Adaptively Simulation-Secure Attribute-Hiding Predicate Encryption

by

Pratish Datta[1]

joint work with

Tatsuaki Okamoto[1] and **Katsuyuki Takashima**[2]

[1]NTT Secure Platform Laboratories
3-9-11 Midori-cho, Musashino-shi,
Tokyo, 180-8585 Japan

[2]Mitsubishi Electric
5-1-1 Ofuna, Kamakura,
Kanagawa, 247-8501 Japan

ASIACRYPT 2018
December 02–06, 2018

Outline

Functional Encryption (FE)

- Setup authority holds a master secret key $\textsc{msk}$ and publishes public system parameters $\textsc{mpk}$.
- An encrypter uses $\textsc{mpk}$ to encrypt message $M \in \mathbb{M}$, creating ciphertext $\textsc{ct}$.
- A decrypter obtains a private decryption key $\textsc{sk}(F)$ for function $F \in \mathcal{F}$, generated using $\textsc{msk}$ by the authority.
- $\textsc{sk}(F)$ can be used to decrypt $\textsc{ct}$ to recover $F(M)$, but nothing more about M.

Various Security Notions for FE

- **Indistinguishability-based (IND) Security**: Distinguishing encryptions of any two messages is infeasible for a group of colluders which do not have a decryption key that decrypts the ciphertexts to distinct values.

- **Simulation-based (SIM) Security**: There exists a polynomial-time simulator that given $F_1(M), \ldots, F_{q_{\text{KEY}}}(M)$ for $M \in \mathbb{M}$, $F_1, \ldots, F_{q_{\text{KEY}}} \in \mathcal{F}$, outputs the view of the colluders given encryption of M and $\text{SK}(F_1), \ldots, \text{SK}(F_{q_{\text{KEY}}})$.
- In general, SIM security is *stronger* than IND security.

Various Security Notions for FE

- **Adaptive (AD) Security**: The adversary is allowed to make ciphertext and decryption key queries at any point of time during the security experiment.
- **Semi-Adaptive (S-AD) Security**: The adversary is restricted to submit its ciphertext queries immediately after viewing the public parameters, and can make decryption key queries only after that.
- **Selective (SEL) Security**: The adversary is bound to declare its ciphertext queries even before the public parameters are generated.

Predicate Encryption (PE)

- Predicate family: $R = \{R(Y, \cdot) : \mathcal{X} \to \{0, 1\} \mid Y \in \mathcal{Y}\}$, $\mathcal{X}, \mathcal{Y}$ = sets of attributes.
- Message space $\mathbb{M} = \mathcal{X} \times \mathcal{M}$, where $\mathcal{M}$ contains the actual payloads.
- Functionality F_{R_Y} associated with predicate $R(Y, \cdot) \in R$:

$$F_{R_Y}(X, \mathsf{msg}) = \left\{ \begin{array}{ll} \mathsf{msg} & \text{if } R(Y, X) = 1 \\ \perp & \text{if } R(Y, X) = 0 \end{array} \right\} \forall (X, \mathsf{msg}) \in \mathbb{M} = \mathcal{X} \times \mathcal{M}.$$

Various Security Notions for PE

- **Strong Attribute Hiding (S-AH)**:
  - Recovering the payload from a ciphertext generated w.r.t $X \in \mathcal{X}$ should be infeasible for a group of colluders not having an authorized decryption key.
  - The ciphertext should conceal X from any group of colluders, even those with authorized decryption keys.
- **Weak Attribute Hiding (W-AH)**: The payload and X should only remain hidden to colluders in possession of unauthorized keys.
- **Payload Hiding (PLH)**: The payload should remain hidden to colluders with unauthorized keys. Also known as attribute-based encryption (ABE).

State of the Art in Attribute-Hiding PE

- Several works developed ABE and W-AH PE schemes supporting *unbounded* collusions even for *general circuits* under *standard* computational assumptions.
- Known *standard*-assumption-based S-AH PE schemes supporting *unbounded* number of *authorized* colluders are restricted to *inner products*.
- It is known that S-AH PE scheme for NC^1 predicates implies indistinguishability obfuscation (IO) for general circuits.

A Motivating Question

Can we design PE *scheme for some sufficiently expressive predicate family (e.g.,* NC^1*) that is secure against an unbounded number of colluders under standard computational assumption such that the* S-AH *guarantee holds for a limited segment (e.g., belonging to some subclass of* NC^1*) of each predicate in the predicate family?*

The Effort of Wee

- In TCC 2017, Wee presented a PE scheme in bilinear groups of prime order secure under the k-LIN assumption.

- $\mathcal{X} = \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n}$, $\mathcal{Y} = \mathcal{F}_{\mathrm{ABP}\circ\mathrm{IP}}^{(q,n',n)}$.

- For any $f \in \mathcal{F}_{\mathrm{ABP}\circ\mathrm{IP}}^{(q,n',n)}$ and $(\vec{x}, \vec{z}) \in \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n}$,

$$f(\vec{x}, \vec{z}) = (f_1(\vec{x}), \ldots, f_n(\vec{x})) \cdot \vec{z},$$

where $f_1, \ldots, f_n : \mathbb{F}_q^{n'} \to \mathbb{F}_q$ are arithmetic branching programs (ABP).

The Attribute-Hiding Characteristics of Wee's PE Scheme

- The predicate family: $R^{\mathrm{ABP} \circ \mathrm{IP}} = \{R^{\mathrm{ABP} \circ \mathrm{IP}}(f, (\cdot, \cdot)) : \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n} \to \{0,1\} \mid f \in \mathcal{F}_{\mathrm{ABP} \circ \mathrm{IP}}^{(q,n',n)}\}$, where

$$R^{\mathrm{ABP} \circ \mathrm{IP}}(f, (\vec{x}, \vec{z})) = \begin{cases} 1 & \text{if } f(\vec{x}, \vec{z}) = 0, \\ 0 & \text{if } f(\vec{x}, \vec{z}) \neq 0. \end{cases}$$

- Other than hiding the payload, CT generated for $(\vec{x}, \vec{z}) \in \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n}$ *conceals* $\vec{z}$ but *not* $\vec{x}$.
- The concealment of $\vec{z}$ is *strong*, i.e., even against colluders possessing authorized keys.
- This security notion is termed as strongly partially-hiding security.

The Advantages and Limitations of Wee's PE Scheme

- This PE scheme simultaneously generalizes ABE for boolean formulas and ABP's, and S-AH inner-product PE (IPE).
- The scheme is strongly partially-hiding against an unbounded number of authorized colluders.
- The security is proven in the SIM framework.
- The downside of this scheme is that it only achieves semi-adaptive security.
- Semi-adaptive security is known to be essentially equivalent to the selective security.
- The known generic conversion from selective to adaptive security does not work for PE schemes not supporting general circuits.

Our Results

- We design a PE scheme for the predicate family $R^{\mathrm{ABP}\circ\mathrm{IP}}$ that achieves SIM-*based adaptively strongly partially hiding* security.
- The scheme supports *any a priori bounded* number of ciphertext queries and *unbounded* number of authorized decryption key queries.
- This is the *best* possible in the SIM-based adaptive security framework.
- This resolves an open problem posed by Wee in TCC 2017.
- The scheme is also *adaptively strongly partially-hiding* in the IND framework against *unbounded* number of ciphertext and authorized decryption key queries.

Our Results

- Our construction is built in *asymmetric* bilinear groups of *prime* order.
- The security is derived under the *simultaneous external decisional linear* (SXDLIN) assumption.
- As a byproduct, we also obtain the *first* SIM-based *adaptively* S-AH IPE scheme supporting unbounded number of authorized colluders.
- We extend the IND-based S-AH methodology of [OT12a, OT12b] to the framework of SIM security and beyond inner products.

[OT12a] : Tatsuaki Okamoto and Katsuyuki Takashima. In EUROCRYPT 2012.
[OT12b] : Tatsuaki Okamoto and Katsuyuki Takashima. In ASIACRYPT 2012.

Comparison with Existing Attribute-Hiding PE Schemes

Schemes	Supported Predicates	IND	SIM	Attribute Hiding	Computational Assumptions
[OT10]	IP∘SP	(poly, poly, poly)-AD	×	Weak (IP-part)	DLIN
[OT12a]	IP	(poly, poly, poly)-AD	×	Strong	DLIN
[Agr17]	GC∘IP	(–, poly, bdd)-S-AD	(–, 1, bdd)-S-AD	Strong (IP-part)	LWE
[Wee17]	ABP∘IP	(–, poly, poly)-S-AD	(–, 1, poly)-S-AD	Strong (IP-part)	k-LIN
Ours	ABP∘IP	(poly, poly, poly)-AD	(poly, bdd, poly)-AD	Strong (IP-part)	SXDLIN

[OT10] : Tatsuaki Okamoto and Katsuyuki Takashima. In CRYPTO 2010.
[OT12a] : Tatsuaki Okamoto and Katsuyuki Takashima. In EUROCRYPT 2012.
[Agr17] : Shweta Agrawal. In CRYPTO 2017.
[Wee17] : Hoeteck Wee. In TCC 2017.

Arithmetic Branching Program ABP

ABP $\Gamma = (V, E, v_0, v_1, \phi)$ computing $f : \mathbb{F}_q^d \to \mathbb{F}_q$:

- (V, E): A directed acyclic graph.
- $v_0, v_1 \in V$: Special vertices called the source and the sink respectively.
- ϕ: A labeling function assigning to each edge in E an affine function in one of the input variables with coefficients in $\mathbb{F}_q$.
- For any $\vec{w} \in \mathbb{F}_q^d$, $f(\vec{w}) = \sum_{P \in \wp} \left[\prod_{e \in P} \phi(e)|_{\vec{w}} \right]$, where $\wp$ is the set of all v_0-v_1 paths P in Γ.

Algorithm $\mathsf{PGB}(f)$ for $f : \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n} \to \mathbb{F}_q \in \mathcal{F}_{\mathrm{ABP}\circ\mathrm{IP}}^{(q,n',n)}$

- Construct the ABP Γ' computing f such that:
  - Γ' has $m + n + 1$ vertices.
  - The variables z_j's only appear on edges leading into the sink vertex.
  - Any vertex has at most one outgoing edge with a label of degree one.
- Using the algorithm of [IK02], compute the matrix representation of Γ',

$$\boldsymbol{L} = \begin{pmatrix} \bigstar & \bigstar & \bigstar & \dots & \bigstar & \bigstar & \dots & \bigstar & 0 \\ -1 & \bigstar & \bigstar & \dots & \bigstar & \bigstar & \dots & \bigstar & 0 \\ 0 & -1 & \bigstar & \dots & \bigstar & \bigstar & \dots & \bigstar & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -1 & \bigstar & \dots & \bigstar & 0 \\ 0 & 0 & 0 & \dots & 0 & -1 & \dots & 0 & z_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & \dots & -1 & z_n \end{pmatrix}_{(m+n)\times(m+n)}$$

with $f(\vec{x}, \vec{z}) = \det(\boldsymbol{L}(\vec{x}, \vec{z}))\forall(\vec{x}, \vec{z}) \in \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n}$, and $\bigstar$'s in the j'^{th} row indicating affine functions in $x_{\rho(j')}$ for all $j' \in [m]$, where $\rho : [m] \to [n']$.

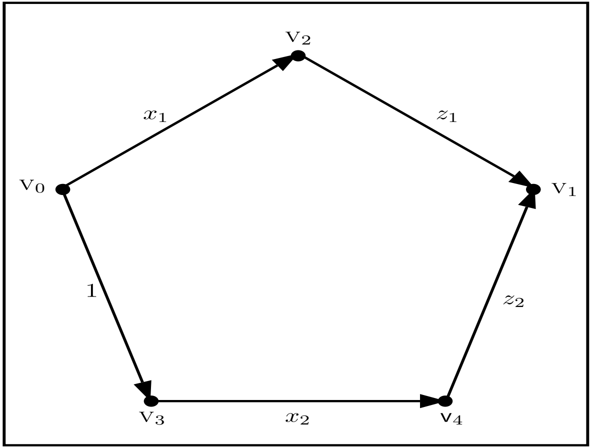
[IK02] : Yuval Ishai and Eyal Kushilevitz. In ICALP 2002.

An Illustrative Example

$$
\begin{aligned}
&f((x_1, x_2), (z_1, z_2)) \\
&= f_1(x_1, x_2)z_1 + f_2(x_1, x_2)z_2 \\
&= x_1 z_1 + x_2 z_2, \\
&\text{where } f_1(x_1, x_2) = x_1,\ f_2(x_1, x_2) = x_2
\end{aligned}
$$

$$
\boldsymbol{L}((x_1, x_2), (z_1, z_2))
$$

$$
= \begin{bmatrix} 1 & x_1 & 0 & 0 \\ -1 & 0 & x_2 & 0 \\ 0 & -1 & 0 & z_1 \\ 0 & 0 & -1 & z_2 \end{bmatrix}
$$



ABP Γ' Computing f

Algorithm PGB(f) for $f : \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n} \to \mathbb{F}_q \in \mathcal{F}_{\mathrm{ABP}\circ\mathrm{IP}}^{(q,n',n)}$ Contd.

- Choose $\vec{r} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{m+n-1}$, and compute

$$\boldsymbol{L}\begin{pmatrix}\vec{r}^{\mathsf{T}}\\ 1\end{pmatrix} = (\alpha_1 x_{\rho(1)} + \gamma_1, \ldots, \alpha_m x_{\rho(m)} + \gamma_m, z_1 + \sigma_1, \ldots, z_n + \sigma_n)^{\mathsf{T}}.$$

- Output $\Big((\{\sigma_j\}_{j\in[n]}, \{\alpha_{j'}, \gamma_{j'}\}_{j'\in[m]}), \rho : [m] \to [n']\Big)$.

- Each of $\{\sigma_j\}_{j\in[n]}, \{\alpha_{j'}, \gamma_{j'}\}_{j'\in[m]}$ are linear functions of $\vec{r}$.

Algorithm REC$(f, \vec{x})$ for $f : \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n} \rightarrow \mathbb{F}_q \in \mathcal{F}_{\mathrm{ABP}\circ\mathrm{IP}}^{(q,n',n)}, \vec{x} \in \mathbb{F}_q^{n'}$

- Generate the matrix representation $\boldsymbol{L} \in \mathbb{F}_q^{(m+n)\times(m+n)}$ of the ABP Γ' computing f.
- Output the cofactors $(\{\Omega'_{j'}\}_{j'\in[m]}, \{\Omega_j\}_{j\in[n]}) \in \mathbb{F}_q^{m+n}$ of all the entries in the last column of $\boldsymbol{L}$ in order.
- The first $m+n-1$ columns of $\boldsymbol{L}$ involve only $\{x_{\iota'}\}_{\iota'\in[n']}$. Hence, all the cofactors are computable.
- Given $(\{\Omega_j\}_{j\in[n]}, \{\Omega'_{j'}\}_{j'\in[m]})$ and $(\{z_j+\sigma_j\}_{j\in[n]}, \{\alpha_{j'}x_{\rho(j')}+\gamma_{j'}\}_{j'\in[m]})$ for any $\vec{z} \in \mathbb{F}_q^n$, recover
$$f(\vec{x},\vec{z}) = \sum_{j'\in[m]} \Omega'_{j'}(\alpha_{j'}x_{\rho(j')}+\gamma_{j'}) + \sum_{j\in[n]} \Omega_j(z_j+\sigma_j).$$

Bilinear Groups

Bilinear group $\mathsf{params}_{\mathbb{G}} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathrm{BPG}}(1^\lambda)$:

- $q \in \mathbb{N}$: Prime integer.
- $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$: Cyclic multiplicative groups of order q with polynomial-time computable group operations.
- $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$: Generators.
- $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$: Mapping satisfying the following:
  - *Bilinearity* : $e(g_1^{\delta}, g_2^{\hat{\delta}}) = e(g_1, g_2)^{\delta\hat{\delta}}$ for all $\delta, \hat{\delta} \in \mathbb{F}_q$.
  - *Non-degeneracy* : $e(g_1, g_2) \neq 1_{\mathbb{G}_T}$, where $1_{\mathbb{G}_T}$ denotes the identity element of the group $\mathbb{G}_T$.
- $\mathsf{params}_{\mathbb{G}}$ is said to be asymmetric if no efficiently computable isomorphism exists between $\mathbb{G}_1$ and $\mathbb{G}_2$.

Dual Pairing Vector Spaces (DPVS)

DPVS $\text{params}_{\mathbb{V}} = (q, \mathbb{V}_1, \mathbb{V}_2, \mathbb{G}_T, \mathbb{A}_1, \mathbb{A}_2, e) \xleftarrow{\text{R}} \mathcal{G}_{\text{DPVS}}(1^\lambda, d, \text{params}_{\mathbb{G}})$:

- $q \in \mathbb{N}$: Prime integer.
- $\mathbb{V}_t = \mathbb{G}_t^d$ for $t \in [2]$: d-dimensional vector spaces over $\mathbb{F}_q$ under vector addition and scalar multiplication defined componentwise.
- $\mathbb{A}_t = \{\boldsymbol{a}^{(t,\ell)} = (\overbrace{1_{\mathbb{G}_t}, \ldots, 1_{\mathbb{G}_t}}^{\ell-1}, g_t, \overbrace{1_{\mathbb{G}_t}, \ldots, 1_{\mathbb{G}_t}}^{d-\ell})\}_{\ell \in [d]}$ of $\mathbb{V}_t$ for $t \in [2]$: Canonical bases, where $1_{\mathbb{G}_t}$ = identity element of $\mathbb{G}_t$.
- $e : \mathbb{V}_1 \times \mathbb{V}_2 \to \mathbb{G}_T$, $e(\boldsymbol{v}, \boldsymbol{w}) = \prod_{\ell \in [d]} e(g_1^{v_\ell}, g_2^{w_\ell}) \in \mathbb{G}_T$ for all $\boldsymbol{v} = (g_1^{v_1}, \ldots, g_1^{v_d}) \in \mathbb{V}_1$, $\boldsymbol{w} = (g_2^{w_1}, \ldots, g_2^{w_d}) \in \mathbb{V}_2$.
- e satisfies the following:
  - *Bilinearity* : $e(\delta\boldsymbol{v}, \widehat{\delta}\boldsymbol{w}) = e(\boldsymbol{v}, \boldsymbol{w})^{\delta\hat{\delta}}$ for all $\delta, \widehat{\delta} \in \mathbb{F}_q$, $\boldsymbol{v} \in \mathbb{V}_1$, and $\boldsymbol{w} \in \mathbb{V}_2$.
  - *Non-degeneracy* : If $e(\boldsymbol{v}, \boldsymbol{w}) = 1_{\mathbb{G}_T}$ for all $\boldsymbol{w} \in \mathbb{V}_2$, then $\boldsymbol{v} = (\overbrace{1_{\mathbb{G}_1}, \ldots, 1_{\mathbb{G}_1}}^{d})$. Similar statement also holds with the vectors $\boldsymbol{v}$ and $\boldsymbol{w}$ interchanged.

Dual Orthonormal Basis Generator $\mathcal{G}_{\text{OB}}(1^\lambda, N, (d_1, \ldots, d_N))$

- Generate $\text{params}_{\mathbb{G}} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\text{BPG}}(1^\lambda)$.
- Sample $\psi \xleftarrow{\mathsf{U}} \mathbb{F}_q \backslash \{0\}$ and compute $g_T = e(g_1, g_2)^\psi$.
- For $\imath \in [N]$, perform the following:
  - Generate $\text{params}_{\mathbb{V}_\imath} = (q, \mathbb{V}_{\imath,1}, \mathbb{V}_{\imath,2}, \mathbb{G}_T, \mathbb{A}_{\imath,1}, \mathbb{A}_{\imath,2}, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\text{DPVS}}(1^\lambda, d_\imath, \text{params}_{\mathbb{G}})$.
  - Sample $\boldsymbol{B}^{(\imath)} = (b^{(\imath)}_{\ell,k}) \xleftarrow{\mathsf{U}} \mathsf{GL}(d_\imath, \mathbb{F}_q)$.
  - Compute $\boldsymbol{B}^{*(\imath)} = (b^{*(\imath)}_{\ell,k}) = \psi((\boldsymbol{B}^{(\imath)})^{-1})^{\intercal}$.
  - For all $\ell \in [d_\imath]$, let $\vec{b}^{(\imath,\ell)}$ and $\vec{b}^{*(\imath,\ell)}$ be the ℓ^{th} rows of $\boldsymbol{B}^{(\imath)}$ and $\boldsymbol{B}^{*(\imath)}$.
  - Compute $\boldsymbol{b}^{(\imath,\ell)} = (\vec{b}^{(\imath,\ell)})_{\mathbb{A}_{\imath,1}}, \boldsymbol{b}^{*(\imath,\ell)} = (\vec{b}^{*(\imath,\ell)})_{\mathbb{A}_{\imath,2}}$ for $\ell \in [d_\imath]$, and set $\mathbb{B}_\imath = \{\boldsymbol{b}^{(\imath,1)}, \ldots, \boldsymbol{b}^{(\imath,d_\imath)}\}, \mathbb{B}^*_\imath = \{\boldsymbol{b}^{*(\imath,1)}, \ldots, \boldsymbol{b}^{*(\imath,d_\imath)}\}$.
  - $\mathbb{B}_\imath$ and $\mathbb{B}^*_\imath$ are dual orthonormal in the sense that for all $\ell, \ell' \in [d_\imath]$,

$$e(\boldsymbol{b}^{(\imath,\ell)}, \boldsymbol{b}^{*(\imath,\ell')}) = \begin{cases} g_T, & \text{if } \ell = \ell', \\ 1_{\mathbb{G}_T}, & \text{otherwise.} \end{cases}$$

- Set $\text{params} = (\{\text{params}_{\mathbb{V}_\imath}\}_{\imath \in [N]}, g_T)$.
- Return $(\text{params}, \{\mathbb{B}_\imath, \mathbb{B}^*_\imath\}_{\imath \in [N]})$.

PHPE.Setup$(1^\lambda, 1^{n'}, 1^n)$

- Generate $(\text{params}, \{\mathbb{B}_\imath, \mathbb{B}^*_\imath\}_{\imath\in[n'+n]}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\text{OB}}(1^\lambda, n'+n, (\overbrace{9,\ldots,9}^{n'+n}))$.
- For $\imath \in [n'+n]$, set

$$\widehat{\mathbb{B}}_\imath = \{\boldsymbol{b}^{(\imath,1)}, \boldsymbol{b}^{(\imath,2)}, \boldsymbol{b}^{(\imath,9)}\},$$
$$\widehat{\mathbb{B}}^*_\imath = \{\boldsymbol{b}^{*(\imath,1)}, \boldsymbol{b}^{*(\imath,2)}, \boldsymbol{b}^{*(\imath,7)}, \boldsymbol{b}^{*(\imath,8)}\}.$$

- Output $\text{MPK} = (\text{params}, \{\widehat{\mathbb{B}}_\imath\}_{\imath\in[n'+n]})$ and $\text{MSK} = \{\widehat{\mathbb{B}}^*_\imath\}_{\imath\in[n'+n]}$.

PHPE.Encrypt$(\text{MPK}, (\vec{x}, \vec{z}) \in \mathbb{F}_q^{n'} \times \mathbb{F}_q^{n})$

- Sample $\omega \xleftarrow{\mathsf{U}} \mathbb{F}_q$.
- For $\iota' \in [n']$, sample $\varphi'_{\iota'} \xleftarrow{\mathsf{U}} \mathbb{F}_q$, and compute

$$\boldsymbol{c}'^{(\iota')} = (\omega(1, x_{\iota'}), \vec{0}^4, \vec{0}^2, \varphi'_{\iota'})_{\mathbb{B}_{\iota'}}.$$

- For $\iota \in [n]$, sample $\varphi_{\iota} \xleftarrow{\mathsf{U}} \mathbb{F}_q$, and compute

$$\boldsymbol{c}^{(\iota)} = (\omega(1, z_{\iota}), \vec{0}^4, \vec{0}^2, \varphi_{\iota})_{\mathbb{B}_{n'+\iota}}.$$

- Output $\text{CT} = (\vec{x}, \{\boldsymbol{c}'^{(\iota')}\}_{\iota' \in [n']}, \{\boldsymbol{c}^{(\iota)}\}_{\iota \in [n]})$.

PHPE.KeyGen$(\mathrm{MPK}, \mathrm{MSK}, f \in \mathcal{F}^{(q,n',n)}_{\mathrm{ABP}\circ\mathrm{IP}})$

- Generate $\left((\{\sigma_j\}_{j\in[n]}, \{\alpha_{j'}, \gamma_{j'}\}_{j'\in[m]}), \rho : [m] \to [n']\right) \xleftarrow{\mathsf{R}} \mathsf{PGB}(f)$.
- Sample $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$.
- For $j' \in [m]$, sample $\vec{\kappa}'^{(j')} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2$, and compute

$$\boldsymbol{k}'^{(j')} = ((\gamma_{j'}, \alpha_{j'}), \vec{0}^4, \vec{\kappa}'^{(j')}, 0)_{\mathbb{B}^*_{\rho(j')}}.$$

- For $j \in [n]$, sample $\vec{\kappa}^{(j)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2$, and compute

$$\boldsymbol{k}^{(j)} = ((\sigma_j, \zeta), \vec{0}^4, \vec{\kappa}^{(j)}, 0)_{\mathbb{B}^*_{n'+j}}.$$

- Output $\mathrm{SK}(f) = (f, \{\boldsymbol{k}'^{(j')}\}_{j'\in[m]}, \{\boldsymbol{k}^{(j)}\}_{j\in[n]})$.

PHPE.Decrypt$(\mathrm{MPK}, \mathrm{SK}(f) = (f, \{\boldsymbol{k}'^{(j')}\}_{j' \in [m]}, \{\boldsymbol{k}^{(j)}\}_{j \in [n]}),$ $\mathrm{CT} = (\vec{x}, \{\boldsymbol{c}'^{(\iota')}\}_{\iota' \in [n']}, \{\boldsymbol{c}^{(\iota)}\}_{\iota \in [n]}))$

- Compute $\Lambda'_{j'} = e(\boldsymbol{c}'^{(\rho(j'))}, \boldsymbol{k}'^{(j')}) = g_T^{\omega(\alpha_{j'} x_{\rho(j')} + \gamma_{j'})}$ for $j' \in [m]$, and $\Lambda_j = e(\boldsymbol{c}^{(j)}, \boldsymbol{k}^{(j)}) = g_T^{\omega(\zeta z_j + \sigma_j)}$ for $j \in [n]$.
- Determine $(\{\Omega'_{j'}\}_{j' \in [m]}, \{\Omega_j\}_{j \in [n]}) = \mathsf{REC}(f, \vec{x})$.
- Compute $\Lambda = \Big(\prod_{j' \in [m]} \Lambda_{j'}'^{\Omega'_{j'}}\Big)\Big(\prod_{j \in [n]} \Lambda_j^{\Omega_j}\Big) = g_T^{\omega\zeta f(\vec{x}, \vec{z})}$.
- If $R^{\mathrm{ABP}\circ\mathrm{IP}}(f, (\vec{x}, \vec{z})) = 1$, i.e., $f(\vec{x}, \vec{z}) = 0$, then $\Lambda = 1_{\mathbb{G}_T}$, while if $R^{\mathrm{ABP}\circ\mathrm{IP}}(f, (\vec{x}, \vec{z})) = 0$, i.e., $f(\vec{x}, \vec{z}) \neq 0$, then $\Lambda \neq 1_{\mathbb{G}_T}$ with all but negligible probability $2/q$, i.e., except when $\omega = 0$ or $\zeta = 0$.
- Output 1, if $\Lambda = 1_{\mathbb{G}_T}$, and 0, otherwise.

Concluding Remarks and Open Problems

- We achieved SIM-based S-AH security against *adaptive* adversaries for PE schemes supporting *expressive* predicate families under standard computational assumption in bilinear groups.

- We designed a SIM-based *adaptively* strongly partially-hiding PE (PHPE) scheme for predicates computing ABP's on public attributes, followed by an IP on private attributes.

- The proposed scheme is proven secure for *any a priori bounded* number of ciphertexts and *unbounded* number of authorized decryption keys.

- An intriguing *open problem* is to identify the largest predicate class for which S-AH PE scheme supporting unbounded number of authorized decryption key queries can be realized from a standard computational assumption.

Thanking Note