Structure-Preserving Signatures on Equivalence Classes From Standard Assumptions

Mojtaba Khalili[‡], Daniel Slamanig[§], Mohammad Dakhilalian[‡]

ASIACRYPT 2019, Kobe, Japan, December 11, 2019

‡ Isfahan University of Technology

§ AIT Austrian Institute of Technology

Talk Outline

- Structure-Preserving Signatures and Applications
- Structure-Preserving Signatures on Equivalence Classes
- Overview of the State-of-the-Art
- Our Approach
- Take Home & Open Questions

Structure-Preserving Signatures and Applications

Structure-Preserving Signatures (SPS) [AFGHO10]

Bilinear groups

$\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are cyclic groups of prime order p

- $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$

Structure-Preserving Signatures (SPS) [AFGHO10]

Bilinear groups

$\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are cyclic groups of prime order p

- $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$

Structure-Preserving Signatures (SPS)

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}) :\ \mathsf{pk} \in \mathbb{G}_i^k$ with $s \in \{1, 2\}$

Structure-Preserving Signatures (SPS) [AFGHO10]

Bilinear groups

$\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are cyclic groups of prime order p

- $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$

Structure-Preserving Signatures (SPS)

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par})$: $\mathsf{pk} \in \mathbb{G}_i^k$ with $s \in \{1, 2\}$

$\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$: $m \in \mathbb{G}_i^n$; $\sigma \in \mathbb{G}_1^u \times \mathbb{G}_2^v$

Structure-Preserving Signatures (SPS) [AFGHO10]

Bilinear groups

$\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are cyclic groups of prime order p

- $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$

Structure-Preserving Signatures (SPS)

$(\mathrm{sk}, \mathrm{pk}) \leftarrow \mathrm{KeyGen}(\mathrm{par})$: $\mathrm{pk} \in \mathbb{G}_i^k$ with $s \in \{1, 2\}$

$\sigma \leftarrow \mathrm{Sign}(\mathrm{sk}, m)$: $m \in \mathbb{G}_i^n$; $\sigma \in \mathbb{G}_1^u \times \mathbb{G}_2^v$

$\{0, 1\} \leftarrow \mathrm{Verify}(\mathrm{pk}, m, \sigma)$: Only uses

pairing-product equations

$$\prod_i \prod_j e(A_i, \hat{B}_j)^{a_{ij}} = Z, \quad \text{and}$$

group membership tests.

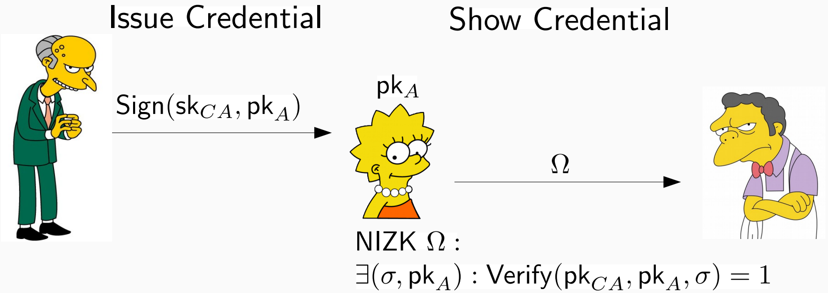
Applications of SPS

Compatible with efficient Groth-Sahai (GS) NIZK proofs

Numerous privacy-preserving applications

(Delegatable) anonymous credentials, group signatures, traceable signatures, blind signatures, anonymous e-cash, verifiable shuffles (e-voting), etc.

Example: Simple Anonymous Credentials



Structure-Preserving Signatures on Equivalence Classes

Structure-Preserving Signatures on Equivalence Classes [HS14]

SPS that signs an equivalence class $[\mathbf{m}]_{\mathcal{R}}$.

- Produce signature given some representative
- Signature for one class is signature for **every** representative of that class

Structure-Preserving Signatures on Equivalence Classes [HS14]

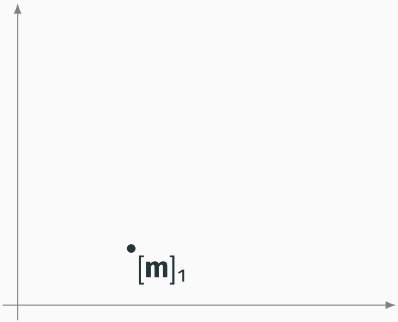
SPS that signs an equivalence class $[\mathbf{m}]_{\mathcal{R}}$.

- Produce signature given some representative
- Signature for one class is signature for **every** representative of that class

The equivalence relation $\sim_{\mathcal{R}}$

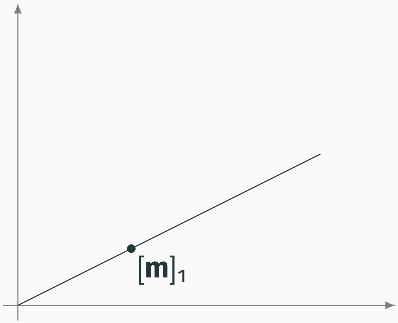
$$\mathbf{m} \in (\mathbb{G}_i^*)^\ell \sim_{\mathcal{R}} \mathbf{n} \in (\mathbb{G}_i^*)^\ell \Leftrightarrow \exists \mu \in \mathbb{Z}_p^* : \mathbf{m} = \mu \mathbf{n}$$

Structure-Preserving Signatures on Equivalence Classes



- Vector $[\mathbf{m}]_1$ of group elements

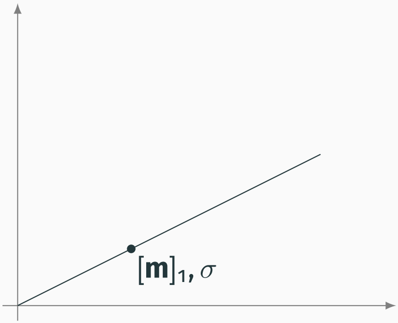
Structure-Preserving Signatures on Equivalence Classes



- Vector $[\mathbf{m}]_1$ of group elements
- EQ classes

  $\sim_{\mathcal{R}}$ mutual ratios of DLOGs

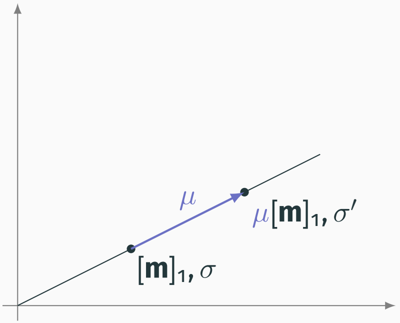
Structure-Preserving Signatures on Equivalence Classes



- Vector $[\mathbf{m}]_1$ of group elements
- EQ classes

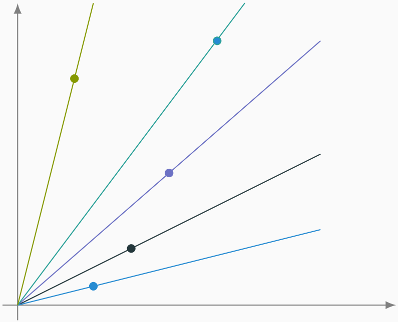
  $\sim_{\mathcal{R}}$ mutual ratios of DLOGs
- Sign representative

Structure-Preserving Signatures on Equivalence Classes



- Vector $[\mathbf{m}]_1$ of group elements
- EQ classes
  $\sim_{\mathcal{R}}$ mutual ratios of DLOGs
- Sign representative
- Switch representative using μ publicly
- Adapt signature to σ' publicly

Structure-Preserving Signatures on Equivalence Classes



- Vector $[\mathbf{m}]_1$ of group elements
- EQ classes
  $\sim_{\mathcal{R}}$ mutual ratios of DLOGs
- Sign representative
- Switch representative using μ publicly
- Adapt signature to σ' publicly

Unlinkability on message space

- No advantage in distinguishing classes given representatives

Unlinkability of signatures (Adaption)

- Adapted signatures indistinguishable from fresh ones

Applications of SPS-EQ

Turned out to be a very versatile tool

- Avoid GS NIZKs
- Instead randomize message and adapt signature

Applications of SPS-EQ

Turned out to be a very versatile tool

- Avoid GS NIZKs
- Instead randomize message and adapt signature

- (Delegatable) anonymous credentials [HS14, DHS15, FHS19, CL19]
- Self-blindable certificates [BHKS18]
- Round-optimal blind signatures [FHS15, FHKS16]
- Group signatures [DS18, BHKS18, CS18, BHS19]
- Verifiably encrypted signatures [HRS15]
- Access control encryption [FGKO17]
- Scalable mix-nets [HPP19]

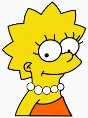
Example: Simple Anonymous Credentials v2.0

Issue Credential

Show Credential

$\mathsf{Sign}(\mathsf{sk}_{CA}, \mathsf{pk}_A)$

$\mathsf{pk}_A = (g, g^x)$



$(g, g^x),\ \mathsf{Sign}(g, g^x)$

Switch representative using μ

$(g^\mu, g^{\mu x}),\ \mathsf{Sign}(g^\mu, g^{\mu x})$

Formal Framework

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\\\allow others pars beyond **BG**

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\\\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

$(\sigma, \tau) \leftarrow \mathsf{Sign}([\mathbf{m}]_i, \mathsf{sk})$ \\\\allow tag τ

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\\\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

$(\sigma, \tau) \leftarrow \mathsf{Sign}([\mathbf{m}]_i, \mathsf{sk})$ \\\\allow tag τ

$([\mathbf{m}']_i, \sigma') \leftarrow \mathsf{ChgRep}([\mathbf{m}]_i, (\sigma, \tau), \mu, \mathsf{pk})$

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\\\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

$(\sigma, \tau) \leftarrow \mathsf{Sign}([\mathbf{m}]_i, \mathsf{sk})$ \\\\allow tag τ

$([\mathbf{m}']_i, \sigma') \leftarrow \mathsf{ChgRep}([\mathbf{m}]_i, (\sigma, \tau), \mu, \mathsf{pk})$

$\{0, 1\} \leftarrow \mathsf{Verify}([\mathbf{m}]_i, (\sigma, \tau), \mathsf{pk})$ \\\\w/o tag τ

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

$(\sigma, \tau) \leftarrow \mathsf{Sign}([\mathbf{m}]_i, \mathsf{sk})$ \\allow tag τ

$([\mathbf{m}']_i, \sigma') \leftarrow \mathsf{ChgRep}([\mathbf{m}]_i, (\sigma, \tau), \mu, \mathsf{pk})$

$\{0, 1\} \leftarrow \mathsf{Verify}([\mathbf{m}]_i, (\sigma, \tau), \mathsf{pk})$ \\w/o tag τ

$\{0, 1\} \leftarrow \mathsf{VKey}(\mathsf{sk}, \mathsf{pk})$

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

$(\sigma, \tau) \leftarrow \mathsf{Sign}([\mathbf{m}]_i, \mathsf{sk})$ \\allow tag τ

$([\mathbf{m}']_i, \sigma') \leftarrow \mathsf{ChgRep}([\mathbf{m}]_i, (\sigma, \tau), \mu, \mathsf{pk})$

$\{0, 1\} \leftarrow \mathsf{Verify}([\mathbf{m}]_i, (\sigma, \tau), \mathsf{pk})$ \\w/o tag τ

$\{0, 1\} \leftarrow \mathsf{VKey}(\mathsf{sk}, \mathsf{pk})$

Definition of SPS-EQ

SPS-EQ

$\mathsf{par} \leftarrow \mathsf{ParGen}(1^\lambda)$ \\allow others pars beyond **BG**

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(\mathsf{par}, \ell)$

$(\sigma, \tau) \leftarrow \mathsf{Sign}([\mathbf{m}]_i, \mathsf{sk})$ \\allow tag τ

$([\mathbf{m}']_i, \sigma') \leftarrow \mathsf{ChgRep}([\mathbf{m}]_i, (\sigma, \tau), \mu, \mathsf{pk})$

$\{0, 1\} \leftarrow \mathsf{Verify}([\mathbf{m}]_i, (\sigma, \tau), \mathsf{pk})$ \\w/o tag τ

$\{0, 1\} \leftarrow \mathsf{VKey}(\mathsf{sk}, \mathsf{pk})$

Tag-based schemes have one-time randomizability

- Only (σ, τ) from **Sign** can be put into **ChgRep** (enough for almost all applications)

EUF-CMA Security

$\text{par} \leftarrow \text{ParGen}(1^\lambda)$

$(\text{sk},\text{pk}) \leftarrow \text{KeyGen}(\text{par}, \ell)$

$\xleftarrow{\text{pk}}$

$\xrightarrow{[\mathbf{m}]_1 \in (\mathbb{G}_1^*)^\ell}$

$\xleftarrow[\text{Sign}([\mathbf{m}]_1, \text{sk})]{}$

$\xrightarrow[([\mathbf{m}]_1^*, \sigma^*)]{}$

Win :

$\text{Verify}([\mathbf{m}]_1^*, \sigma^*, \text{pk}) = 1 \quad \wedge$

$[\mathbf{m}]_{\mathcal{R}}^* \neq [\mathbf{m}]_{\mathcal{R}}$

Security Properties: Unforgeability

Weak EUF-CMA Security [FG18]

par $\leftarrow$ ParGen(1^λ)

(sk,pk)$\leftarrow$ KeyGen(par, ℓ)

pk $\longleftarrow$

$\mathrm{dlog}([\mathbf{m}]_1) \in (\mathbb{Z}_p^*)^\ell$ $\longrightarrow$

$\longleftarrow$ Sign($[\mathbf{m}]_1$, sk)

$\longrightarrow$ $([\mathbf{m}]_1^*, \sigma^*)$

Win :
Verify($[\mathbf{m}]_1^*, \sigma^*$, pk) $= 1 \quad \wedge$
$[\mathbf{m}]_\mathcal{R}^* \neq [\mathbf{m}]_\mathcal{R}$

Unlinkability of Messages and Signatures

Unlinkability of Messages

$$[\mathbf{m}]_1 \leftarrow_{\$} [\mathbf{m}]_{\mathcal{R}} \quad \approx \quad [\mathbf{m}]_1 \leftarrow_{\$} (\mathbb{G}_1^*)^\ell$$

Unlinkability of Messages and Signatures

Unlinkability of Messages

$$[\mathbf{m}]_1 \leftarrow_{\$} [\mathbf{m}]_{\mathcal{R}} \quad \approx \quad [\mathbf{m}]_1 \leftarrow_{\$} (\mathbb{G}_1^*)^\ell$$

Assume DDH in $\mathbb{G}_1^*$ ✓

Unlinkability of Messages and Signatures

Unlinkability of Messages

$$[\mathbf{m}]_1 \leftarrow_{\$} [\mathbf{m}]_{\mathcal{R}} \quad \approx \quad [\mathbf{m}]_1 \leftarrow_{\$} (\mathbb{G}_1^*)^{\ell}$$

Assume DDH in $\mathbb{G}_1^*$ ✓

Unlinkability of Signatures (Adaption)

$$(\mu[\mathbf{m}]_1, \mathsf{Sign}(\mu[\mathbf{m}]_1, \mathsf{sk})) \quad \approx \quad \mathsf{ChgRep}([\mathbf{m}]_i, \mathsf{Sign}([\mathbf{m}]_1, \mathsf{sk}), \mu, \mathsf{pk})$$

Unlinkability of Messages and Signatures

Unlinkability of Messages

$$[\mathbf{m}]_1 \leftarrow_\$ [\mathbf{m}]_\mathcal{R} \quad \approx \quad [\mathbf{m}]_1 \leftarrow_\$ (\mathbb{G}_1^*)^\ell$$

Assume DDH in $\mathbb{G}_1^*$ ✓

Unlinkability of Signatures (Adaption)

$$(\mu[\mathbf{m}]_1, \mathsf{Sign}(\mu[\mathbf{m}]_1, \mathsf{sk})) \quad \approx \quad \mathsf{ChgRep}([\mathbf{m}]_i, \mathsf{Sign}([\mathbf{m}]_1, \mathsf{sk}), \mu, \mathsf{pk})$$

Keys and/or signature generated honestly or maliciously?

Turns out to be quite subtle for applications

Security Properties: Adaption

	Keys	Signatures
Honest	$\mathsf{VKey}(\mathsf{sk}, \mathsf{pk}) = 1$ (HK)	
Malicious		

Security Properties: Adaption

	Keys	Signatures
Honest	$\mathsf{VKey}(\mathsf{sk}, \mathsf{pk}) = 1$ (HK)	
Malicious	$\mathsf{Verify}(\ldots, \mathsf{pk}) = 1$ (MK)	

Security Properties: Adaption

	Keys	Signatures
Honest	$\mathsf{VKey}(\mathsf{sk}, \mathsf{pk}) = 1$ (HK)	$\sigma \leftarrow \mathsf{Sign}(\ldots)$ (HS)
Malicious	$\mathsf{Verify}(\ldots, \mathsf{pk}) = 1$ (MK)	

Security Properties: Adaption

	Keys	Signatures
Honest	$\mathsf{VKey}(\mathsf{sk}, \mathsf{pk}) = 1$ (HK)	$\sigma \leftarrow \mathsf{Sign}(\ldots)$ (HS)
Malicious	$\mathsf{Verify}(\ldots, \mathsf{pk}) = 1$ (MK)	$\mathsf{Verify}(\cdot, \sigma, \cdot) = 1$ (MS)

Security Properties: Adaption

	Keys	Signatures
Honest	$\mathsf{VKey}(\mathsf{sk}, \mathsf{pk}) = 1$ (HK)	$\sigma \leftarrow \mathsf{Sign}(\ldots)$ (HS)
Malicious	$\mathsf{Verify}(\ldots, \mathsf{pk}) = 1$ (MK)	$\mathsf{Verify}(\cdot, \sigma, \cdot) = 1$ (MS)

In addition: Honest parameter model **(HP)**

- **par** generated honestly, but keys can be generated maliciously
- (MK,MS) in HP gives (HK,MS)

Security Properties: Adaption

	Keys	Signatures
Honest	$\mathsf{VKey}(\mathsf{sk}, \mathsf{pk}) = 1$ (HK)	$\sigma \leftarrow \mathsf{Sign}(\dots)$ (HS)
Malicious	$\mathsf{Verify}(\dots, \mathsf{pk}) = 1$ (MK)	$\mathsf{Verify}(\cdot, \sigma, \cdot) = 1$ (MS)

In addition: Honest parameter model **(HP)**

- **par** generated honestly, but keys can be generated maliciously
- (MK,MS) in HP gives (HK,MS)

- (HK,HS) introduced in [FG18]
- (HK,MS) and (MK,MS) introduced in [FHS15]

Overview of the State-of-the-Art

Overview: State-of-the-Art

Scheme	Unforgeability	Assumption	Adaption
[FHS15]	EUF-CMA	GGM	MK, MS

Overview: State-of-the-Art

Scheme	Unforgeability	Assumption	Adaption
[FHS15]	EUF-CMA	GGM	MK, MS
[FG18]	Weak EUF-CMA*	DLIN	HK, HS**

*Weak EUF-CMA sufficient for most applications

**Adaption under honest keys and signatures (HK, HS) too weak for most applications (see Paper for details)

Overview: State-of-the-Art

Scheme	Unforgeability	Assumption	Adaption
[FHS15]	EUF-CMA	GGM	MK, MS
[FG18]	Weak EUF-CMA*	DLIN	HK, HS**
This work	EUF-CMA	SXDH	MK, MS (HP)***

*Weak EUF-CMA sufficient for most applications

**Adaption under honest keys and signatures (HK, HS) too weak for most applications (see Paper for details)

***Sufficient for almost all applications

Overview: State-of-the-Art

Scheme	Unforgeability	Assumption	Adaption
[FHS15]	EUF-CMA	GGM	MK, MS
[FG18]	Weak EUF-CMA*	DLIN	HK, HS**
This work	EUF-CMA	SXDH	MK, MS (HP)***

*Weak EUF-CMA sufficient for most applications

**Adaption under honest keys and signatures (HK, HS) too weak for most applications (see Paper for details)

***Sufficient for almost all applications

EUF-CMA Secure SPS-EQ from Standard Assumptions

Our Approach

Common technique to construct (tightly secure) SPS under standard assumptions

- One-time (SP) MAC $\longmapsto$ Many-time (SP) MAC $\longmapsto$ SPS

Numerous works [BKP14,KW15,KPW15,GHK17,GHKP18,AJOPRW19]

Common technique to construct (tightly secure) SPS under standard assumptions

- One-time (SP) MAC $\longmapsto$ Many-time (SP) MAC $\longmapsto$ SPS

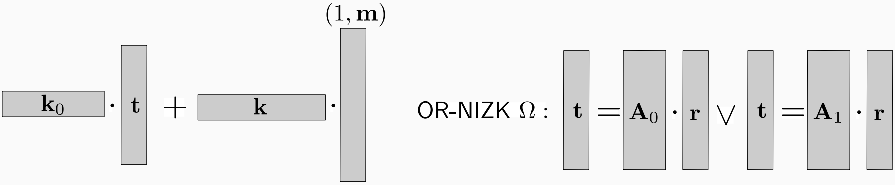
Numerous works [BKP14,KW15,KPW15,GHK17,GHKP18,AJOPRW19]

- Weakly EUF-CMA secure SPS-EQ in [FG18] use [BKP14] as starting point
- We use [GHKP18] as a starting point

Starting from the MAC of [GHKP18]

$$\mathbf{k}_0 \cdot \mathbf{t} + \mathbf{k} \cdot (1, \mathbf{m})$$

OR-NIZK Ω : $\mathbf{t} = \mathbf{A}_0 \cdot \mathbf{r} \bigvee \mathbf{t} = \mathbf{A}_1 \cdot \mathbf{r}$



Hurdles to overcome

- Make MAC linear to switch within class
- Have malleable and perfectly randomizable proofs Ω

Starting from the MAC of [GHKP18]

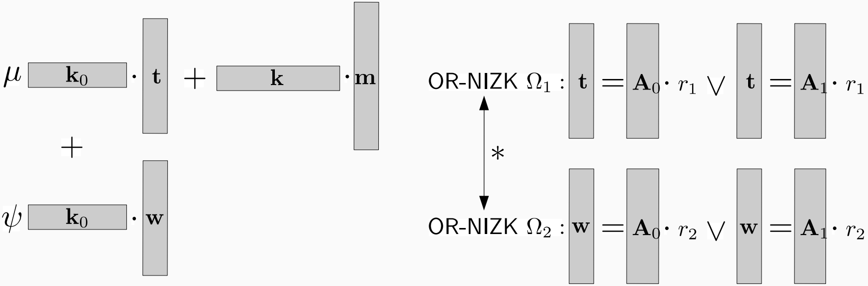
$$\mathbf{k}_0 \cdot \mathbf{t} + \mathbf{k} \cdot (\cancel{1}, \mathbf{m})$$

$$\text{OR-NIZK } \Omega : \mathbf{t} = \mathbf{A}_0 \cdot \mathbf{r} \vee \mathbf{t} = \mathbf{A}_1 \cdot \mathbf{r}$$

Hurdles to overcome

- Make MAC linear ✓ to switch within class
- Have malleable and perfectly randomizable proofs Ω

Doubling of a modified MAC of [GHKP18]



First steps

- Add second “MAC” (to empty message), which acts as tag
- $*$ Doubling OR-NIZK, sharing randomness
- Fix $\boldsymbol{k} = \mathbf{1}$ ($\mathbf{A}_0$, $\mathbf{A}_1$ vectors) – instantiation from SXDH only

Achieving Malleability and Perfect Randomizability

Modify the OR-NIZK of [GHKP18]

Problem

- [GHKP18] fixes $[\mathbf{z}]_2$ in CRS and provide $[\mathbf{z}_0]_2$ and $[z_1]_2$ s.t. $[\mathbf{z}]_2 = [\mathbf{z}_0]_2 + [\mathbf{z}_1]_2$ and at least one is in span$(\mathbf{z})$

Achieving Malleability and Perfect Randomizability

Modify the OR-NIZK of [GHKP18]

Problem

- [GHKP18] fixes $[\mathbf{z}]_2$ in CRS and provide $[\mathbf{z}_0]_2$ and $[z_1]_2$ s.t. $[\mathbf{z}]_2 = [\mathbf{z}_0]_2 + [\mathbf{z}_1]_2$ and at least one is in span$(\mathbf{z})$

Replace this part with a homomorphic QA-NIZK [JR14]

- Show that one of $[\mathbf{z}_0]_2$ and $[\mathbf{z}_1]_2$ is in span$(\mathbf{D} + \mathbf{z})$
- Preservers the soundness of OR-NIZK

Achieving Malleability and Perfect Randomizability

Modify the OR-NIZK of [GHKP18]

Problem

- [GHKP18] fixes $[\mathbf{z}]_2$ in CRS and provide $[\mathbf{z}_0]_2$ and $[z_1]_2$ s.t. $[\mathbf{z}]_2 = [\mathbf{z}_0]_2 + [\mathbf{z}_1]_2$ and at least one is in span$(\mathbf{z})$

Replace this part with a homomorphic QA-NIZK [JR14]

- Show that one of $[\mathbf{z}_0]_2$ and $[\mathbf{z}_1]_2$ is in span$(\mathbf{D} + \mathbf{z})$
- Preservers the soundness of OR-NIZK

Malleable ✓ and perfectly randomizable ✓ proofs

Now supports additive update of the two OR-NIZK yielding a perfectly distributed fresh proof for witness $\boldsymbol{r}' = \boldsymbol{r}_1 + \psi \boldsymbol{r}_2$ and word $[\mathbf{t}']_1 = \mu[\mathbf{t}]_1 + \psi[\mathbf{w}]_1$

Concrete Comparison

Scheme	$\lvert$Signature$\rvert$	$\lvert$PK$\rvert$	Ass.	Red. Loss
[FHS15]	$2\lvert\mathbb{G}_1\rvert + 1\lvert\mathbb{G}_2\rvert$	$\ell\lvert\mathbb{G}_2\rvert$	GGM	-
[FG18]	$(4\ell+2)\lvert\mathbb{G}_1\rvert + 4\lvert\mathbb{G}_2\rvert$	$(4\ell+2)\lvert\mathbb{G}_2\rvert$	DLIN	$\mathcal{O}(Q)$
This work	$8\lvert\mathbb{G}_1\rvert + 9\lvert\mathbb{G}_2\rvert$	$3\ell\lvert\mathbb{G}_2\rvert$	SXDH	$\mathcal{O}(\log Q)^*$

Concrete Comparison

Scheme	$\|$Signature$\|$	$\|$PK$\|$	Ass.	Red. Loss
[FHS15]	$2\|\mathbb{G}_1\| + 1\|\mathbb{G}_2\|$	$\ell\|\mathbb{G}_2\|$	GGM	-
[FG18]	$(4\ell + 2)\|\mathbb{G}_1\| + 4\|\mathbb{G}_2\|$	$(4\ell + 2)\|\mathbb{G}_2\|$	DLIN	$\mathcal{O}(Q)$
This work	$8\|\mathbb{G}_1\| + 9\|\mathbb{G}_2\|$	$3\ell\|\mathbb{G}_2\|$	SXDH	$\mathcal{O}(\log Q)^*$

*Tightness inherited from [GHKP18]

Concrete Applications

- Group signatures in [DS18] and [BHKS18]

Concrete Applications

- Group signatures in [DS18] and [BHKS18]
- Access control encryption (ACE) in [FGKO17]

Concrete Applications

- Group signatures in [DS18] and [BHKS18]
- Access control encryption (ACE) in [FGKO17]
- Self-blindable certificates [BHKS18]

Concrete Applications

- Group signatures in [DS18] and [BHKS18]
- Access control encryption (ACE) in [FGKO17]
- Self-blindable certificates [BHKS18]
- Attribute-based credentials wo malicious issuer (or with a CRS) [HS14, FHS19]

Concrete Applications

- Group signatures in [DS18] and [BHKS18]
- Access control encryption (ACE) in [FGKO17]
- Self-blindable certificates [BHKS18]
- Attribute-based credentials wo malicious issuer (or with a CRS) [HS14, FHS19]
- Shortest round-optimal blind signatures with a CRS; improving by about a factor of 4 (using the template in [FHS15,FHKS16])

Take Home & Open Questions

Conclusion

Take Home

- SPS-EQ are a versatile tool for privacy-preserving applications
- First EUF-CMA secure SPS-EQ under standard assumptions (SXDH)

Conclusion

Take Home

- SPS-EQ are a versatile tool for privacy-preserving applications
- First EUF-CMA secure SPS-EQ under standard assumptions (SXDH)

Open Questions

- Apply our idea to other SPS to improve efficiency and/or support other assumptions
- Construct SPS-EQ under standard assumption that support malicious keys wo HP, i.e., (MK,MS)
  - Constructions wo CRS seem very hard
  - Untrusted CRS?

Thank you! Questions?

@drl3c7er

Supported by EU ECSEL

and FWF/netidee SCIENCE PROFET