Zero-Knowledge Arguments for Matrix-Vector Relations and Lattice-Based Group Encryption

Benoît Libert[1,3] San Ling[2] Fabrice Mouhartem[1] Khoa Nguyen[2] Huaxiong Wang[2]

Journées C2 2018,
11 octobre 2018

[1]École Normale Supérieure de Lyon

[2]Nanyang Technological University

[3]CNRS

Privacy-Preserving Cryptography

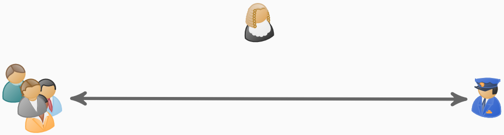
Goal: Provide functionalities while keeping users anonymous

Privacy-Preserving Cryptography

Goal: Provide functionalities while keeping users anonymous

Examples: Group Signatures, Anonymous Credentials, e-Cash, ...

Privacy-Preserving Cryptography



Goal: Provide functionalities while keeping users anonymous

Examples: Group Signatures, Anonymous Credentials, e-Cash, ...

Main ingredients:

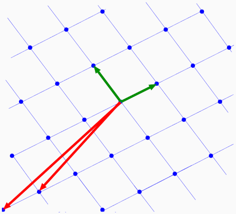
- Digital signatures;
- Public-Key encryption;
- Supporting Zero-Knowledge proofs.

Lattice-Based Cryptography [Ajt96, Reg05]

Lattice

A lattice is a discrete subgroup of $\mathbb{R}^n$. Can be seen as integer linear combinations of a finite set of vectors.

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

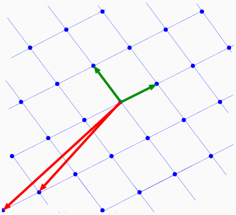


Lattice-Based Cryptography [Ajt96, Reg05]

Lattice

A lattice is a discrete subgroup of $\mathbb{R}^n$. Can be seen as integer linear combinations of a finite set of vectors.

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



Why?

- Simple and asymptotically efficient;
- **Still** conjectured quantum-resistant;
- Connection between average-case and worst-case problems;
- Powerful functionalities (e.g., FHE).

$\rightarrow$ Finding a short non-zero vector in a lattice is hard.

Hardness Assumptions: SIS and LWE [Ajt96, Reg05]

Small Integer Solution

$$\mathbf{x} \, \mathbf{A} = \mathbf{0} \ [q]$$

Goal: Given $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, find $\mathbf{x} \in \mathbb{Z}^m \backslash \{\mathbf{0}\}$ small

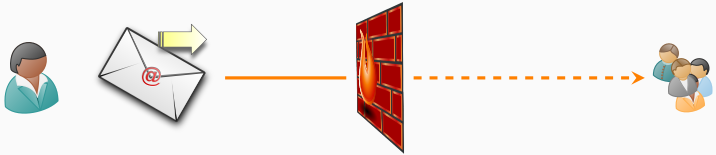
Learning With Errors

$$\left(\underbrace{\mathbf{A}}_{n} \Big\}^{m} , \; \mathbf{A}\,\mathbf{s} + \mathbf{e} \right)$$

$\mathbf{s} \hookleftarrow \mathbb{Z}_q^n$ $\quad$ $\mathbf{e}$ small error

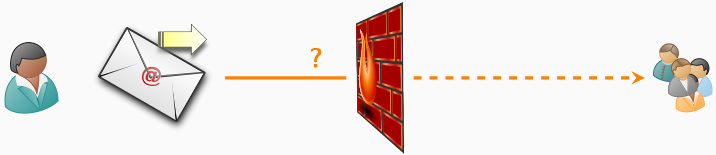
Goal: Given $\left(\mathbf{A}, \mathbf{A}\,\mathbf{s} + \mathbf{e}\right)$, find $\mathbf{s} \in \mathbb{Z}_q^n$

Firewall Filtering [LYJP14]



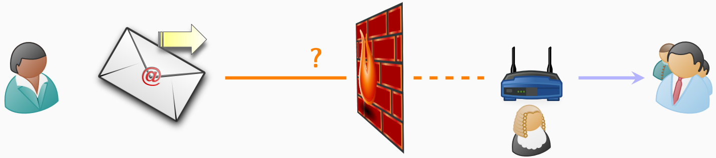
- A user wants to send a message to a group behind a firewall

Firewall Filtering [LYJP14]



- A user wants to send a message to a group behind a firewall
- The recipient of the message can be a sensitive information

Firewall Filtering [LYJP14]



- A user wants to send a message to a group behind a firewall
- The recipient of the message can be a sensitive information
- The router can lift anonymity to route messages

Group Encryption [KTY07]

Group encryption allows encrypting while proving that:

1. The ciphertext is well-formed and intended for some registered group member who will be able to decrypt;
2. The opening authority will be able identify the receiver if necessary;
3. The plaintext satisfies certain properties.

Group Encryption [KTY07]

Group encryption allows encrypting while proving that:

1. The ciphertext is well-formed and intended for some registered group member who will be able to decrypt;
2. The opening authority will be able identify the receiver if necessary;
3. The plaintext satisfies certain properties.

Possible applications

- Firewall filtering
- Anonymous trusted third parties
- Cloud storage services
- Hierarchical group signatures [TW05]

Group Encryption vs Group Signatures

Encryption analogue of group signatures.

Group Encryption vs Group Signatures

Encryption analogue of group signatures.

Group signatures [CvH91]: Group users can anonymously sign messages on behalf of the whole group.

Group encryption [KTY07]: Sender can encrypt a message to an anonymous group member.

Group Encryption vs Group Signatures

Encryption analogue of group signatures.

Group signatures [CvH91]: Group users can anonymously sign messages on behalf of the whole group.

Group encryption [KTY07]: Sender can encrypt a message to an anonymous group member.

Accountability

Group members are kept accountable for their actions: an opening authority can un-anonymize the signatures/ciphertexts if necessary. needs arise.

History

2007 Introduction of group encryption by Kiayias, Tsiounis and Yung

- Modular design from anonymous PKE, signatures and interactive ZK proofs
- Instantiation using number-theoretic assumptions

History

2007 Introduction of group encryption by Kiayias, Tsiounis and Yung

- Modular design from anonymous PKE, signatures and interactive ZK proofs
- Instantiation using number-theoretic assumptions

2009 Non-interactive GE in the standard model from pairings by Cathalo, Libert and Yung

History

2007 Introduction of group encryption by Kiayias, Tsiounis and Yung

- Modular design from anonymous PKE, signatures and interactive ZK proofs
- Instantiation using number-theoretic assumptions

2009 Non-interactive GE in the standard model from pairings by Cathalo, Libert and Yung

2013 Various improvements from El Aimani and Joye

History

2007 Introduction of group encryption by Kiayias, Tsiounis and Yung

- Modular design from anonymous PKE, signatures and interactive ZK proofs
- Instantiation using number-theoretic assumptions

2009 Non-interactive GE in the standard model from pairings by Cathalo, Libert and Yung

2013 Various improvements from El Aimani and Joye

2014 Refined traceability mechanism from Libert, Yung, Peters and Joye.

History

2007 Introduction of group encryption by Kiayias, Tsiounis and Yung

- Modular design from anonymous PKE, signatures and interactive ZK proofs
- Instantiation using number-theoretic assumptions

2009 Non-interactive GE in the standard model from pairings by Cathalo, Libert and Yung

2013 Various improvements from El Aimani and Joye

2014 Refined traceability mechanism from Libert, Yung, Peters and Joye.

✗ All existing realizations of GE rely on number-theoretic assumptions

? Construction from other assumptions, e.g., lattice-based?

Outline

Introduction

Toward Realizing Group Encryption

Zero-Knowledge Arguments for Group Encryption

In the Lattice-Based World

Remark

Group signatures and group encryption rely on the same building blocks

- Digital signatures;
- Public-Key encryption;
- Supporting Zero-Knowledge proofs.

Remark

Group signatures and group encryption rely on the same building blocks

- Digital signatures;
- Public-Key encryption;
- Supporting Zero-Knowledge proofs.

However, there are several realizations of group signatures:
[GKV10, CNR12, LLLS13, NNZ15, LNW15, LLNW16, LNM16, LLMN16]

In the Lattice-Based World

Remark

Group signatures and group encryption rely on the same building blocks

- Digital signatures;
- Public-Key encryption;
- Supporting Zero-Knowledge proofs.

However, there are several realizations of group signatures: [GKV10, CNR12, LLLS13, NNZ15, LNW15, LLNW16, LNM16, LLMN16]

? What is the main difficulty?

Zero-Knowledge Proofs for Lattices

Two main proof systems in lattice-based cryptography:

Schnorr-like [Sch89]: On Ring-LWE [Lyu08], concise but not expressive.
Algebraic

Stern-like [Ste93]: On LWE, heavy but expressive.
Combinatorial

Zero-Knowledge Proofs for Lattices

Two main proof systems in lattice-based cryptography:

Schnorr-like [Sch89]: On Ring-LWE [Lyu08], concise but not expressive.
Algebraic

Stern-like [Ste93]: On LWE, heavy but expressive.
Combinatorial

Both deal with "linear relations", i.e., of the form

$$\mathbf{X} \cdot \mathbf{s} = \mathbf{y} \bmod q$$

Zero-Knowledge Proofs for Lattices

Two main proof systems in lattice-based cryptography:

Schnorr-like [Sch89]: On Ring-LWE [Lyu08], concise but not expressive.
Algebraic

Stern-like [Ste93]: On LWE, heavy but expressive.
Combinatorial

Both deal with "linear relations", i.e., of the form

$$\mathbf{X} \cdot \mathbf{s} = \mathbf{y} \bmod q$$

Examples: (I)**SIS** and **LWE** relations are linear

The Case of Group Signatures

A modular design for GS [BMW03]: **sign-then-encrypt-then-prove**

- Each user has a signature σ on its identity *id* issued by the group manager (GM)

The Case of Group Signatures

A modular design for GS [BMW03]: **sign-then-encrypt-then-prove**

- Each user has a signature σ on its identity *id* issued by the group manager (GM)
- To generate a signature, the user encrypts *id* to $\mathbf{c}$ under opening authority (OA) public key

The Case of Group Signatures

A modular design for GS [BMW03]: **sign-then-encrypt-then-prove**

- Each user has a signature σ on its identity *id* issued by the group manager (GM)
- To generate a signature, the user encrypts *id* to **c** under opening authority (OA) public key
- Then, user proves that:
  1. He has a secret valid pair (id, σ), w.r.t. $\mathsf{vk}_{\mathrm{GM}}$
  2. **c** is a valid encryption of *id*, w.r.t. $\mathsf{pk}_{\mathrm{OA}}$

The Case of Group Signatures

A modular design for GS [BMW03]: **sign-then-encrypt-then-prove**

- Each user has a signature σ on its identity *id* issued by the group manager (GM)
- To generate a signature, the user encrypts *id* to $\mathbf{c}$ under opening authority (OA) public key
- Then, user proves that:
  1. He has a secret valid pair (id, σ), w.r.t. $\mathsf{vk}_{\mathrm{GM}}$ **ISIS**
  2. $\mathbf{c}$ is a valid encryption of *id*, w.r.t. $\mathsf{pk}_{\mathrm{OA}}$ **LWE**

✓ Known techniques allow realizing the ZK proofs

Main Difficulty in Group Encryption

A modular design [KTY07]:

- Each member has a key pair $(\mathsf{pk}, \mathsf{sk})$ for an anonymous encryption scheme
- GM signs each pk and publishes (pk, σ)

Main Difficulty in Group Encryption

A modular design [KTY07]:

- Each member has a key pair $(\mathsf{pk}, \mathsf{sk})$ for an anonymous encryption scheme
- GM signs each pk and publishes (pk, σ)
- Sender uses pk to encrypt a message μ satisfying $\mathcal{R}$, obtains $\mathbf{c}$
- Sender also encrypts pk under $\mathsf{pk}_{\mathrm{OA}}$, obtains $\mathbf{c}_{\mathrm{OA}}$

Main Difficulty in Group Encryption

A modular design [KTY07]:

- Each member has a key pair $(\mathsf{pk}, \mathsf{sk})$ for an anonymous encryption scheme
- GM signs each pk and publishes (pk, σ)
- Sender uses pk to encrypt a message μ satisfying $\mathcal{R}$, obtains $\mathbf{c}$
- Sender also encrypts pk under $\mathsf{pk}_{\mathrm{OA}}$, obtains $\mathbf{c}_{\mathrm{OA}}$
- Then, sender proves that:
  1. $\mathbf{c}$ is a correct encryption of some message μ under some pk
  2. Sender knows a valid signature σ on pk, w.r.t. $\mathsf{vk}_{\mathrm{GM}}$; $\mathbf{c}_{\mathrm{OA}}$ is a correct encryption of pk, w.r.t. $\mathsf{pk}_{\mathrm{OA}}$; μ satisfies $\mathcal{R}$.

Main Difficulty in Group Encryption

A modular design [KTY07]:

- Each member has a key pair $(\mathsf{pk}, \mathsf{sk})$ for an anonymous encryption scheme
- GM signs each pk and publishes (pk, σ)
- Sender uses pk to encrypt a message μ satisfying $\mathcal{R}$, obtains $\mathbf{c}$
- Sender also encrypts pk under $\mathsf{pk}_{\mathrm{OA}}$, obtains $\mathbf{c}_{\mathrm{OA}}$
- Then, sender proves that:
  1. **$\mathbf{c}$ is a correct encryption of some message μ under some pk**
  2. Sender knows a valid signature σ on pk, w.r.t. $\mathsf{vk}_{\mathrm{GM}}$; $\mathbf{c}_{\mathrm{OA}}$ is a correct encryption of pk, w.r.t. $\mathsf{pk}_{\mathrm{OA}}$; μ satisfies $\mathcal{R}$.

We have to handle relations with **hidden-but-certified** matrix:

$$\mathbf{X} \cdot \mathbf{s} + \mathbf{e} = \mathbf{b} \bmod q$$

We call this "quadratic relations".

Zero-Knowledge Arguments for Group Encryption

Stern's Protocol [Ste93]

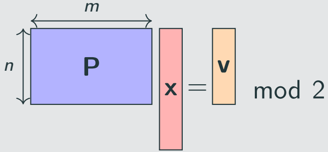
Stern's protocol is a ZK proof for Syndrome Decoding Problem.

Stern's Protocol [Ste93]

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

Syndrome Decoding Problem

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

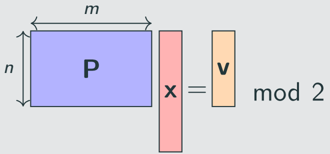


Stern's Protocol [Ste93]

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

Syndrome Decoding Problem

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



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

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

Recent uses of Stern-like protocols in lattice-based crypto:
[LNW15, LLNW16, LMN16, LLNMW16, LLNMW17, LLNW17]

Stern's Ideas

Syndrome Decoding Problem

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

$$\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod 2$$

Stern's Ideas

Syndrome Decoding Problem

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

$$\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod 2$$

1. **Permuting:** Proving the witness constraint using random permutation
   - Send the verifier $\pi(\mathbf{x})$
   - $\mathbf{x}$ is binary of hamming weight w *iff* $\pi(\mathbf{x})$ does

   ☞ The randomness of π protects the value of $\mathbf{x}$

Stern's Ideas

Syndrome Decoding Problem

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

$$\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod 2$$

1. **Permuting:** Proving the witness constraint using random permutation
   - Send the verifier $\pi(\mathbf{x})$
   - $\mathbf{x}$ is binary of hamming weight w *iff* $\pi(\mathbf{x})$ does
   - ☞ The randomness of π protects the value of $\mathbf{x}$
2. **Masking:** Proving linear equation using a random mask $\mathbf{r}$
   - Send the verifier $\mathbf{y} = \mathbf{x} + \mathbf{r}$ and show that $\mathbf{P} \cdot \mathbf{y} = \mathbf{v} + \mathbf{P} \cdot \mathbf{r}$

Stern's Ideas

Syndrome Decoding Problem

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

$$\mathbf{P} \cdot \mathbf{x} = \mathbf{v} \bmod 2$$

1. **Permuting:** Proving the witness constraint using random permutation
   - Send the verifier $\pi(\mathbf{x})$
   - $\mathbf{x}$ is binary of hamming weight w *iff* $\pi(\mathbf{x})$ does
   - ☞ The randomness of π protects the value of $\mathbf{x}$
2. **Masking:** Proving linear equation using a random mask $\mathbf{r}$
   - Send the verifier $\mathbf{y} = \mathbf{x} + \mathbf{r}$ and show that $\mathbf{P} \cdot \mathbf{y} = \mathbf{v} + \mathbf{P} \cdot \mathbf{r}$

Idea: We will

2.1 Pre-process the given quadratic relation

2.2 Exploit permuting to prove the relation

Dealing with Quadratic Relations: First Step

Goal: Express $\mathbf{X} \cdot \mathbf{s}$ as $\mathbf{Q} \cdot \mathbf{z}$

Dealing with Quadratic Relations: First Step

Goal: Express $\mathbf{X} \cdot \mathbf{s}$ as $\mathbf{Q} \cdot \mathbf{z}$

Idea: Binary decomposition

Dealing with Quadratic Relations: First Step

Goal: Express $\mathbf{X} \cdot \mathbf{s}$ as $\mathbf{Q} \cdot \mathbf{z}$

Idea: Binary decomposition

- $\mathbf{z}$ is still **quadratic**: each z_i is a product of a bit from $\mathbf{X}$ with a bit from $\mathbf{s}$
- The component bits additionally satisfy other relations

Dealing with Quadratic Relations: First Step

Goal: Express $\mathbf{X} \cdot \mathbf{s}$ as $\mathbf{Q} \cdot \mathbf{z}$

Idea: Binary decomposition

- $\mathbf{z}$ is still **quadratic**: each z_i is a product of a bit from $\mathbf{X}$ with a bit from $\mathbf{s}$
- The component bits additionally satisfy other relations

Goal

Prove that a secret bit z is of form $z = c_1 \cdot c_2$, while preserving the possibility to show that the component bits c_1 and c_2 satisfy other equations.

Dealing with Quadratic Relations: Second Step

Goal

Prove that a secret bit z is of form $z = c_1 \cdot c_2$, while preserving the possibility to show that the component bits c_1 and c_2 satisfy other equations.

Dealing with Quadratic Relations: Second Step

Goal

Prove that a secret bit z is of form $z = c_1 \cdot c_2$, while preserving the possibility to show that the component bits c_1 and c_2 satisfy other equations.

Idea: Two-bit based permutations

Define bit-extension $\mathrm{ext}(\cdot, \cdot)$ and permutation $T_{b_1, b_2}(\cdot)$ s.t.

$$\mathbf{v} = \mathrm{ext}(c_1, c_2) \iff T_{b_1, b_2}(\mathbf{v}) = \mathrm{ext}(c_1 \oplus b_1, c_2 \oplus b_2)$$

Dealing with Quadratic Relations: Second Step

Goal

Prove that a secret bit z is of form $z = c_1 \cdot c_2$, while preserving the possibility to show that the component bits c_1 and c_2 satisfy other equations.

Idea: Two-bit based permutations

Define bit-extension $\mathrm{ext}(\cdot,\cdot)$ and permutation $T_{b_1,b_2}(\cdot)$ s.t.

$$\mathbf{v} = \mathrm{ext}(c_1, c_2) \iff T_{b_1,b_2}(\mathbf{v}) = \mathrm{ext}(c_1 \oplus b_1, c_2 \oplus b_2)$$

1. Extend $z = c_1 \cdot c_2$ to $\mathbf{v} = \mathrm{ext}(c_1, c_2)$

Dealing with Quadratic Relations: Second Step

Goal

Prove that a secret bit z is of form $z = c_1 \cdot c_2$, while preserving the possibility to show that the component bits c_1 and c_2 satisfy other equations.

Idea: Two-bit based permutations

Define bit-extension $\mathrm{ext}(\cdot,\cdot)$ and permutation $T_{b_1,b_2}(\cdot)$ s.t.

$$\mathbf{v} = \mathrm{ext}(c_1, c_2) \iff T_{b_1,b_2}(\mathbf{v}) = \mathrm{ext}(c_1 \oplus b_1, c_2 \oplus b_2)$$

1. Extend $z = c_1 \cdot c_2$ to $\mathbf{v} = \mathrm{ext}(c_1, c_2)$
2. Permute $\mathbf{v}$ with random bits b_1, b_2, and give the verifier the permuted vector $T_{b_1,b_2}(\mathbf{v})$

Dealing with Quadratic Relations: Second Step

Goal

Prove that a secret bit z is of form $z = c_1 \cdot c_2$, while preserving the possibility to show that the component bits c_1 and c_2 satisfy other equations.

Idea: Two-bit based permutations

Define bit-extension $\mathrm{ext}(\cdot,\cdot)$ and permutation $T_{b_1,b_2}(\cdot)$ s.t.

$$\mathbf{v} = \mathrm{ext}(c_1, c_2) \iff T_{b_1,b_2}(\mathbf{v}) = \mathrm{ext}(c_1 \oplus b_1, c_2 \oplus b_2)$$

1. Extend $z = c_1 \cdot c_2$ to $\mathbf{v} = \mathrm{ext}(c_1, c_2)$
2. Permute $\mathbf{v}$ with random bits b_1, b_2, and give the verifier the permuted vector $T_{b_1,b_2}(\mathbf{v})$
3. To prove that the **same** bits c_1, c_2 appear in other equations: set up similar mechanisms at their other appearances, and use the **same** b_1, b_2.

Group Encryption: Putting Everything Together

Ingredients

- Anonymous encryption — [ABB10] IBE + [CHK04] transform
- Signature scheme — [LLMNW16]
- Supporting ZK proofs — [This Talk]

[KTY07]'s modular construction $\Rightarrow$ first group encryption construction from (classical) lattice assumptions proven secure in the standard model

Conclusion

- Our results:
  - Zero-Knowledge arguments for "quadratic relations":

$$\mathbf{X} \cdot \mathbf{s} + \mathbf{e} = \mathbf{b} \bmod q.$$

  $\rightarrow$ Building block for cryptography: may be of independent interest
  - First lattice-based group encryption scheme

Questions?