Structure-Preserving Cryptography

(Invited talk in Asiacrypt 2015)

Masayuki ABE
NTT Secure Platform Laboratories
Modular Design in General

Modular design:
A design approach that subdivides a system into smaller parts that can be independently created and then used in different systems.

(mostly from Wikipedia)
Modular Design in General

Benefit:
Reduction in cost
less customization,
shorter learning time
Modular Design in General

Benefit:
Reduction in cost
  less customization,
  shorter learning time
Flexibility in design
Modular Design in General

**Benefit:**
Reduction in cost
  - less customization,
  - shorter learning time
Flexibility in design
Augmentation
  - adding new solution by merely plugging in a new module,
  - making the manufacturing process more adaptive to change
Modular Design in General

**Downside:**
Low quality modular systems are not optimized for performance. This is usually due to the cost of putting up interfaces between modules.
Blind Signatures [Fis06]

\[ P_{\text{user}} \]

\[ m' \leftarrow \text{Commit}(m; z) \]

\[ P_{\text{signer}} \]

\[ (sk, vk) \]

\[ s \leftarrow \text{Sign}_{sk}(m') \]

\[ P_{\text{verifier}} \]

\[ 1 \leftarrow \text{Verify}_{pk}(m', s) \]

\[ \sigma : \text{NIZKPoK for} \]

\[ (m, \sigma) \]

unlinkable
Generic construction in cryptography is a design approach that constructs a cryptographic system by combining smaller and abstract cryptographic primitives that conform to independent security notions.

Benefit:
Reduction in cost: Simper, easy-to-follow security proofs.
Flexibility in design: Off-the-shelf building blocks. Choice of assumptions.
Augmentation: New solution by plugging in a new building block
Generic construction in cryptography is a design approach that constructs a cryptographic system by combining smaller and abstract cryptographic primitives that conform to independent security notions.

**SECURITY**
- **BENEFITS**
  - Easy-to-read security proofs, choice of assumptions, etc.

**FUNCTIONALITY**
- **BENEFITS**
  - Less learning time, use of existing schemes, efficiency improvement, etc.

**MODULALITY**
Generic Construction in Cryptography

Downside:
Mainly used to show feasibility under minimal assumptions, or to show the underlying ideas. Often hard or ignored to find an efficient instantiation.

- Quest for minimal assumptions, etc.
- Use of too powerful inefficient building blocks to have interoperability.
**Structure-Preserving Cryptography** is a framework for efficiently instantiating generic constructions using bilinear groups as a common ground for building blocks.

### Bilinear Groups

\[ \Lambda = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, G_1, G_2) \]

\( \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T \) : groups of prime order \( p \)

\( G_1, G_2, e(G_1, G_2) \) generate \( \mathbb{G}_1, \mathbb{G}_2, \) and \( \mathbb{G}_T \), respectively

\[ e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T \]

\[ \forall X \in \mathbb{G}_1, \forall Y \in \mathbb{G}_2, \forall a, b \in \mathbb{Z}_p, \quad e(X^a, Y^b) = e(X, Y)^{ab} \]
Structure-Preserving Schemes

A cryptographic scheme is structure-preserving if:

- (Group elements as interface) All public objects such as public-keys, messages, commitments, etc, merely consist of elements in $G_1$ and $G_2$.

- (Correctness via PPEs) Verifying relations of interest can be done only by group operations, membership testing, and evaluating pairing product equations of the form

$$\prod_{i=1}^{n} e(A_i, Y_i) \prod_{i=1}^{m} e(X_i, B_i) \prod_{i=1}^{m} \prod_{j=1}^{n} e(X_i, Y_j)^{c_{ij}} = T.$$  

where $c_{ij}$ and $T$ are system-defined constants.
A cryptographic scheme is structure-preserving if:

- **(Group elements as interface)**
  
  All public objects such as public keys, messages, commitments, etc., merely consists of elements in $G_1$ and $G_2$.

- **(Correctness via PPEs)**
  
  Verifying relations of interest can be done only by group operations, membership testing, and evaluating pairing product equations of the form

  \[
  \prod_{i}^{n} e(A_i, Y_i) \prod_{i}^{m} e(X_i, B_i) \prod_{i}^{m} \prod_{j}^{n} e(X_i, Y_j)^{c_{ij}} = T.
  \]

  where $c_{ij}$ and $T$ are system-defined constants.

Aim for high interoperability and use of efficient Groth-Sahai proof system.
Structure-Preserving Schemes

A cryptographic scheme is structure-preserving if:

- **(Group elements as interface)**
  All public objects such as public keys, messages, commitments, etc., merely consists of elements in $G_1$ and $G_2$.

- **(Correctness via PPEs)**
  Verifying relations of interest can be done only by group operations, membership testing, and evaluating pairing product equations of the form:

  $$
  \prod_{i=1}^{n} e(A_i, Y_i) \prod_{i=1}^{m} e(X_i, B_i) \prod_{i=1}^{m} \prod_{j=1}^{n} e(X_i, c_{ij})
  $$

  where $c_{ij}$ and $T$ are system-defined constants.

Aim for high interoperability and use of efficient Groth-Sahai proof system.

Constraints that make it hard to build efficient building blocks.
Difficulty in Designing SP Primitives

Example: Signature Scheme

Key generation:  \( sk \rightarrow vk \)

Signing:  \( m, r, sk \rightarrow \sigma \)

Verification:  \( m \rightarrow \sigma \rightarrow vk \)

Must find a one-way structure within the same group (without hash functions).
(In)Feasibility of SP Primitives

Feasible
- NIWI, NIZK, CCA PKE, Non-shrinking Commitments, Shrinking TCR Commitments, (One-time/ Homomorphic/ Automorphic/ Equivalence Class) Signatures, Oblivious Transfer

Unknown
- ID-based Encryption, Functional Encryption, ...

Infeasible
- Unique signatures, (V)PRF, Deterministic Encryption, Shrinking CR Commitments
Known Structure-Preserving Primitives

Proof Systems
- NIWI, NIZK [Gro06, GS08, GSW10, EG14]
- Properties of GS-proofs [BCCKLS09, Fu11, CKLM12]
- Simulation-Sound NIZK [Gro06, CCS08, HJ12]

Signatures
- Constructions [Gro06, GH08, CLY09, AFGHO10, AHO10, AGHO11, CK11, ACD+12, CDH12, CK12, ADKNO13, LPJY13, ALP13, AGOTia14, AGOTib14, HS14, LJ14, CM14, BFFSST15, LPY15, AKOTi15, KPW15, Groth15]
- Bounds [AGHO11, AGO11, AGOTia14, AGOTib14]

Public-Key Encryption
- CPA [ElG85, HK07, Sha07]
- CCA2 [CHKLN11]

Commitments
- Constructions [Gro09, AFGHO10, AHO10, AHO12, AKOTi15]
- Bounds [AHO12]

Oblivious Transfer
- Construction [DDvMNP15]
Known Structure-Preserving Primitives

Proof Systems
- NIWI, NIZK [Gro06, GS08]
- Properties of GS-proofs [BCCKLS09, Fuc11, CKLM12]
- Simulation-Sound NIZK [Gro06, CCS08, HJ12]

Signatures
- Constructions [Gro06, GH08, CLY09, AFGHO10, AHO10, AGHO11, CK11, ACD+12, CDH12, CK12, ADKNO13, LPJY13, ALP13, AGOTia14, AGOTib14, HS14, LJ14, CM14, BFFSST15, LPY15, AKOTi15, KPW15, Groth15]
- Bounds [AGHO11, AGO11, AGOTia14, AGOTib14]

Public-Key Encryption
- CPA [EIG85, HK07, Sha07]
- CCA2 [CHKLN11]

Commitments
- Constructions [Gro09, AFGHO10, AHO10, AHO12, AKOTi15]
- Bounds [AHO12]

Oblivious Transfer
- Construction [DDvMNP15]
Structure-Preserving Signatures

- The public keys, messages, and signatures consist of elements of $G_1$ and $G_2$, and

- signature verification is done only by group operations, membership testing, and evaluation of pairing product equations.
# Advances of Research on SPS

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Assumptions</th>
<th>Signature Size</th>
<th>Group Type</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Fuc09]</td>
<td>$q$-DHSDH</td>
<td>$21k + 11$</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>[CLY09]</td>
<td>$q$-HSDH, Flex-DH, DLIN</td>
<td>$9k + 4$</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>[AFGH010]</td>
<td>$q$-SFP</td>
<td>7</td>
<td>Any</td>
<td>Unilateral</td>
</tr>
<tr>
<td>[AGHO11]</td>
<td>$q$-type, SXDH</td>
<td>6</td>
<td>III</td>
<td></td>
</tr>
<tr>
<td></td>
<td>interactive, SXDH</td>
<td>4</td>
<td>III</td>
<td>Unilateral</td>
</tr>
<tr>
<td>[AGOTi_{a14}]</td>
<td>interactive</td>
<td>3</td>
<td>Any</td>
<td>$k = 1$</td>
</tr>
<tr>
<td>[AGOTi_{b14}]</td>
<td>interactive</td>
<td>3</td>
<td>III</td>
<td>$k = 1$</td>
</tr>
<tr>
<td></td>
<td>interactive</td>
<td>2</td>
<td>II</td>
<td>$k = 1$</td>
</tr>
<tr>
<td>[Gro06]</td>
<td>DLIN</td>
<td>$O(k)$</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>[CK12]</td>
<td>DLIN</td>
<td>$24k + 100 + 9x$</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SXDH, RCDH</td>
<td>$18k + 77 + 6x$</td>
<td>III</td>
<td>Unilateral</td>
</tr>
<tr>
<td>[CDH12]</td>
<td>DLIN</td>
<td>$6k + 53$</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>[ACDKNO12]</td>
<td>DLIN</td>
<td>17</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SXDH, XDLIN</td>
<td>14</td>
<td>III</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>11</td>
<td>III</td>
<td>Unilateral</td>
</tr>
<tr>
<td>[ADKNO13]</td>
<td>DLIN</td>
<td>14</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>[KPW15]</td>
<td>2-LIN</td>
<td>10</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SXDH</td>
<td>7</td>
<td>III</td>
<td>Unilateral</td>
</tr>
</tbody>
</table>

$(|\text{msg}| = k, 2^x \text{ sigs.})$
Static v.s. q-Type Assumptions

Static assumptions

• Simple as DLIN, SXDH
• Widely used.

Security of [ADKNO13, KPW15] is reduced to static assumptions with loss factor of $1/q$ and $1/q^2$.

q-Type Assumptions

• Consists of huge number of group elements
• Some are widely used, some are just ad-hoc.

Security of [AFGHO10, AGHO11] is tightly reduced to q-type assumptions.
Static v.s. q-Type Assumptions

Static Assumptions

• Simple as DLIN, SXDH
• Widely used.
• Tight generic security
  \[ \text{CDH: } O(\ell^2)/|G| \]

q-Type Assumptions

• Consist of huge number of group elements
• Some are widely used, some are just ad-hoc.
• Loose generic security?
  \[ q-\text{SDH: } O(q \ell^2)/|G| \]

Any generic adversary after \( \ell \) steps wins at most with this probability.

Given: \( G, G^x, G^{x^2}, ..., G^{x^q} \)
Find: \( G^{\frac{1}{x+c}}, c \)
q-Type Assumption: Example

Simultaneous Flexible Pairing Assumption (SFP) [AFGHO10]

Given: \((A, A', B, B', G_z, H_z, G_r, H_u) \in \mathbb{G}^8\) and 
\((Z_i, R_i, S_i, T_i, U_i, V_i, W_i) \in \mathbb{G}^7\) for \(i = 1, \ldots, q\) constrained that

\[
e(A, A') = e(G_z, Z_i) e(G_r, R_i) e(S_i, T_i)
\]

\[
e(B, B') = e(H_z, Z_i) e(H_u, U_i) e(V_i, W_i)
\]

Find: \((Z, R, S, T, U, V, W)\) with \(Z \not\in \{Z_1, \ldots, Z_q\}\) and \(Z \neq 1\)
Tight Generic Hardness of SFP

**Theorem.** [AGFHO15] Any generic algorithm $A$ performing up to $\ell$ group operations and pairings solves the $q$-SFP problem with probability at most $O(\ell^2 + q^2)/|\mathbb{G}|$. 
Tight Generic Hardness of SFP

**Theorem.** [AGFHO15] Any generic algorithm $A$ performing up to $\ell$ group operations and pairings solves the $q$-SFP problem with probability at most $\mathcal{O}(\ell^2 + q^2)/|G|$.

- **CDH:** $\mathcal{O}(\ell^2)/|G|$  
  [Sho97]

- **q-SDH:** $\mathcal{O}(q \ell^2)/|G|$  
  [BB04, Che06]
Tight Generic Hardness of SFP

Theorem.[AGFHO15] Any generic algorithm $A$ performing up to $\ell$ group operations and pairings solves the $q$-SFP problem with probability at most $O(\ell^2 + q^2)/|\mathbb{G}|$.

Proof intuition:
Recall that reference answers satisfy

$$e(A, A') = e(G_z, Z_i) e(G_r, R_i) e(S_i, T_i)$$
$$e(B, B') = e(H_z, Z_i) e(H_u, U_i) e(V_i, W_i).$$

Taking the discrete log wrt $G_r$, the relations are written as

$$a a' = g_z z_i + 1 r_i + s_i t_i$$
$$b b' = h_z z_i + h_u u_i + v_i w_i.$$
Proof intuition (cont’d):

The $j$-th group element viewed by the generic adversary is represented by a function $F_j$ of a linear combination of variables $a, a', b, b', g_z, 1, h_z, h_u, z_i, r_i, s_i, t_i, u_i, v_i, w_i$ where

$$r_i := a a' - g_z z_i - s_i t_i$$
$$u_i := (b b' - h_z z_i - v_i w_i)/h_u.$$ 

For every $j, j' < \ell + q$, $F_j - F_j'$ and $F_j \cdot F_j'$ are Laurent polynomials with total degree of at most some small constants. Thus any of them vanishes at a random assignment to the variables only with probability at most $2C_{\ell+q}(\text{const } / |G|) = O(\ell^2 + q^2) / |G|$ as claimed.
## Lower Bounds on Signature Size

<table>
<thead>
<tr>
<th>Group type</th>
<th>Messages</th>
<th>Lower bounds</th>
<th></th>
<th>Upper Bounds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Interactive</td>
<td>Non-interactive</td>
<td>Interactive</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7 [KPW15]</td>
</tr>
<tr>
<td>Type-II</td>
<td>$M \in G_1$, Bilateral</td>
<td>3 [AGOTib14]</td>
<td></td>
<td>3*1 [AGOTia14]</td>
</tr>
<tr>
<td></td>
<td>$M \in G_2$</td>
<td>2 [AGOTib14]</td>
<td></td>
<td>2*2 [AGOTib14]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 [AGOTib14]</td>
</tr>
<tr>
<td>Type-I</td>
<td>N/A</td>
<td>3 [AGOTia14]</td>
<td></td>
<td>3*2 [AGOTia14]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7 [AFGHO10]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10 [KPW15]</td>
</tr>
</tbody>
</table>

*2: Single-element message.
Extensions of SPS

Linearly homomorphic SPS [LPJY13]
  • Application to Quasi-adaptive NIZK

Selectively randomizable SPS [AGOTi14a]
  • Flexibly change signatures from strongly unforgeable to randomizable ones

SPS for equivalence classes [HS14,FHS15]
  • Can sign on equivalence classes defined by vector of group elements
  • Application to optimal-round blind signatures without using GS-proofs

Fully SPS [AKOTi15,Gro15(tomorrow!)]
  • Even secret-keys are group elements
Open Problems on SPS

- Find more lower bounds for the case of non-interactive assumptions.
- Separately show lower bounds for static assumptions. Are they different from those for q-type assumptions?
- Show constant-size SPS with a tight reduction to simple assumptions.
Structure-Preserving Commitments

- The commitment keys, messages, commitments, and opening informations consist of elements of $\mathbb{G}_1$ and $\mathbb{G}_2$, and

- opening verification is done only by group operations, membership testing, and evaluation of pairing product equations.
State of The Art

Performance of Structure-Preserving Commitment Schemes

| Group   | Scheme   | $|msg|$ | $|key|$ | $|com|$ | $|open|$ | #(PPE) | asmpt. |
|---------|----------|-------|--------|--------|--------|--------|--------|
| Type-I  | [Gro09]  | $k$   | $2k + 4$ | $2_{(\mathbb{G}_T)}$ | 2      | 2      | STP    |
|         | [AHO10]  | $k$   | $2k + 2$ | $2_{(\mathbb{G}_T)}$ | 2      | 2      | SDP    |
|         | [CLY09]  | $k$   | 5       | $3k$   | $3k$   | $3k$   | DLIN   |
|         | [AHO10]  | $k$   | $2k + 2$ | $2k + 2$ | 2      | 2      | SDP    |
|         | [AHO12]  | $k$   | $2k + 3$ | $k + 2$ | 2      | 2      | SDP    |
| Type-III| [AHO12]  | $(0, k)$ | $(k + 1, 1)$ | $(1, k)$ | $(0, 1)$ | 1      | DBP    |

$|msg| < |com|$ 

Abe, Haralambiev, Ohkubo, “Group to Group Commitments Do Not Shrink”, Eurocrypt 2012
Theorem 9 [AHO12]. If the discrete-logarithm problem is hard in the base groups, key generation and commitment algorithms are algebraic, and $|\text{com}| < |\text{msg}|$, then the commitment scheme is not binding.

Binding Property (Collision Resistance)

$ck$

$A_{cr}$

$(\text{com}, \text{msg}, \text{open}, \text{msg}', \text{open}')$

$1 = \text{TC.Vrf}(ck, \text{com}, \text{msg}, \text{open}) = \text{TC.Vrf}(ck, \text{com}, \text{msg}', \text{open}')$
Theorem [AKOTi14]. There exists a shrinking homomorphic trapdoor structure-preserving commitment scheme that is chosen message target collision resistant (CMTCR) if there exist a one-time non-adaptive chosen message secure structure-preserving partially one-time signature scheme (POS) and a γ target collision resistant trapdoor commitment scheme (γ-TC) exists.

Notion of CMTCR

\[ \text{A}_{\text{cmtcr}} \]

\[ ck \]

\[ msg_i \]

\[ (i^*, msg^*, open^*) \]

\[ com_i, open_i \]

\[ (com_i, open_i) \leftarrow \text{TC.Com}(ck, msg_i; r) \]

\[ 1 = \text{TC.Vrf}(ck, com_i^*, msg^*, open^*) \]
Impossibility Argument for CR

If TC.Com is algebraic, commitment $\vec{C}$ and opening $\vec{D}$ are computed by linear combination of message $\vec{M}$ and commitment key $\vec{V}$. Coefficient matrix $B$ may depend on $\vec{M}$, $\vec{V}$ and internal random coins.

$$\begin{pmatrix} \vec{C} \\ \vec{D} \end{pmatrix} = \begin{pmatrix} B_1 & B_2 \\ B_3 & B_4 \end{pmatrix} \begin{pmatrix} \vec{M} \\ \vec{V} \end{pmatrix}$$

If $|\vec{C}| < |\vec{M}|$, there exists $\vec{M}' (\neq \vec{M})$ that $\vec{C} = (\vec{B}_1 | \vec{B}_2)(\vec{M}' | \vec{V})^T$. To compute such $\vec{M}'$, matrix $\vec{B}$ must be known to the adversary.
Impossibility Argument for CR

If TC.Com is algebraic, commitment $\bar{C}$ and opening $\bar{D}$ are computed by linear combination of message $\hat{M}$ and commitment key $\hat{V}$. Coefficient matrix $B$ may depend on $\hat{M}$, $\hat{V}$ and internal random coins.

$$
\begin{align*}
\begin{pmatrix}
\bar{C} \\
\bar{D}
\end{pmatrix}
&= 
\begin{pmatrix}
B_1 & B_2 \\
B_3 & B_4
\end{pmatrix}
\begin{pmatrix}
\hat{M} \\
\hat{V}
\end{pmatrix}
\end{align*}
$$

If $|\bar{C}| < |\hat{M}|$, there exists $\hat{M}'(\neq \hat{M})$ that $\bar{C} = (\bar{B}_1 | \bar{B}_2)(\hat{M}' | \hat{V})^T$. To compute such $\hat{M}'$, matrix $\bar{B}$ must be known to the adversary.
Impossibility Argument for CR

If TC.Com is algebraic, commitment $\vec{C}$ and opening $\vec{D}$ are computed by linear combination of message $\vec{M}$ and commitment key $\vec{V}$. Coefficient matrix $B$ may depend on $\vec{M}$, $\vec{V}$ and internal random coins.

$$
\begin{bmatrix}
\vec{C} \\
\vec{D}
\end{bmatrix}
= 
\begin{bmatrix}
B_1 & B_2 \\
B_3 & B_4
\end{bmatrix}
\begin{bmatrix}
\vec{M} \\
\vec{V}
\end{bmatrix}
$$

If $|\vec{C}| < |\vec{M}|$, there exists $\vec{M}' (\neq \vec{M})$ that $\vec{C} = \vec{M}'$

To compute such $\vec{M}'$, matrix $\vec{B}$ must be known.

The impossibility argument does not apply to CM-TCR.

In the CR game, it is the adversary that chooses the random coins. Hence it knows matrix $B$.

In CM-TCR game, it is the challenger that chooses the random coins. Thus matrix $B$ is unknown to the adversary.
Shrinking SPTC : Generic Construction

Building blocks

- Partial one-time signature scheme (POS)
- $\gamma$-target collision resistant commitment scheme ($\text{TC}_\gamma$)

\[ \text{Com}(ck, msg) \]

1. \( (M^{(1)}, \ldots, M^{(\kappa)}) \in \mathcal{M}_\text{pos}^{\kappa} \leftarrow msg \)
2. \( (vk_{\text{pos}}, sk_{\text{pos}}) \leftarrow \text{POS.Key}(gk) \)
3. For \( i = 1, \ldots, \kappa \)
   \[
   (ovk_{\text{pos}}^{(i)}, osk_{\text{pos}}^{(i)}) \leftarrow \text{POS.Ovk}(gk) \\
   \sigma_{\text{pos}}^{(i)} \leftarrow \text{POS.Sign}(sk_{\text{pos}}, osk_{\text{pos}}^{(i)}, M^{(i)})
   \]
4. \( (\text{com}_{\text{gbc}}, \text{open}_{\text{gbc}}) \leftarrow \text{TC}_\gamma.\text{Com}(sk_{\text{gbc}}, osk_{\text{gbc}}^{(1)}, \ldots, osk_{\text{pos}}^{(\kappa)}) \)
5. \( \text{com} := \text{com}_{\text{gbc}} \)
   \[
   \text{open} := (\text{open}_{\text{gbc}}, vk_{\text{pos}}, ovk_{\text{pos}}^{(1)}, \ldots, ovk_{\text{pos}}^{(\kappa)}, \sigma_{\text{gbc}}^{(1)}, \ldots, \sigma_{\text{pos}}^{(\kappa)})
   \]
Shrinking SPTC : sketch

Concrete SPTC from POS and TC\(\gamma\) in Type-III groups

Commitment-key : \(ck = (G, \tilde{G}, \tilde{X}, \tilde{X}_1, \ldots, \tilde{X}_k, \tilde{Y}_1, \ldots, \tilde{Y}_\ell)\)

Message : \(M = (\tilde{M}_1, \ldots, \tilde{M}_{k, \ell})\)

Commitment : \(C = \tilde{G}_u\)

Opening : \(D = (R, \{\tilde{Z}_j, \tilde{R}_j\}_{j=1}^k, A_1, \ldots, A_k, G_z, G_1, \ldots, G_\ell)\)

Verification(\(ck, M, C, D\))

\[
e(G, \tilde{G}_u) = e(R, \tilde{G}) e(G_z, \tilde{X}) \prod_{i=1}^k e(A_i, \tilde{X}_i) \prod_{i=1}^\ell e(G_i, \tilde{Y}_i)
\]

for \(j = 1, \ldots, k\)

\[
e(A_j, \tilde{G}) = e(G_z, \tilde{Z}_j) e(G, \tilde{R}_j) \prod_{i=1}^\ell e(G_i, \tilde{M}_{(j-1)\ell+i})
\]

\[
\begin{array}{|c|c|}
\hline
\text{Message} & k \cdot \ell \\
\text{Commit key} & 1 + k + \ell \\
\text{Trapdoor} & 1 + k + \ell \text{ (in } \mathbb{Z}_p) \\
\text{Commitment} & 1 \\
\text{Opening info} & 2 + 3k + \ell \\
\text{# of PPEs} & 1 + k \\
\hline
\end{array}
\]
Usefulness of CM-TCR

CM-TCR is still useful in constructing CMA-secure signature schemes.

CMA-security Game

\[ \mathcal{A}_{cma} \]

\[ \frac{pk}{\downarrow} \]

\[ (\sigma^*, msg^*) \]

Adversary’s choice

\[ msg_i \]

\[ (com_{tc}, open_{tc}) \leftarrow TC.Com(ck_{tc}, msg_i; r) \]

\[ \sigma_{xsig} \leftarrow xSIG.Sign(sk_{xsig}, com_{tc}) \]

\[ \sigma_i := (\sigma_{xsig}, com_{tc}, open_{tc}) \]

Challenger’s choice (not seen by the adversary)

\[ msg^* \notin \{msg_i\} \]

\[ 1 \leftarrow SIG.Vrf(pk, msg^*, \sigma^*) \]
Applications
# Works on Structure-Preserving Crypto

## Proof Systems
- NIWI, NIZK [Gro06, GS08, GSW10, EG14]
- Properties of GS-proofs [BCCKLS09, Fuc11, CKLM12]
- Simulation-Sound NIZK [Gro06, CCS08, HJ12]

## Signatures
- Constructions [Gro06, GH08, CLY09, AFGHO10, AHO10, AGHO11, CK11, ACD+12, CDH12, CK12, ADKNO13, LPJY13, ALP13, AGOTia14, AGOTib14, HS14, LJ14, CM14, BFFSST15, LPY15, AGKOTi15, KPW15, Groth15]
- Bounds [AGHO11, AGO11, AGOTia14, AGOTib14]

## Public-Key Encryption
- CPA [ElG85, HK07, Sha07]
- CCA2 [CHKLN11]

## Commitments
- Constructions [Gro09, AFGHO10, AHO10, AHO12, AKOTi15]
- Bounds [AHO12]

## Oblivious Transfer
- Construction [DDvMNP15]

## Applications (Blind signature, Group signature, Credential system, etc,...)
- [AFGHO10, CHKLN11, Kris11, ALP12, LPY12, HJ12, CKLM12, FKMV12, AJ13, BFG13, LPJY13, KR13, CMA13, SEHKMO13, ZLG13, ACDN14, LJYP14, LPJM14, AEHS14, LPDW14, AGBSS14, HRS15, FHS15, KM15, Ghada15]
General Idea for Group Signatures

“Group/Traceable Signature” = “Signature” + “Revocation Mechanism”

- Guarantees integrity of messages.
- Authenticate the signer.

Glued by NIZK that guarantees correct computation while hiding privacy related objects in each part.

[AFGHO10] Signatures
[BB04] one-time Signatures

Groth-Sahai Proof System

Encryption/Anonymous Tag System

- Opening, Tracing
- Claiming, Denial
## Comparison

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Construction Type</th>
<th>Claim &amp; Trace</th>
<th>Deny Function</th>
<th>Anonymity w/ Trace</th>
<th>Anonymity Level</th>
<th>Concurrent Join</th>
<th>Sig. Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>[AHO10]</td>
<td>generic</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>CCA</td>
<td>yes</td>
<td>58</td>
</tr>
<tr>
<td>[LY09]</td>
<td>Tailor-made</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>CPA</td>
<td>no</td>
<td>83</td>
</tr>
<tr>
<td>[ACHO11]</td>
<td>generic</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>CCA</td>
<td>yes</td>
<td>107</td>
</tr>
</tbody>
</table>

Table: Summary of properties among group signature and traceable signature schemes that provide non-frameability (the signature size counts the number of group elements).
Wrap Up
Conclusion

• There are proof systems, signatures, encryption, and commitments over bilinear groups that are structure-preserving and thus interoperable each other.

• They can be used for modular construction of intricate cryptographic tasks. And the efficiency of the resulting scheme can be evaluated with concrete figures.
  
  • There is room for hand-crafted optimization by carefully choosing which elements are hidden and which are put in the clear.

• There are interesting open problems both in practice (efficiency improvements) and theory (lower bounds), and missing important tools like IBE, FE, etc.
Authors of papers that study or use SP primitives.