Delegatable Anonymous Credentials From Mercurial Signatures With Stronger Privacy

Scott Griffy[1], Anna Lysyanskaya[1], Omid Mir[2], Octavio Pérez Kempner[3], Daniel Slamanig[4]

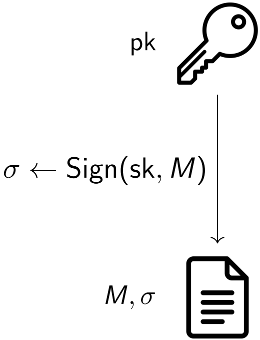
[1]Brown University, [2]Austrian Institute of Technology, [3]NTT Social Informatics Laboratories, [4]Universität der Bundeswehr München

[GLM+24]

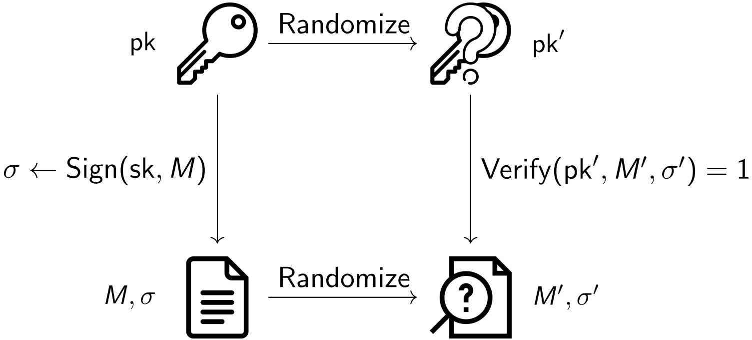
Overview of talk

1. Mercurial signatures (MS) overview
2. MS Background
3. Open problem with existing MS construction
4. Our contributions (fixing this problem and more)

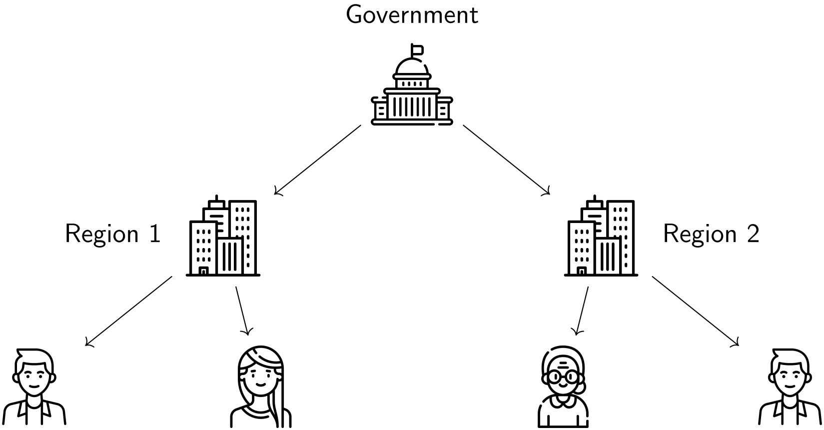
Mercurial Signatures



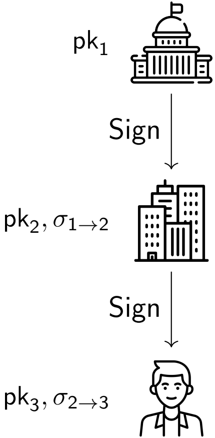
Mercurial Signatures



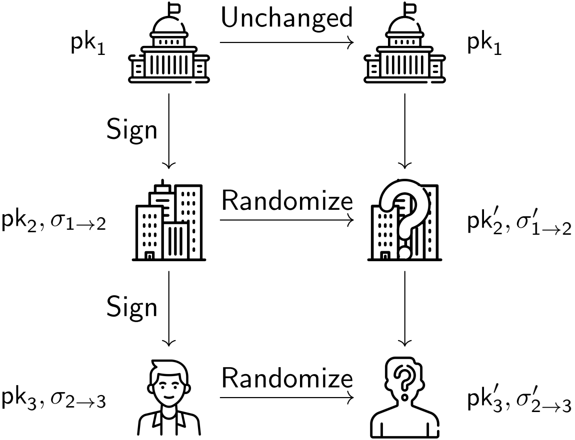
Delegatable anonymous credentials



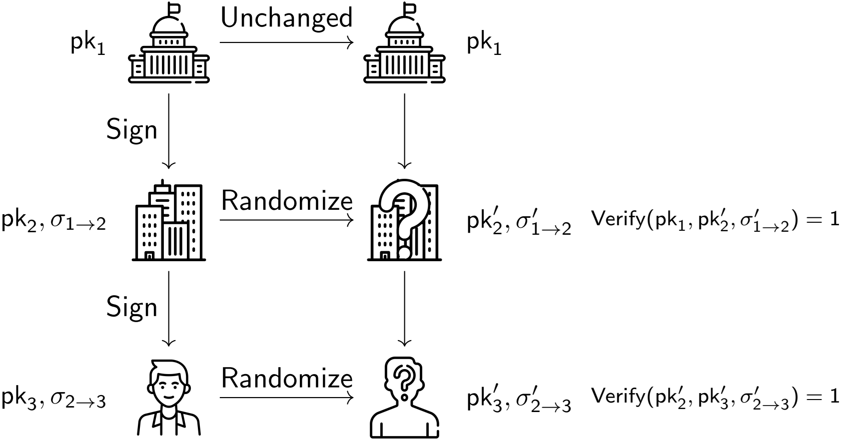
Delegatable anonymous credentials from mercurial signatures



Delegatable anonymous credentials from mercurial signatures



Delegatable anonymous credentials from mercurial signatures



Mercurial Signatures Background

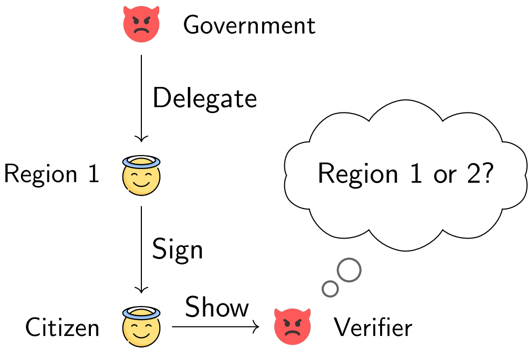
Structure-preserving-signatures on equivalence classes - [FHS19]

Signatures with flexible public keys - [BHKS18]

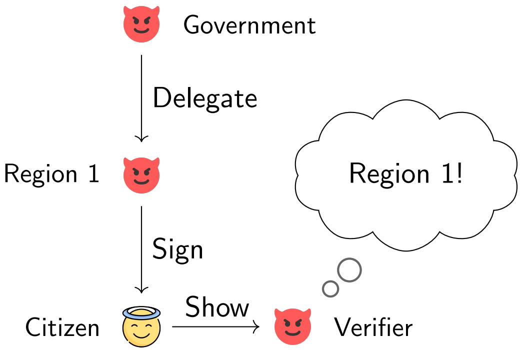
Mercurial Signatures
- [CL19, CLPK22, MB**G**$^{+}$23, PM23, ANKT24, CL21]

Delegatable anonymous credentials
- [BCC^{+}09, MSBM23, BB18, AN11]

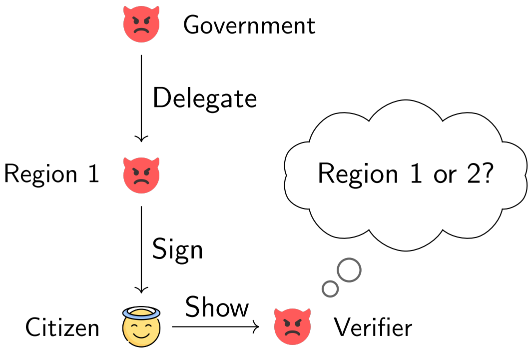
Anonymity in CL19



Open problem in CL19



Our main contribution



Our contributions

1. Mercurial signature scheme with stronger privacy
   - Structured CRS
   - Type-III bilinear pairings + GGM (same setting as CL19)
2. DAC construction with strong privacy from our signatures
3. Revocation of public keys (including intermediate issuers)
4. Serially updating CRS (similiar to [GKM$^+$18])

Mercurial signatures

Definition (Mercurial signatures (basic part))

$\mathsf{KeyGen}(\mathsf{pp}) \to (\mathsf{pk}, \mathsf{sk})$
$\mathsf{Sign}(\mathsf{sk}, M) \to \sigma$
$\mathsf{Verify}(\mathsf{pk}, M, \sigma) \to (0 \text{ or } 1)$

Mercurial signatures - Randomizability

Definition (Randomization correctness)

Given key, $(\mathsf{pk}, \mathsf{sk})$, verfying message and signature, (M, σ), conversion factor ρ and:
$\mathsf{pk}' \leftarrow \mathsf{ConvertPK}(\mathsf{pk}; \rho)$
$\sigma' \leftarrow \mathsf{ConvertSig}(\sigma; \rho)$
It holds that:
$\mathsf{Verify}(\mathsf{pk}', M, \sigma') = 1$

Mercurial signatures - Randomizability

Definition (Randomization correctness)

Given key, $(\mathsf{pk}, \mathsf{sk})$, verfying message and signature, (M, σ), conversion factor ρ and:

$\mathsf{pk}' \leftarrow \mathsf{ConvertPK}(\mathsf{pk}; \rho)$

$\sigma' \leftarrow \mathsf{ConvertSig}(\sigma; \rho)$

It holds that:

$\mathsf{Verify}(\mathsf{pk}', M, \sigma') = 1$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger Adversary

😇 😈

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$
$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$b \leftarrow\$ \{1, 2\}; \rho \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger   Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$b \leftarrow\$ \{1, 2\}; \rho \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$$\xrightarrow{\mathsf{pk}_1, \mathsf{pk}'_b}$$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger | Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$b \leftarrow\!\!\$ \{1, 2\}; \rho \leftarrow\!\!\$ \mathbb{Z}_p^*$

$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\xrightarrow{\mathsf{pk}_1, \mathsf{pk}'_b}$

$\xleftarrow{M}$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger | Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$
$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$
$b \leftarrow\!\!\$ \{1, 2\}; \rho \leftarrow\!\!\$ \mathbb{Z}_p^*$
$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\xrightarrow{\mathsf{pk}_1, \mathsf{pk}'_b}$

$\xleftarrow{M}$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$
$\sigma_2 = \mathsf{Sign}(\mathsf{sk}_2, M)$
$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger — Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$
$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$
$b \leftarrow\!\!\$ \{1, 2\}; \rho \leftarrow\!\!\$ \mathbb{Z}_p^*$
$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\xrightarrow{\mathsf{pk}_1, \mathsf{pk}'_b}$

$\xleftarrow{M}$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$
$\sigma_2 = \mathsf{Sign}(\mathsf{sk}_2, M)$
$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

$\xrightarrow{\sigma_1, \sigma'_b}$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger | Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$b \leftarrow\!\!\$ \{1, 2\}; \rho \leftarrow\!\!\$ \mathbb{Z}_p^*$

$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\xrightarrow{\mathsf{pk}_1, \mathsf{pk}'_b}$

$\xleftarrow{M}$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$

$\sigma_2 = \mathsf{Sign}(\mathsf{sk}_2, M)$

$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

$\xrightarrow{\sigma_1, \sigma'_b}$

$\xleftarrow{b^*}$

Mercurial signatures - Randomizability

Definition (Public key class-hiding [CL19])

Challenger | Adversary

$(\mathsf{sk}_1, \mathsf{pk}_1) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$(\mathsf{sk}_2, \mathsf{pk}_2) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$

$b \leftarrow\$ \{1, 2\}; \rho \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}'_b \leftarrow \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\xrightarrow{\mathsf{pk}_1, \mathsf{pk}'_b}$

$\xleftarrow{M}$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$

$\sigma_2 = \mathsf{Sign}(\mathsf{sk}_2, M)$

$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

$\xrightarrow{\sigma_1, \sigma'_b}$

$\xleftarrow{b^*}$

Adversary wins if $b^* = b$

Mercurial signatures from bilinear pairings [CL19]

Parameter generation (Setup):

Public parameters are a description of a bilinear pairing $\mathrm{pp} = (e, P, \hat{P})$. $e(P^a, \hat{P}^b) = e(P, \hat{P})^{ab}$. $\langle P \rangle = \mathbb{G}_1$, $\langle \hat{P} \rangle = \mathbb{G}_2$. $\#\mathbb{G}_1 = \#\mathbb{G}_2 = p$.

Mercurial signatures from bilinear pairings [CL19]

Parameter generation (Setup):

Public parameters are a description of a bilinear pairing $\mathsf{pp} = (e, P, \hat{P})$. $e(P^a, \hat{P}^b) = e(P, \hat{P})^{ab}$. $\langle P \rangle = \mathbb{G}_1$, $\langle \hat{P} \rangle = \mathbb{G}_2$. $\#\mathbb{G}_1 = \#\mathbb{G}_2 = p$.

Key generation (KeyGen):

$\mathsf{sk} = (x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2}) \in \mathbb{G}_2$

Mercurial signatures from bilinear pairings [CL19]

Parameter generation (Setup):

Public parameters are a description of a bilinear pairing $\mathsf{pp} = (e, P, \hat{P})$. $e(P^a, \hat{P}^b) = e(P, \hat{P})^{ab}$. $\langle P \rangle = \mathbb{G}_1$, $\langle \hat{P} \rangle = \mathbb{G}_2$. $\#\mathbb{G}_1 = \#\mathbb{G}_2 = p$.

Key generation (KeyGen):

$\mathsf{sk} = (x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2}) \in \mathbb{G}_2$

Randomization (ConvertPK):

$\rho \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

Mercurial signatures from bilinear pairings [CL19]

Parameter generation (Setup):

Public parameters are a description of a bilinear pairing $\mathsf{pp} = (e, P, \hat{P})$. $e(P^a, \hat{P}^b) = e(P, \hat{P})^{ab}$. $\langle P \rangle = \mathbb{G}_1$, $\langle \hat{P} \rangle = \mathbb{G}_2$. $\#\mathbb{G}_1 = \#\mathbb{G}_2 = p$.

Key generation (KeyGen):

$\mathsf{sk} = (x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2}) \in \mathbb{G}_2$

Randomization (ConvertPK):

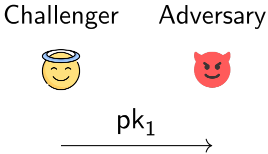
$\rho \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

Sign, Verify, ConvertSig, ChangeRep, ConvertSK

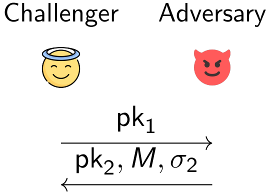
Our (adversarial) public key class-hiding definition

Definition (Adversarial Public key class-hiding (this paper))



Our (adversarial) public key class-hiding definition

Definition (Adversarial Public key class-hiding (this paper))



Our (adversarial) public key class-hiding definition

Definition (Adversarial Public key class-hiding (this paper))

Challenger 😇 Adversary 😈

Challenger → Adversary: pk_1

Adversary → Challenger: $\mathsf{pk}_2, M, \sigma_2$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$

$\mathsf{pk}'_b = \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

Our (adversarial) public key class-hiding definition

Definition (Adversarial Public key class-hiding (this paper))

Challenger | Adversary

$\xrightarrow{\mathsf{pk}_1}$

$\xleftarrow{\mathsf{pk}_2, M, \sigma_2}$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$

$\mathsf{pk}'_b = \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

$\xrightarrow{\sigma_1, \mathsf{pk}'_b, \sigma'_b}$

Our (adversarial) public key class-hiding definition

Definition (Adversarial Public key class-hiding (this paper))

Challenger | Adversary

Adversary → Challenger...

Challenger → Adversary: pk_1

Adversary → Challenger: $\mathsf{pk}_2, M, \sigma_2$

$\sigma_1 = \mathsf{Sign}(\mathsf{sk}_1, M)$

$\mathsf{pk}'_b = \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$

$\sigma'_b = \mathsf{ConvertSig}(\sigma_b; \rho)$

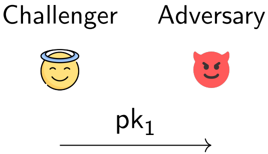
Challenger → Adversary: $\sigma_1, \mathsf{pk}'_b, \sigma'_b$

Adversary → Challenger: b^*

Adversary wins if $b^* = b$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):



Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

Challenger　　Adversary

$$\xrightarrow{\mathsf{pk}_1}$$

$$(x_1, x_2) \leftarrow_{\$} \mathbb{Z}_p^*$$

$$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

Challenger | Adversary

Challenger → Adversary: pk_1

Adversary: $(x_1, x_2) \xleftarrow{\$} \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

Adversary → Challenger: $\mathsf{pk}_2, M, \sigma_2$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

Challenger | Adversary

$$\xrightarrow{\mathsf{pk}_1}$$

$(x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

$$\xleftarrow{\mathsf{pk}_2, M, \sigma_2}$$

...$\mathsf{pk}'_b = \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$...

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

Challenger | Adversary

$\xrightarrow{\mathsf{pk}_1}$

$(x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

$\xleftarrow{\mathsf{pk}_2, M, \sigma_2}$

...$\mathsf{pk}'_b = \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)$...

$\xrightarrow{\sigma_1, \mathsf{pk}'_b, \sigma'_b}$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

Challenger | Adversary

$\xrightarrow{\mathsf{pk}_1}$

$(x_1, x_2) \leftarrow\!\!\$ \; \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

$\xleftarrow{\mathsf{pk}_2, M, \sigma_2}$

$\ldots\mathsf{pk}'_b = \mathsf{ConvertPK}(\mathsf{pk}_b; \rho)\ldots$

$\xrightarrow{\sigma_1, \mathsf{pk}'_b, \sigma'_b}$

Let $\mathsf{pk}'_b = (\hat{Z}_1, \hat{Z}_2)$

If $b = 1$: $(\hat{Z}_1)^{x_2/x_1} \neq \hat{Z}_2$

If $b = 2$: $(\hat{Z}_1)^{x_2/x_1} = \hat{Z}_2$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

$(x_1, x_2) \leftarrow\!\!\$\; \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

$(x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

$\mathsf{pk}'_2 = (\hat{X}'_1, \hat{X}'_2) = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

$(x_1, x_2) \leftarrow\!\!\$\ \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

$\mathsf{pk}_2' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

$\mathsf{dlog}_{\hat{X}_1}(\hat{X}_2) = x_2/x_1 = \mathsf{dlog}_{\hat{X}_1'}(\hat{X}_2')$

Our (adversarial) public key class-hiding definition

CL19 does not meet this new definition (attack):

$(x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}_2 = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$

$\mathsf{pk}_2' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

$\mathrm{dlog}_{\hat{X}_1}(\hat{X}_2) = x_2/x_1 = \mathrm{dlog}_{\hat{X}_1'}(\hat{X}_2')$

$(y_1, y_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk}_1 = (\hat{Y}_1, \hat{Y}_2) = (\hat{P}^{y_1}, \hat{P}^{y_2})$

$\mathrm{dlog}_{\hat{Y}_1}(\hat{Y}_2) = y_2/y_1 \neq \mathrm{dlog}_{\hat{X}_1'}(\hat{X}_2')$

Intuition of our MS construction

Structured CRS:

$b_1, b_2 \leftarrow\!\!\$\ \mathbb{Z}_p$

$\mathsf{pp} = (\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2})$

Intuition of our MS construction

Structured CRS:

$b_1, b_2 \leftarrow\$ \mathbb{Z}_p$

$\mathsf{pp} = (\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2})$

Key generation:

$\mathsf{sk} = (x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$

Intuition of our MS construction

Structured CRS:

$b_1, b_2 \leftarrow\$ \mathbb{Z}_p$

$\mathsf{pp} = (\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2})$

Key generation:

$\mathsf{sk} = (x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$

Key randomization:

$\mathsf{pk}' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

Intuition of our MS construction

Structured CRS:

$b_1, b_2 \leftarrow\!\!\$\; \mathbb{Z}_p$

$\mathsf{pp} = (\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2})$

Key generation:

$\mathsf{sk} = (x_1, x_2) \leftarrow\!\!\$\; \mathbb{Z}_p^*$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$

Key randomization:

$\mathsf{pk}' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

$(\hat{X}_1')^{x_2/x_1} = \hat{P}^{\rho b_1 x_2} \neq \hat{X}_2' = \hat{P}^{\rho b_2 x_2}$

Intuition of our MS construction

Structured CRS:
$b_1, b_2 \leftarrow\$ \mathbb{Z}_p$
$\mathsf{pp} = (\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2})$
Key generation:
$\mathsf{sk} = (x_1, x_2) \leftarrow\$ \mathbb{Z}_p^*$
$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$
Key randomization:
$\mathsf{pk}' = (\hat{X}_1', \hat{X}_2') = (\hat{X}_1^{\rho}, \hat{X}_2^{\rho})$

$(\hat{X}_1')^{x_2/x_1} = \hat{P}^{\rho b_1 x_2} \neq \hat{X}_2' = \hat{P}^{\rho b_2 x_2}$

Sign, Verify, ConvertSig, ChangeRep, ConvertSK

More intuition of our MS construction (starting from CL19)

Messages and keys:

$M = (M_1, M_2) \in (\mathbb{G}_1)^2$

$\mathsf{sk} = (x_1, x_2) \in (\mathbb{Z}_p^*)^2$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2}) \in (\mathbb{G}_2)^2$

More intuition of our MS construction (starting from CL19)

Messages and keys:
$M = (M_1, M_2)$
$\mathsf{sk} = (x_1, x_2)$
$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$
Signature generation:
Sample $y \leftarrow\$ \mathbb{Z}_p^*$

$$\sigma = \left(Z = (\prod_{i=1}^{2} M_i^{x_i})^y, Y = P^{1/y}, \hat{Y} = \hat{P}^{1/y}\right)$$

More intuition of our MS construction (starting from CL19)

Messages and keys:
$M = (M_1, M_2)$
$\mathsf{sk} = (x_1, x_2)$
$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$
Signature generation:
Sample $y \leftarrow\!\!\$\ \mathbb{Z}_p^*$

$$\sigma = \left(Z = (\prod_{i=1}^{2} M_i^{x_i})^y, Y = P^{1/y}, \hat{Y} = \hat{P}^{1/y} \right)$$

$$e(Z, \hat{Y}) = e(\prod_{1}^{2} M_i^{x_i}, \hat{P})$$

More intuition of our MS construction (starting from CL19)

Messages and keys:
$M = (M_1, M_2)$
$\mathsf{sk} = (x_1, x_2)$
$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{P}^{x_1}, \hat{P}^{x_2})$
Signature generation:
Sample $y \leftarrow\!\!\$\ \mathbb{Z}_p^*$

$$\sigma = \left(Z = (\prod_{i=1}^{2} M_i^{x_i})^y, Y = P^{1/y}, \hat{Y} = \hat{P}^{1/y} \right)$$

Verification:

$$\prod_{i=1}^{2} e(M_i, \hat{X}_i) = e(Z, \hat{Y}) \text{ and } e(Y, \hat{P}) = e(P, \hat{Y})$$

More intuition of our MS construction

Structured CRS:
$b_1, b_2 \leftarrow\$ \mathbb{Z}_p^*$
$\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2}$
Messages and keys:
$M = (M_1, M_2)$
$\mathsf{sk} = (x_1, x_2)$
$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$
Signature generation:
Sample $y \leftarrow\$ \mathbb{Z}_p^*$

$$\sigma = \left(Z = (\prod_{i=1}^{2} M_i^{x_i})^y, Y = P^{1/y}, \hat{Y} = \hat{P}^{1/y} \right)$$

Verification:

$$\prod_{i=1}^{2} e(M_i, \hat{X}_i) \neq e(Z, \hat{Y}) \text{ and } e(Y, \hat{P}) = e(P, \hat{Y})$$

More intuition of our MS construction

Structured CRS:

$b_1, b_2 \leftarrow\!\$\ \mathbb{Z}_p^*$

$(A_1 = P^{b_1}, A_2 = P^{b_2}) \in \mathbb{G}_1$

$\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2}$

Messages and keys:

$M = (M_1, M_2, M_3, M_4) = (P^{m_1}, P^{m_2}, A_1^{m_1}, A_2^{m_2})$

$\mathsf{sk} = (x_1, x_2)$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$

Signature generation:

Sample $y \leftarrow\!\$\ \mathbb{Z}_p^*$

$$\sigma = \left(Z = (\prod_{i=3}^{4} M_i^{x_i})^y, Y = P^{1/y}, \hat{Y} = \hat{P}^{1/y} \right)$$

Verification:

$\prod_{i=1}^{2} e(M_i, \hat{X}_i) = e(Z, \hat{Y})$ and $e(Y, \hat{P}) = e(P, \hat{Y})$

Intuition of our MS construction

Structured CRS:

$a_1, a_2, b_1, b_2 \leftarrow\$ \mathbb{Z}_p^*$

$A_1 = P^{a_1}, A_2 = P^{a_2}, A_3 = P^{a_1 b_1}, A_4 = P^{a_2 b_2}$

$\hat{B}_1 = \hat{P}^{b_1}, \hat{B}_2 = \hat{P}^{b_2}$

Messages and keys:

$M = (M_1, M_2, M_3, M_4) = (A_1^{m_1}, A_2^{m_2}, A_3^{m_1}, A_4^{m_2})$

$\mathsf{sk} = (x_1, x_2)$

$\mathsf{pk} = (\hat{X}_1, \hat{X}_2) = (\hat{B}_1^{x_1}, \hat{B}_2^{x_2})$

Signature generation:

Sample $y \leftarrow\$ \mathbb{Z}_p^*$

$$\sigma = \left(Z = (\prod_{i=3}^{4} M_i^{x_i})^y, Y = P^{1/y}, \hat{Y} = \hat{P}^{1/y} \right)$$

Verification:

$$\prod_{i=1}^{2} e(M_i, \hat{X}_i) = e(Z, \hat{Y}) \text{ and } e(Y, \hat{P}) = e(P, \hat{Y})$$

More challenges (that we had to solve)

1. Ensuring correctness with $\hat{B}_1$ and $\hat{B}_2$
2. Verifying that keys use $\hat{B}_1$ and $\hat{B}_2$
   - Double size of public keys (adding $\hat{X}_3$ and $\hat{X}_4$) and use "verification bases" (V_1, V_2, V_3, V_4) to verify by computing: $e(V_1, \hat{X}_1) = e(V_3, \hat{X}_3) \wedge e(V_2, \hat{X}_2) = e(V_4, \hat{X}_4)$
3. Extending the scheme to multiple levels for DAC
   - Ensure that different levels have correlated structure to retain correctness

Additional Features

1. Use recognizable [CL19] to create revocation tokens:
   - $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}_{\mathsf{GLMPS24}}(\mathsf{pp})$
   - $(\mathsf{sk}_{\mathsf{tok}}, \mathsf{pk}_{\mathsf{tok}}) \leftarrow \mathsf{KeyGen}_{\mathsf{CL19}}(\mathsf{pp})$
   - $\sigma_{\mathsf{tok}} \leftarrow \mathsf{Sign}_{\mathsf{CL19}}(\mathsf{sk}_{\mathsf{tok}}, \mathsf{pk})$
2. Use Schnorr proofs to update $\mathsf{pp} = (\hat{B}_1, \hat{B}_2)$ to $\mathsf{pp}' = (\hat{B}'_1, \hat{B}'_2)$ by exponentiating generators with new scalars (b'_1, b'_2) and proving:
   - $\pi = NIZK[b'_1, b'_2 : \hat{B}'_1 = \hat{B}_1^{b'_1} \wedge \hat{B}'_2 = \hat{B}_2^{b'_2}]$

Summary

1. Mercurial signature scheme with stronger privacy
2. DAC construction with strong privacy from our signatures
3. Revocation of public keys (including intermediate issuers)
4. Serially updating CRS (similiar to [GKM$^+$18])

Future work:

- Remove the structure CRS (trusted setup) entirely

Thank you!

Delegatable Anonymous Credentials From Mercurial Signatures With Stronger Privacy

`https://eprint.iacr.org/2024/1216`

Scott Griffy, Anna Lysyanskaya, Omid Mir, Octavio Pérez Kempner, Daniel Slamanig

BROWN NTT AUSTRIAN INSTITUTE OF TECHNOLOGY Universität der Bundeswehr München

Tolga Acar and Lan Nguyen.
Revocation for delegatable anonymous credentials.
In Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi, editors, *PKC 2011*, volume 6571 of *LNCS*, pages 423–440. Springer, Berlin, Heidelberg, March 2011.

Masayuki Abe, Masaya Nanri, Octavio Perez Kempner, and Mehdi Tibouchi.
Interactive threshold mercurial signatures and applications.
Cryptology ePrint Archive, Paper 2024/625, 2024.
https://eprint.iacr.org/2024/625.

Johannes Blömer and Jan Bobolz.
Delegatable attribute-based anonymous credentials from dynamically malleable signatures.
In Bart Preneel and Frederik Vercauteren, editors, *ACNS 18International Conference on Applied Cryptography and Network Security*, volume 10892 of *LNCS*, pages 221–239. Springer, Cham, July 2018.

Mira Belenkiy, Jan Camenisch, Melissa Chase, Markulf Kohlweiss, Anna Lysyanskaya, and Hovav Shacham.
Randomizable proofs and delegatable anonymous credentials.
In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 108–125. Springer, Berlin, Heidelberg, August 2009.

Michael Backes, Lucjan Hanzlik, Kamil Kluczniak, and Jonas Schneider.
Signatures with flexible public key: Introducing equivalence classes for public keys.
In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 405–434. Springer, Cham, December 2018.

Elizabeth C. Crites and Anna Lysyanskaya.
Delegatable anonymous credentials from mercurial signatures.
In Mitsuru Matsui, editor, *CT-RSA 2019*, volume 11405 of *LNCS*, pages 535–555. Springer, Cham, March 2019.

Elizabeth C. Crites and Anna Lysyanskaya.
Mercurial signatures for variable-length messages.

PoPETs, 2021(4):441–463, October 2021.

Aisling Connolly, Pascal Lafourcade, and Octavio Perez-Kempner.
Improved constructions of anonymous credentials from structure-preserving signatures on equivalence classes.
In Goichiro Hanaoka, Junji Shikata, and Yohei Watanabe, editors, *PKC 2022, Part I*, volume 13177 of *LNCS*, pages 409–438. Springer, Cham, March 2022.

Georg Fuchsbauer, Christian Hanser, and Daniel Slamanig.
Structure-preserving signatures on equivalence classes and constant-size anonymous credentials.
Journal of Cryptology, 32(2):498–546, April 2019.

Jens Groth, Markulf Kohlweiss, Mary Maller, Sarah Meiklejohn, and Ian Miers.
Updatable and universal common reference strings with applications to zk-SNARKs.

In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 698–728. Springer, Cham, August 2018.

Scott Griffy, Anna Lysyanskaya, Omid Mir, Octavio Perez Kempner, and Daniel Slamanig.
Delegatable anonymous credentials from mercurial signatures with stronger privacy.
Cryptology ePrint Archive, Report 2024/1216, 2024.

Omid Mir, Balthazar Bauer, Scott Griffy, Anna Lysyanskaya, and Daniel Slamanig.
Aggregate signatures with versatile randomization and issuer-hiding multi-authority anonymous credentials.
pages 30–44, 2023.

Omid Mir, Daniel Slamanig, Balthazar Bauer, and Rene Mayrhofer.
Practical delegatable anonymous credentials from equivalence class signatures.

In Michelle Mazurek and Micah Sherr, editors, *Proceedings on Privacy Enhancing Technologies (PoPETs)*, volume 2023 of *Proceedings on Privacy Enhancing Technologies*, page 488–513, Germany, July 2023. de Gruyter.
The 23rd Privacy Enhancing Technologies Symposium ; Conference date: 10-07-2023 Through 15-07-2023.

Colin Putman and Keith M. Martin.
Selective delegation of attributes in mercurial signature credentials.
In Elizabeth A. Quaglia, editor, *19th IMA International Conference on Cryptography and Coding*, volume 14421 of *LNCS*, pages 181–196. Springer, Cham, December 2023.