Set (Non-)Membership NIZKs from Determinantal Accumulators

Helger Lipmaa,

Roberto Parisella,

Simula UiB, Norway

A privacy problem

$S = \{1,3,4,7,9,13,19,21\}$ Public set

Alice: the prover

Does Alice have a number in S?

Bob: the verifier

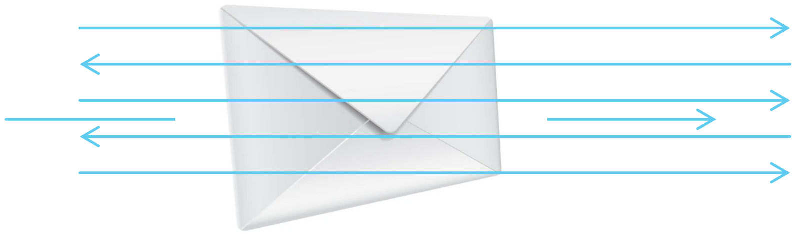
$7 \in S$ (secret)

Secret number

Only if Alice's number is in S

Alice $\left(Enc; (7, rand)\right)$

Bob (Enc)



Accept or reject

Encrypt the secret number

Alice and Bob interact

$S = \{1,3,4,7,9,13,19,21\}$

$\mathcal{L} = \{x = Enc\,(7, rand) \colon w = 7, rand\}$

Security

Alice

Honest
$7 \in S$

Bob

Accept

- **Completeness:** honest prover always convinces the verifier.

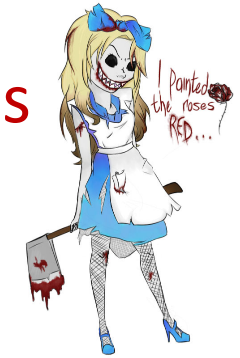
$S = \{1,3,4,7,9,13,19,21\}$ $\mathcal{L} = \{x = Enc\,(7, rand): w = 7, rand\}$

Security

Alice

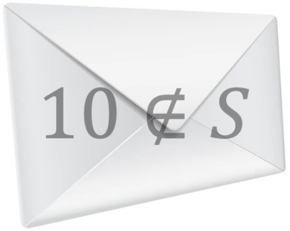
Malicious

$10 \notin S$



Bob

Reject



- Completeness: honest prover always convinces the verifier.
- Soundness: malicious prover cannot convince the verifier.

$S = \{1,3,4,7,9,13,19,21\}$

$\mathcal{L} = \{x = Enc\ (7, rand): w = 7, rand\}$

Security

Alice

Honest

$7 \in S$

Bob

Malicious

Knows only that $x \in S$

- Completeness: honest prover always convinces the verifier.
- Soundness: malicious prover cannot convince the verifier.
- Zero-knowledge: the verifier learns nothing about the witness

$S = \{1,3,4,7,9,13,19,21\}$

$\mathcal{L} = \{x = Enc\,(7, rand) : w = 7, rand\}$

Set Non-Membership NIZK

Trusted third party

Non-Interactive

crs

Alice $\left(crs, Enc; (x, rand)\right)$

$x \notin S$

Bob $(crs, Enc\ )$

π

Completeness, Soundness, Zero-knowledge

Succinctness (constant proof size and verifier complexity)

Set (non)-Membership NIZK

No random oracle
No GGM/AGM

crs

Alice $\left(crs, Enc; (x, rand)\right)$

$x \in S$ or $x \notin S$

π

Bob $(crs, Enc\ )$

Completeness, Soundness, Zero-knowledge

Succinctness (constant proof size and verifier complexity)

Set Membership NIZK With Signatures

- crs explicitly depends on the set S.

- It seems to disallow Non-membership

Accumulators Non ZK Set (Non-)Membership

$S = \{a_1, \dots, a_m\}$

$crs = Z_S, apk$

$Z_S = commit(S, apk)$

Alice (crs, x)

x, ϕ

Bob $(crs\ )$

Memb.verify$(x, \phi, Z_S, apk) = 1$ ⟹ $x \in S$

Non.Memb.verify$(x, \phi, Z_S, apk) = 1$ ⟹ $x \notin S$

Set Membership NIZK With Accumulators

$S = \{a_1, \ldots, a_m\}$

$crs = apk$
+ NIZK system crs

$Z_S = commit(S, apk)$

Alice $\left(crs, Enc; (x, rand)\right)$

π, Enc_ϕ

Bob $(crs, Enc\ )$

Prove that

Memb.verify$\left(Enc, Enc_\phi, Z_S, apk\right) = 1$

Non.Memb.verify$\left(Enc, Enc_\phi, \phi, Z_S, apk\right) = 1$

- crs depends only from $|S|$.

- It allows Non-membership proof

Falsifiable Set-membership (without ROM)

Constructions			
Primitives			
Signature or accumulators			
Communication and computational complexity			
Assumptions			

Cryptographic groups

- Bracket notation for additive groups

$$\mathcal{G} = \langle g \rangle := [1],$$
$$[x] \in \mathcal{G}: [x] = x[1] \; (= x\, g),$$

- Hardness assumptions

1. $x \leftarrow [x]$ is hard (discrete logarithm assumption)
2. $[x\, y] \leftarrow ([x], [y])$ is hard (CDH assumption)

Bilinear Pairing Groups

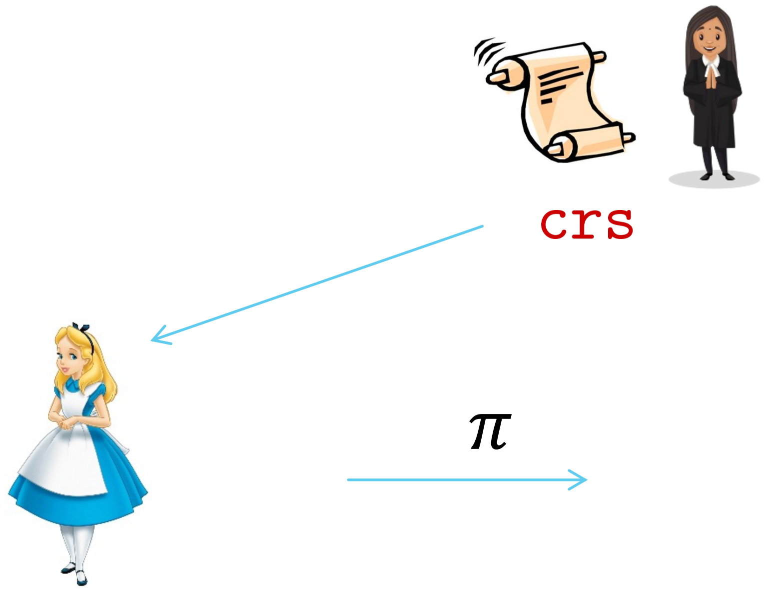
- Three additive groups cryptographic groups

$$(p, \mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_T, [1]_1, [1]_2, \cdot)$$

p is the order of each group

1. $[x]_1 \cdot [y]_2 = [x\, y]_T$
2. $[x]_1 \leftrightarrow [x]_2$ is hard (type III pairings: no efficient isomorphism between groups)

Groth-Sahai Set Membership NIZKs



Underlying primitive + GS crs

$$ver(crs, x, \pi) = 0$$

Primitive PPE verification

+

Groth-Sahai for ZK

Pairing-based Set-membership (without ROM)

Constructions	AN11	DGP+19
Primitives	AN accumulator Groth-Sahai	BB signatures Groth-Sahai
Signature or accumulators		
Communication and computational complexity		
Assumptions		

A matrix C is a a **QDR (Quasi-Determinantal Representation)** of a polynomial F if

1. **Affine map**: each entry of C is an affine function
2. **F -rank**: $Det\left(C(\vec{x})\right) = F(\vec{x})$

Determinantal representation

3. **First column dependence**

- $\mathcal{L}_{\{pk,C\}} = \left\{[ct]_1 : \exists r, \vec{x}, Enc_{pk}(\vec{x}; r) = [ct]_1 \wedge \det\left(C(\vec{x})\right) = 0\right\}$

ElGamal (linear homomorphic)

[CLPO21] NIZK

$[e]_2$

Prover $([e]_2, [ct]_1, r, \vec{x})$

Verifier $([e]_2, [ct]_1)$

Compute $\vec{\gamma}$

$[ct_\gamma]_1 \leftarrow Enc_{pk}(\vec{\gamma})$

Compute $[\vec{\delta}, \vec{z}]_2$

$[ct_\gamma]_1, [\vec{\delta}, \vec{z}]_2$ →

Accept if

$$[\vec{\gamma}]_1 \cdot [1]_2 + [C(\vec{x})]_1 \cdot \begin{bmatrix} e \\ \vec{\delta} \end{bmatrix}_2 = [0]_T$$

Check **encrypted version**

First column dependence

$\vec{\gamma}$ in $[\cdot]_1$, e in $[\cdot]_2$ → **Soundness**

CLPØ Set Membership NIZKs

crs

$[Enc_{\vec{\gamma}}, Enc_{\phi}]_1, [\vec{\delta}, \vec{z}]_2$

Determinantal accumulator key($\boldsymbol{apk}$) + $[e]_2$

$$[\vec{\gamma}]_1 \cdot [1]_2 + [(C\ (x, \phi))]_1 \cdot \begin{bmatrix} e \\ \vec{\delta} \end{bmatrix}_2 = [0]_T$$

Determinantal verification

\+

CLPØ for ZK

$$\mathcal{L}_{\{pk, S, apk\}} = \{[ct]_1 : \exists r, x\ .\ Enc_{pk}(x; r) = [ct]_1 \wedge Det\big(C(x, \phi)\big) = 0\}$$

CLPØ ≫ Groth-Sahai

[CH20,CLPO21,GKP22,LP23]

- Language defined in $\mathcal{G}_1$ only
  - $\mathcal{G}_1$ complexity $\approx \frac{1}{2}\, \mathcal{G}_2$ complexity
  - ElGamal can always be used
- Simple design and automatic optimization
- Shorter, uniformly random crs

But …

- Less standard, new (falsifiable) assumptions

sound Accumulators

Needed for sound Falsifiable NIZK

$S = \{a_1, \ldots, a_m\}$

$crs = Z_S, apk$

$Z_S = commit(S, apk)$

Alice (crs, x)

$[x]_1$ x, ϕ, ϕ

Bob $(crs\ )$

Memb.verify$(x, \phi, Z_S, apk) = 1$ ⟹ $x \in S$

Non.Memb.verify$(x, \phi, Z_S, apk) = 1$ ⟹ $x \notin S$

$[\,\cdot\,]_1$-sound GS friendly accumulator

- Add a a GS proof of a “knowledge equation”

Source of inefficiency

+1 equation, +1 committed variable

$[\,\cdot\,]_1$-sound determinantal accumulator

- Almost for free (not affect proof size).

Big efficiency gain

Conclusion

- We define the notion of determinatal primitives (friendly with CLPØ NIZK framework)
- We propose a new determinantal accumulator
- We propose a set (non-)membership NIZK in the standard model, with efficiency comparable with corresponding NIZK in the ROM
- We give more evidence that the CLPØ framework is a valid route to improve over GS

Thanks for your attention

Check the full version
On eprint
Questions?

Bibliography:
[CH20]: Shorter Non-Interactive Zero-Knowledge Arguments and ZAPs for Algebraic Languages
[CLPO21]: Efficient NIZKs for Algebraic Sets
[GKP22]: NIWI and New Notions of Extraction for Algebraic Languages
[LP22]: full version of this paper (eprint)