A Lattice-Based Group Signature Scheme with Message-Dependent Opening

Benoît Libert* **Fabrice Mouhartem*** Khoa Nguyen†

*École Normale Supérieure de Lyon, France

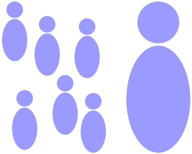
†Nanyang Technological University, Singapore

ACNS, Guildford — June 20th, 2016

Example

Public Transportation

Some user wants to take public transportations.



Example

Public Transportation

Some user wants to take public transportations.

Example

Public Transportation

Some user wants to take public transportations.

- Authenticity & Integrity

Example

Public Transportation

Some user wants to take public transportations.

- Authenticity & Integrity
- Anonymity

Example

Public Transportation

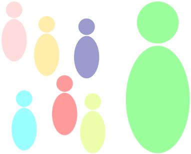
Some user wants to take public transportations.

- Authenticity & Integrity
- Anonymity
- Traceability

Example

Public Transportation

Some user wants to take public transportations.



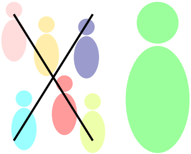
signature →

- Authenticity & Integrity
- Anonymity
- Traceability

Example

Public Transportation

Some user wants to take public transportations.



signature

- Authenticity & Integrity
- Anonymity
- Traceability 

- Avoid opening abuses

Message Dependent Opening

(Sakai et al. Pairing'12)

Group Signature: cryptographic primitive allowing a user to anonymously sign messages on behalf of a group.

Message Dependent Opening

(Sakai et al. Pairing'12)

Group Signature: cryptographic primitive allowing a user to anonymously sign messages on behalf of a group.

An opening authority (OA) can un-anonymize a given signature.

Message Dependent Opening

(Sakai et al. Pairing'12)

Group Signature: cryptographic primitive allowing a user to anonymously sign messages on behalf of a group.

An opening authority (OA) can un-anonymize a given signature.

Problem

Many applications (public transportations, anonymous access control,. . .) require privacy even against authorities.

Message Dependent Opening

(Sakai et al. Pairing'12)

Group Signature: cryptographic primitive allowing a user to anonymously sign messages on behalf of a group.

An opening authority (OA) can un-anonymize a given signature.

Problem

Many applications (public transportations, anonymous access control,. . .) require privacy even against authorities.

$\rightarrow$ **Idea:** Add another authority to restrict the power of the OA.

Message Dependent Opening

(Sakai et al. Pairing'12)

Group Signature: cryptographic primitive allowing a user to anonymously sign messages on behalf of a group.

An opening authority (OA) can un-anonymize a given signature.

Problem

Many applications (public transportations, anonymous access control,...) require privacy even against authorities.

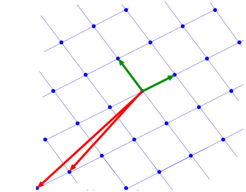
$\rightarrow$ **Idea:** Add another authority to restrict the power of the OA.

The admitter delivers tokens that allow OA to open all signatures for specific messages.

Lattice-Based Cryptography

A **Lattice** is the set of integer linear combination of independent vectors called a basis

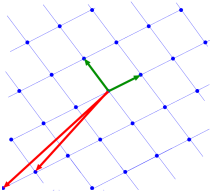
$$\Lambda(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \left\{ \sum_{i \leq n} a_i \mathbf{b}_i \;\middle|\; \forall i, a_i \in \mathbb{Z} \right\}$$



Lattice-Based Cryptography

A **Lattice** is the set of integer linear combination of independent vectors called a basis

$$\Lambda(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \left\{ \sum_{i \leq n} a_i \mathbf{b}_i \;\middle|\; \forall i, a_i \in \mathbb{Z} \right\}$$



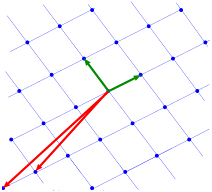
Advantages

Simple, efficient, conjectured resistant to a quantum adversary, links between average-case and wost-case problems, expressive...

Lattice-Based Cryptography

A **Lattice** is the set of integer linear combination of independent vectors called a basis

$$\Lambda(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \left\{ \sum_{i \leq n} a_i \mathbf{b}_i \;\middle|\; \forall i, a_i \in \mathbb{Z} \right\}$$



Advantages

Simple, efficient, conjectured resistant to a quantum adversary, links between average-case and wost-case problems, expressive. . .

Average-case problems: SIS and LWE

Outline

State of the art

- Introduction by Chaum and van Heyst (Eurocrypt'91)
- First scalable solution.
  Ateniese-Camenisch-Joye-Tsudik (Crypto'00)
- GS-MDO. Sakai *et al.* (Pairing'12)
  ↳ Relation with IBE
- Efficient GS-MDO in the ROM. Ohara *et al.* (AsiaCCS'13)
- Scheme in the standard model. Libert-Joye (CT-RSA'14)

State of the art

- Introduction by Chaum and van Heyst (Eurocrypt'91)
- First scalable solution.
  Ateniese-Camenisch-Joye-Tsudik (Crypto'00)
- Lattice-based scheme.
  Gordon-Katz-Vaikuntanathan (Asiacrypt'10)
- GS-MDO. Sakai *et al.* (Pairing'12)
  ↳ Relation with IBE
- Efficient GS-MDO in the ROM. Ohara *et al.* (AsiaCCS'13)
- Scheme in the standard model. Libert-Joye (CT-RSA'14)
- Efficient lattice-based signature (LNW and NZZ PKC'15)

No lattice-based GS-MDO so far

GPV IBE

(Gentry-Peikert-Vaikuntanathan; STOC'08)

Identity Based Encryption: To encrypt $\mathbf{m} \in \{0,1\}^\ell$ under id

Setup Generate $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with a trapdoor $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$

$$\mathsf{mpk} = \mathbf{A}, \qquad \mathsf{msk} = \mathbf{T_A}.$$

GPV IBE

(Gentry-Peikert-Vaikuntanathan; STOC'08)

Identity Based Encryption: To encrypt $\mathbf{m} \in \{0,1\}^\ell$ under id

Setup Generate $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ with a trapdoor $\mathbf{T_A} \in \mathbb{Z}^{m\times m}$

$$\mathsf{mpk} = \mathbf{A}, \qquad \mathsf{msk} = \mathbf{T_A}.$$

Extract Let $\mathbf{G} = \mathcal{H}(\mathsf{id})$.
Use $\mathbf{T_A}$ compute small norm matrix $\mathbf{E} \in \mathbb{Z}^{m\times \ell}$ s.t.

$$\mathbf{A} \cdot \mathbf{E} = \mathbf{G} \bmod q$$

GPV IBE

(Gentry-Peikert-Vaikuntanathan; STOC'08)

Identity Based Encryption: To encrypt $\mathbf{m} \in \{0,1\}^{\ell}$ under id

Setup Generate $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with a trapdoor $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$

$$\mathsf{mpk} = \mathbf{A}, \qquad \mathsf{msk} = \mathbf{T_A}.$$

Extract Let $\mathbf{G} = \mathcal{H}(\mathsf{id})$.

Use $\mathbf{T_A}$ compute small norm matrix $\mathbf{E} \in \mathbb{Z}^{m \times \ell}$ s.t.

$$\mathbf{A}\,\mathbf{E} = \mathbf{G} \bmod q$$

Encrypt Sample $(\mathbf{s}, \mathbf{e}_1, \mathbf{e}_2) \hookleftarrow \chi^n \times \chi^m \times \chi^{\ell}$ and output

$$\mathbf{c} = \left(\mathbf{A}^T \cdot \mathbf{s} + \mathbf{e}_1, \mathbf{G}^T \cdot \mathbf{s} + \mathbf{e}_2 + \lfloor q/2 \rfloor \cdot \mathbf{m}\right).$$

GPV IBE

(Gentry-Peikert-Vaikuntanathan; STOC'08)

Identity Based Encryption: To encrypt $\mathbf{m} \in \{0,1\}^{\ell}$ under id

Setup Generate $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with a trapdoor $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$

$$\mathsf{mpk} = \mathbf{A}, \qquad \mathsf{msk} = \mathbf{T_A}.$$

Extract Let $\mathbf{G} = \mathcal{H}(\mathsf{id})$.
Use $\mathbf{T_A}$ compute small norm matrix $\mathbf{E} \in \mathbb{Z}^{m \times \ell}$ s.t.

$$\mathbf{A} \, \mathbf{E} = \mathbf{G} \bmod q$$

Encrypt Sample $(\mathbf{s}, \mathbf{e}_1, \mathbf{e}_2) \hookleftarrow \chi^n \times \chi^m \times \chi^{\ell}$ and output

$$\mathbf{c} = \left(\mathbf{A}^T \cdot \mathbf{s} + \mathbf{e}_1, \mathbf{G}^T \cdot \mathbf{s} + \mathbf{e}_2 + \lfloor q/2 \rfloor \cdot \mathbf{m} \right).$$

Decrypt $\mathbf{m}' = \boxed{1 \mid 2 \mid 4 \mid \cdots \mid 2^{\ell-1}} \cdot \left\lfloor (\mathbf{c}_2 - \mathbf{E}^T \cdot \mathbf{c}_1) \cdot (q/2) \right\rceil$

Boyen's Signature

(PKC'10)

To sign a message $M = m_1 \cdots m_\ell \in \{0,1\}^\ell$:

KeyGen: Generate matrix $\mathbf{A}$ with trapdoor $\mathbf{T_A}$, random matrices $\mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$.

Boyen's Signature

(PKC'10)

To sign a message $M = m_1 \cdots m_\ell \in \{0,1\}^\ell$:

KeyGen: Generate matrix $\mathbf{A}$ with trapdoor $\mathbf{T}_\mathbf{A}$, random matrices $\mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$.

Sign: Using $\mathbf{T}_\mathbf{A}$, compute short $\mathbf{z} \in \mathbb{Z}^{2m} = \sigma$ s.t.

$$\underbrace{\left[\, \mathbf{A} \;\middle|\; \mathbf{A}_0 + \sum_{i=1}^{\ell} m_i \cdot \mathbf{A}_i \,\right]}_{\mathbf{A}_M} \mathbf{z} = \mathbf{u} \qquad (*)$$

Boyen's Signature

(PKC'10)

To sign a message $M = m_1 \cdots m_\ell \in \{0,1\}^\ell$:

KeyGen: Generate matrix $\mathbf{A}$ with trapdoor $\mathbf{T}_\mathbf{A}$, random matrices $\mathbf{A}_0, \ldots, \mathbf{A}_\ell \in \mathbb{Z}_q^{n\times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$.

Sign: Using $\mathbf{T}_\mathbf{A}$, compute short $\mathbf{z} \in \mathbb{Z}^{2m} = \sigma$ s.t.

$$\underbrace{\left[\,\mathbf{A} \;\middle|\; \mathbf{A}_0 + \sum_{i=1}^{\ell} m_i \cdot \mathbf{A}_i\,\right]}_{\mathbf{A}_M} \mathbf{z} = \mathbf{u} \qquad (*)$$

Verify: Test $\|\mathbf{z}\| \leq \beta$.
Compute $\mathbf{A}_M$ to check relation $(*)$.

Stern's Protocol
(Crypto'93)

Stern's protocol is a ZK proof for Syndrome Decoding Problem.

Stern's Protocol
(Crypto'93)

Stern's protocol is a ZK proof for Syndrome Decoding Problem.

Syndrome Decoding Problem

Given $\mathbf{P} \in \mathbb{Z}_2^{m\times n}$ and $\mathbf{v} \in \mathbb{Z}_2^{n}$, find $\mathbf{x}$ s.t. $\mathrm{w}(x) = w$ and

$$ {}_{m}\overset{n}{\mathbf{P}}\;\mathbf{x} = \mathbf{v} \mod 2 $$

Stern's Protocol
(Crypto'93)

Stern's protocol is a ZK proof for Syndrome Decoding Problem.

Syndrome Decoding Problem

Given $\mathbf{P} \in \mathbb{Z}_2^{m \times n}$ and $\mathbf{v} \in \mathbb{Z}_2^n$, find $\mathbf{x}$ s.t. $\mathrm{w}(x) = w$ and

$$m \; \overset{n}{\boxed{\mathbf{P}}} \; \boxed{\mathbf{x}} = \boxed{\mathbf{v}} \mod 2$$

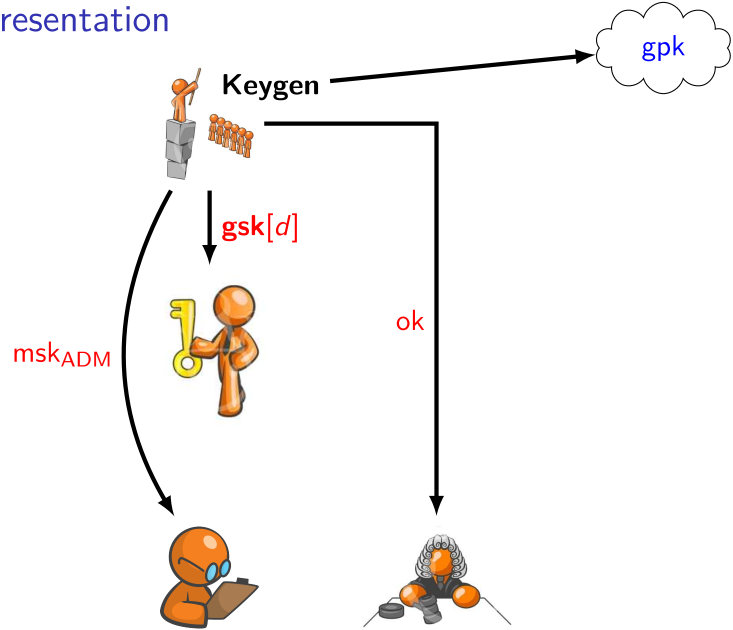
[KTX08]: mod 2 $\rightarrow$ mod q

[LNSW13]: Extend Stern's protocol for SIS and LWE statements

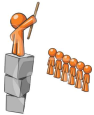
Recent uses of Stern-like protocols in lattice-based crypto:
[LNW15], [LLNW16], [LLNMW16]

Outline

Presentation

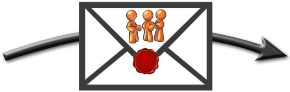


Presentation



Sign

gsk[d]

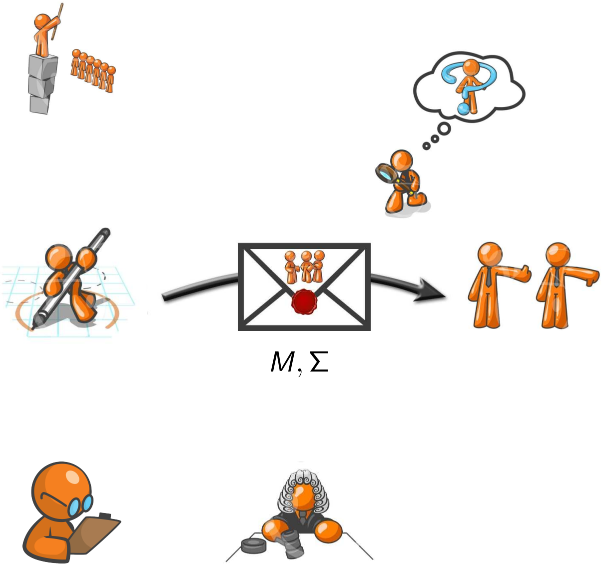


M, Σ

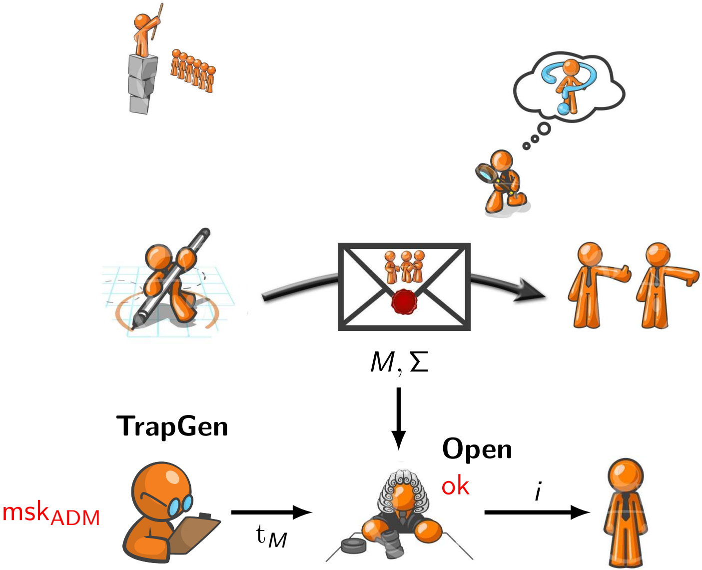
Verify

gpk

Presentation



Presentation



Definition

Group Signature with Message Dependent Opening

It is a tuple of algorithms (**Keygen**, **Sign**, **Verify**, **TrapGen**, **Open**).

Definition

Group Signature with Message Dependent Opening

It is a tuple of algorithms (**Keygen**, **Sign**, **Verify**, **TrapGen**, **Open**).

- **KeyGen:** run by a trusted entity

  `Input:` security parameter λ and group size N
  `Output:` public parameters gpk, opening authority's secret key ok admitter's master secret key $\mathsf{msk}_{\mathsf{ADM}}$,

Definition

Group Signature with Message Dependent Opening

It is a tuple of algorithms (**Keygen**, **Sign**, **Verify**, **TrapGen**, **Open**).

- **Sign** and **Verify** proceed as in standard digital signatures

Definition

Group Signature with Message Dependent Opening

It is a tuple of algorithms (**Keygen**, **Sign**, **Verify**, **TrapGen**, **Open**).

- **TrapGen:**

  `Input:` *Admitter*'s secret $\mathsf{msk}_{\mathsf{ADM}}$ and message M.
  `Output:` token t_M for message M

Definition

Group Signature with Message Dependent Opening

It is a tuple of algorithms (**Keygen**, **Sign**, **Verify**, **TrapGen**, **Open**).

- **TrapGen:**

  `Input:` *Admitter*'s secret $\mathsf{msk}_{\mathsf{ADM}}$ and message M.
  `Output:` token t_M for message M

- **Open:**

  `Input:` **OA**'s secret ok, a token t_M, message M and Σ
  `Output:` identity i or error $\perp$

Generic Construction of Group Signatures

(Bellare-Micciancio-Warinschi; Eurocrypt'03)

E is an encryption scheme, S is a signature scheme.

- **Keygen.** $1^\lambda, 1^N \to \mathbf{gsk}, \mathrm{ok}, \mathrm{gpk}$
  - $(S.vk, S.sk) \leftarrow S.Keygen$
  - $\mathbf{gsk}[d] \leftarrow S.Sign_{S.sk}(d)$ for $d = 1, \ldots, N$
  - $\mathrm{ok} \leftarrow E.sk$
  - $\mathrm{gpk} = (E.pk, S.vk)$

Generic Construction of Group Signatures

(Bellare-Micciancio-Warinschi; Eurocrypt'03)

E is an encryption scheme, S is a signature scheme.

- **Keygen.** $1^\lambda, 1^N \rightarrow \mathbf{gsk}, \mathrm{ok}, \mathrm{gpk}$
  - $(S.vk, S.sk) \leftarrow S.Keygen$
  - $\mathbf{gsk}[d] \leftarrow S.Sign_{S.sk}(d)$ for $d = 1, \ldots, N$
  - $\mathrm{ok} \leftarrow E.sk$
  - $\mathrm{gpk} = (E.pk, S.vk)$
- **Sign:** $d, \mathbf{gsk}[d], M \rightarrow (C, \pi)$
  - $C \leftarrow E.Enc(d)$
  - $\pi \leftarrow$ proof of knowledge of a pair $(d, S.Sign(d))$ w.r.t. $S.vk$ and correctness of the encryption (embed M into the proof)

Generic Construction of Group Signatures

(Bellare-Micciancio-Warinschi; Eurocrypt'03)

E is an encryption scheme, S is a signature scheme.

- **Keygen.** $1^\lambda, 1^N \to \mathbf{gsk}, \mathrm{ok}, \mathrm{gpk}$
  - $(S.vk, S.sk) \leftarrow S.Keygen$
  - $\mathbf{gsk}[d] \leftarrow S.Sign_{S.sk}(d)$ for $d = 1, \ldots, N$
  - $\mathrm{ok} \leftarrow E.sk$
  - $\mathrm{gpk} = (E.pk, S.vk)$
- **Sign:** $d, \mathbf{gsk}[d], M \to (C, \pi)$
  - $C \leftarrow E.Enc(d)$
  - $\pi \leftarrow$ proof of knowledge of a pair $(d, S.Sign(d))$ w.r.t. $S.vk$ and correctness of the encryption (embed M into the proof)
- **Verify:** $(C, \pi) \to \{0, 1\}$
  - Check the proof π

Generic Construction of Group Signatures

(Bellare-Micciancio-Warinschi; Eurocrypt'03)

E is an encryption scheme, S is a signature scheme.

- **Keygen.** $1^\lambda, 1^N \rightarrow \mathbf{gsk}, \mathrm{ok}, \mathrm{gpk}$
  - $(S.vk, S.sk) \leftarrow S.Keygen$
  - $\mathbf{gsk}[d] \leftarrow S.Sign_{S.sk}(d)$ for $d = 1, \ldots, N$
  - $\mathrm{ok} \leftarrow E.sk$
  - $\mathrm{gpk} = (E.pk, S.vk)$
- **Sign:** $d, \mathbf{gsk}[d], M \rightarrow (C, \pi)$
  - $C \leftarrow E.Enc(d)$
  - $\pi \leftarrow$ proof of knowledge of a pair $(d, S.Sign(d))$ w.r.t. $S.vk$ and correctness of the encryption (embed M into the proof)
- **Verify:** $(C, \pi) \rightarrow \{0, 1\}$
  - Check the proof π
- **Open:** $(C, \pi), M, \mathrm{ok} \rightarrow \{1, \ldots, N\} \cup \perp$
  - Decrypt C with $\mathrm{ok} = E.sk$

Outline

Construction

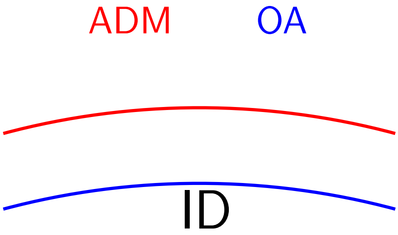
- The scheme is built upon [LNW15] group signature scheme
  - It is a lattice-based scheme
  - Use GPV-IBE + CHK to encrypt identity

Construction

- The scheme is built upon [LNW15] group signature scheme
  - It is a lattice-based scheme
  - Use GPV-IBE + CHK to encrypt identity

Main Idea

Use two-layer encryption

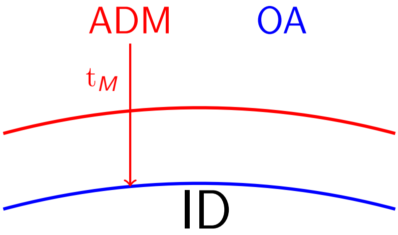


Construction

- The scheme is built upon [LNW15] group signature scheme
  - It is a lattice-based scheme
  - Use GPV-IBE + CHK to encrypt identity

Main Idea

Use two-layer encryption

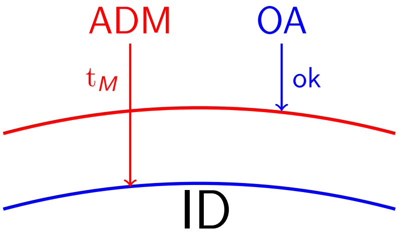


Construction

- The scheme is built upon [LNW15] group signature scheme
  - It is a lattice-based scheme
  - Use GPV-IBE + CHK to encrypt identity

Main Idea

Use two-layer encryption

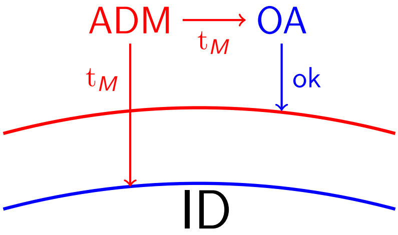


Construction

- The scheme is built upon [LNW15] group signature scheme
  - It is a lattice-based scheme
  - Use GPV-IBE + CHK to encrypt identity

Main Idea

Use two-layer encryption

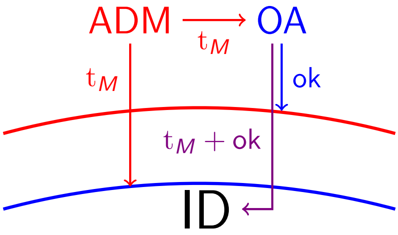


Construction

- The scheme is built upon [LNW15] group signature scheme
  - It is a lattice-based scheme
  - Use GPV-IBE + CHK to encrypt identity

Main Idea

Use two-layer encryption



General Construction

(Sakai et al. Pairing'12)

- **KeyGen:**
  - Generate keys $(\mathsf{pk}, \mathsf{sk})$ for signature and

$$\mathbf{gsk}[d] = \mathsf{Sig.Sign}_{\mathsf{sk}}(d)$$

  - Generate two key pairs for the GPV IBE:

$$(\mathbf{B}, \mathbf{T_B}) \text{ and } (\mathbf{C}, \mathbf{T_C})$$

  - Output

$$\mathsf{ok} = \mathbf{T_B} \qquad \mathsf{msk}_{\mathsf{ADM}} = \mathbf{T_C}$$

$$\mathbf{gsk} \qquad \mathsf{gpk} = (\mathsf{pk}, \mathbf{B}, \mathbf{C}, \mathsf{OTS}, \mathcal{H})$$

General Construction

(Sakai et al. Pairing'12)

- **Sign:** $d, M \mapsto$
  - $C \leftarrow Enc(d)$, using CHK
  - $\widehat{C} \leftarrow IBE.Enc_M(C)$
  - Prove possession of $(d, \mathbf{gsk}[d])$ and that everything is correct

  ↳ $\Sigma = (\widehat{C}, \pi)$

General Construction

(Sakai et al. Pairing'12)

- **Sign:** $d, M \mapsto$
  - $C \leftarrow Enc(d)$, using CHK
  - $\widehat{C} \leftarrow IBE.Enc_M(C)$
  - Prove possession of $(d, \mathbf{gsk}[d])$ and that everything is correct

  ↳ $\Sigma = (\widehat{C}, \pi)$

- **Verify:**
  - Check π + CHK's signature

General Construction

(Sakai et al. Pairing'12)

- **Sign:** $d, M \mapsto$
  - $C \leftarrow Enc(d)$, using CHK
  - $\widehat{C} \leftarrow IBE.Enc_M(C)$
  - Prove possession of $(d, \mathbf{gsk}[d])$ and that everything is correct

  ↳ $\Sigma = (\widehat{C}, \pi)$

- **Verify:**
  - Check π + CHK's signature

- **TrapGen:**
  - $\mathrm{t}_M \leftarrow IBE.Derive(M)$

General Construction

(Sakai et al. Pairing'12)

- **Sign:** $d, M \mapsto$
  - $C \leftarrow Enc(d)$, using CHK
  - $\widehat{C} \leftarrow IBE.Enc_M(C)$
  - Prove possession of $(d, \textbf{gsk}[d])$ and that everything is correct

  ↳ $\Sigma = (\widehat{C}, \pi)$

- **Verify:**
  - Check π + CHK's signature

- **TrapGen:**
  - $\mathrm{t}_M \leftarrow IBE.Derive(M)$

- **Open:**
  - $C \leftarrow IBE.Dec_{\mathrm{t}_M}(\widehat{C})$
  - $d \leftarrow Dec(C)$

Our Realization: Difficulties

Problem

We need to prove double-encryption relations in ZK

Our Realization: Difficulties

Problem

We need to prove double-encryption relations in ZK

Technique. Adapt Stern's protocol as in [LLMNW16]

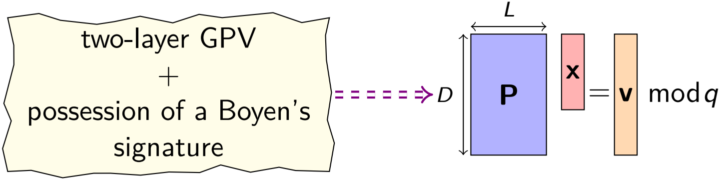
Our Realization: Difficulties

Problem

We need to prove double-encryption relations in ZK

Technique. Adapt Stern's protocol as in [LLMNW16]

- Possible because relations can be transformed into



with $\mathbf{x} \in \{-1, 0, 1\}^L$ and $\mathbf{v} \in \mathbb{Z}_q^D$.

Conclusion

- We provide:
  - A lattice-based group signature scheme with message dependent opening
  - Security in the ROM under standard lattice assumptions
  - A modular technique that extends [LNW15]
- We can easily adapt the technique of [LLMNW16] for dynamic group signatures to get message-dependent openings for dynamic groups

Questions?