Simple RingCT Transactions

BOSTON UNIVERSITY

# MPC-in-the-head [IKOS07]

$\text{Prover}_{x,w}$

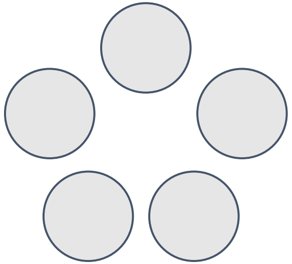
$\text{Verifier}_{x}$

BOSTON UNIVERSITY

# MPC-in-the-head [IKOS07]

Prover$_{x,w}$

Verifier$_{x}$

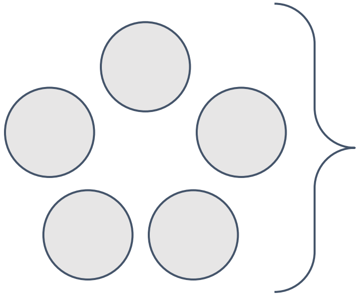


MPC over Relation Circuit

# MPC-in-the-head [IKOS07]

Prover$_{x,w}$

Verifier$_{x}$

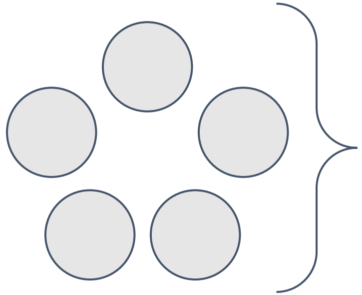


a = Com(Views) $\xrightarrow{a}$

MPC over Relation Circuit

# MPC-in-the-head [IKOS07]

Prover$_{x,w}$



a = Com(Views)

MPC over Relation Circuit

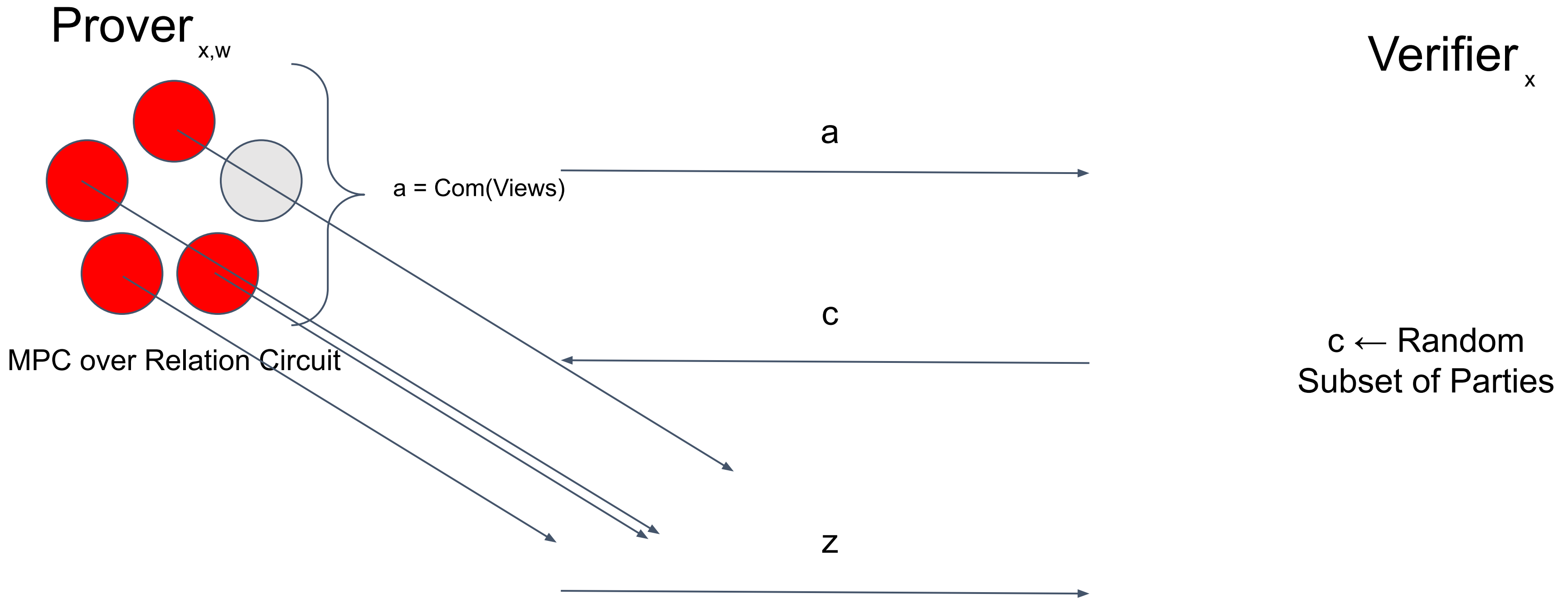
a

c

Verifier$_{x}$

c ← Random Subset of Parties

# MPC-in-the-head [IKOS07]

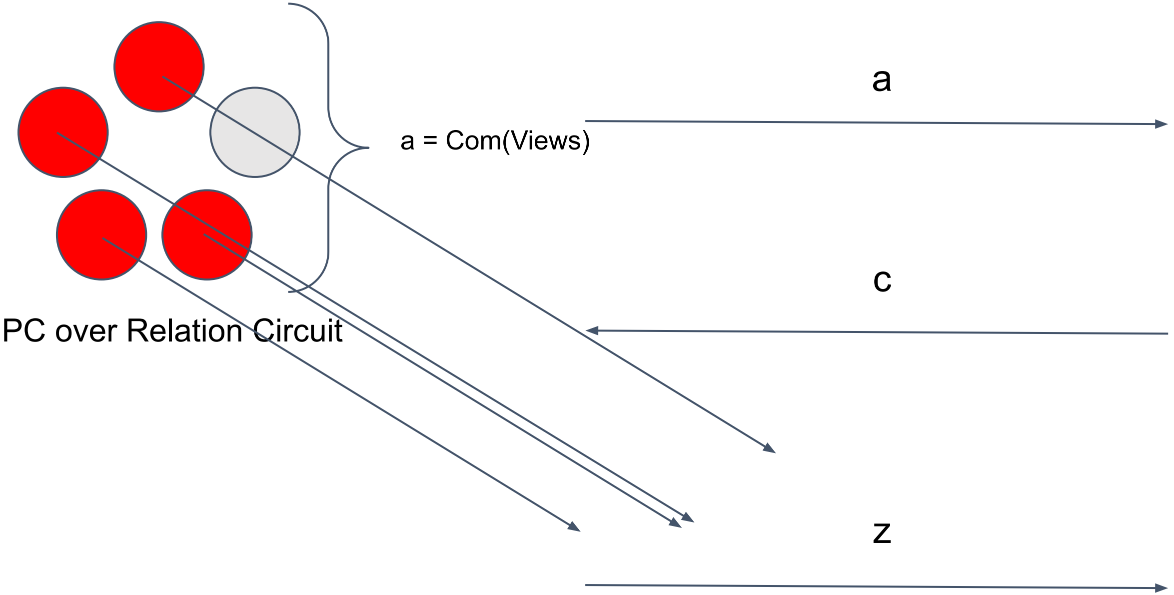



BOSTON UNIVERSITY

# MPC-in-the-head [IKOS07]

Prover $_{x,w}$

Verifier $_{x}$

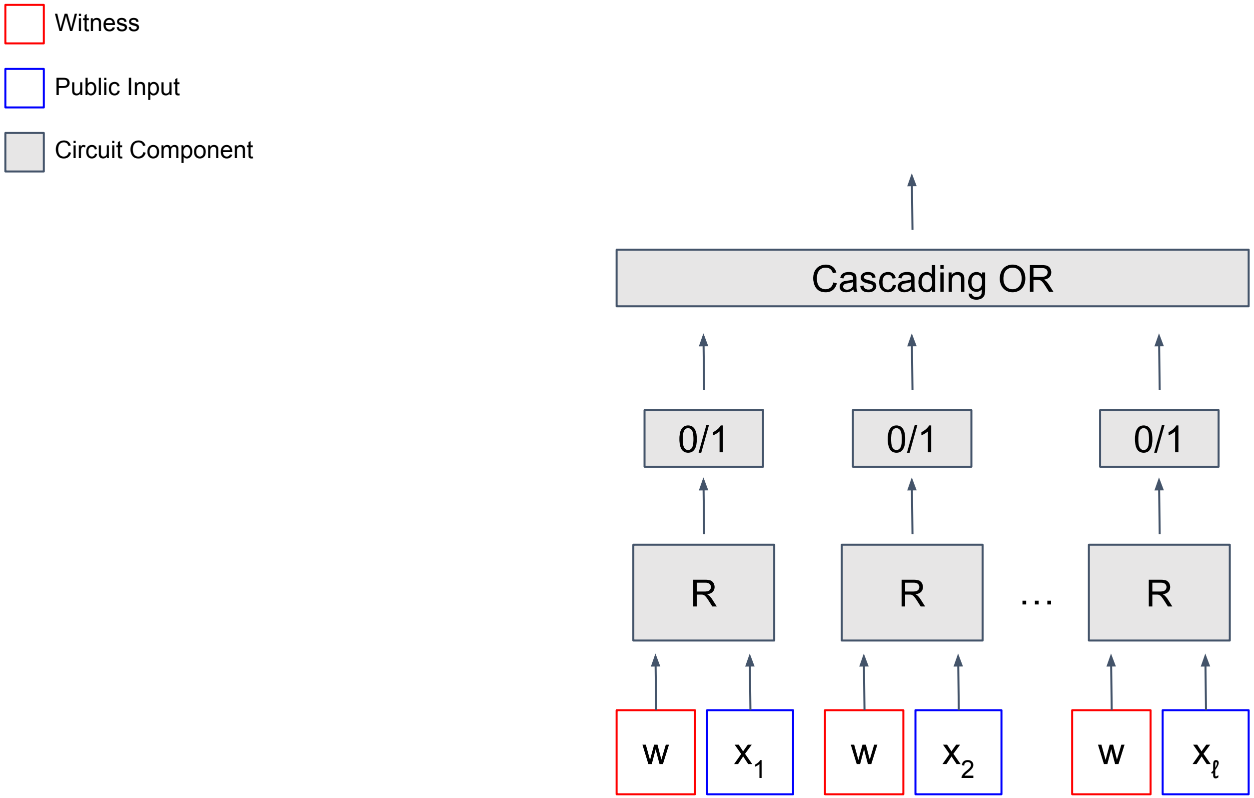



c ← Random
Subset of Parties

Verify Consistency
of Views

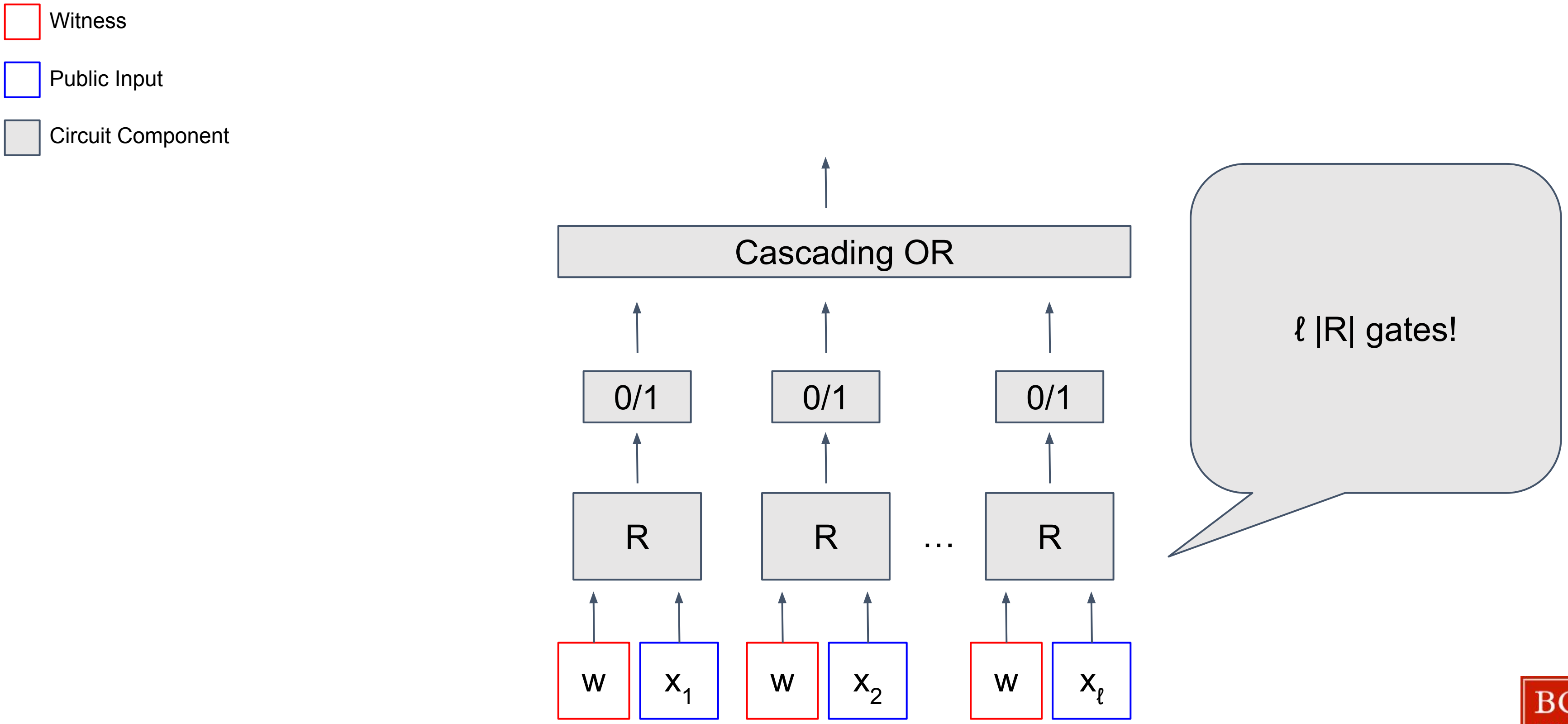
# Representation 1: Naive Repetition

$R(x_1,w)=1$ or $R(x_2,w)=1$ or ... or $R(x_\ell,w)=1$



# Representation 1: Naive Repetition

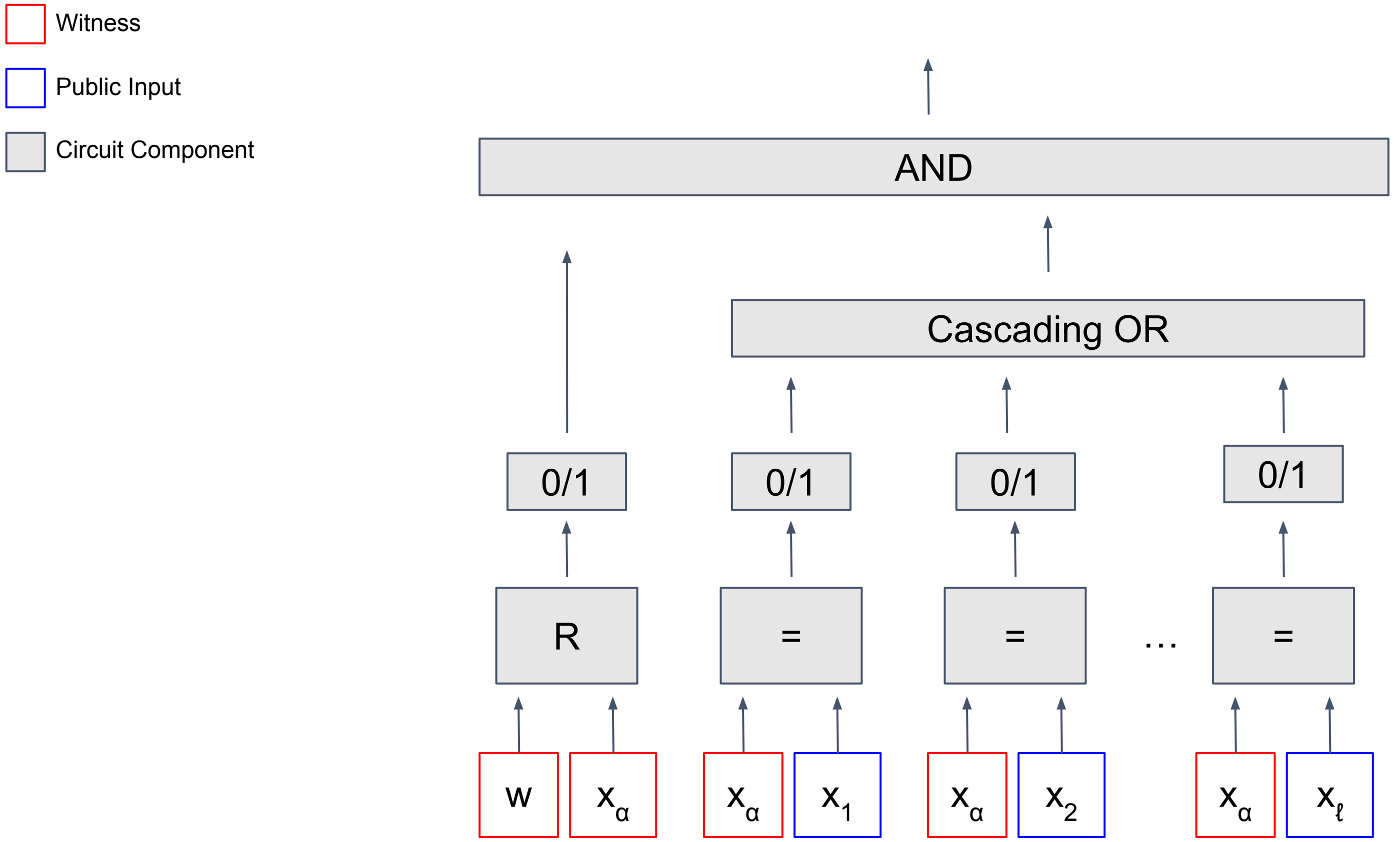
$R(x_1,w)=1$ or $R(x_2,w)=1$ or ... or $R(x_\ell,w)=1$




BOSTON UNIVERSITY

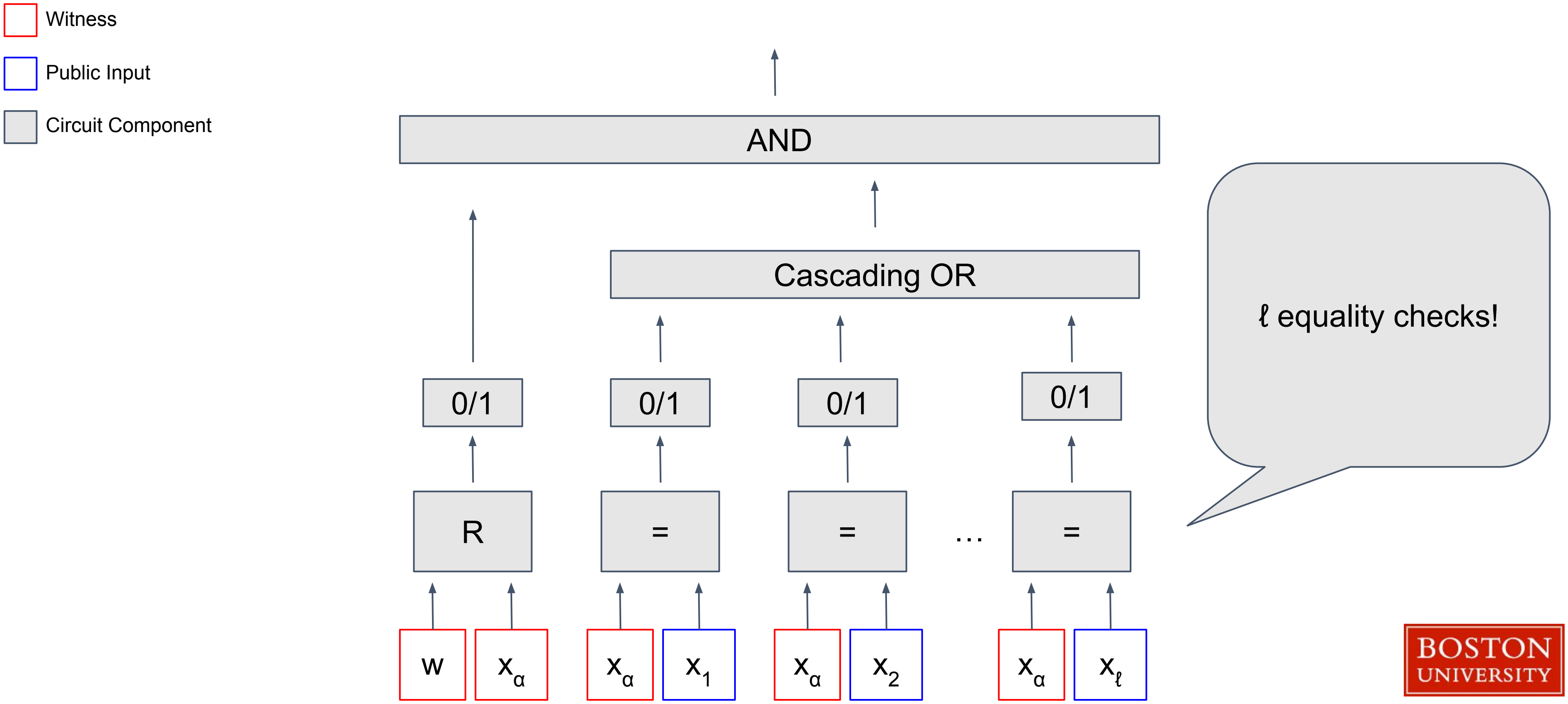
# Representation 2: Equality Check

$R(x_1,w)=1$ or $R(x_2,w)=1$ or ... or $R(x_\ell,w)=1$



# Representation 2: Equality Check

$R(x_1,w)=1$ or $R(x_2,w)=1$ or ... or $R(x_\ell,w)=1$



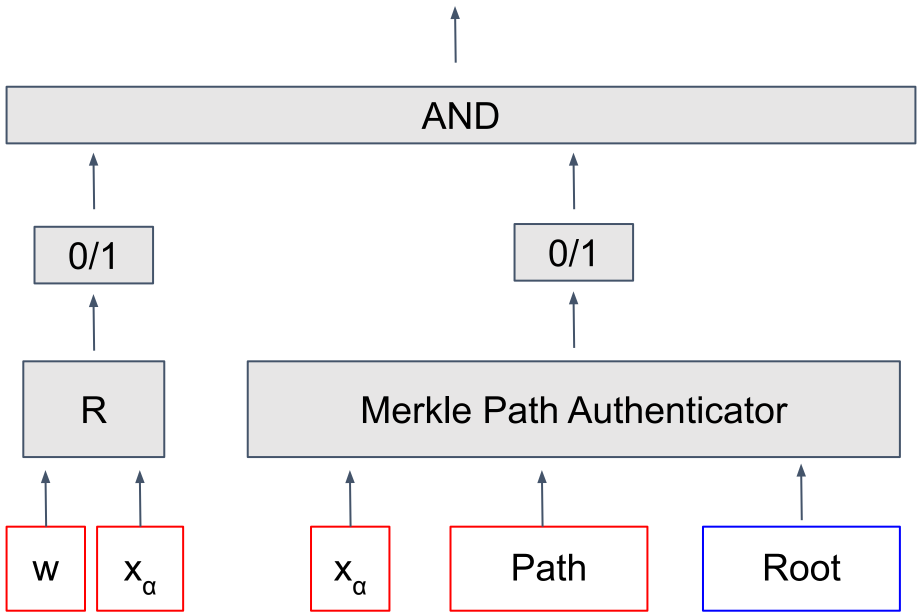
# Representation 3: Merkle Tree

$R(x_1,w)=1$ or $R(x_2,w)=1$ or ... or $R(x_\ell,w)=1$

Witness

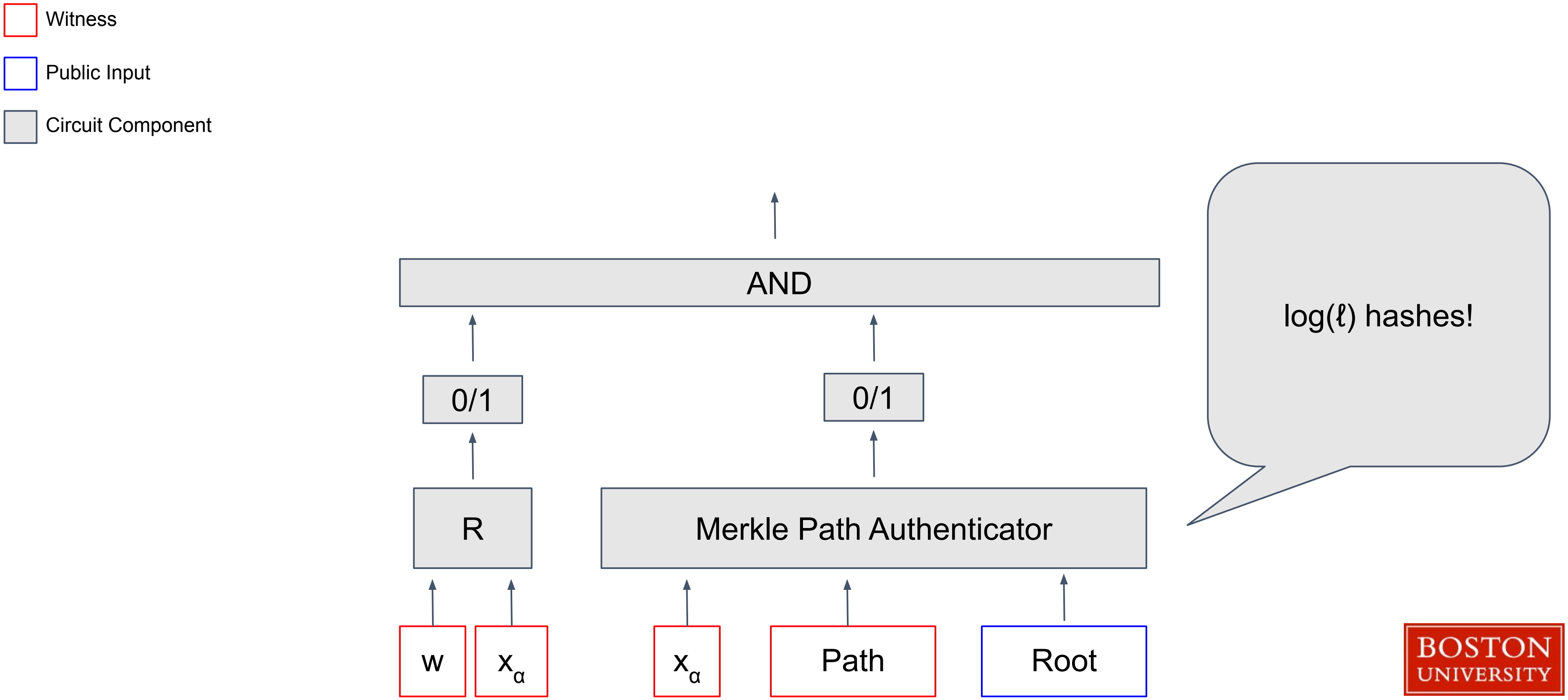
Public Input

Circuit Component

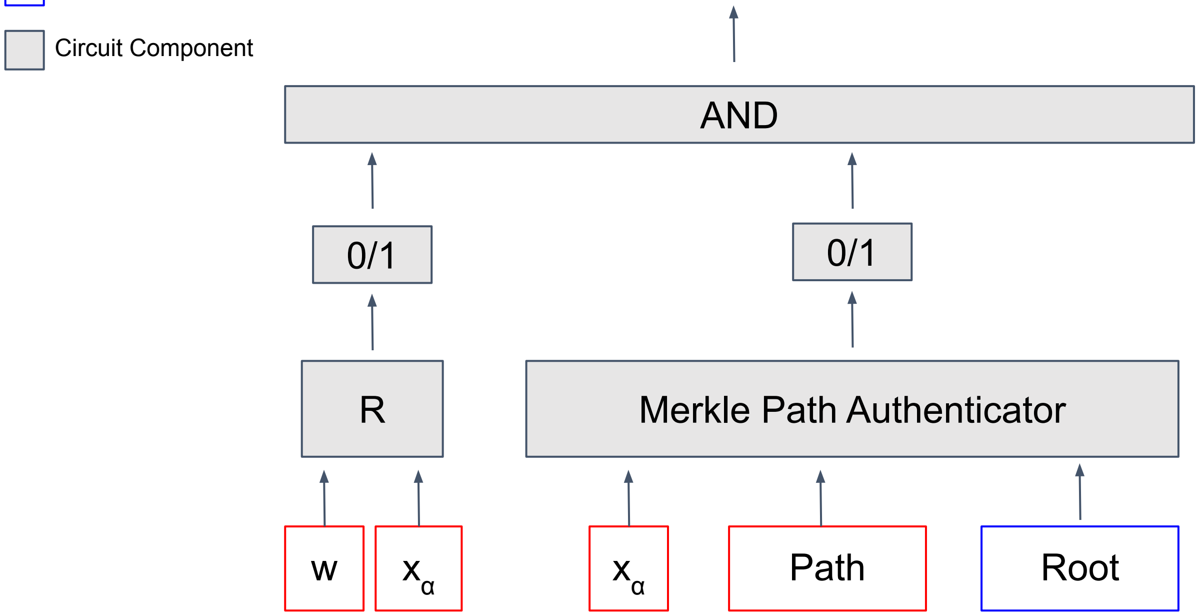


# Representation 3: Merkle Tree

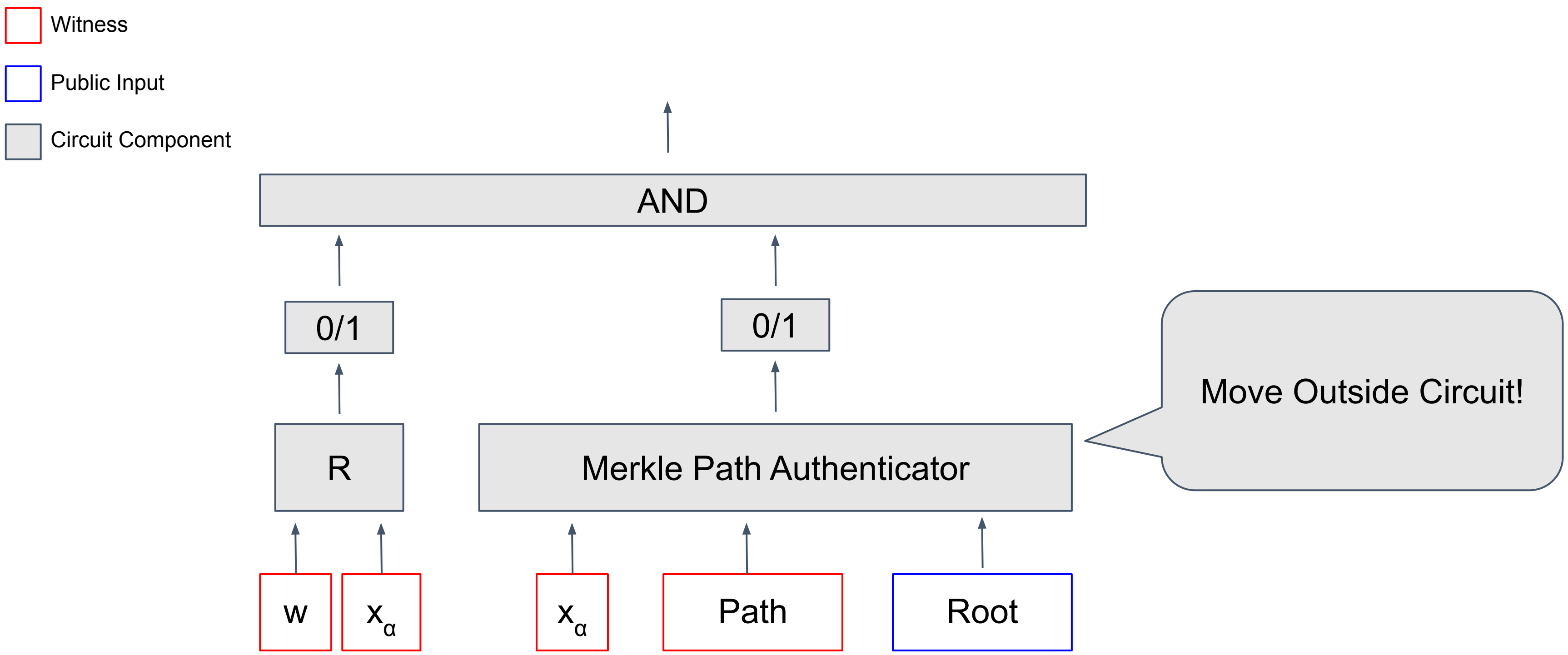
$R(x_1,w)=1$ or $R(x_2,w)=1$ or ... or $R(x_\ell,w)=1$



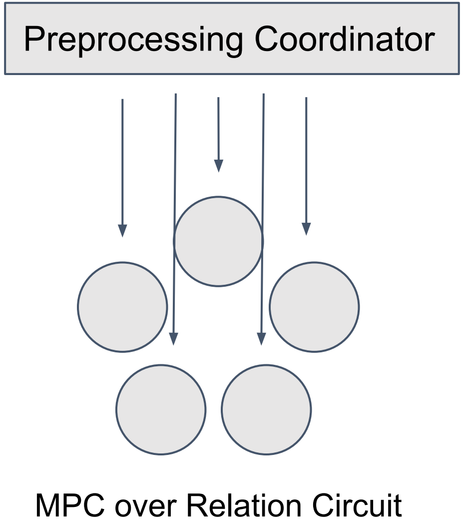
# Our Approach



# Our Approach

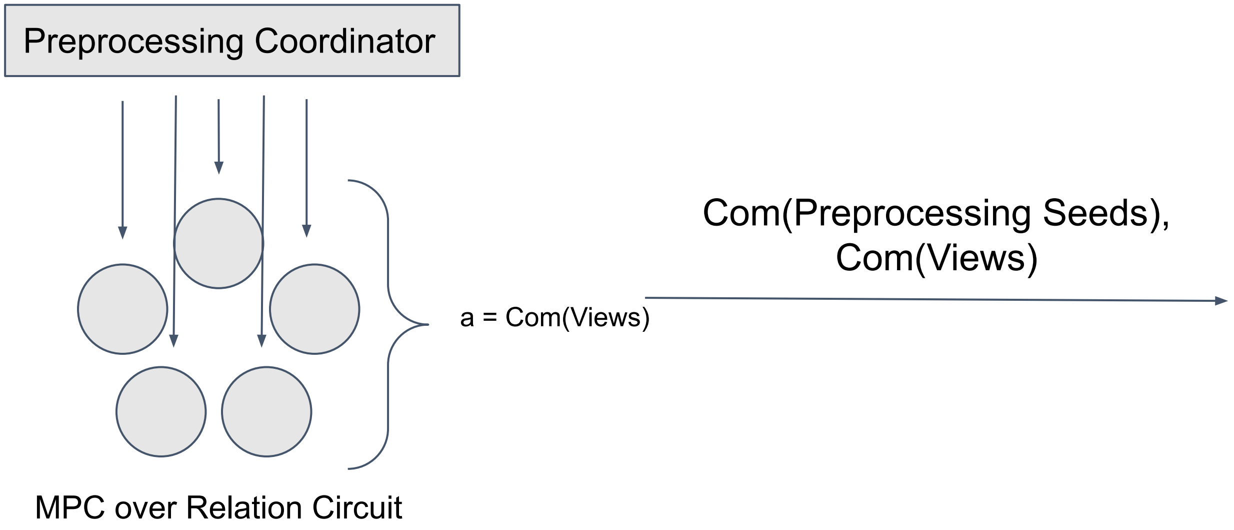


# Our Approach: 1. Integrate Preprocessing [KKW18]

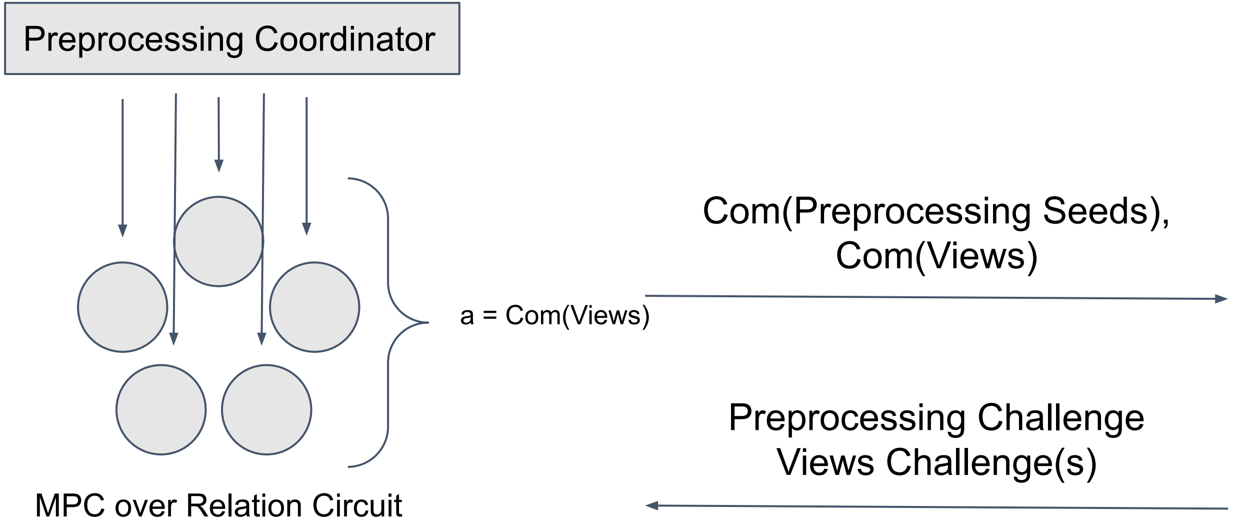



BOSTON UNIVERSITY

# Our Approach: 1. Integrate Preprocessing [KKW18]

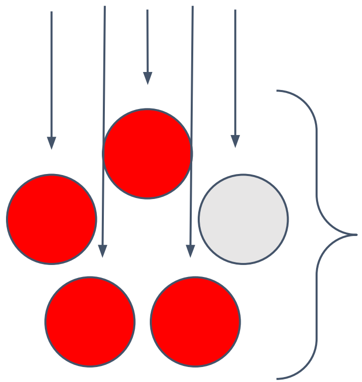


# Our Approach: 1. Integrate Preprocessing [KKW18]



# Our Approach: 1. Integrate Preprocessing [KKW18]

Preprocessing Coordinator



MPC over Relation Circuit

a = Com(Views)

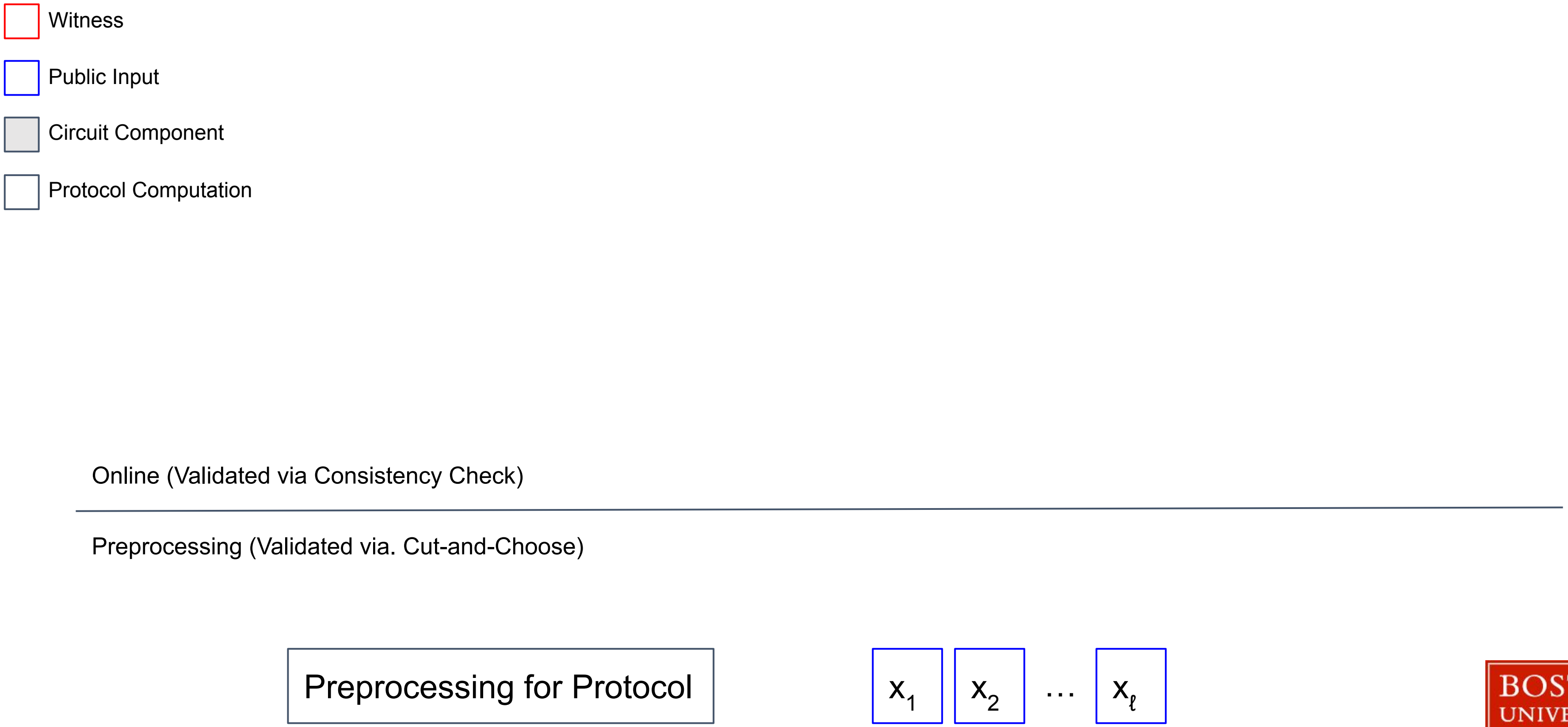
Com(Preprocessing Seeds),
Com(Views)

Preprocessing Challenge
Views Challenge(s)

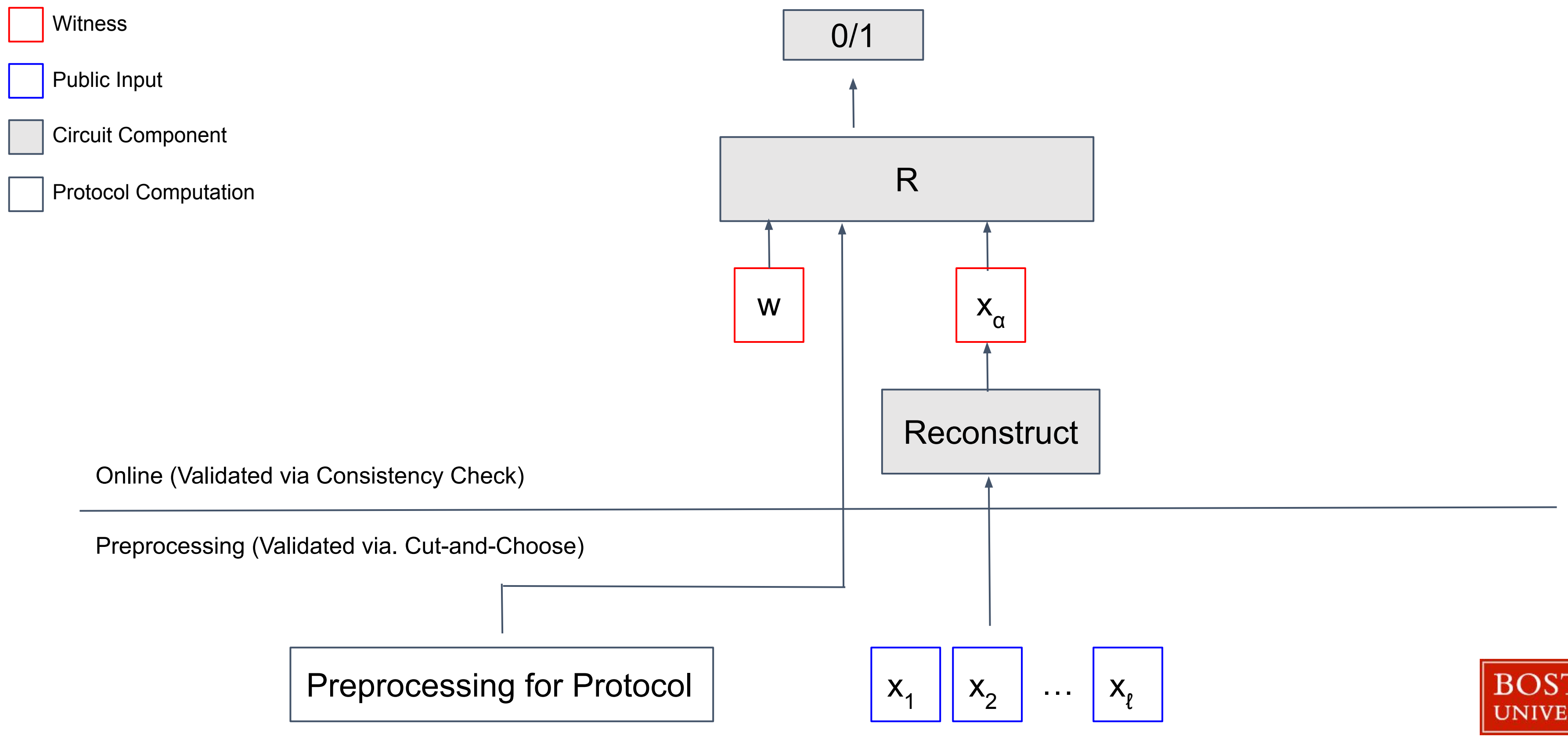
Open(Prepreocessing)
Open(Views)

1. Verify Correctness of Preprocessing
2. Verify Consistency of Views

# Our Approach: 2. Move Set Membership To Privacy Free Preprocessing

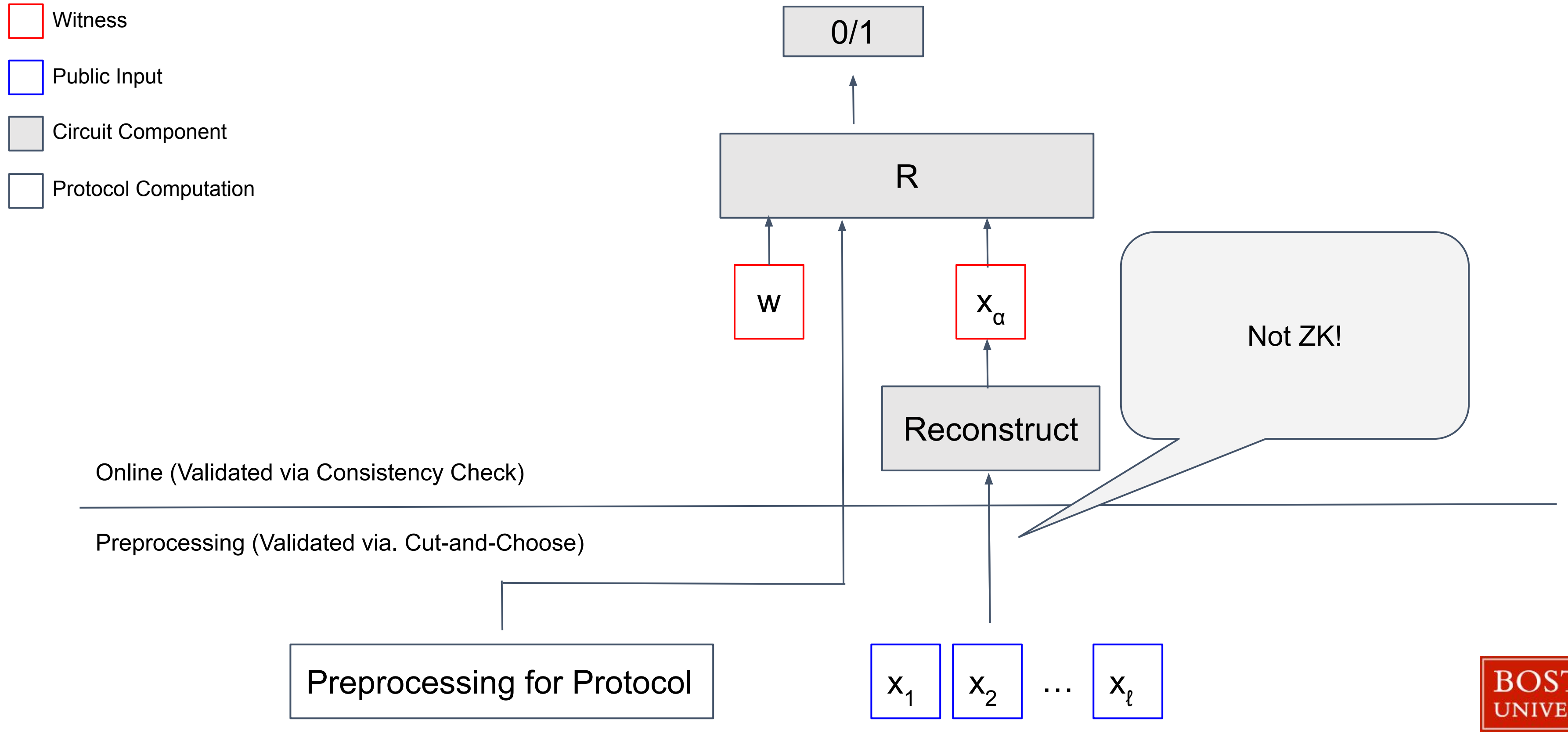


# Our Approach: 2. Move Set Membership To Privacy Free Preprocessing

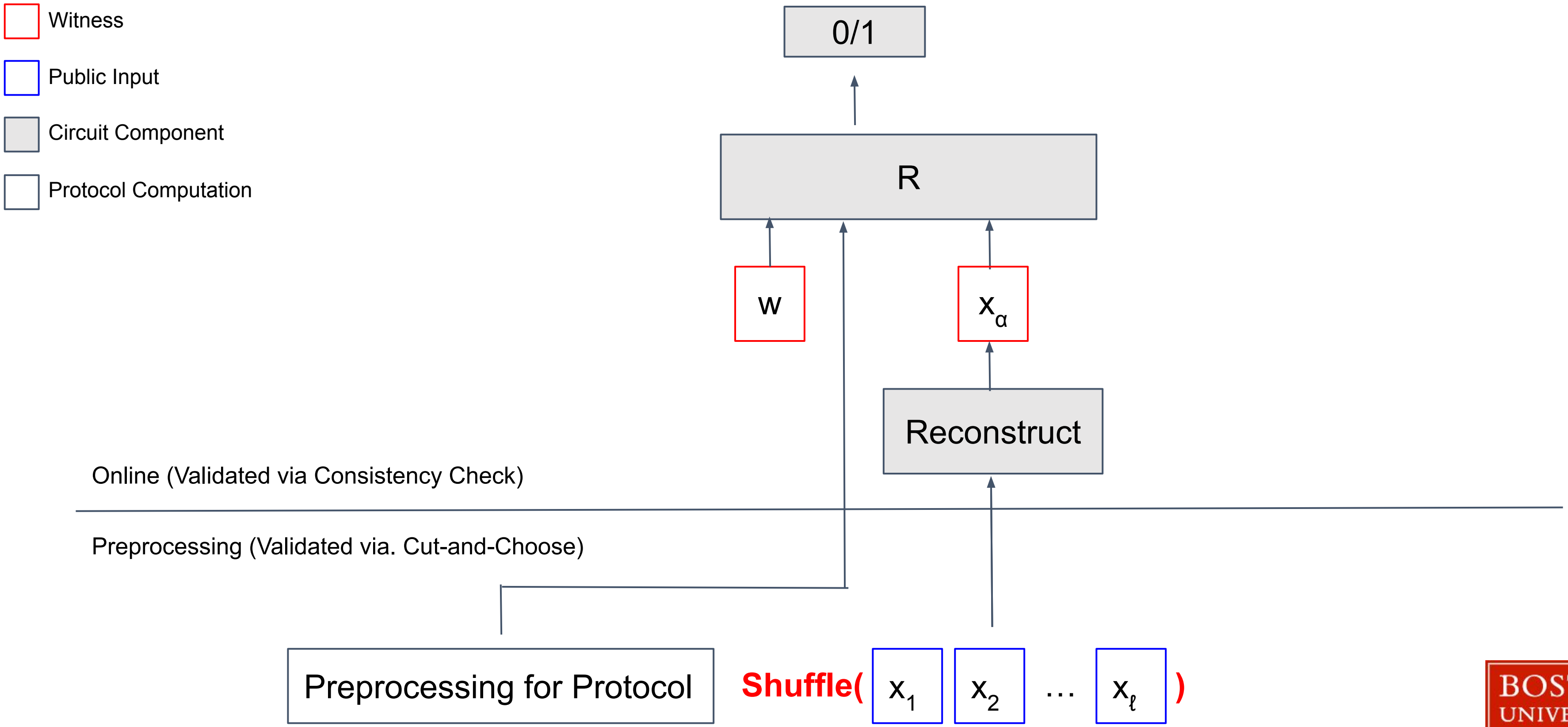



BOSTON UNIVERSITY

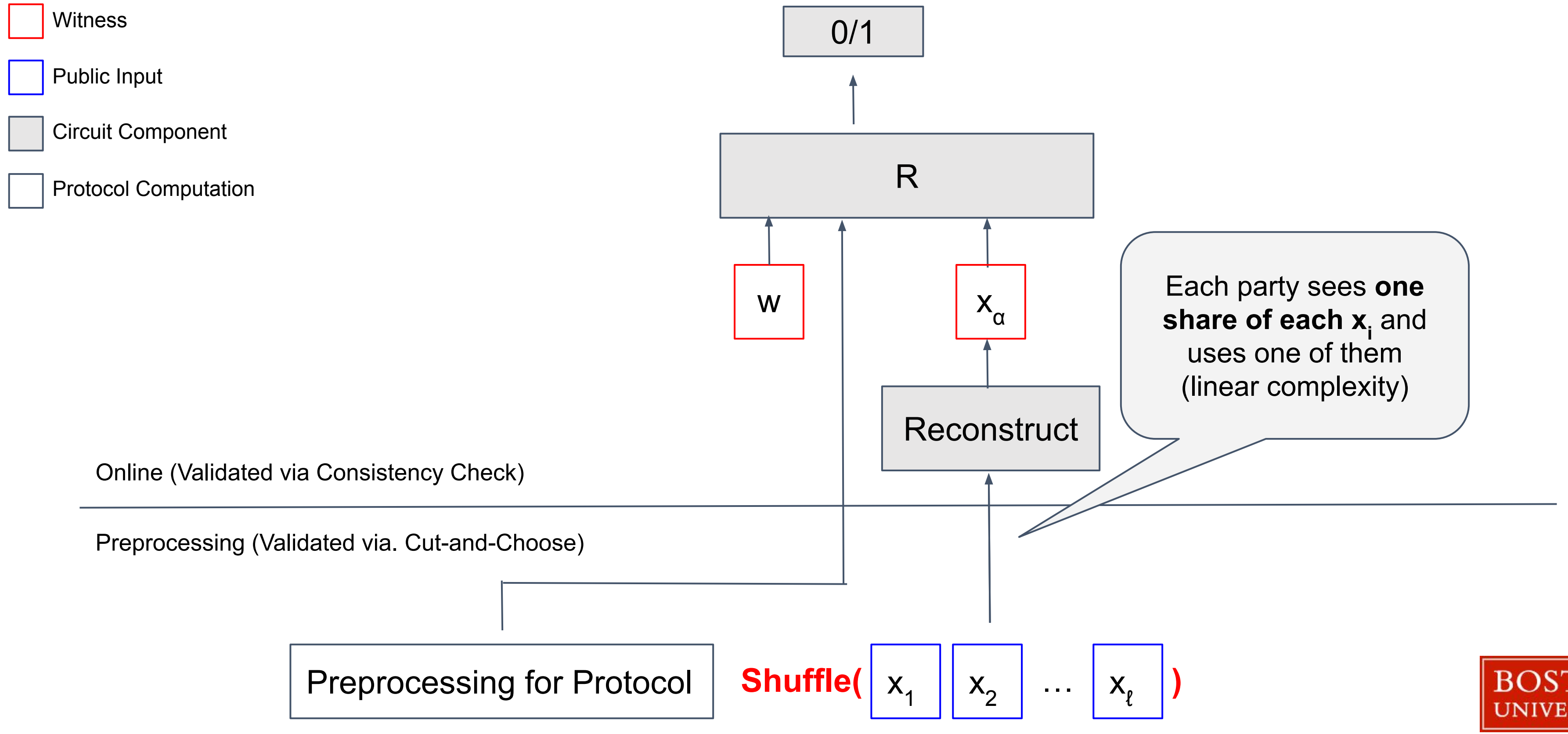
# Our Approach: 2. Move Set Membership To Privacy Free Preprocessing



# Our Approach: 3. Getting Soundness and Zero-Knowledge



# Our Approach: 3. Getting Soundness and Zero-Knowledge




BOSTON UNIVERSITY

# Our Approach: 4. Binding Efficiently with Accumulator

- Witness
- Public Input
- Circuit Component
- Protocol Computation

Online (Validated via Consistency Check)

Preprocessing (Validated via. Cut-and-Choose)

Accumulator

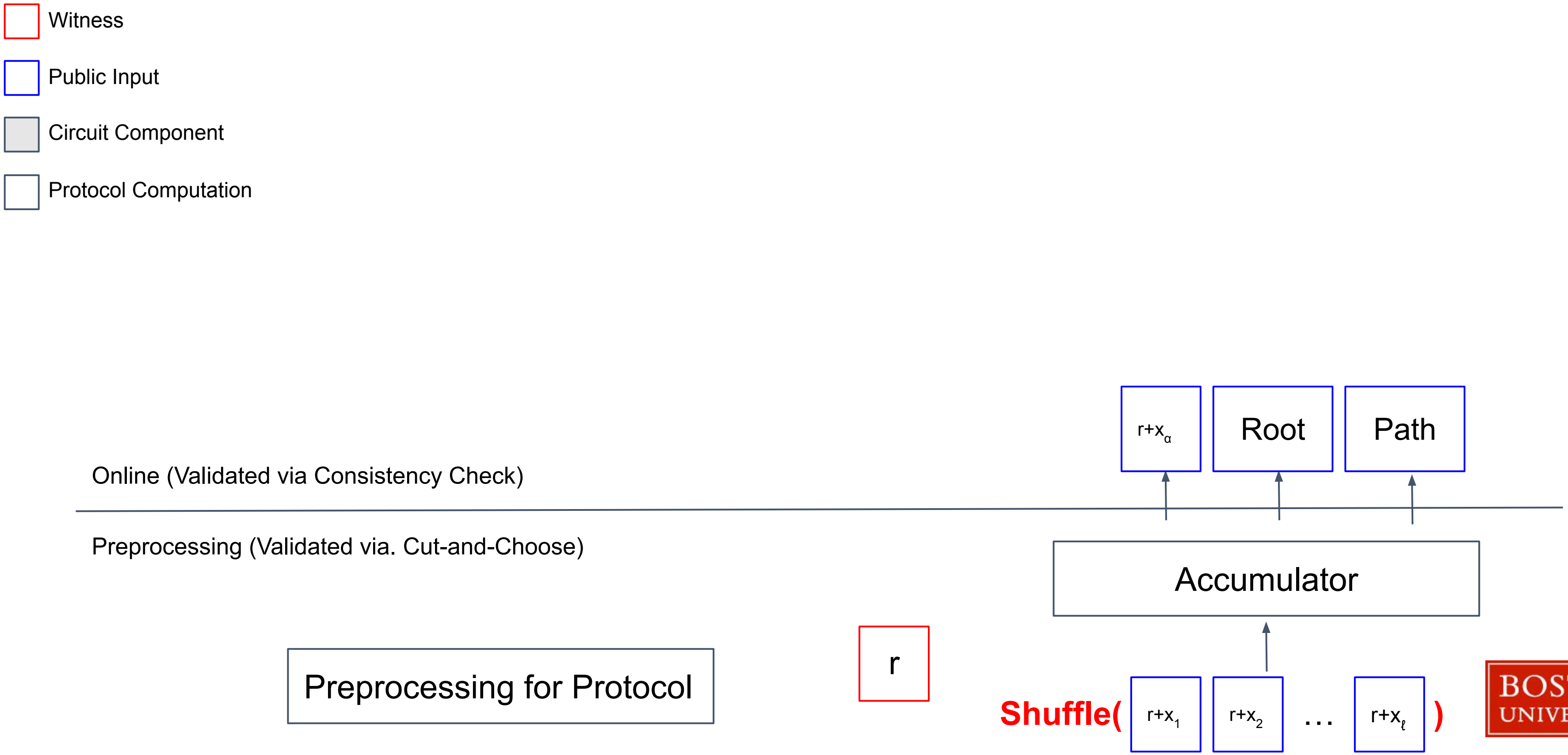
Preprocessing for Protocol

r

**Shuffle(** $r+x_1$ $r+x_2$ … $r+x_\ell$ **)**

BOSTON UNIVERSITY

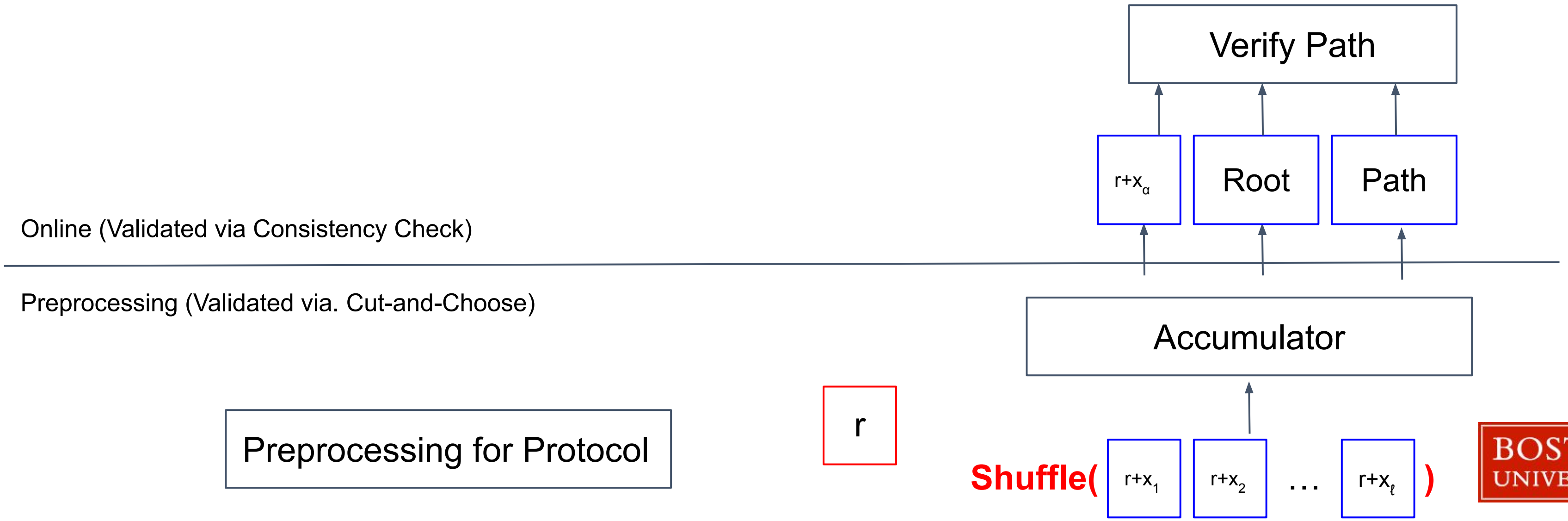
# Our Approach: 4. Binding Efficiently with Accumulator




BOSTON UNIVERSITY

# Our Approach: 4. Binding Efficiently with Accumulator

- Witness
- Public Input
- Circuit Component
- Protocol Computation




BOSTON UNIVERSITY

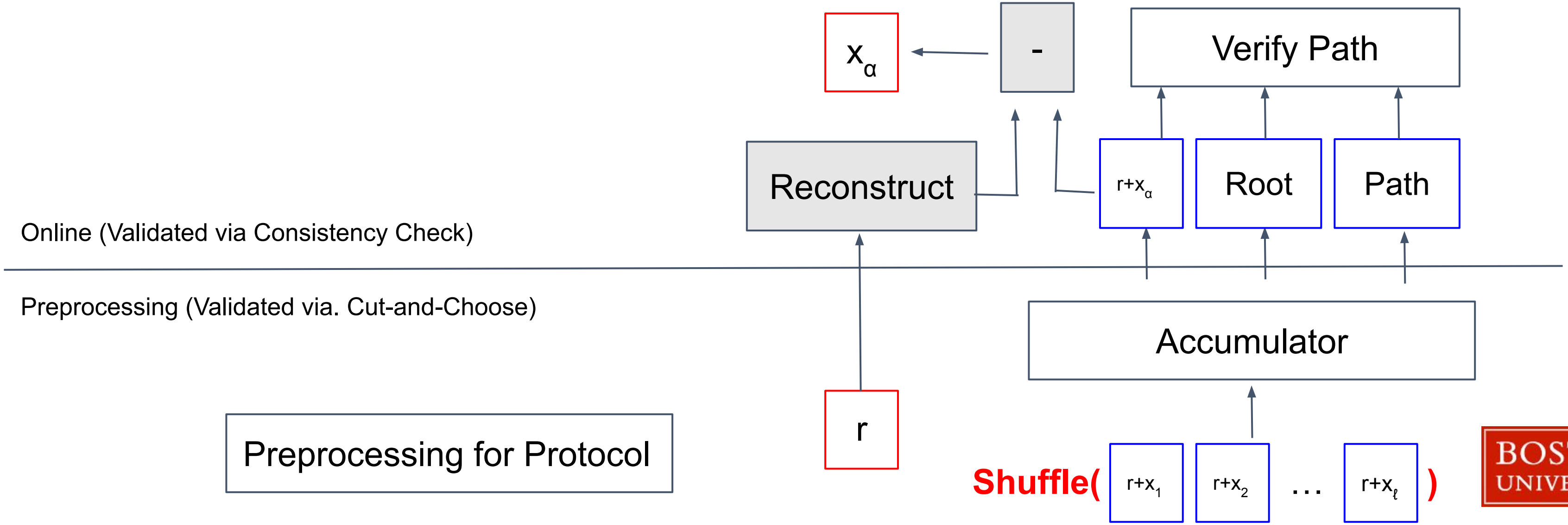
# Our Approach: 4. Binding Efficiently with Accumulator

Witness

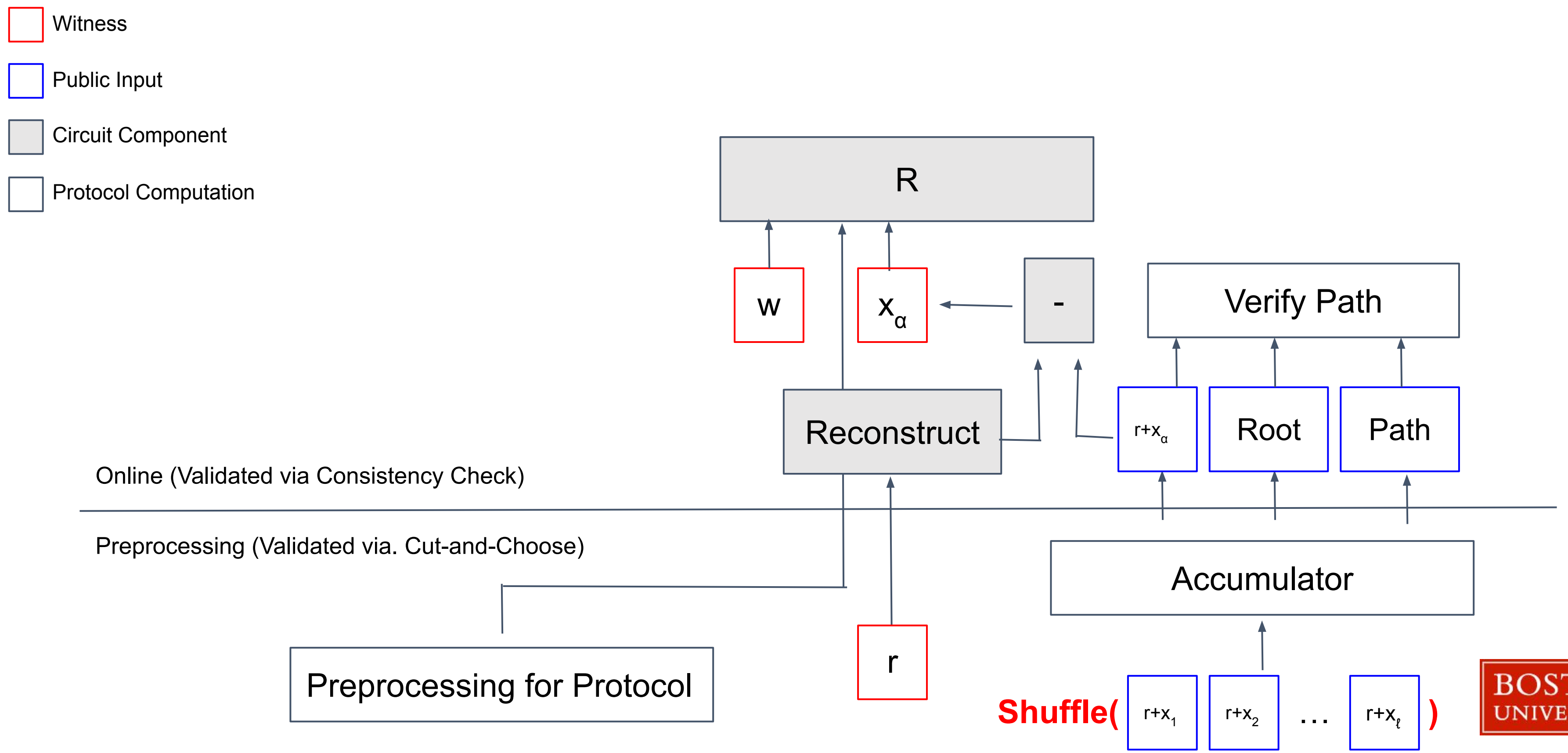
Public Input

Circuit Component

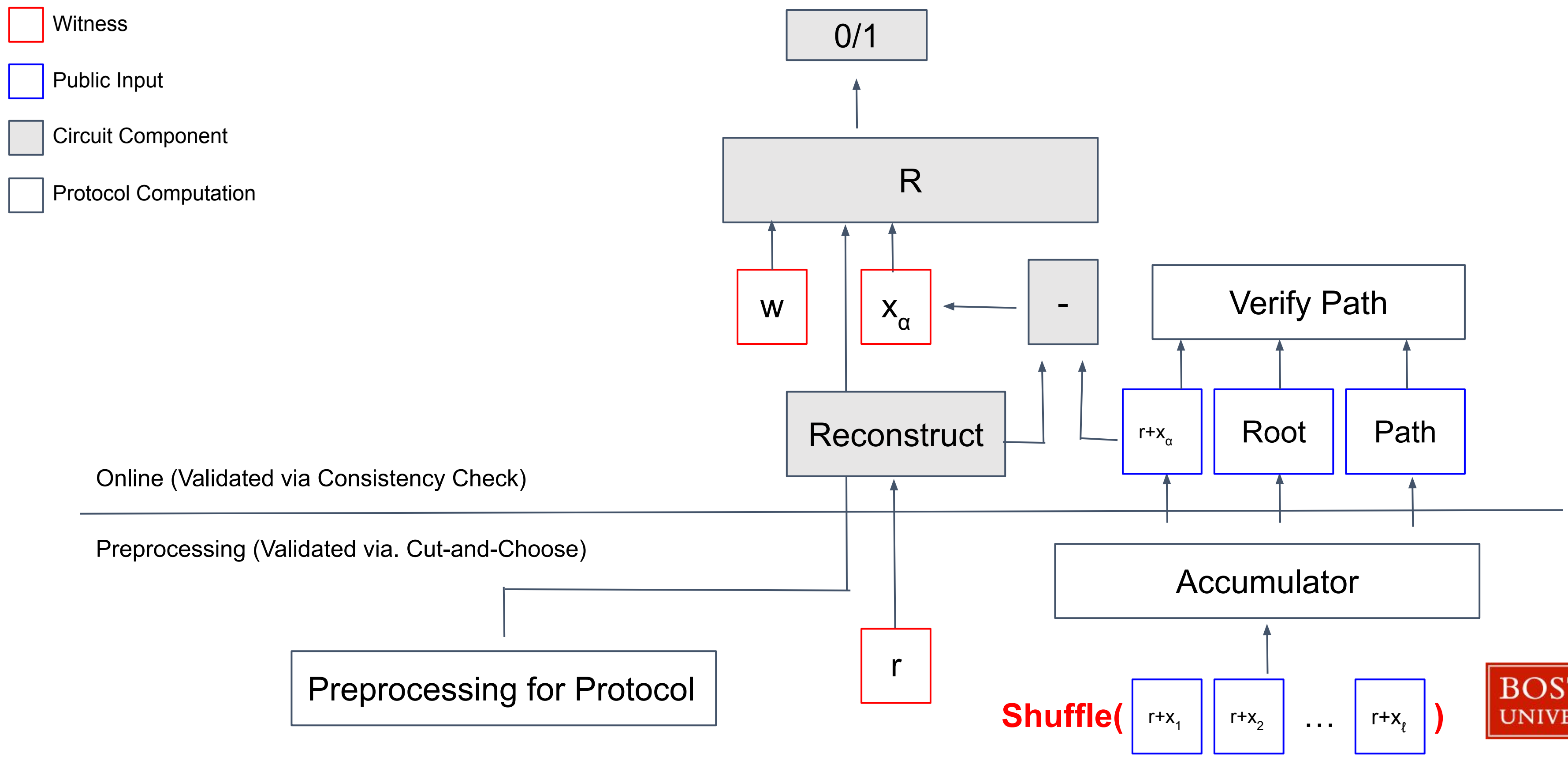
Protocol Computation



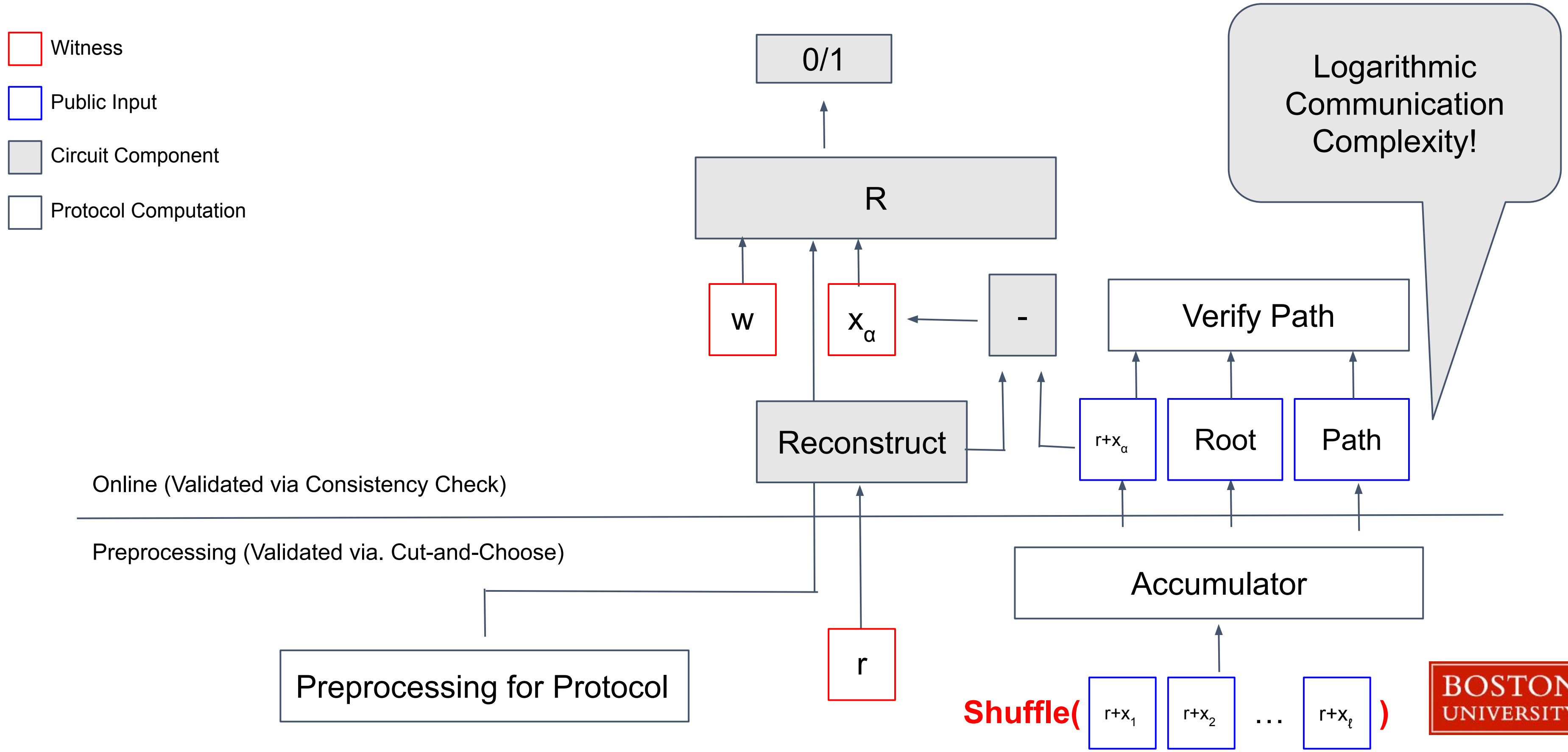
# Our Approach: 4. Binding Efficiently with Accumulator



# Our Approach: 4. Binding Efficiently with Accumulator



# Our Approach: 4. Binding Efficiently with Accumulator




BOSTON UNIVERSITY

# NIZK-based PQ Signatures [GMO16, CDGORRSZ17,KKW18]

$$\text{NIPoK}\{(\text{sk}) : \text{m and pk} = \text{PRF}_{\text{sk}}(0)\}$$

# NIZK-based PQ Ring Signatures [KKW18]

$$\text{NIPoK}\{(\text{sk},\text{pk'}) : \text{m and pk'} = \text{PRF}_{\text{sk}}(0) \text{ and pk'} \in \{\text{pk}_1, \text{pk}_2, \dots, \text{pk}_\ell\}\}$$

BOSTON UNIVERSITY

# NIZK-based PQ Signatures [GMO16, CDGORRSZ17,KKW18]

$$\mathrm{NIPoK}\{(sk) : m \text{ and } pk = \mathrm{PRF}_{sk}(0)\}$$

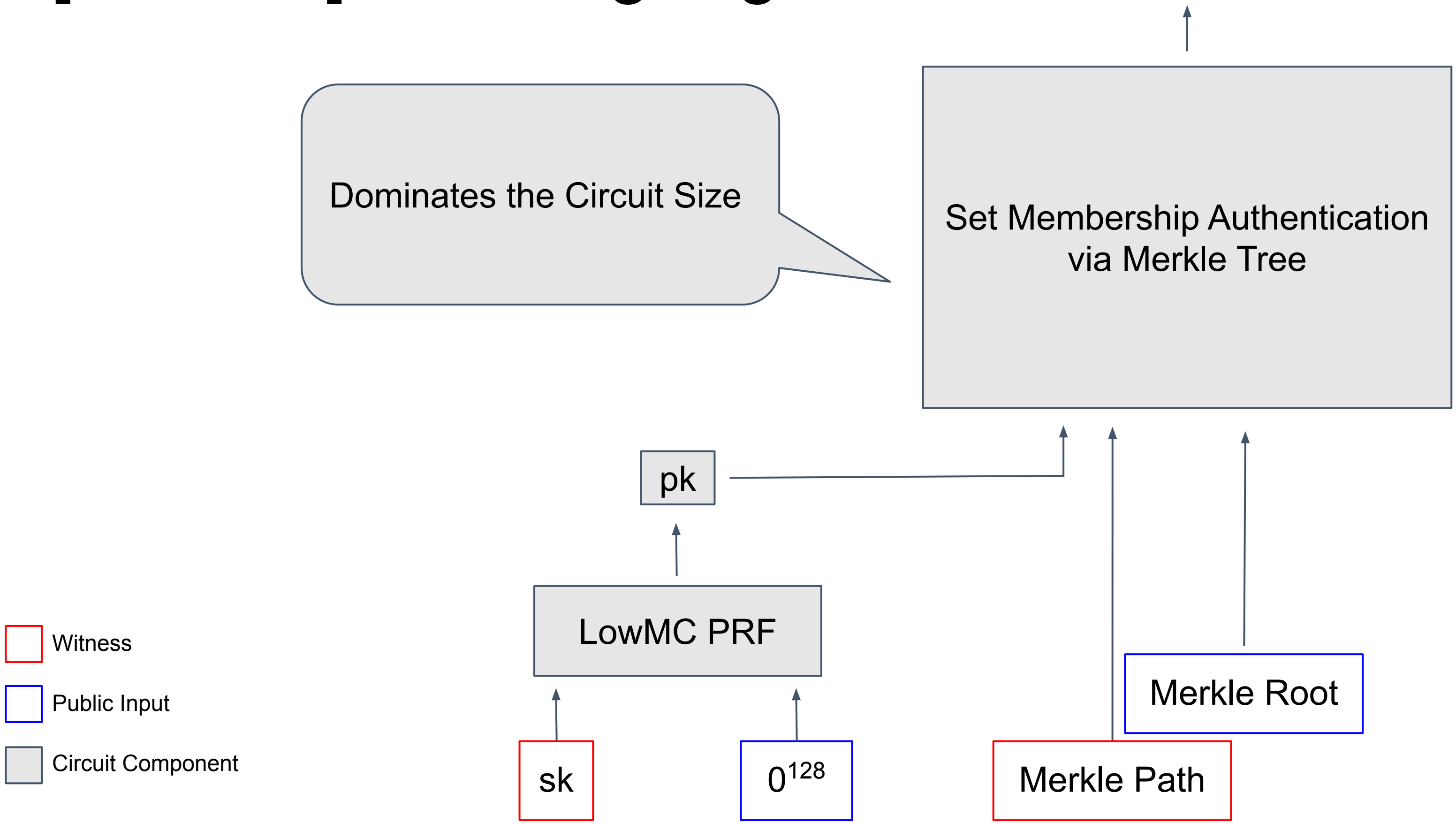
# NIZK-based PQ Ring Signatures [KKW18]

$$\mathrm{NIPoK}\{(sk,pk') : m \text{ and } pk' = \mathrm{PRF}_{sk}(0) \text{ and } pk' \in \{pk_1, pk_2, \ldots , pk_\ell\}\}$$

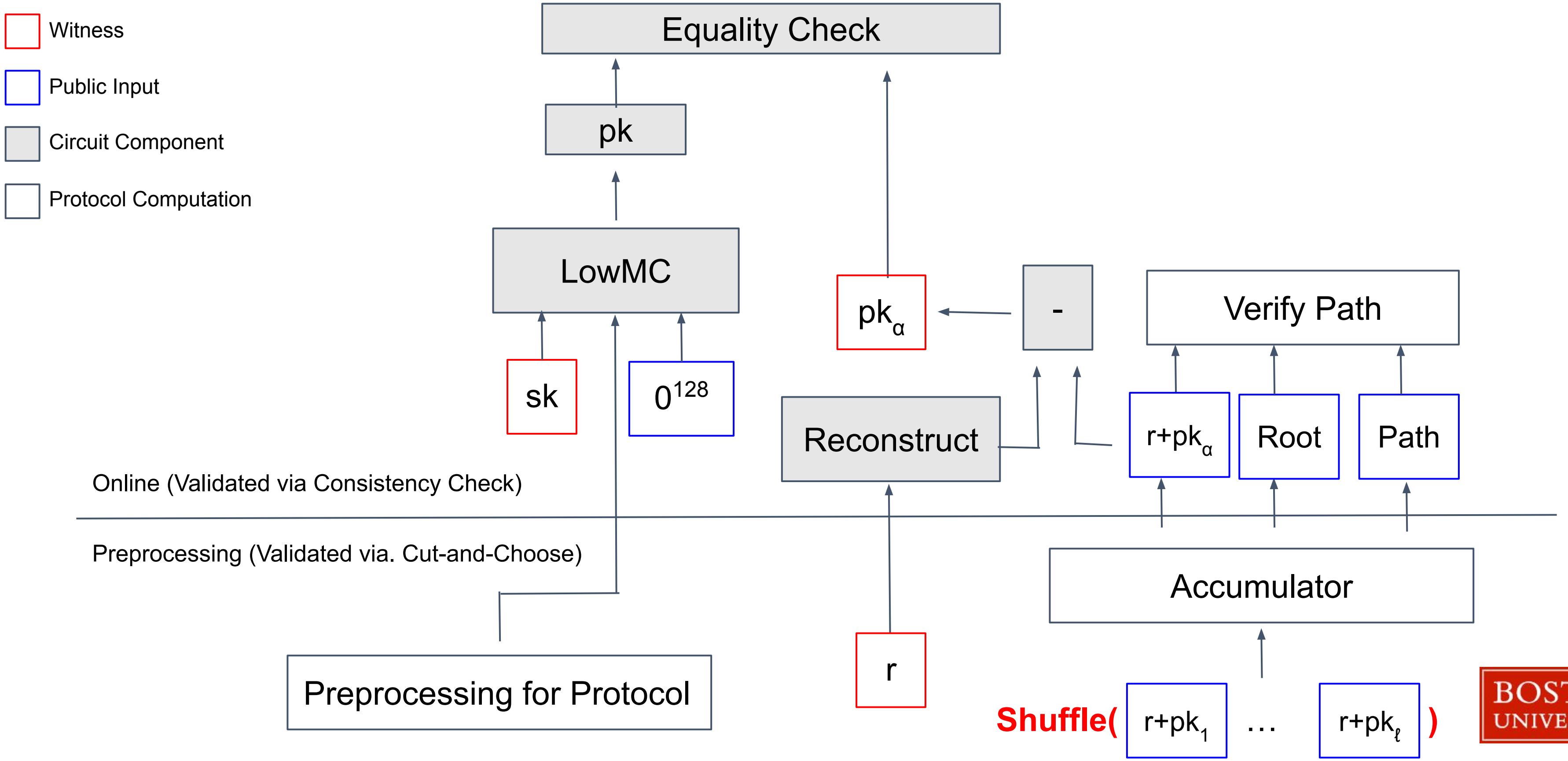
Set Membership

BOSTON UNIVERSITY

# [KKW18] PQ Ring Signatures



# Our PQ Ring Signatures



| Ring size: | $2^7$ | $2^{10}$ | $2^{13}$ | Assumption |
|---|---|---|---|---|
| Derler et al. [13] | 982 KB | 1352 KB | 1722 KB | Symmetric Key |
| Katz et al. [33] | 285 KB | 388 KB | 492 KB | Symmetric Key |
| **This Work** | **52 KB** | **56 KB** | **60 KB** | Symmetric Key |

| Ring size: | $2^3$ | $2^6$ | $2^{12}$ | Assumption |
|---|---|---|---|---|
| Libert et al. [39] | 52 MB | 94 MB | 179 MB | SIS |
| Torres et al. [51] | $> 124$ KB | $> 900$ KB | 61 MB | Ring-SIS |
| Esgin et al. [14] | 41 KB | 58 KB | 256 KB | M-LWE & M-SIS |
| **This Work** | **46 KB** | **50 KB** | **59 KB** | Symmetric Key |

BOSTON UNIVERSITY

# Also in the Paper

- Non-Black Box Integration into existing MPC-in-the-Head protocols
- Super Simple & Efficient PQ RingCT

# Thanks!

https://eprint.iacr.org/2021/1656.pdf

Aarushi Goel (JHU), Mathias Hall-Andersen (Aarhus), **Gabriel Kaptchuk (BU)**, and Matthew Green (JHU)

BOSTON UNIVERSITY

| Ring size: | $2^7$ | | $2^{10}$ | | $2^{13}$ | |
|---|---|---|---|---|---|---|
| | $\lvert\sigma\rvert$ | $t$ | $\lvert\sigma\rvert$ | $t$ | $\lvert\sigma\rvert$ | $t$ |
| Derler et al. [13] | 982 KB | — | 1352 KB | — | 1722 KB | — |
| Katz et al. [33] | 285 KB | — | 388 KB | — | 492 KB | — |
| **This Work** (Server) | 52 KB | 126 ms | 56 KB | 210 ms | 60 KB | 1980 ms |
| **This Work** (Laptop) | 52 KB | 2163 ms | 56 KB | 3437 ms | 60 KB | 16080 ms |

Server: Xeon E5-2695 (18 Cores, 2.10 GHz)

Server: Xeon E5-2666 (10 Cores, 2.60 GHz)

| Ring/group size: | $2^7$ | | $2^{10}$ | | $2^{13}$ | |
|---|---|---|---|---|---|---|
| | $\lvert\sigma\rvert$ | $t$ | $\lvert\sigma\rvert$ | $t$ | $\lvert\sigma\rvert$ | $t$ |
| Derler et al. [21] | 982 KB | — | 1.35 MB | — | 1.72 MB | — |
| Here | 285 KB | 2.0 s | 388 KB | 2.8 s | 492 KB | 3.6 s |
| Boneh et al. [12] | 1.37 MB | — | 1.85 MB | — | — | — |
| Here | 315 KB | 2.3 s | 418 KB | 3.0 s | 532 KB | 3.8 s |

BOSTON UNIVERSITY

# Super Simple PQ RingCT Transactions

Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin
2. Double-spend protection
3. Output coin well formed
4. Range proofs

# Super Simple PQ RingCT Transactions

Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin
2. Double-spend protection
3. Output coin well formed
4. Range proofs

Existing Approaches:

Take independent approaches and duct tape together

BOSTON UNIVERSITY

# Super Simple PQ RingCT Transactions

## Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin
2. Double-spend protection
3. Output coin well formed
4. Range proofs

<u>Existing Approaches:</u>

Take independent approaches and duct tape together

<u>Our Approach:</u>

Throw it into a ZK Proof and don't worry about it

# Super Simple PQ RingCT Transactions

Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin
2. Double-spend protection
3. Output coin well formed
4. Range proofs

Existing Approaches:

Take independent approaches and duct tape together

Our Approach:

Throw it into a ZK Proof and don't worry about it

BOSTON UNIVERSITY

# Super Simple PQ RingCT Transactions

Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin (ring signature)
2. Double-spend protection
3. Output coin well formed
4. Range proofs

Existing Approaches:

Take independent approaches and duct tape together

Our Approach:

Throw it into a ZK Proof and don't worry about it

# Super Simple PQ RingCT Transactions

Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin (ring signature)
2. Double-spend protection (LowMC as PRF)
3. Output coin well formed
4. Range proofs

Existing Approaches:

Take independent approaches and duct tape together

Our Approach:

Throw it into a ZK Proof and don't worry about it

BOSTON UNIVERSITY

# Super Simple PQ RingCT Transactions

## Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin (ring signature)
2. Double-spend protection (LowMC as PRF)
3. Output coin well formed (Trivial addition)
4. Range proofs

Existing Approaches:

Take independent approaches and duct tape together

Our Approach:

Throw it into a ZK Proof and don't worry about it

# Super Simple PQ RingCT Transactions

Parts of a PQ RingCT Construction:

1. Demonstrating authorization to spend hidden coin (ring signature)
2. Double-spend protection (LowMC as PRF)
3. Output coin well formed (Trivial addition)
4. Range proofs (Do addition without overflow)

<u>Existing Approaches:</u>

Take independent approaches and duct tape together

<u>Our Approach:</u>

Throw it into a ZK Proof and don't worry about it

BOSTON UNIVERSITY