

Towards a Publicly-Verifiable Mix-net Providing Everlasting Privacy



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Denise Demirel, Jeroen van de Graaf, Johannes Buchmann

Authors



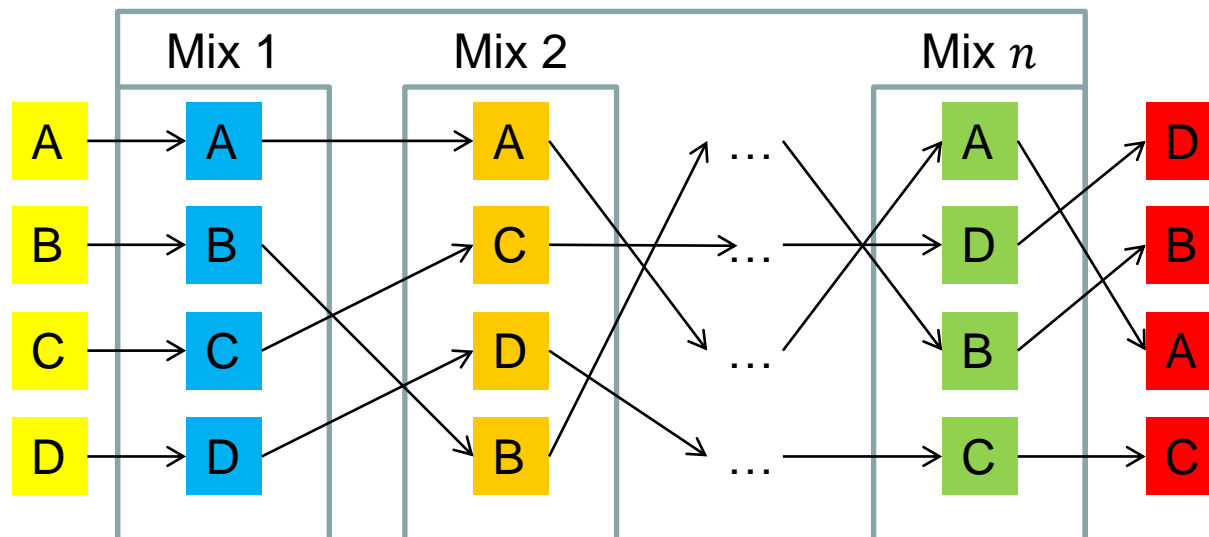
Denise Demirel, Department of Computer Science,
Cryptography and Computer Algebra Group,
Technische Universität Darmstadt, Germany

Jeroen van de Graaf, Departamento de Ciência de
Computação, Universidade Federal de Minas Gérias, Brazil

Johannes Buchmann, Department of Computer Science,
Cryptography and Computer Algebra Group,
Technische Universität Darmstadt, Germany

Introduction

- Mix-nets were introduced by David Chaum in 1981
- Reencryption mix-nets [1993, Park et al.] allow third parties to verify the correctness of the shuffling procedure



- Electronic Voting, electronic exam systems, online auctions

Public-Key Encryption

Probabilistic public-key encryption algorithm (Gen, Enc, Dec)

- Semantical (CCA) security

- Homomorphic such that

$$\forall m, m' \in G, \forall r, r' \in H: Enc(m, r) \cdot Enc(m', r') = Enc(m +_G m', r +_H r')$$

It follows that messages can be “reencrypted”

$$ReEnc(u, r') = Enc(m, r) \cdot Enc(0_G, r') = Enc(m, r + r')$$

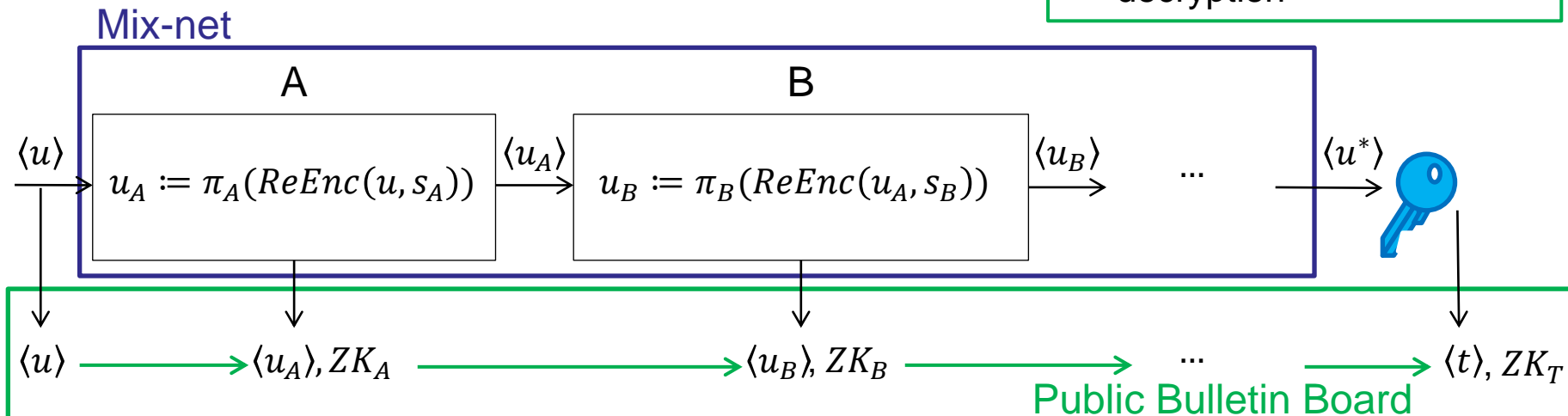
- (k, n) -threshold decryption

Reencryption Mix-net

- Input $u = Enc(t, s)$, with message t and randomness s
- $ReEnc(u, s') = Enc(t, s) \cdot Enc(0_G, s') = Enc(t, s + s')$

Public Verification Process

- ZK-Proof of correct shuffling
- ZK-Proof of correct decryption



Computational Privacy

- Homomorphic public-key cryptography, e.g., Paillier, Elgamal
- Computational assumptions
- Current implementations have an expiration date
- Violates principle of free and secret suffrage
- Possible solution: using an unconditional hiding commitment scheme to encode the published audit information

Commitment Scheme

A (non-interactive) commitment scheme $(GenCom, Com, Unv)$

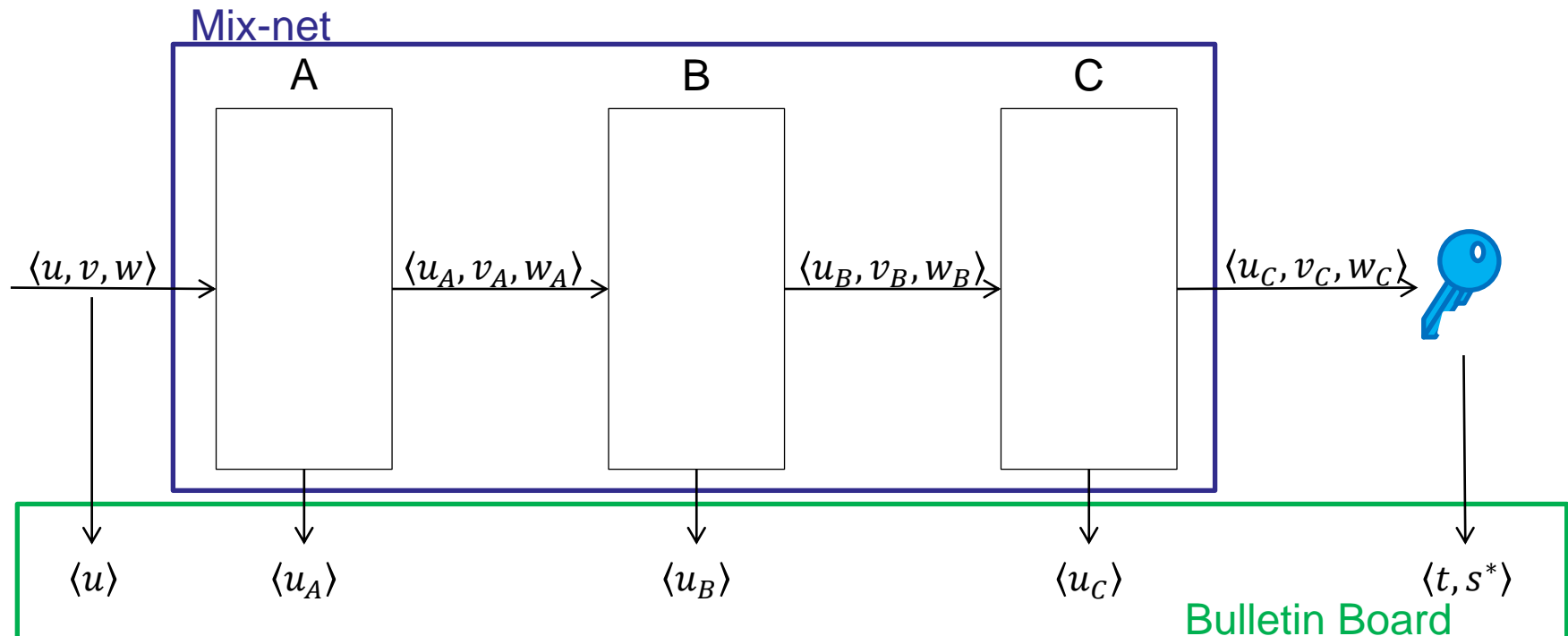
- $GenCom(1^\kappa)$ defines message space and randomization space for security parameter κ .
- $c = Com(t, s) \in C$ generates commitment $c \in C$ to $t \in M$ and $s \in R$.
- $Unv(c, t, s)$ returns t if $c = Com(t, s)$ and \perp if not.

Mix-net Providing Everlasting Privacy Towards the Public

Input

$u = Com(t, s)$, commitment to message t with decommitment s

$v = Enc_M(t), w = Enc_R(s)$ opening values encrypted with public-key cryptography



Mix-net Providing Everlasting Privacy Towards the Public

Input

$u = Com(t, s)$, commitment to message t with randomness s

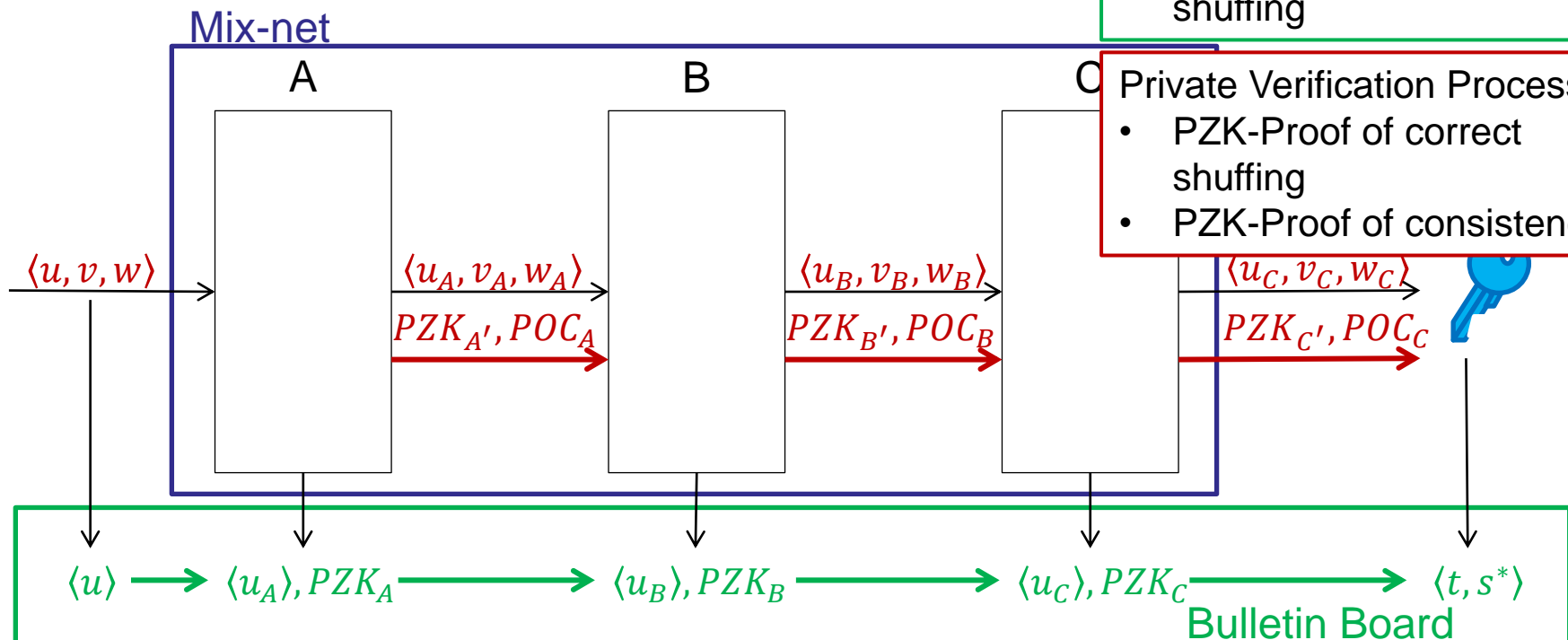
$v = Enc_M(t), w = Enc_R(s)$ opening values encrypted with M

Public Verification Process

- PZK-Proof of correct shuffling

Private Verification Process

- PZK-Proof of correct shuffling
- PZK-Proof of consistency



Assumptions

Correctness:

- The random challenge bits are unpredictable
- The authorities cannot break the computational binding property of the cryptographic primitives for the parameters chosen before the whole process is completed

Robustness:

- “ (k, n) -threshold”- decryption at least k out of n key holders participate in the decryption process
- **The authorities carry out the private verification process**

Assumptions (2)

Privacy:

- The authorities cannot break the computational assumption of the encryption scheme.
- At least one mix is honest and keeps the association between its input and output values secret.
- Using “ (k, n) -threshold”- decryption at least $(n - k + 1)$ out of n key holders keep their key portion secret.
- **Private channels can be used to send the encrypted opening values.**

Properties

Correctness: Changes on the messages will be detected with overwhelming probability even if all authorities collaborate.

Robustness: The protocol always terminates successfully. If one authority cheats, it will get caught with overwhelming probability.

Privacy: During the process, the messages remains secret as long as a minimum number of authorities act honest.

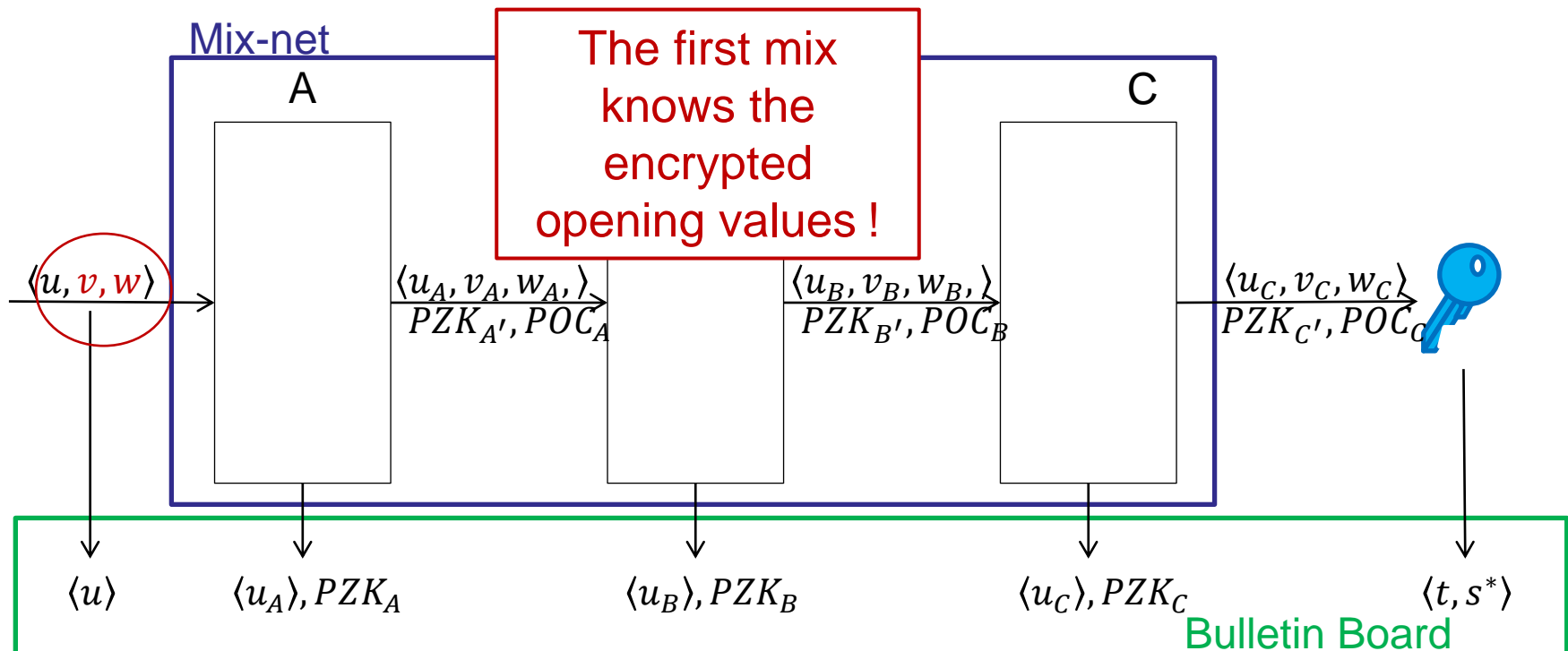
Everlasting privacy towards observers: All published data do not reveal any information about the messages.

Everlasting Privacy Towards the Authorities

Input

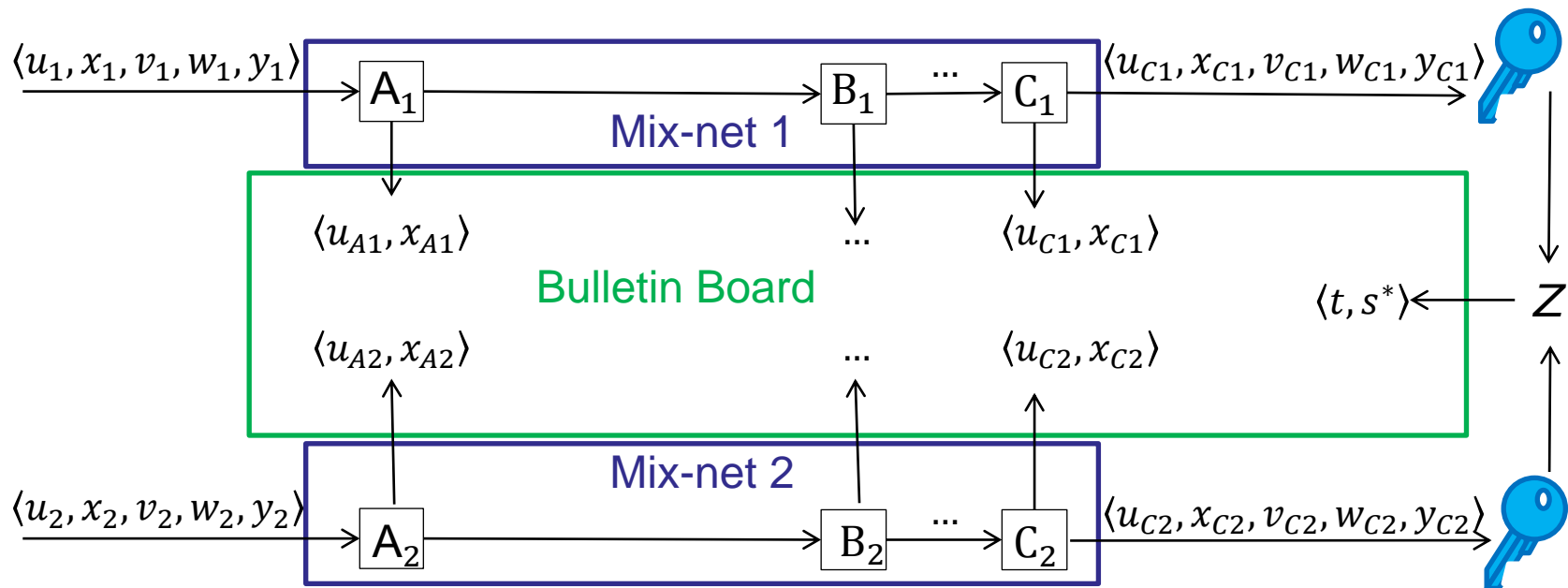
$u = Com(t, s)$, commitment to message t with randomness s

$v = Enc_M(t), w = Enc_R(s)$ opening values encrypted with public-key cryptography

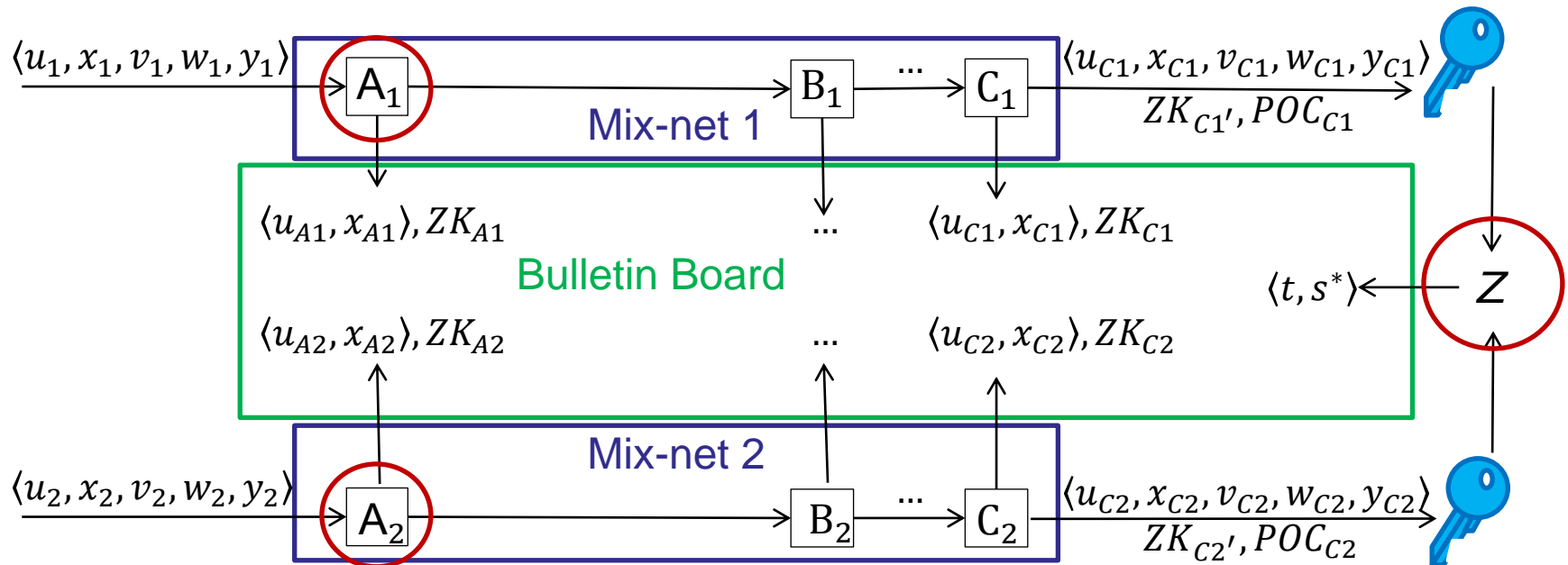


Secret Sharing

- Split message t : $t_1 + t_2 = t$
- Choose ID
- Submit $u_1, x_1, v_1, x_1 = Com(ID, r_1), y_1 = Enc_R(r_1)$ to Mix-net 1 and $u_2, x_2, v_2, x_2 = Com(ID, r_2), y_2 = Enc_R(r_2)$ to Mix-net 2

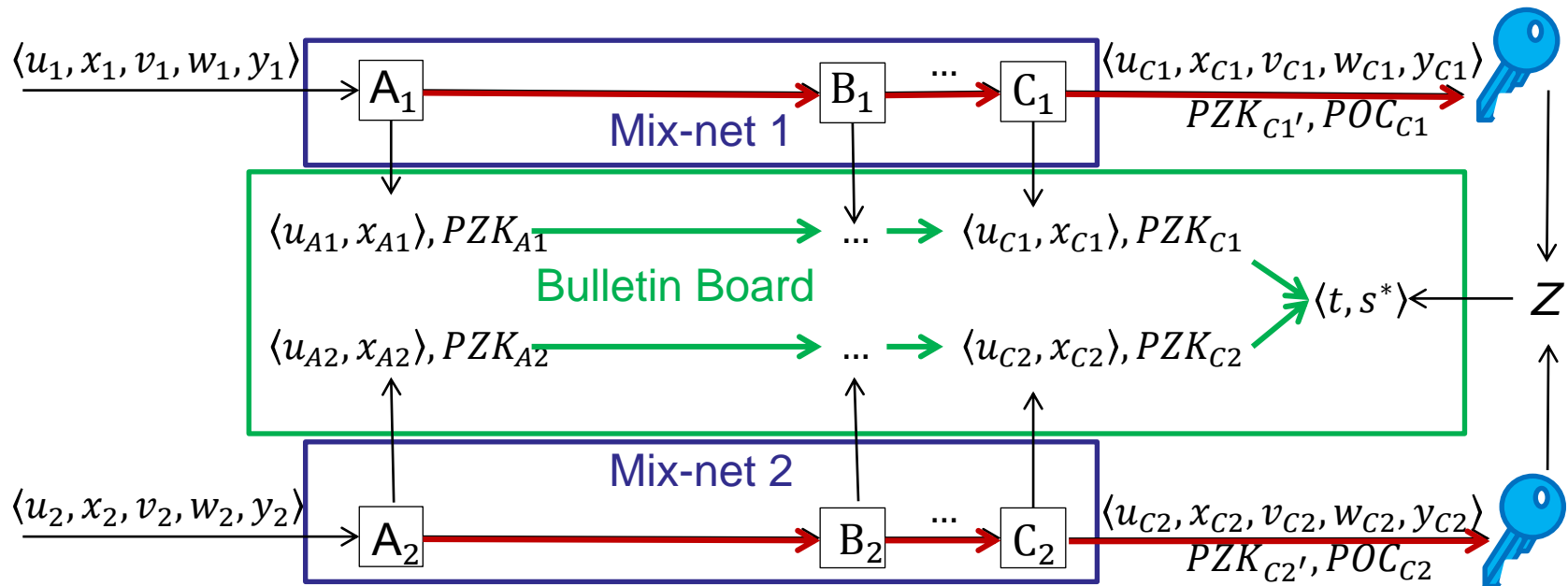


No Single Point of Failure



Secret Sharing

- Split message t : $t_1 + t_2 = t$
- Choose ID
- Submit $u_1, x_1, v_1, x_1 = Com(ID, r_1), y_1 = Enc_R(r_1)$ to Mix-net 1 and $u_2, x_2, v_2, x_2 = Com(ID, r_2), y_2 = Enc_R(r_2)$ to Mix-net 2



Commitment Scheme - Requirements

- **Correctness:** For any $t \in M, s \in R: \text{Unv}(\text{Com}(t, s), t, s) = t$
- **Non-Interactive**
- **Computationally Binding:** Given $c = \text{Com}(t, s)$, for any PPT A the probability to find (t', s') with $t \neq t'$ such that $\text{Com}(t, s) = \text{Com}(t', s')$ is negligible in κ .
- **Unconditionally Hiding:** Distribution of $\text{Com}(t, s)$ and $\text{Com}(t', s')$ must be identical when $s, s' \in R$ are chosen uniformly at random.
- **Homomorphic:** For all $t, t' \in M$ and $s, s' \in R$

$$\text{Com}(t, s) \cdot_C \text{Com}(t', s') = \text{Com}(t +_M t', s +_R s')$$

It follows that messages can be “rerandomize”

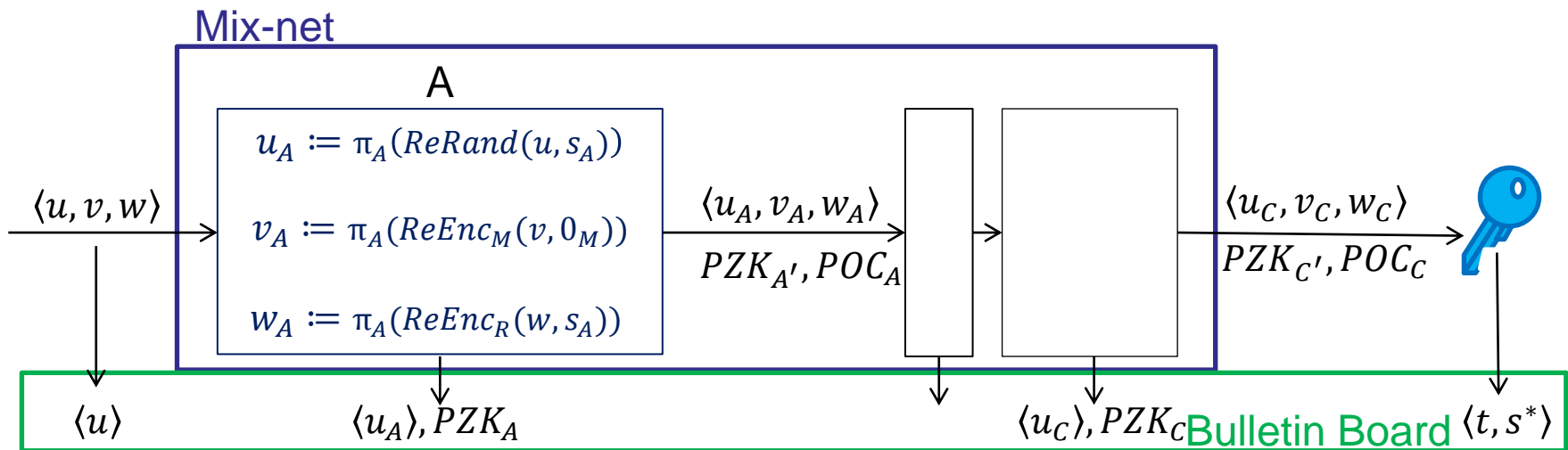
$$\text{ReRand}(c, s') = \text{Com}(t, s) \cdot_C \text{Com}(0_M, s') = \text{Com}(t, s +_R s')$$

1. Commitment scheme (homomorphic, unconditionally hiding)
2. Two instances of the encryption scheme
 1. Enc_M which is homomorphic over message space M
 2. Enc_R which is homomorphic over randomization space R
 - Paillier encryption and adapted Pedersen Commitments [MN07]
 - EC groups with asymmetric pairing (E.Cuvelier, O.Pereira, T.Peters)
3. Perfect zero-knowledge proof of correct shuffling and consistency
 - Non-interactive zero-knowledge shuffle argument, e.g., [Groth2010] or [LZ12].
 - Cut-and-choose based shuffling proof [SK95]

Mixing Process

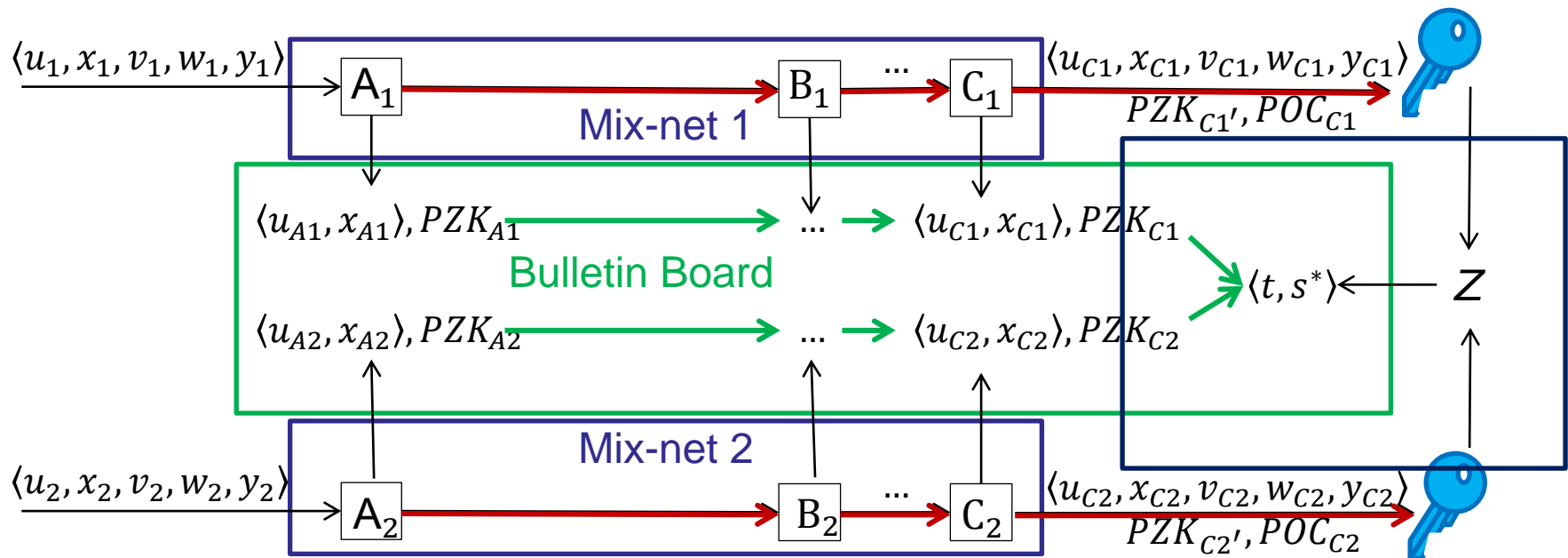
Anonymisation

- Rerandomize public output $u'_A = u \cdot Com(\mathbf{0}_M, \mathbf{s}_A) = Com(t, s + \mathbf{s}_A)$
- Reencrypt message $v'_A = v \cdot Enc_M(\mathbf{0}_M) = Enc_M(t)$
- Adapt decommitment value $w'_A = w_i \cdot Enc_R(\mathbf{s}_A) = Enc_R(s + \mathbf{s}_A)$
- Permutation $\langle u_A, v_A, w_A \rangle = \langle \pi_A(u'_A), \pi_A(v'_A), \pi_A(w'_A) \rangle$



Secret Sharing

- Split message t : $t_1 + t_2 = t$
- Choose ID
- Submit $u_1, x_1, v_1, x_1 = Com(ID, r_1), y_1 = Enc_R(r_1)$ to Mix-net 1 and $u_2, x_2, v_2, x_2 = Com(ID, r_2), y_2 = Enc_R(r_2)$ to Mix-net 2



Matching

- Reveal decommitment value: $Dec_R(y_{Ci}) = s_i'^*$,
- Reveal association: $ID^* = x_{Ci} \cdot Com(0_M, -s_i'^*) = Com(ID, 0_R)$
- Match shares with same ID^* and publish opening values

Proof of Correct Matching

- $x_{C1}: Com(ID, s_1'^*)$ and $x_{C2}: Com(ID, s_2'^*)$,
- $Com(ID, s_1'^*) = Com(ID, s_2'^*) \cdot Com(0_M, s_1'^* - s_2'^*)$
- Proof of correct matching by showing knowledge of “rerandomization” value $(s_1'^* - s_2'^*)$, e.g., by cut-and-choose.

Summary

Simple mix-net providing everlasting privacy towards the public

Complex mix-net (using secret sharing) providing everlasting privacy towards the authorities

Future work

- Introduce mix-net to Prêt à Voter and Split-Ballot
- Quantum resistant commitment and encryption scheme

Thank you

Questions?