Practical “Signatures with Efficient Protocols” from Simple Assumptions

Benoît Libert$^1$  Fabrice Mouhartem$^1$
Thomas Peters$^2$  Moti Yung$^3$

$^1$École Normale Supérieure de Lyon, France
$^2$Université Catholique de Louvain, Belgium
$^3$Snapchat & Columbia University, USA

AsiaCCS, Xi’an – June 2nd 2016
Privacy-Preserving Cryptography

**Important Goal:** Anonymous authentication.
Privacy-Preserving Cryptography

**Important Goal:** Anonymous authentication.

e.g. e-voting, e-cash, group signatures, anonymous credentials...
Privacy-Preserving Cryptography

**Important Goal:** Anonymous authentication.

e.g. e-voting, e-cash, group signatures, anonymous credentials...

Requires

- A signature scheme
- Zero-knowledge (ZK) proof
Privacy-Preserving Cryptography

**Important Goal:** Anonymous authentication.

e.g. e-voting, e-cash, group signatures, anonymous credentials...

Requires

- A signature scheme
- Zero-knowledge (ZK) proof compatible with this signature
Privacy-Preserving Cryptography

**Important Goal:** Anonymous authentication.

e.g. e-voting, e-cash, **group signatures**, anonymous credentials...

Requires

- A signature scheme
- Zero-knowledge (ZK) proof compatible with this signature
Digital Signatures

Signature Scheme

Signer

Message

Signature

Verifier

Sign

Verify

Guarantees authenticity and integrity
Digital Signatures

Guarantees \textit{authenticity} and \textit{integrity}.
Group Signatures

Bob wants to take public transportations.
Group Signatures

Bob wants to take public transportations.
Group Signatures

Bob wants to take public transportations.

- Authenticity & Integrity
Group Signatures

Bob wants to take public transportations.

- Authenticity & Integrity
- Anonymity
Group Signatures

Bob wants to take public transportations.

- Authenticity & Integrity
- Anonymity
- Dynamicity

![Image of a bus ticket machine with a signature]

Join
Group Signatures

Bob wants to take public transportations.

- **Authenticity & Integrity**
- **Anonymity**
- **Dynamicity**
- **Traceability**

![Image of a transaction with a signature]
Commitments

Digital equivalent of a sealed box.
Commitments

Digital equivalent of a sealed box.

Properties

Commitments provide

- **Binding** property: once sealed, a value cannot be changed
Commitments

Digital equivalent of a sealed box.

Properties

Commitments provide

- **Binding** property: once sealed, a value cannot be changed
- **Hiding** property: nobody can tell what is inside the box without the key
Signature with Efficient Protocols

Signature Scheme with Efficient Protocols
(Camenisch-Lysyanskya, SCN’02)

- Signer
  - Sign
  - Message
- Verifier
  - Verify
  - Message
- Signature

- Signature
Signature with Efficient Protocols

Signature Scheme with Efficient Protocols (Camenisch-Lysyanskya, SCN’02)

- Signature
- Sign committed values
Signature with Efficient Protocols

Signature Scheme with Efficient Protocols (Camenisch-Lysyanskya, SCN’02)

- **Signer**
  - Sign
  - Message

- **Verifier**
  - Verify
  - Message
  - Open
  - Signature
  - ZKPoK
  - PoK

- **Steps**
  - Signature
  - Sign committed values
  - Proof of Knowledge (PoK) of (Message; Signature)
Pairing-Based Cryptography

Pairing

\[ e : G \times \hat{G} \rightarrow G_T \]

s.t. for \( g \in G, \hat{g} \in \hat{G} \)

\[ e(g^a, \hat{g}^b) = e(g, \hat{g})^{ab} \]
Pairing-Based Cryptography

Pairing

\[ e : \mathbb{G} \times \hat{\mathbb{G}} \longrightarrow \mathbb{G}_T \]

s.t. for \( g \in \mathbb{G}, \hat{g} \in \hat{\mathbb{G}} \)

\[ e(g^a, \hat{g}^b) = e(g, \hat{g})^{ab} \]

Hardness assumptions:

- **SXDH**: DDH holds in \( \mathbb{G} \) and \( \hat{\mathbb{G}} \) with \( \mathbb{G} \neq \hat{\mathbb{G}} \)
  - **DDH**: given \((g, g^a, g^b, g^c)\), tells whether \( c = a \cdot b \) or \( c \in_R \mathbb{Z}_p \)

- **SDL**: given \((g, \hat{g}, g^a, \hat{g}^a)\), compute \( a \in \mathbb{Z}_p \) with \( p = |\mathbb{G}| \)

→ Well studied, fixed-size assumptions.
Standard Assumptions

Standard assumptions are static and non-interactive assumptions.
Standard Assumptions

Standard assumptions are **static** and **non-interactive** assumptions.

**vs**

**Static** (or fixed-size) assumptions

- **DDH**: $(g^a, g^b, g^c) \mapsto$ tells if $c = ab$ or $c$ random.

**q-type** assumptions

- **q-DH-Inversion**: $(g^{x^0}, g^{x^2}, \ldots, g^{x^q}) \mapsto g^{x^{q+1}}$
Standard Assumptions

Standard assumptions are **static** and **non-interactive** assumptions.

vs

**Static** (or fixed-size) assumptions

- **DDH:** \((g^a, g^b, g^c) \mapsto\) tells if \(c = ab\) or \(c\) random.

**q-type** assumptions

- **q-DH-Inversion:**
  \[
  (g^x, g^{x^2}, \ldots, g^{x^q}) \mapsto g^{x^{q+1}}
  \]

\(q\) usually represents the number of adversarial queries.
Large values of \(q\) may lead to attacks (Cheon (Eurocrypt’06))
Standard Assumptions

Standard assumptions are **static** and **non-interactive** assumptions.

**Static** (or fixed-size) assumptions
- **DDH**:
  
  
  \[(g^a, g^b, g^c) \mapsto \text{tells if } c = ab \text{ or } c \text{ random.}\]

**q-type** assumptions
- **q-DH-Inversion**:

  
  \[(g^x, g^{x^2}, \ldots, g^{x^q}) \mapsto g^{x+1}

  \]

  
  
  \[q\] usually represents the number of adversarial queries.

  
  Large values of \(q\) may lead to attacks (Cheon (Eurocrypt’06))

**Non-interactive** assumptions
- **DL**:

  
  \[g^a \in G \mapsto a \in \mathbb{Z}_p\]

**Interactive** assumptions
- **One-more-DL**:

  
  Given oracle access to \((g^{a_i} \mapsto a_i)\), finds \((b_i)_i\) given \((g^{b_i})_i\)
Outline

Introduction

Our Signature Scheme

Dynamic Group Signature

Conclusion
Signature

Signature Scheme:

- Constant-size
Signature

Signature Scheme:

- Constant-size 4 group elements
Signature

Signature Scheme:

- Constant-size 4 group elements
- Multi-block
Signature

Signature Scheme:

- Constant-size 4 group elements
- Multi-block
- Randomizable
Signature with Efficient Protocols

Signature Scheme:

- Constant-size \(4\) group elements
- Multi-block
- Randomizable

Compatible with Efficient Protocols
Signature with Efficient Protocols

Signature Scheme:

- Constant-size 4 group elements
- Multi-block
- Randomizable

Compatible with Efficient Protocols

- Sign committed messages
Signature with Efficient Protocols

Signature Scheme:

- Constant-size 4 group elements
- Multi-block
- Randomizable

Compatible with Efficient Protocols

- Sign committed messages
- ZK-Prove the knowledge of a valid message-signature pair
Linear Subspace Membership

We say that \( \vec{v} \in \text{Span}(\text{Rows}(M)) \) if there exists \( \vec{w} \in \mathbb{Z}_p^t \) satisfying

\[
\vec{v} = g^{\vec{w} \cdot M} \in \mathbb{G}^n
\]
Linear Subspace Membership

We say that \( \vec{v} \in \text{Span}(\text{Rows}(M)) \) if there exists \( \vec{w} \in \mathbb{Z}_p^t \) satisfying

\[
\vec{v} = g^{\vec{w} \cdot M} \in \mathbb{G}^n
\]

First Quasi-Adaptive Non-Interactive-ZK (QA-NIZK) proofs were proposed by Libert-Peters-Joye-Yung (Eurocrypt'14)
Linear Subspace Membership

We say that \( \vec{v} \in \text{Span}(\text{Rows}(M)) \) if there exists \( \vec{w} \in \mathbb{Z}_p^t \) satisfying

\[
\vec{v} = g^{\vec{w} \cdot M} \in G^n
\]

First **Quasi-Adaptive Non-Interactive-ZK** (QA-NIZK) proofs was proposed by Libert-Peters-Joye-Yung (Eurocrypt’14)

**Quasi-Adaptive** (Jutla-Roy (Asiacrypt’13)) means that the common reference string (crs) may depend on the language (here the matrix \( M \))
Proof System for Linear Subspace Membership

Use of Kiltz-Wee Quasi-Adaptive Non-Interactive ZK proofs (QA-NIZK) to prove linear subspace membership.
Proof System for Linear Subspace Membership

Use of Kiltz-Wee Quasi-Adaptive Non-Interactive ZK proofs (QA-NIZK) to prove linear subspace membership.

Kiltz-Wee QA-NIZK (Eurocrypt’15)

Given \( M = (\vec{M}_1, \ldots, \vec{M}_t)^T \in \mathbb{G}^{t \times n} \), \( \pi \in \mathbb{G} \) prove that \( \vec{v} \in \text{Span}(\text{Rows}(M)) \) for some witness \( \vec{w} \).

Which is constant-size.
Our Signature Scheme

\[ pk = (cp, crs, \vec{v} = (v_1, \ldots, v_\ell, w) \in_R \mathbb{G}^{\ell+1}, \Omega = h^\omega) \quad sk = \omega \]

\[
M = \begin{pmatrix}
g & 1 & \cdots & 1 & 1 & 1 & 1 & \cdots & 1 & h \\
v_1 & g & 0 & \cdots & 0 & h & 0 & \cdots & 0 & 1 \\
\vdots & 0 & \ddots & 0 & \vdots & 0 & \ddots & 0 & \ddots & \vdots \\
v_\ell & 0 & \cdots & g & 0 & 0 & \cdots & h & 0 & 1 \\
w & 0 & \cdots & 0 & g & 0 & \cdots & 0 & h & 1
\end{pmatrix}
\]
Our Signature Scheme

\[ pk = (cp, crs, \vec{v} = (v_1, \ldots, v_\ell, w) \in \mathbb{G}^{\ell+1}, \Omega = h^\omega) \quad sk = \omega \]

\[
M = \begin{pmatrix}
g & 1 & \cdots & 1 & 1 & 1 & 1 & \cdots & 1 & h \\
v_1 & g & 0 & \cdots & 0 & h & 0 & \cdots & 0 & 1 \\
\vdots & 0 & \ddots & 0 & \vdots & 0 & \ddots & 0 & \vdots & \vdots \\
v_\ell & 0 & \cdots & g & 0 & 0 & \cdots & h & 0 & 1 \\
w & 0 & \cdots & 0 & g & 0 & \cdots & 0 & h & 1
\end{pmatrix}
\]

\[
\begin{align*}
\sigma_1 &= g^{\omega}(v_1^{m_1} \cdots v_\ell^{m_\ell} w)^s \\
\sigma_2 &= g^s \\
\sigma_3 &= h^s
\end{align*}
\]

+ \( \pi \): ZK proof that

\[
(\sigma_1, \sigma_1^{m_1}, \ldots, \sigma_2^{m_\ell}, \sigma_2, \sigma_3^{m_1}, \ldots, \sigma_3^{m_\ell}, \sigma_3, \Omega) \in \text{Span}(\text{Rows}(M))
\]
Our Signature Scheme

\[ pk = (cp, crs, \vec{v} = (v_1, \ldots, v_\ell, w) \in_R \mathbb{G}^{\ell+1}, \Omega = h^\omega) \quad sk = \omega \]

\[ M = \begin{pmatrix}
  g & 1 & \cdots & 1 & 1 & 1 & 1 & \cdots & 1 & h \\
 v_1 & g & 0 & \cdots & 0 & h & 0 & \cdots & 0 & 1 \\
 \vdots & 0 & \ddots & 0 & \vdots & 0 & \ddots & 0 & \vdots & \vdots \\
 v_\ell & 0 & \cdots & g & 0 & 0 & \cdots & h & 0 & 1 \\
 w & 0 & \cdots & 0 & g & 0 & \cdots & 0 & h & 1 \\
\end{pmatrix} \]

\[ \sigma_1 = g^\omega (v_1^{m_1} \cdots v_\ell^{m_\ell} w)^s \quad \sigma_2 = g^s \quad \sigma_3 = h^s \]

\[ \cdot (v_1^{m_1} \cdots v_\ell^{m_\ell} w)^{s'} \cdot g^{s'} \cdot h^{s'} \]

+ \pi: \text{ZK proof that} \quad \sigma_1, \sigma_2^{m_1}, \ldots, \sigma_2^{m_\ell}, \sigma_3^{m_1}, \ldots, \sigma_3^{m_\ell}, \sigma_3, \Omega \in \text{Span}(\text{Rows}(M)) \]
Properties

Security

The signature scheme is secure under chosen-message attack under SXDH.
Properties

Security

The signature scheme is secure under \textit{chosen-message attack} under \textit{SXDH}.

Efficient protocols

There exist \textit{practical} protocols for:

- signing committed messages
## Security

The signature scheme is secure under **chosen-message attack** under **SXDH**.

## Efficient protocols

There exist **practical** protocols for:

- signing committed messages
- proving knowledge of a valid message-signature pair
Outline

Introduction

Our Signature Scheme

Dynamic Group Signature

Conclusion
Definition

Dynamic Group Signature

It is a tuple of algorithms (Setup, Join, Sign, Verify, Open).
Definition

Dynamic Group Signature

It is a tuple of algorithms (Setup, Join, Sign, Verify, Open).

- **Setup**: done by a trusted entity
  
  Input: security parameter $\lambda$, bound on group size $N$
  Output: public parameters $Y$, group manager’s secret key $S_{GM}$, the opening authority’s secret key $S_{OA}$
Definition

Dynamic Group Signature

It is a tuple of algorithms \((\text{Setup, Join, Sign, Verify, Open})\).

- **Join**: interactive protocols between \(U_i \leftrightarrow GM\).

  Provides \((\text{cert}_i, \text{sec}_i)\) to \(U_i\).

  Where \(\text{cert}_i\) attests the secret \(\text{sec}_i\).

  Updates the list of users and membership certificates.
Definition

Dynamic Group Signature

It is a tuple of algorithms \((\text{Setup}, \text{Join}, \text{Sign}, \text{Verify}, \text{Open})\).

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

- **Open:**
  
  Input: OA’s secret \(S_{OA}\), \(M\) and \(\Sigma\)
  
  Output: \(i\) or \(\bot\)
Security Notions

Three security notions

- **Anonymity** Only OA can open a signature
Security Notions

Three security notions

- **Anonymity** Only OA can open a signature

- **Traceability** Security of honest GM against malicious users who want to escape from traceability
Security Notions

Three security notions

- **Anonymity** Only OA can open a signature
- **Traceability** Security of honest GM against malicious users who want to escape from traceability
- **Non-frameability** Security of honest members against malicious GM/OA authorities
Security Notions

Three security notions

- **Anonymity** Only OA can open a signature

- **Traceability** Security of honest GM against malicious users who want to escape from traceability

- **Non-frameability** Security of honest members against malicious GM/OA authorities

CCA/CPA security refers to anonymity
Security Notions

Three security notions

- **Anonymity** Only OA can open a signature
- **Traceability** Security of honest GM against malicious users who want to escape from traceability
- **Non-frameability** Security of honest members against malicious GM/OA authorities

CCA/CPA security refers to anonymity

→ Decryption queries correspond to opening queries
Generic Construction

- **Keygen** $\rightarrow S_{GM}, S_{OA}, \mathcal{Y}$
  - $S_{GM} \leftarrow \text{Sign.sk}$
  - $S_{OA} \leftarrow \text{Enc.sk}$
  - $\mathcal{Y} \leftarrow (\text{Sign.pk}, \text{Enc.pk})$
Generic Construction

- **Keygen** $\rightarrow S_{GM}, S_{OA}, Y$
  - $S_{GM} \leftarrow \text{Sign.sk}$
  - $S_{OA} \leftarrow \text{Enc.sk}$
  - $Y \leftarrow (\text{Sign.pk}, \text{Enc.pk})$

- **Join**
  - $\text{cert}_i \leftarrow \text{GM obliviously sign identity sec}_i = \text{ID chosen by } \mathcal{U}_i$
Generic Construction

- **Keygen** → $S_{GM}, S_{OA}, \mathcal{Y}$
  - $S_{GM} \leftarrow \text{Sign.sk}$
  - $S_{OA} \leftarrow \text{Enc.sk}$
  - $\mathcal{Y} \leftarrow \text{(Sign.pk, Enc.pk)}$

- **Join**
  - $cert_i \leftarrow \text{GM obliviously sign identity sec}_i = \text{ID chosen by} \ U_i$

- **Sign** → $(C, \pi)$
  - $\widetilde{cert} \leftarrow U_i$ re-randomize $cert_i$
  - $C \leftarrow \text{Encrypt(} \widetilde{cert}; \ r \text{)}$
  - $\pi \leftarrow \text{ZKoK of (ID; } \widetilde{cert}, r \text{)}$
Generic Construction

- **Keygen** → \( S_{GM}, S_{OA}, \mathcal{Y} \)
  - \( S_{GM} \leftarrow \text{Sign.sk} \)
  - \( S_{OA} \leftarrow \text{Enc.sk} \)
  - \( \mathcal{Y} \leftarrow (\text{Sign.pk}, \text{Enc.pk}) \)

- **Join**
  - \( \text{cert}_i \leftarrow \text{GM obliviously sign identity sec}_i = \text{ID chosen by } \mathcal{U}_i \)

- **Sign** → \((C, \pi)\)
  - \( \text{cert} \leftarrow \mathcal{U}_i \text{ re-randomize } \text{cert}_i \)
  - \( C \leftarrow \text{Encrypt} (\text{cert}; \ r) \)
  - \( \pi \leftarrow \text{ZKoK} \text{ of (ID; cert, r)} \)

Use of the previous signature with efficient protocols.
Results

Security

The scheme is traceable, resistant to framing attacks and CCA-anonymous in the ROM under SXDH and SDL assumptions.
Results

**Security**

The scheme is traceable, resistant to framing attacks and CCA-anonymous in the ROM under SXDH and SDL assumptions.

**In the random oracle model for efficiency reasons.**

(Libert-Peters-Yung’15 signature has 19+8 group elements)
Results

Security

The scheme is traceable, resistant to framing attacks and CCA-anonymous in the ROM under SXDH and SDL assumptions.

In the random oracle model for efficiency reasons. (Libert-Peters-Yung’15 signature has 19+8 group elements)

<table>
<thead>
<tr>
<th>Name</th>
<th>Signature length</th>
<th>Assumptions</th>
<th>Group Type</th>
<th>Anonymity</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>G,Z_p bitsize</td>
<td>q-SDH + DLIN</td>
<td>Static</td>
<td>CPA</td>
</tr>
<tr>
<td>BBS04</td>
<td>3,6,2304</td>
<td>q-SDH + XDH</td>
<td>Dynamic</td>
<td>CCA</td>
</tr>
<tr>
<td>DP06</td>
<td>4,5,2304</td>
<td>interactive + SDL</td>
<td>Dynamic</td>
<td>CCA-</td>
</tr>
<tr>
<td>BCNSW10</td>
<td>3,2,1280</td>
<td>interactive</td>
<td>Dynamic</td>
<td>CCA-</td>
</tr>
<tr>
<td>PS16</td>
<td>2,2,1024</td>
<td>interactive</td>
<td>Dynamic</td>
<td>CCA-</td>
</tr>
<tr>
<td>Ours</td>
<td>7,3,2560</td>
<td>SXDH + SDL</td>
<td>Dynamic</td>
<td>CCA</td>
</tr>
</tbody>
</table>

Table: Comparison between different group signature schemes

CCA- means selfless-CCA-anonymity
Conclusion

We propose:

- A group signature built on **well studied assumptions** with comparable signature length with other schemes
  - Almost as efficient as Delerablée-Pointcheval’06

- A rather efficient signature with efficient protocols that can be used for other privacy-friendly protocols

- An implementation is in progress
Conclusion

We propose:

■ A group signature built on **well studied assumptions** with comparable signature length with other schemes
  ▶ Almost as efficient as Delerablée-Pointcheval’06

■ A rather efficient signature with efficient protocols that can be used for other privacy-friendly protocols

■ An implementation is in progress

Thank you for your attention.
Any Question?