Certificate-based Smooth Projective Hashing and Its Applications

Sujuan Li[1], Yi Mu[2], and Mingwu Zhang[3]
(Corresponding author: Sujuan Li)
School of Mathematical and Physical Sciences, Nanjing Tech University[1]
No.30 Puzhu Road Pukou District, Nanjing, China
(Email: lisujuan1978@126.com)
School of Computer and Information Technology, University of Wollongong Australia[2]
Northfields Ave, Wollongong, Australia
School of Computer Science, Hubei University of Technology[3]
No.28 Nanli Road Hongshan District, Wuhan, China
(Received Oct. 21, 2016; revised and accepted Feb. 1, 2017)

Abstract

Smooth projective hashing was firstly introduced by Cramer and Shoup (EuroCrypt'02) as a tool to construct efficient chosen-ciphertext-secure public key encryption schemes. Since then, they have found many other applications, such as password-based authenticated key exchange, oblivious transfer, zero-knowledge arguments *et al.* Certificate-based encryption (CBE) not only eliminates third-party queries and heavy certificate management problem in traditional public-key encryption, but also solves key escrow problem for identity-based encryption. We introduce the new concept of certificate-based smooth projective hashing (CB-SPH). Under the security model for the leakage-resilient certificate-based encryption (LR-CBE), we show how to construct a general leakage-resilient certificate-based encryption scheme using the certificate-based smooth projective hashing. Based on these theoretical constructions, we present two concrete CB-SPH instantiations under the DBDH assumption and the DLWE assumption respectively. Based on these CB-SPH instantiations, we can construct leakage resilient CBE schemes.

Keywords: Certificate-based Smooth Projective Hashing; DBDH; DLWE; Key-leakage Resilient

1 Introduction

Traditional cryptographic schemes assume that the secret keys are completely hidden from the adversaries. However side-channel attacks [21, 36] and cold boot attacks [3] indicate that the conventional attack model fails to capture some attacks in the real world. We classify these attacks as key leakage attacks in which the attackers may obtain some partial information about the secret states of the cryptosystems. To stand against such attacks, there has been a surge of interest in creating leakage resilient cryptographic schemes [3, 5, 8, 10, 13, 15, 22, 23, 26, 28, 32, 33, 35, 37]. The feature of a leakage-resilient cryptosystem is that it remains secure even when some secret internal information including the secret key is leaked to the adversary.

Smooth projective hashing (SPH) was firstly introduced by Cramer and Shoup [12]. Originally it is a tool for constructing adaptively chosen ciphertext (CCA2) secure public key encryption (PKE). Lately SPH was found that it can be applied to construct different cryptographic schemes such as password-based authenticated key exchange, oblivious transfer, zero-knowledge arguments *et al.* In Crypto'09 Naor and Segev [32] found that SPH can be applied to the leakage resilient PKE. Under the subset membership problem (SMP), they presented a general construction of leakage-resilient PKE scheme with the help of universal SPH creatively. They extended the framework of key-leakage to the setting of chosen ciphertext attacks (Akavia *et al.* [3] formalized the first framework for modeling the security of leakage-resilient PKE). Based on the Decisional Deffie-Hellman (DDH) assumption they gave two practical PKE schemes against key leakage with 1/4 and 1/6 leakage ratio respectively. Based on Naor and Segev's work [32], there are more researches paid on the leakage-resilient PKE afterwards. Nguyen *et al.* [33] explored stateless/stateful leakage-resilient public key encryption from the SPH. Kurosawa *et al.* [23] presented another general method of constructing CCA-secure PKE with key leakage, which is based on the universal$_2$ SPH. They also gave two concrete public key encryption schemes under the DCR assumption and the DLIN assumption respectively. In AsiaCrypt'13 Qin and Liu [35] presented a new general construction of PKE

scheme against leakage-resilient chosen-ciphertext attacks (LR-CCA), from any SPH and any one-time lossy filter (OT-LF).

In constructing Identity-based encryption (IBE) schemes against key leakage attacks, Alwen *et al.* [5] firstly gave the concept of ID-based smooth projective hashing (IB-SPH) and presented three IB-SPH instantiations derived from existing IBE schemes [7, 18, 19]. Based on these instantiations they also got three leakage resilient IBE schemes. With the help of the dual system encryption technology, Lewko *et al.* [24] got fully secure IBE, HIBE, and ABE systems which are resilient to bounded leakage from each of many secret keys per user, as well as many master keys. Li *et al.* [27] gave a leakage resilient IBE scheme based on the IB-SPH extracted from Coron's IBE scheme [11].

1.1 Our Motivation

Smooth projective hashing plays an important role in public key cryptosystems. Informally, a smooth projective hashing is a family of keyed hash functions. Its special construction achieves many applications such as constructing chosen-ciphertext-secure encryption, leakage-resilient encryption *et al.* From the aspect of assemble construction, many research works have been paid to the properties of the language such as conjunction or disjunction of languages [1, 6] and the properties of the smooth projective hash such as homomorphic smooth projective hashing [41] or dual projective hashing [40].

The certificate-based encryption (CBE) was firstly proposed by Gentry [17] in EuroCrypt'03. In CBE schemes, a user produces public and private keys, and applies a corresponding certificate which is given from the trusted certificate authority (CA). It is possible to obtain a plaintext only in the case that the user has the private key and certificate simultaneously. CBE not only eliminates third-party queries and heavy certificate management problem in traditional public-key encryption, but also solves key escrow problem for identity-based encryption. CBE has attracted more concern due to its advantages. Recently, many CBE schemes [16, 29, 31, 42] have been proposed. In the key leakage resilient setting, Yu *et al.* [39] proposed the first leakage resilient certificate-based public key encryption using the dual system encryption technology. But their construction is in the composite order group which costs much more than in a prime order group. Afterwards Yu *et al.* [38] proposed another leakage resilient certificate-based public key encryption under the DBDH assumption in the random oracle. Li *et al.* [25] proposed a continuous leakage resilient certificate-based encryption with the help of secret sharing technology.

Focusing on the construction of the secret key in the smooth projective hashing, we can see some useful new smooth projective hashing which can be operated under some new conditions. Yang *et al.* [37] proposed an updatable smooth projective hashing which can be used to design general constructions of public key encryption schemes against continuous key leakage. Considering the secret key is delegated by the key generate center, Alwen *et al.* [5] introduce an identity-based smooth projective hashing which can be used to design the leakgae resilient identity-based encryption schemes.

To explore the smooth projective hashing with applications in certificate-based cryptosystems and obtain the leakage resilient certificate-based public key encryption schemes in the prime order group, we propose the new notion of certificate-based smooth projective hashing, borrowing the ideas of Yang et al [37] and Alwen et al [5]. In this paper we focus on the certificate-based smooth projective hashing and its application in leakage resilient encryptions which can also be extended to the continuous leakage resilient settings.

1.2 Our Contribution

In brief, our contribution is described as follows:

1) In the certificate-based settings we firstly give a definition of generalized certificate-based smooth projective hashing (CB-SPH). In order to guarantee its security we verify the smooth and projective properties.

2) Based on the definition and security properties of CB-SPH, we show how to convert smooth CB-SPH to leakage resilient one and show how to construct leakage-resilient certificate-based encryption schemes.

3) As a concrete example, in a prime-order group we present the first practical CB-SPH construction under the DBDH assumption in the standard model. To achieve a leakage-resilient certificate-based encryption, the construction using CB-SPH tool is much more efficient and practical comparing with the construction using dual system encryption technology in the composite order group.

4) Under the decisional learning with errors assumption, we firstly present a lattice-based CB-SPH instantiation which also can be transferred into a leakage resilient certificate-based encryption.

1.3 Organization

The rest of the article is organized as follows. We review some preliminaries that are used in this article in Section 2. In Section 3, we present the general construction of the certificate-based smooth projective hashing with projection and smoothness properties. In Section 4, we introduce the security model and the generic construction for the leakage-resilient certificate-based encryption. Two concrete certificate-based smooth projective hashings based on the DBDH assumption and the DLWE assumption respectively are shown in Section 5. Lastly, we give a conclusion and future work in Section 6.

2 Preliminaries

In this section, we present some basic notions and tools that will be used in our constructions and security proofs. We formally state some decisional assumptions and present the notion of average-case strong randomness extractors.

2.1 Computational Assumptions

Let $\mathsf{BLGroupGen}$ be a PPT algorithm that takes as input a security parameter κ and output a tuple (G, G_1, g, e). Let G and G_1 be the two cyclic groups of order p for some large prime p. A map $e : G \times G \to G_1$ is a bilinear pairing. Let g be a random generator of G. The following DBDH assumption is given in $(G, G_1, g, e) \leftarrow \mathsf{BLGroupGen}(\kappa)$.

Definition 1. *(Decisional Bilinear Diffie-Hellman Assumption) We define the decisional bilinear Diffie-Hellman (DBDH) problem as: Given (G, g, g^a, g^b, g^c) and a random element $Z \in G_1$ as input, output 1 if $Z = e(g, g)^{abc}$ and output 0 otherwise. We say that the (t, ϵ)-DBDH assumption holds if no t-time algorithm has advantage at least ϵ in solving the DBDH problem.*

Definition 2. *(Decisional Learning With Errors Assumption) We define the decisional learning with errors (DLWE) problem as: For an integer $p \geq 2$ and some probability distribution χ over Z_p, an integer dimension $n \in Z^+$, and a vector $\boldsymbol{s} \in Z_p^n$, define $A_{s,\chi}$ as the distribution over $Z_p^n \times Z_p$ of the variable $(\boldsymbol{a}, \boldsymbol{a}^T \boldsymbol{s} + t)$ where $\boldsymbol{a} \in_R Z_p^n$ and $t \leftarrow \chi$. The decisional learning with errors (DLWE) assumption holds if the distribution $A_{s,\chi}$ and the uniform distribution over $Z_p^n \times Z_p$ are computationally indistinguishable, where $\boldsymbol{s} \in_R Z_p$.*

2.2 Random Extractor

The statistical distance between two random variables X and Y over a finite domain Ω is $\mathrm{SD}(X,Y) = \frac{1}{2} \sum_{\omega \in \Omega} |\Pr[X = \omega] - \Pr[Y = \omega]|$. We write $X \approx_\epsilon Y$ to denote $\mathrm{SD}(X,Y) \leq \epsilon$, and $X \approx Y$ to denote that the statistical distance is negligible. The min-entropy of a random variable X is $H_\infty(X) = -\log(max_x \Pr[X = x])$.

We use the notion of average min-entropy [14] which captures the remaining unpredictability of a random variable X conditioned on another random variable Y, formally defined as:

$$\widetilde{H}_\infty(X|Y) = -\log(E_{y \in Y}[2^{-H_\infty(X|Y=y)}])$$

where $E_{y \in Y}$ denotes the expected value over all values of Y .

Lemma 1. *[14] For any random variables X, Y, Z, if Y has 2^r possible values, then*

$$\widetilde{H}_\infty(X|(Y,Z)) \geq \widetilde{H}_\infty(X|Z) - r.$$

Specially,

$$\widetilde{H}_\infty(X|Y) \geq H_\infty(X) - r.$$

Definition 3. *[14] A function Ext:$\{0,1\}^u \times \{0,1\}^t \to \{0,1\}^v$ is an average-case (m, ϵ)-strong extractor if for all pairs of random variables (X, Z) such that $X \in \{0,1\}^u$ and $\widetilde{H}_\infty(X|Z) \geq m$ it holds that*

$$SD((\mathsf{Ext}(X, R), R, Z), (U_v, R, Z)) \leq \epsilon.$$

where R is uniform in $\{0,1\}^t$.

The definition of universal hashing [5] and the leftover-hash lemma [34] are given as follows.

Definition 4. *(ρ-Universal Hashing) A family $\mathcal{H}$, consisting of deterministic functions $h : \{0,1\}^u \to \{0,1\}^v$, is ρ-universal hash family if for any $m_1 \neq m_2 \in \{0,1\}^u$ we have $Pr_{h \leftarrow \mathcal{H}}[h(m_1) = h(m_2)] \leq \rho$.*

Lemma 2. *(Leftover-Hash Lemma [34]) Assume that the family $\mathcal{H}$ of functions $h : \{0,1\}^u \to \{0,1\}^v$ is a ρ-universal hash family. Then the randomized extractor $\mathsf{Ext}(x, r) = h(x)$, where h is uniform over $\mathcal{H}$, is an (m, ϵ)-extractor as long as $m \geq v + 2\log(1/\epsilon)$ and $\rho \leq \frac{1}{2^v}(1 + \epsilon^2)$.*

This lemma implies that the universal hash functions are also good extractors.

3 Certificate-based Smooth Projective Hashing (CB-SPH)

As we have discussed in the introduction, smooth projective hashing serves as a good framework to unify many PKE schemes based on decisional assumptions. Before we introduce the new notion of CB-SPH, we firstly recall the basic conception about SPH.

3.1 Smooth Projective Hashing

The Smooth projective hashing (SPH) consists of two ingredients, namely the subset membership problem (SMP), which will be extended to the distribution distinguishable problem (DDP) for including the lattice-based hardness assumption, and the projective hash function (PHF).

3.1.1 Subset Membership Problem

The SMP defines a set X and a language $L \subset X$, from which a member x can be efficiently sampled with a witness w. We give the formal definition of SMP in [12].

Definition 5. *(Subset Membership Problem). A subset membership problem S specifies a collection $(S_\kappa)_{\kappa \geq 0}$ of distributions. For every value of a security parameter $\kappa \geq 0$, S_κ is a probability distribution over instance descriptions.*

An instance description $\Lambda = (X, W, PK, L, R)$ specifies the following:

- Finite non-empty sets X, W, PK, and two collections of distributions $L = (L_{pk})_{pk \in PK}$ and $X \setminus L = (X \setminus L_{pk})_{pk \in PK}$ over X.
- A collection of binary relations $R = (R_{pk})_{pk \in PK}$ defined over $X \times W$. For $x \in X$ and $w \in W$ and some $pk \in PK$ such that $x, w \in R_{pk}$, we say that w is a witness for x.

A subset membership problem is hard if it is computationally impossible to distinguish random members $x \in L$ from random non-members $x \in X \setminus L$.

3.1.2 Distribution Distinguishable Problem

To *include lattice-based PKE scheme* which implement the smooth projective hashing technique, we borrow the definition of distribution distinguishable problem (DDP) in [9] which relaxes some restrictions (More details refer to [9]).

Definition 6. *(Distribution Distinguishable Problem). A distribution distinguishable problem* D *specifies a collection* $(D_\kappa)_{\kappa \geq 0}$ *of distributions. For every value of a security parameter* $\kappa \geq 0$, D_κ *is a probability distribution over instance descriptions.*

An instance description $\Gamma = (X, W, PK, A, B, R)$ specifies the following:

- Finite non-empty sets X, W, PK, and two collections of distributions $A = (A_{pk})_{pk \in PK}$ and $B = (B_{pk})_{pk \in PK}$ over X where $X = A \cup B$.
- A collection of binary relations $R = (R_{pk})_{pk \in PK}$ defined over $X \times W$. For $x \in X$ and $w \in W$ and some $pk \in PK$ such that $x, w \in R_{pk}$, we say that w is a witness for x.

$\Gamma = (X, W, PK, A, B, R)$ indicates that the instance Γ specifies X, W, PK, A, B and R. D provides the three following algorithms:

SampDDP(κ): Input a security parameter κ, and output the public and secret key pair (pk, sk) and an instance description Γ according to the distribution D_κ.

SampA(pk): Output $x \leftarrow A_{pk}$ along with a witness $w \in W$ such that $(x, w) \in R_{pk}$. This is sampling with witness algorithm.

SampB(pk): Output $x \leftarrow B_{pk}$. This is sampling without the witness algorithm.

It is only requiring that algorithms $\mathsf{SampDDP}$ and SampA should be efficient. A distribution problem D is said to be hard if A_{pk} and B_{pk} are computationally indistinguishable for any probability polynomial-time adversary.

3.1.3 Projective Hash Function

The PHF with projection $\alpha : SK \to PK$ is a family of hash functions H indexed by SK with domain X in SMP or domain A in DDP. For we will give a lattice-based SPH in the rest of this paper, we mainly discuss in the range of DDP.

Definition 7. *(Projective Hash Function). Let* X, Y, SK, PK *be finite non-empty sets, and* A_{pk} *be a collection of distributions indexed by* PK. *Here* X, PK, A_{pk} *are defined as in DDP above. Let* $H = \{H_{sk} : X \to Y\}_{sk \in SK}$ *be a family of functions indexed by* SK. *Let* $\alpha : SK \to PK$ *be a projection from* SK *to* PK. *We say* $\mathsf{H} = (H, SK, PK, X, A_{pk}, Y, \alpha)$ *a projective hash function if for any* $sk \in SK$ *and* $pk = \alpha(sk)$, *the action of* H_{sk} *on* $x \leftarrow A_{pk}$ *is approximately determined by* $\alpha(sk)$.

3.1.4 Generalized Smooth Projective Hashing

A generalized smooth projective hashing which encompasses the lattice-based smooth projective hashing technology combines DDP D with PHF H as following four algorithms:

Setup(κ): Run $\mathsf{SampDDP}(\kappa)$ to generate a master public/secret key pair (mpk, msk) and an instance description $\Gamma = (X, W, PK, A, B, R)$ of D, pich a projective hash function $\mathsf{H} = (H, SK, PK, X, A_{pk}, Y, \alpha)$. mpk will be used in the following algorithms.

KeyGen(κ): Pick $sk \leftarrow_R SK$, compute $pk \leftarrow \alpha(sk)$. Output the public/secret key pair (pk, sk).

Priv(sk, x): Take as input a private key sk and $x \in X$, and output $y \in Y$ such that $y = H_{sk}(x)$. It is the private evaluation algorithm.

Pub(pk, x, w): Take as input pk and $x \in A_{pk}$ with a witness $w \in W$, and output $y \in Y$. It is the public evaluation algorithm.

In the rest of this paper we still call the generalized smooth projective hashing smooth projective hashing (SPH) for short.

3.2 Certificate-based Smooth Projective Hashing

In this part we will introduce the formal definition of certificate-based smooth projective hashing (CB-SPH) and define some properties for CB-SPH. Firstly we describe the DDP and PHF in the certificate-based settings.

3.2.1 Distribution Distinguishable Problem

We only extend the algorithm $\mathsf{SampDDP}$ as below.

SampDDP(κ): Input a security parameter κ, and output a master public and secret key pair (mpk, msk) and an instance description Γ according to the distribution D_κ.

Therefore, the instance description Γ needs to be attached by the master public key set MPK. We get $\Gamma = (X, W, MPK, PK, A, B, R)$.

3.2.2 Projective Hash Function

The definition of projective hash function are the same as that of SPH besides H is added the master public key set MPK and the user's identity set ID i.e. $\mathsf{H} = (H, SK, MPK, ID, PK, X, A_{pk}, Y, \alpha)$.

3.2.3 Certificate-based Smooth Projective Hashing

A certificate-based smooth projective hashing (CB-SPH) P combines DDP D with PHF H in certificate-based settings. It also includes the five algorithms as follow:

Setup(κ): The authenticated center (CA) runs $\mathsf{SampDDP}(\kappa)$ to generate a master public/secret key pair (mpk, msk). The following algorithms all take mpk as input.

UserKeyGen(id): The user takes as input the master public key mpk and the identity id. It outputs the user's private key $sk_{id} = \sigma_1(id)$. Then using the master public key mpk, the identity id and the user private key sk_{id}. It outputs the user's public key $pk_{id} = \alpha(sk_{id})$.

CerGen(msk, id): For an identity id, the CA first calculates $H(id, pk_{id}) = id'$. Then, it takes as input the master public key mpk, the master secret key msk, id' and the public pk_{id}. It outputs the user?s certificate $Cert_{id} = \sigma_2(id, msk)$.

Pub(pk, id, x, w): It takes as input $id \in ID$, $x \in A_{pk}$ and a witness $w \in W$ for x, and output $y \in Y$. It is the public evaluation algorithm.

Priv($x, sk_{id}, Cert_{id},$): It takes as input user's private key sk_{id}, user's certificate $Cert_{id}$ and $x \in X$, and output $y \in Y$ such that $y = H_{sk_{id}, Cert_{id}}(x)$. It is the private evaluation algorithm.

In the CB-SPH structure, we extend the KeyGen algorithm according to the certificate-based requirements. The user's decryption key is divided into two parts. One is the user's private key generated by the user himself. And another one is the user's certificate produced by the CA. Under such extension, the private key also need to keep the projective connection with the public key.

We require a certificate-based smooth projective hashing to satisfy the following properties.

1) **Soundness**.
For any $id \in ID$, the user's private key sk_{id}, the user's certificate $Cert_{id}$, the (master) public key mpk, pk and $x \leftarrow A_{pk}$, we have

$$\text{Priv}(x, sk_{id}, Cert_{id}) = \text{Pub}(id, pk, x, w).$$

2) **Indistinguishability**.
We define the following indistinguishable game which is called IND game for short. The interactive game between the adversary $\mathcal{A}$ and the challenger $\mathcal{B}$ is described as follows.

Setup: The challenger $\mathcal{B}$ runs this algorithm to generate the master public key mpk and master secret key msk respectively. $\mathcal{B}$ sends mpk and *even* msk to $\mathcal{A}$.
Phase 1: $\mathcal{A}$ maintains two lists: L_{key}, L_{Cert} and performs the following queries in an adaptive fashion in this phase.

- Private Key queries: $\mathcal{A}$ produces an identity id and requests the corresponding private key sk_{id}. If the item for identity id does not exist in the list L_{Key}, $\mathcal{B}$ runs the $\mathsf{UserKeyGen}$ algorithm to generate the user's private key $sk_{id} = \sigma_1(mpk, id)$.
- Certificate queries: The adversary $\mathcal{A}$ asks the certificate $Cert_{id}$ for the identity id. If the item for identity id does not exist in the list L_{Cert}, B runs the UserKeyGen algorithm to generate where $Cert_{id} = \sigma_2(mpk, id, msk)$

Challenge Stage: The adversary $\mathcal{A}$ selects an arbitrary challenge identity $id^* \in ID$ randomly and possibly one for which it has seen the private key sk_{id^*} and the certificate $Cert_{id^*}$. The challenger $\mathcal{B}$ chooses $\beta \leftarrow \{0, 1\}$ randomly.
If $\beta = 1$, the challenger computers $(x, w) \leftarrow \mathsf{SampA}(mpk, id^*, pk_{id^*})$.
If $\beta = 0$, the challenger computers $x \leftarrow \mathsf{SampB}(mpk, id^*, pk_{id^*})$.
The challenger gives x to the adversary $\mathcal{A}$.
Phase 2: $\mathcal{A}$ makes a sequence of queries with $id \in ID$ adaptively as in phase 1.
Output: The adversary $\mathcal{A}$ output a bit $\beta' \in \{0, 1\}$. We say that $\mathcal{A}$ wins the game if $\beta' = \beta$.

Note that, during the **setup** phase, the challenger $\mathcal{B}$ sends the master key msk to the attacker $\mathcal{A}$ because $\mathcal{A}$ can even know the private key and the user's certificate in the following stage. We define the advantage of $\mathcal{A}$ in distinguishing honest/dishonest ciphertexts to be

$$\text{Adv}^{IND}_{CB-SPH,\mathcal{A}}(\kappa) = |\Pr[\mathcal{A} \text{ wins}] - \tfrac{1}{2}|.$$

Definition 8. *(Indistinguishability) A CB-SPH satisfies the indistinguishability property if no polynomially-time adversary $\mathcal{A}$ has a non-negligible advantage in the above game.*

3) **Projection**
A CB-SPH is projective if for any $id \in ID$,

$$\Pr[\text{Pub}(id, pk, x, w) \neq \text{Priv}(x, sk_{id}, Cert_{id})] \leq \text{negl}(\kappa)$$

where w is a witness for x and $sk_{id}, Cert_{id}$ is the user private key and certificate respectively. The probability is taken over the choice of $x \leftarrow A_{pk}$.

4) **Smoothness/Leakage-Smoothness**.
The following two properties are mainly ensuring that there are many possibilities for $\mathsf{Pub}(x, \cdot)$ of an dishonest ciphertext $x \leftarrow B_{pk}$, which are left undetermined by the public parameters of the system.

Definition 9. *(ϵ-Smooth CB-SPH) We say that a CB-SPH is ϵ-smooth if for any $id \in ID$*

$$\boldsymbol{SD}((R, id, x, y), (R, id, x, y')) \leq \epsilon,$$

where R is the ensemble of (mpk, msk), $x \leftarrow B_{pk}, y \leftarrow Priv(x, sk_{id}, Cert_{id})$ and $y' \leftarrow U_Y$.

Definition 10. *(l-Leakage-Resilient ϵ-Smooth CB-SPH) We say that a CB-SPH is l-leakage-resilient ϵ-smooth if for any fixed parameters produced by the above algorithms of CB-SPH and any possibly randomized function $f(\cdot)$ with l-bit output, we have:*

$$\boldsymbol{SD}((R, id, f(sk_{id}, Cert_{id}), x, y),$$

$$(R, id, c, f(sk_{id}, Cert_{id}), x, y')) \leq \epsilon$$

where R is the ensemble of (mpk, msk), $x \leftarrow B_{pk}, y \leftarrow Priv(x, sk_{id}, Cert_{id})$ and $y' \leftarrow U_Y$.

3.3 Generic Construction of Leakage-Resilient CB-SPH

We now show how to convert a CB-SPH (Setup, $\mathsf{UserKeyGen}$, CerGen, Priv, Pub) into a leakage-resilient one using an average-case randomness extractor Ext: $Y \times S \rightarrow \{0,1\}^v$ with seeds set $S = \{0,1\}^\mu$. We modified the algorithms SampA and SampB of DDP D as follows:

- $\overline{\mathsf{SampA}}(pk)$: sample $(x, w) \leftarrow \mathrm{SampA}(pk)$, pick a seed $d \leftarrow_R \{0,1\}^\mu$, and output $\bar{x} = (x, w, d)$.
- $\overline{\mathsf{SampB}}(pk)$: sample $x \leftarrow \mathrm{SampB}(pk)$, pick a seed $d \leftarrow_R \{0,1\}^\mu$, and output $\bar{x} = (x, d)$.

We keep the algorithms Setup, $\mathsf{UserKeyGen}$ and CerGen unchanged, define:

- $\overline{\mathsf{Priv}}(\bar{x}, sk_{id}, Cert_{id})$: parse $\bar{x}$ as (x, w, d), compute $y \leftarrow \mathsf{Priv}(x, sk_{id}, Cert_{id})$, and output $\bar{y} \leftarrow \mathsf{Ext}(y, d)$.
- $\overline{\mathsf{Pub}}(id, pk, \bar{x}, w)$: parse $\bar{x}$ as (x, w, d), compute $y \leftarrow \mathsf{Pub}(id, pk, x, w)$, and output $\bar{y} \leftarrow \mathsf{Ext}(y, d)$.

We will show a theorem which instruct that the transformed CB-SPH (Setup, $\mathsf{UserKeyGen}$, CerGen, $\overline{\mathsf{Priv}}$, $\overline{\mathsf{Pub}}$) is leakage-resilient smooth for some parameters.

Theorem 1. *Given an ϵ-smooth CB-SPH, let Ext: $Y \times S \rightarrow \{0,1\}^v$ be a average-case $(\log|Y| - l, \epsilon_{ext})$-extractor where S is a seeds set $\{0,1\}^\mu$, then the above transformation produces an l-leakage $(\epsilon + \epsilon_{ext})$-smooth CB-SPH.*

Proof. For a ϵ-smooth CB-SPH, we have

$$\mathbf{SD}((R, id, x, y), (R, id, x, y')) \leq \epsilon,$$

where R is the ensemble of (mpk, msk), $x \leftarrow B_{pk}, y \leftarrow \mathrm{Priv}(x, sk_{id}, Cert_{id},)$ and $y' \leftarrow U_Y$. It implies that $\widetilde{H}_\infty(y|R, id, x) \approx \log|Y|$. In the presence of leakage, an adversary has access to at most l bits of leakage from the private key sk_{id} and the certificate $Cert_{id}$. Based on Lemma 1, $\widetilde{H}_\infty(y|(R, f(sk_{id}, Cert_{id}), id, x)) \geq \widetilde{H}_\infty(y|(R, id, x)) - l = \log|Y| - l$.

According to the definition of a $(\log|Y| - l, \epsilon_{ext})$ randomness extractor, we have

$$SD((R, id, f(dk_{id}), x, \bar{y}), (R, id, f(dk_{id}), x, \bar{y'})) \leq \epsilon + \epsilon_{ext}$$

where dk_{id} is the ensemble of $(sk_{id}, Cert_{id})$.

For $\bar{x} = (x, s)$ where s is chosen independently from $\{0,1\}^\mu$,

$$SD((R, id, f(dk_{id}), \bar{x}, \bar{y}), (R, id, f(dk_{id}), \bar{x}, \bar{y'})) \leq \epsilon + \epsilon_{ext}$$

where dk_{id} is the ensemble of $(sk_{id}, Cert_{id})$. So the transformed CB-SPH is l-leakage $(\epsilon + \epsilon_{ext})$-smooth. □

4 Leakage-Resilient Certificate-based Encryption

4.1 Definition

A certificate-based public key encryption scheme Π is defined by five algorithms [17]: Setup, CerGen, $\mathsf{UserKeyGen}$, $\mathsf{Encrypt}$ and $\mathsf{Decrypt}$. Given M the message space, based on the structure of the CB-SPH, the description of the leakage resilient certificate-based Encryption is as follows.

- The first three algorithms are the same as the Setup, $\mathsf{UserKeyGen}$, CerGen algorithms in the CB-SPH. The following algorithms all take mpk as input.
- $\mathsf{Encrypt}(id, pk, m)$: Taking as input a message $m \in M$, a message sender runs this algorithm and return a ciphertext c.
- $\mathsf{Decrypt}(c, sk_{id}, Cert_{id})$: Taking as input the ciphertext c, the user runs this algorithm to return a message m using the user's private key sk_{id} and the certificate $Cert_{id}$.

Soundness of Decryption
For any $id \in ID$, any $m \in M$ and any other parameters produced by the above algorithms, we have

$$\Pr[m \neq m' \mid c \leftarrow \mathsf{Encrypt}(id, pk, m), m' \leftarrow \mathsf{Decrypt}(c, sk_{id}, Cert_{id})] \leq negl(\kappa).$$

4.2 Semantic Security with Key Leakage

As widely known, there are two types of adversaries with different capabilities in CBE, called Type I and Type II respectively.

Type I Adversary: This type of adversary $\mathcal{A}_I$ simulates the uncertified user. Such adversary has the ability to substitute a public key for any user and learn at most $l(l \in N)$ bits for leaked secret information for the cryptographic primitive, but has no access to the master secret.

Type II Adversary: This type of adversary $\mathcal{A}_{II}$ acts an honest-but-curious certifier with the master key. Such adversary has the ability to obtain a certificate of every user and learn at most $l(l \in N)$ bits for leaked secret information for the cryptographic primitive, but is prohibited to replace any user's public keys.

There are also two types of adversaries in the leakage-resilient CBE. Here, we give the semantic security model of the leakage-resilient CBE. We define the semantic security game parameterized by the security parameter κ and a leakage parameter l.

Refer to the security model of [38], we present a LR-CBE security model. This model is described via IND-LR-CPA Game. We consider the security based on the game against leakage-resilient and adaptive chosen plaintext attacks (IND-LR-CPA).

IND-LR-CPA Game: The following is the interactive game between any probabilistic polynomial-time l-key-leakage adversary $\mathcal{A}$ of Type I or Type II and a challenger $\mathcal{B}$.

Setup: The challenger $\mathcal{B}$ takes as input a security parameter 1^κ and implements algorithm $\mathsf{Setup}(1^\kappa)$. It keeps master key msk secret and returns mpk to the attacker $\mathcal{A}$.

Phase 1: $\mathcal{A}$ makes queries adaptively, $\mathcal{B}$ handles as follows:

- Certificate queries $\mathsf{Cer}(id)$: (For Type I attacker only) $\mathcal{A}$ chooses an identity id and gives it to $\mathcal{B}$. $\mathcal{B}$ computes the corresponding certificate
$$Cert_{id} = \sigma_2(mpk, msk, id)$$
and sends it to $\mathcal{A}$.
- Private Key Extraction queries $\mathsf{PrK}(id)$: $\mathcal{A}$ produces an identity id and requests the corresponding private key sk_{id}. If the user id's public key has not been replaced then $\mathcal{B}$ responds with the user private key $sk_{id} = \sigma_1(mpk, id)$. If $\mathcal{A}$ has already replaced the user id's public key, then $\mathcal{B}$ does not provide the corresponding private key to $\mathcal{A}$.
- Request Public Key queries $\mathsf{PK}(id)$: $\mathcal{A}$ produces an identity id to $\mathcal{B}$ and requests id's public key. $\mathcal{B}$ responds by returning the public key pk for the user id computing $sk_{id} = \sigma_1(mpk, id)$ and $pk = \alpha(mpk, id, sk_{id})$.
- Public Key Replacement $\mathsf{PKR}(id)$: (For Type I attacker only)$\mathcal{A}$ can repeatedly replace the public key pk the corresponding to the user identity id with any value pk' of $\mathcal{A}$'s choice.
- Leakage queries $\mathsf{LK}(id, \mathsf{Leakage}(dk_{id})$: $\mathcal{A}$ produces an identity id and requests the corresponding the leakage information of its decryption key dk_{id} where $dk_{id} = (sk_{id}, Cert_{id})$. For any randomized function $f(\cdot)$ with l-bit output, $\mathcal{B}$ returns $f(dk_{id})$. The only restriction is that all of the leakage information about dk_{id} is l bits. For the details it is divided into two aspects as follows.
  For Type I attacker $\mathcal{A}_I$, due to his ability of replacing the public key, can get the correlated leakage information $Cert'_{id}$ about the user's certificate $Cert_{id}$ besides knowing the private key sk_{id} where the length amount of $Cert'_{id}$ and sk_{id} is at most l-bit.
  For Type II attacker $\mathcal{A}_{II}$, who has the master secret key msk, can get the leakage information sk'_{id} of the secret value sk_{id} besides knowing the certificate $Cert_{id}$ where the length amount of sk'_{id} and $Cert_{id}$ is at most l bits.

Challenge Stage: The adversary $\mathcal{A}$ selects an arbitrary challenge identity $id^* \in ID$ which appeared in at most l-bit leakage query. $\mathcal{A}$ also selects two equal-length messages $m_0, m_1 \in M$. The challenger $\mathcal{B}$ chooses $\beta \leftarrow \{0,1\}$ randomly, computes $c \leftarrow \text{Encrypt}(id^*, pk, m_b)$ and sends it to the adversary $\mathcal{A}$.

Phase 2: $\mathcal{A}$ adaptively makes a new sequence of queries with $id \neq id^* \in ID$ adaptively as in phase 1 except that the adversary cannot perform the leakage query.

Output: The adversary $\mathcal{A}$ outputs a bit $\beta' \in \{0,1\}$. We say that $\mathcal{A}$ wins the game if $\beta' = \beta$.

We define the advantage of $\mathcal{A}$ in the semantic security game with l-bit key-leakage to be

$$\text{Adv}_{\mathcal{A},\Pi}^{LR-CPA}(\kappa, l) = |\Pr[\mathcal{A} \text{ wins}] - \tfrac{1}{2}|.$$

Definition 11. *(Leakage-resilient CBE) A CBE scheme is l-leakage-resilient if*

1) *It satisfies the soundness of decryption;*
2) *The advantage of any PPT adversary $\mathcal{A}$ in the semantic security game with l-bit key leakage is*
$$\text{Adv}_{\Pi,\mathcal{A}}^{LR-CPA}(\kappa, l) = negl(\kappa);$$
3) *The relative leakage ratio of the scheme is defined to be $\alpha = \frac{l}{|dk_{id}|}$ where $|dk_{id}|$ denotes the bit size of the full decryption key sk_{id} and $Cert_{id}$.*

4.3 Construction of Leakage-Resilient CBE

It is almost natural that constructing leakage resilient CBE from l-leakage smooth CB-SPH (Setup, CerGen, UserKeyGen, $\overline{\mathsf{Priv}}$, $\overline{\mathsf{Pub}}$).

Given an l-leakage smooth CB-SPH, we can construct an l-leakage resilient CBE with an identity set ID and a message set M by using the hashing value as a one-time-pad to encrypt a message directly. Recall that a CBE scheme consists of PPT algorithms (Setup, CerGen, UserKeyGen, Encrypt, Decrypt). The syntax of the first three steps is the same as that in the leakage smooth CB-SPH, and Encrypt, Decrypt have the following syntax:

- $\mathsf{Encrypt}(pk, id, m)$: Compute $(x, w, d) \leftarrow \overline{\mathsf{SampA}}(pk)$, $y \leftarrow \overline{\mathsf{Pub}}(pk, id, x, w)$ and set $z = y \oplus m$. Output $c = (x, z)$.
- $\mathsf{Decrypt}(c, sk_{id}, Cert_{id})$: Parse $c = (x, z)$. Compute $y \leftarrow \overline{\mathsf{Priv}}(x, sk_{id}, Cert_{id})$. Output $m = z \oplus y$.

Theorem 2. *Given an l-leakage smooth CB-SPH, then the above construction comes to an l-leakage resilient CBE.*

Proof. For the security analysis, we proceed via a sequence of indistinguishable games. We start with Game 0 as in the real experiment and end up with a game where the view of $\mathcal{A}$ is statistically independent of the challenge bit b:

Game 0: The first game is the semantic security game with l-bit key-leakage. In the challenge stage of Game 0, the adversary selects two length-equal messages $m_0, m_1 \in M$ and a challenge identity $id^* \in ID$, the challenger chooses $b \leftarrow \{0, 1\}$ and computes $c \leftarrow \mathsf{Encrypt}(id^*, m_b)$ which we parse as $c = (x, z)$ where

$$(x, w, d) \leftarrow \overline{\mathsf{SampA}}(pk),\ y \leftarrow \overline{\mathsf{Pub}}(id^*, x, w) \text{ and set } z = y \oplus m_b.$$

Game 1: We modify the challenge stage of Game 0 as following:

$$(x, w, d) \leftarrow \overline{\mathsf{SampA}}(pk),\ \tilde{y} \leftarrow \overline{\mathsf{Priv}}(x, sk_{id^*}, Cert_{id^*}) \text{ and set } \tilde{z} = \tilde{y} \oplus m_b.$$

For the *projective property* of CB-SPH, we have $\tilde{y} = y$ with non-negligible probability. We claim that Game 0 and Game 1 are statistically indistinguishable.

Game 2: In the challenge stage of Game 2, we modify the challenge process by using a dishonest encapsulation algorithm to compute the ciphertext $c = (x, z)$ where

$$x \leftarrow \overline{\mathsf{SampB}}(pk), \tilde{y} \leftarrow \overline{\mathsf{Priv}}(x, sk_{id^*}, Cert_{id^*}), \tilde{z} = \tilde{y} \oplus m_b.$$

Game 1 and Game 2 are computationally indistinguishable due to *the hardness of the SMP* of the CB-SPH.

Game 3: The challenge ciphertext $c = (x, z)$ is computed by

$$x \leftarrow \overline{\mathsf{SampB}}(pk), \tilde{z} \leftarrow U_Y.$$

We claim that Game 2 and Game 3 are statistically indistinguishable due to the *smooth property* of the CB-SPH.

In general, Game 0 and Game 3 are indistinguishable for any PPT adversary. Obviously, the advantage of any adversary in Game 3 is negligible in κ. Therefore, the advantage of any PPT adversary in Game 0 is negligible in κ. □

5 Instantiations of CB-SPH

In this section, we present two instantiations of CB-SPH from the standard DBDH assumption and the DLWE assumption respectively.

5.1 CB-SPH Based on the DBDH Assumption

Briefly we recall the instance description DDP on the DBDH assumption which will be embedded into the concrete CB-SPH instantiation.

5.1.1 DDP Based on the DBDH Assumption

Let D be a distribution distinguish problem based on the DBDH assumption. It includes:

SampDDP(κ): Run the bilinear group algorithm $\mathsf{BLGroupGen}(\kappa)$ to generate $\mathrm{PP} = (e, p, G, G_1)$ where G, G_1 are both p-prime order groups and $e : G \times G \to G_1$, choose $g, g_2, h \leftarrow_R G$, sets $pk = (e, G, G_1, g, g_2, h)$; Outputs an instance description $\Gamma = (X, W, PK, A_{pk}, B_{pk}, R_{pk})$ of D where $X = G \times G_1, W = Z_p, R_{pk} = \{(x, t) \in X \times W : ((g^t, e(g_2, h)^t), t)\}$, two collections of distributions A_{pk} and B_{pk} are defined by SampA and SampB as follow:

SampA(pk): Pick $t \leftarrow_R Z_p^*$, output $x = (g^t, e(g_2, h)^t) \leftarrow A_{pk}$ and $t \in W$.

SampB(pk): Pick $t, t' \leftarrow_R Z_p^*$, output $x = (g^t, e(g_2, h)^{t'}) \leftarrow B_{pk}$.

In the certificate-based settings we need some more public parameters, just like the master public key mpk, the user's identity id and the public key pk. The extensive details are described as below.

SampDDP(κ): Run bilinear group algorithm $\mathsf{BLGroupGen}(\kappa)$ to generate $PP = (e, p, G, G_1)$. Choose $g, g_1, g_2, h \leftarrow_R G$, where $g_1 = g^a$ $(a \in Z_p^*)$, sets $mpk = (e, G, G_1, g, g_1, g_2, h), msk = a$; Outputs an instance description

$$\Gamma = (X, W, MPK, ID, PK, A_{pk}, B_{pk}, R_{pk})$$

of D where $X = (G \times G_1)^2, W = (Z_p)^2, R_{pk} = \{(x,(s,t)) \in X \times W : ((g_1^s g^{-s \cdot id}, e(g,g)^s, g^t, e(g_2,h)^t), w = (s,t))\}$.

SampA(mpk, id, pk): Pick $s, t \leftarrow_R Z_p^*(s \neq t)$, and output

$$x = (g_1^s g^{-s \cdot id}, e(g,g)^s, g^t, e(g_2,h)^t) \leftarrow A_{pk}$$

and $(s,t) \in W$.

SampB(mpk, id, pk): Pick $s, t, t' \leftarrow_R Z_p^*(t \neq t')$, and output

$$x = (g_1^s g^{-s \cdot id}, e(g,g)^s, g^t, e(g_2,h)^{t'}) \leftarrow B_{pk}.$$

5.1.2 Projective Hash Function

Let $\mathsf{H} = (H, SK, CERT, PK, X, A_{pk}, Y, \alpha)$ be a corresponding projective hash function, where $SK = Z_p^2, CERT = Z_p \times G, Y = G_1$. For $sk_{id} = (x,y), Cert_{id} = (Cert_1, Cert_2)$ and $x = (c_0, c_1, c_2, c_3)$, H is defined as $H_{sk_{id}, Cert_{id}}(x) = e(c_0, Cert_2) c_1^{Cert_1} e(c_2,h)^y c_3^x$.

5.1.3 DBDH-based CB-SPH

Let P be a CB-SPH for D asscociating H, which includes the following algorithms:

Setup(κ): The CA runs $\mathsf{SampDDP}(\kappa)$ to generate the master public keys $mpk = (e, p, G, G_1, g, g_1, g_2, h)$ where $g_1 = g^a \in G$ for a random number $a \in Z_p^*$. mpk will be used in the following algorithms. The master secret key is $msk = a$.

CerGen(id, msk): The CA picks $Cert_1 \in Z_p^*$ randomly and computes

$$Cert_2 = (hg^{-Cert_1})^{\frac{1}{a - id}}.$$

Then the CA returns $Cert_{id} = (Cert_1, Cert_2)$ as the user's certificate and send it to the user.

UserKeyGen(id): The user picks $x, y \in Z_p^*$ at random and gets the user private key $sk_{id} = (x,y)$. Then the corresponding public key is $pk = \alpha(sk_{id}) = g_2{}^x g^y$.

Pub(id, pk, x, w): Choose $x = (c_0, c_1, c_2, c_3) \leftarrow \mathsf{SampA}(mpk, id, pk)$ and $w = (s,t) \in W$ where $s, t \in Z_p^*$ and $s \neq t$ and compute

$$y = e(g,h)^s \cdot e(h, pk)^t.$$

Priv($x, sk_{id}, Cert_{id}$): According to $x = (c_0, c_1, c_2, c_3) \leftarrow \mathsf{SampA}(mpk, id, pk)$, with the help of the user's private key $sk_{id} = (x,y)$ and certificate $Cert_{id} = (Cert_1, Cert_2)$, compute

$$y = e(c_0, Cert_2) c_1^{Cert_1} e(c_2,h)^y c_3^x.$$

5.1.4 Remark

Our proposed CB-SPH can be transferred to a LR-CBE scheme through the way explained in Section 4. In a CBE scheme, a Type I attacker is allowed to know the private but no information of the certificate of the target identity, and a Type II attacker only knows the user's certificate of the target identity without any information about that user's private key. From our construction of CB-SPH and the use of efficient random extractors, we allow a Type I attacker finds out some secret information of the certificate or a Type II attacker gets some sensitive information of the user's private key. Even armed with such additional leakage information, the adversaries still get negligible advantage in attacking our proposed LR-CBE scheme. Obviously, for any type of adversary, the length of the relative key-leakage of our proposed LR-CBE instantiation is at most $3 \log p (= 2 \log p + \log p)$ by Lemma 2. Thus the relative leakage ratio of the LR-CBE scheme reaches $\frac{3}{4}$ maximally due to $|sk_{id}| + |Cert_{id}| = 4 \log p$.

On the other hand, based on this concrete instantiation we can present a CCA-secure leakage resilient CBE scheme. As we all know, it is usually performed by applying a suitable authentication with encryption. Borrowing the idea of [35], the scheme from the CB-SPH plus the one-time lossy filter (OT-LF) can achieve both the CCA-secure and the best leakage rate.

5.2 CB-SPH Based on the DLWE Assumption

In this section, we use that IBE scheme that was given by Gentry *et al.* [20] to construct a CB-SPH based on the DLWE assumption. It is the first cetificate-based cryprographic structure which can be transferred into a certificete-based encryption. We recall the DLWE-based DDP firstly.

5.2.1 DLWE-based DDP

Let D be a distribution distinguish problem based on the DLWE assumption.

SampDDP(κ): According to the trapdoor generation algorithm $\mathsf{TrapGen}(p, \kappa)$ of [20], it generates $A \in Z_p^{n \times m}$ along with a trapdoor $T \subset \Lambda^{\perp}(A,p)$ such that $\|\widetilde{T}\| \leq O(\sqrt{\kappa log p})$ where $\Lambda^{\perp}(A,p)$ is a set of $\{e \in Z^m$ s.t.$A^T e = \mathbf{0} \mod p\}$. Set $pk = A, sk = T$; Output an instance description $\Gamma = (X, W, PK, A_{pk}, B_{pk}, R_{pk})$ of D, where $X = Z_p^n \times Z_p, W = Z_p^n, PK = Z_p^{n \times m}, R_{pk} = \{((p,v),w) \in X \times W : ((A^T w + t, v), w)$

where error term $t \in \chi^m, v \in Z_p\}$ where χ^m is a noise distribution.

SampA(pk): Pick $w \leftarrow_R Z_p^n, t \leftarrow_R \chi^m, v \leftarrow_R Z_p$, compute $p = A^T w + t$, output $x = (p, v)$ and $w \in W$.

SampB(pk): Pick $p \leftarrow_R Z_p^m$ and $v \leftarrow_R Z_p$, output $x = (p, v)$.

In the certificate-based settings we need some more public parameters, just like master public key mpk, the user's identity id and public key pk. The extensive details are described as below.

SampDDP(κ): It generates $A \in Z_p^{n \times m}$ along with a trapdoor $T \subset \Lambda^{\perp}(A, p)$ according to the trapdoor generation algorithm $\mathsf{TrapGen}(p, \kappa)$ of [20] such that $\|\widetilde{T}\| \leq O(\sqrt{\kappa log p})$ where $\Lambda^{\perp}(A, p)$ is a set of $\{e \in Z^m$ s.t.$A^T e = \mathbf{0} \mod p\}$. Trapdoor function $f_A(x) = Ax \mod p$. Set $mpk = (A, f_A), msk = T$ and the user's public key is $pk = Q_{id} \in_R Z_p^{n \times m}$; Outputs an instance description $\Gamma = (X, W, MPK, PK, A_{pk}, B_{pk}, R_{pk})$ of D, where $X = (Z_p^n)^2 \times Z_p, W = Z_p^n, PK = Z_p^{n \times m}, R_{pk} = \{((p_1, p_2, v), w) \in X \times W : ((Q_{id}^T w + t_1, A^T w + t_2, v), w)$ where error terms $t_1, t_2 \in \chi^m, v \in Z_p\}$.

SampA(mpk, id, pk): Pick $w \leftarrow_R Z_p^n, t_1, t_2 \leftarrow_R \chi^m, v \leftarrow Z_p$, compute $p_1 = Q_{id}^T w + t_1, p_2 = A^T w + t_2$, output $x = (p_1, p_2, v)$ and $w \in W$.

SampB(mpk, id, pk): Pick $p_1, p_2 \leftarrow_R Z_p^m$ and $v \leftarrow_R Z_p$, output $x = (p_1, p_2, v)$.

5.2.2 Projective Hash Function

Let $\mathsf{H} = (H, SK, CERT, PK, X, A_{pk}, Y, \alpha)$ be a corresponding projective hash function, where $SK = Z_p^m, CERT = Z_p^m, Y = Z_2$. For $sk_{id} = e_{id}, Cert_{id} = t_{id}$ and $x = (p_1, p_2, v)$, H is defined as $H_{sk_{id}, Cert_{id}}(x) = y$ as $y = 1$ if $|v - (sk_{id}, Cert_{id})^T \cdot (p_1, p_2)| \leq \frac{p-1}{4}$ and $y = 0$ otherwise.

5.2.3 DLWE-based CB-SPH

Before we introduce the DLWE-based CB-SPH structure we recall some important lemma and algorithms which will be used in the CB-SPH.

We say that a matrix $A \in Z^{m \times m}$ is Z_p-invertible if $A \mod p$ is invertible as a matrix in $Z^{m \times m}$.

Lemma 3. *[4] Let $p > 2$ and a matrix $A \in Z_p^{n \times m}, m > n$. Let T be a basis for $\Lambda^{\perp}(A, p), \sigma \geq \|\widetilde{T}\| \cdot \omega(\sqrt{log m})$. Then for $u \in Z_p^n$, there is a polynomial-time algorithm $\mathsf{SamplePre}(A, T, u, \sigma)$ that returns $x \in \Lambda^u(A, p)$ sampled from a distribution statistically close to $D_{\Lambda^u(A,p),\sigma}$ where $\Lambda^u(A, p)$ is a set of $\{e \in Z^m$ s.t.$A^T e = u \mod p\}$.*

Algorithm [2] Sample $S(1^m)$: Let $\sigma_s = O(\sqrt{\kappa log q} \cdot \omega(log m) \cdot \sqrt{m})$.

- Let T_0 be the canonical basis of the lattice Z^m;
- For $i = 1, 2, \cdots, m$, do $s_i \leftarrow \mathsf{SampleGaussian}(Z^m, T_0, \sigma_s, \mathbf{0})$ uniformly;
- If S is Z_p-invertible, output S; otherwise repeat Step 2.

Let P be a CB-SPH for D associating H, which includes the following algorithms:

Setup(κ): Run $\mathsf{SampDDP}(\kappa)$ to generate the master public keys $mpk = (A, f_A)$ and the master secret key is $msk = T$. Let $H_1 : \{0,1\}^* \rightarrow Z_p^n$.

CerGen(mpk, id, msk): On input identity $id \in \{0,1\}^*$ and master public key A. Let $u = H_1(id) \in Z_p^n$ using the PPT algorithm $\mathsf{SamplePre}(A, T, u, \sigma)$ with trapdoor T to sample $t_{id} \leftarrow f_A^{-1}(u)$ such that $\|t_{id}\| \leq \sigma\sqrt{m}$. The CA returns $Cert_{id} = t_{id}$ as the user's certificate.

UserKeyGen(mpk, id): On input the identity id, then use the algorithm Sample $S(1^m)$ to generate a Z_p-invertible matrix S_{id}, compute $S_{id} Cert_{id} = e_{id}$. Then the user's private key is $sk_{id} = e_{id}$.

Choose a random matrix $Q_{id} \in Z_p^{n \times m}$ and compute $u_1 = Q_{id} sk_{id} \mod p \in Z_p^{n \times m}$. The user computes the corresponding public key $pk = \alpha(sk_{id}) = (Q_{id}, u_1, u)$.

Pub(id, pk, x, w): Choose $x = (p_1, p_2, v) \leftarrow \mathsf{SampA}(mpk, id, pk)$, if $|v - (u_1 + u)^T \cdot w| \leq \frac{p-1}{4}$ then set $y = 1$ else set $y = 0$.

Priv($x, sk_{id}, Cert_{id}$): Choose $x = (p_1, p_2, v) \leftarrow \mathsf{SampA}(mpk, id, pk)$, if $|v - (sk_{id}, Cert_{id})^T \cdot (p_1, p_2)| \leq \frac{p-1}{4}$ then set $y = 1$ else set $y = 0$.

5.2.4 Remark

In this subsection, we focus on how to use the IBE scheme [19] to construct a CB-SHP structure. We introduce a Z_p-invertible matrix $S_{id} \in Z_p^{m \times m}$ as a secret value and store it. It has two properties: 1)its norm is small; 2) its distribution is statistically close to a Gaussian distribution. The certificate $Cert_{id}$ is extracted from a distribution statistically close to a discrete Gaussian distribution by a preimage sampleable function with the master private key T which norm is also norm. For $sk_{id} = S_{id} Cert_{id}$ with the properties of S_{id} and $Cert_{id}$, it is achieved that sk_{id}'s distribution is statistically close to a Gaussian distribution and its norm is also small.

The cryptosystem based on lattice is leakage resilient in character [3], therefore the CBE scheme from the proposed DLWE-based CB-SPH is also leakage resilient for any kind of adversary and the random extractor may be unnecessary in the structure. On the other hand, based on our instantiation we can present CCA-secure CBE against key leakage attack in the random oracle.

6 Conclusion and Future Work

In this paper we presented the new notion of certificate-based smooth projecitve hashing and introduced its applications in leakage resilient encryption. We gave the formal definition of CB-SPH and showed how to transfer CB-SPH to leakage resilient one and further showed how to achieve leakage resilient certificate-based encryption (LR-CBE) schemes. With two concrete CB-SPHs, we put forward the first practical realization of LR-CBE which is based on the DBDH assumption in the standard model and presented a lattice-based CB-SPH under the DLWE assumption in the random oracle. Besides applications in the construction of LR-CBE schemes, we thought the concept of CB-SPH is of independent interest and may have other applications in the study of certificate-based cryptography.

Acknowledgments

This research is supported partially by the National Natural Science Foundation of China under Grants No. 61702259, 61672289 and 61672010, Jiangsu Government Scholarship for Overseas Studies, Postdoctoral Science Foundation of Jiangsu Province (No.1601008A), the Natural Science Fund for Colleges and Universities of Jiangsu Province (No.16KJB520018), the Social Science Fund for Colleges and Universities of Jiangsu Province (No. 2017SJB0201).

References

[1] M. Abdalla, F. Benhamouda, D. Pointcheval, "Disjunctions for smooth projective hashings: New constructions and applications," in *Advances in Cryptology (EUROCRYPT'15)* pp. 69-100, 2015.

[2] S. Agrawal, D. Boneh, X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE," in *Advances in Cryptology (CRYPTO'10)*, pp. 98-115, 2010.

[3] A. Akavia, S. Goldwasser, V. Vaikuntanathan, "Simultaneous hardcore bits and cryptography against memory attacks," in *Proceedings of the 6th Theory of Cryptography Conference (TCC'09)*, pp. 474-495, 2009.

[4] J. Alwen, C.Peiker, "Generating shorter bases for hard random lattices," *Theory of Computing Systems*, vol. 48, no. 3, pp. 535-553, 2011.

[5] J. Alwen, Y. Dodis, M. Naor, G. Segev, S. Walfish, D. Wichs, "Public-key encryption in the bounded-retrieval model," in *Advances in Cryptology (EUROCRYPT'10)*, pp. 113-134, 2010.

[6] F. Benhamouda, O. Blazy, C. Chevalier, "New techniques for SPHFs and efficient one-round PAKE protocols," in *Advances in Cryptology (CRYPTO'13)*, pp. 449-475, 2013.

[7] D. Boneh, C. Gentry, M. Hamburg, "Space-efficient identity based encryption without pairings," in *Foundation of Computer Science (FOCS'07)*, pp. 647-657, 2007.

[8] Z. Brakerski, S. Goldwasser, "Circular and leakage resilient public-key encryption under subgroup indistinguishability (or: quadratic residuosity strikes back)," in *Advances in Cryptology (CRYPTO'10)*, pp. 1-20, 2010.

[9] Y. Chen, Z. Zhang, D. Lin, et al, "Generalized (identity-based) hash proof system and its applications," *Security and Communication Networks,* vol. 9, no. 12, pp. 1698-1716, 2016.

[10] S. Chow, Y. Dodis, Y. Rouselakis, B. Waters, "Practical leakage-resilient identity-based encryption from simple assumptions," in *Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS'10)*, pp. 152-161, 2010.

[11] J. S. Coron, "A variant of Boneh-Franklin IBE with a tight reduction in the random oracle model," *Design, Codes Cryptography*, Vol.50, no.1, pp. 115-133, 2009.

[12] R. Cramer, V. Shoup, "Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption," in *Advances in Cryptology (EUROCRYPT'02)*, pp. 45-64, 2002.

[13] Y. Dodis, K. Haralambiev, A. LOpez-Alt, D. Wichs, "Efficient public-key cryptography in the presence of key leakage," in *Advances in Cryptology (ASCIACRYPT'10)*, pp. 613-631, 2010.

[14] Y. Dodis, R. Ostrovsky, L. Reyzin, A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM Journal on Computing*, vol. 38, no. 1, pp. 97-139, 2008.

[15] S. Dziembowski, S. Faust, "Leakage-resilient cryptography from the inner-product extractor," in *Advances in Cryptology (ASCIACRYPT'11)*, pp. 702-721, 2011.

[16] D. Galindo, P. Morillo, C. Rfols, "Improved certificate-based encryption in the standard model," *Journal of System Software*, vol. 81, no. 7, pp. 1218-1226, 2008.

[17] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in *Advances in Cryptology (EUROCRYPT'03)*, pp. 272-293, 2003.

[18] C. Gentry, "Practical identity-based encryption without random oracles," in *Advances in Cryptology (EUROCRYPT'06)*, pp. 445-464, 2006.

[19] C. Gentry, C. Peikert, V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the 40th annual ACM symposium on Theory of computing (STOC'08)*, pp. 197-206, 2008.

[20] C. Gentry, C. Peikert, V. Vaikuntanathan, "How to use a short basis: Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the 40th Annual ACM Sysmposium on Theory of Computing*, pp. 197-206, 2008.

[21] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, E. W. Felten, "Lest we remember: Cold boot attacks on encryption keys," in *Proceedings of the 17th USENIX Security Symposium*, pp. 45-60, 2008.

[22] E. Kiltz, K. Pietrzak, "Leakage resilient ElGamal encryption," in *Advances in Cryptology (ASIACRYPT'10)*, pp. 595-612, 2010.

[23] K. Kurosawa, R. Nojima, L.T.Phong, "New leakage resilient CCA-secure public key encryption," *Journal of Mathematical Cryptology*, vol. 7, no. 4, pp. 297-312, 2013.

[24] A. Lewko, Y. Rouselakis, B. Waters, "Achieving leakage resilience through dual system encryption," in *Conference on Theory of Cryptography*, pp. 70-88, 2011.

[25] J. Li, Y. Guo, Q. Yu, Y. Lu, Y. Zhang, F. Zhang, "Continuous leakage-resilient certificate-based encryption," *Information Sciences*, vol. 355, pp. 1-14, 2016.

[26] S. Li, Y. Mu, M. Zhang, F. Zhang, "Updatable lossy trapdoor functions and its application in continuous leakage," in *The 9th International Conference on Provable Security (ProvSec'16)*, pp. 309-319, 2016.

[27] S. Li, F. Zhang, "Leakage-resilient identity-based encryption scheme," *International Journal of Grid and Utility Computing,* vol. 4, no. 2/3, pp. 187-196, 2013.

[28] S. Li, F. Zhang, Y. Sun, L. Shen, "Efficient leakage resilient public key encryption from DDH assumption," *Cluster Computing*, vol. 16, pp. 797-806, 2013.

[29] J. K. Liu, J. Zhou, "Efficient certificate-based encryption in the standard model," *Security and Cryptography for Networks*, vol. 5229, pp. 144-155, 2008.

[30] S. Liu, J. Weng, Y. Zhao, "Efficient public key cryptosystem resilient to key leakage chosen ciphertext attacks," in *CT-RSA'13*, pp. 84–100, 2013.

[31] Y. Lu, J. Li, "Efficient certificate-based encryption scheme secure against key replacement attacks in the standard model," *Journal of Information Science Engineer* vol. 30, no. 5, pp. 1553-1568, (2014).

[32] M. Naor, G. Segev, "Public-key cryptosystems resilient to key leakage," in *Advances in Cryptology (CRYPTO'09)*, pp. 18-35, 2009.

[33] M. H. Nguyen, K. Tanaka, K. Yasunaga, "Leakage-resilience of stateless/stateful public-key encryption from hash proofs,", in *Australasian Conference on Information Security and Privacy (ACISP'12)*, pp. 208-222, 2012.

[34] N. Nisan, D. Zuckerman, "Randomness is linear in space," *Journal of Computer System Science*, vol. 52, no. 1, pp. 43-52, 1996.

[35] B. Qin, S. Liu, "Leakage-resilient chosen-ciphertext secure public-key encryption from hash proof system and one-time lossy filter," in *Advances in Cryptology (ASIACRYPT'13)*, pp. 381-400, 2013.

[36] E. K. Reddy, "Elliptic curve cryptosystems and side-channel," *International Journal of Network Security*, vol. 12, no. 3, pp. 151-158, 2011.

[37] R. Yang, Q. Xu, Y. Zhou, R. Zhang, C. Hu, Z. Yu, "Updatable hash proof system and its applications," in *European Symposium on Research in Computer Security (ESORICS)*, pp. 266-285, 2015.

[38] Q. Yu, J. Li, Y. Zhang, "Leakage-resilient certificate-based encryption," *Security and Communication Networks*, vol. 8, no. 18, pp. 3346-3355, 2015.

[39] Q. Yu, J. Li, Y. Zhang, W. Wu, X. Huang, Y. Xiang, "Certificate-based encryption resilient to key leakage," *Journal of Systems and Software*, vol. 116, pp. 101-112, 2015.

[40] H. Wee, "Dual projective hashing and its applications – lossy trapdoor functions and more," in *Advances in Cryptology (EUROCRYPT'12)*, pp. 246-262, 2012.

[41] H. Wee, "KDM-security via homomorphic smooth projective hashing," in *19th International Conference on the Theory and Practice of Public-Key Cryptography (PKC'16)*, pp. 159-179, 2016.

[42] W. Wu, Y. Mu, W. Susilo, X. Huang, L. Xu, "A provably secure construction of certificate-based encryption from certificateless encryption," *Computer Journal*, vol. 55, no. 10, pp. 1157-1168, (2012).

Biography

Sujuan Li is an associate professor of School of Mathematical and Physical Sciences at Nanjing Tech University. She received her Ph.D. degree in applied mathematics from Nanjing Normal University. Her research interests include cryptography and information security.

Yi Mu is a professor of School of Computer and Information Technology at University of Wollongong Australia. He received his Ph.D. degree from Australian National University. His research interests include cryptography and network security.

Mingwu Zhang is a professor of School of Computer at Hubei University of Technology. He received his Ph.D. degree in cryptography from South China Agricultural University. His research interests include cryptography and network security.