A New Leakage-Resilient IBE Scheme in the Relative Leakage Model

Yu Chen, Song Luo, Zhong Chen

A New Leakage-Resilient IBE Scheme in the Relative Leakage Model

Yu Chen*, Song Luo, and Zhong Chen

Information Security Lab, School of EECS, Peking University, Beijing, China
Key Laboratory of High Confidence Software Technologies, Ministry of Education
{chenyu,luosong,chen}@infosec.pku.edu.cn

Abstract. We propose the first leakage-resilient Identity-Based Encryption (IBE) scheme with full domain hash structure. Our scheme is leakage-resilient in the relative leakage model and the random oracle model under the decisional bilinear Diffie-Hellman (DBDH) assumption.

Key words: identity based encryption, leakage-resilient, relative leakage, bilinear Diffie-Hellman assumption

1 Introduction

Cryptographic schemes are used to be analyzed in an attack model in which the internal secret states are completely hidden from the adversary/attacker. However several works [12, 13] indicated that the attack model fails to capture many attacks in the real world, since the attacker may obtain some partial information about the secret states via various *key leakage attacks*. Therefore it is urgent to design leakage-resilient cryptographic schemes which remain provably secure in the strengthened attack model which takes *key leakage attacks* into account.

Recently, the research community pay a lot of attention to construct IBE schemes with leakage-resilience. Alwen et al. [1] presented three leakage-resilient IBE schemes from the Gentry IBE [10], the Boneh-Gentry-Hamburg IBE [4], and Gentry-Peikert-Vaikuntanathan IBE [11], respectively. Among them, the first scheme is secure in the standard model, while the other two schemes are secure in the random oracle model. Chow et al. [6] gave three new leakage-resilient IBE schemes from the Boneh-Boyen IBE [2], the Waters IBE [16], and the Lewko-Waters IBE [14], respectively. All of them are secure in the standard model.

Our Contributions. According to [5], IBE schemes from pairings can be classified into three broad families, the full-domain hash family (e.g.

* Supported by National Natural Science Foundation of China (No.61073156).

Boneh-Franklin IBE [3]), the exponent inversion family (e.g. Gentry-IBE [10]), and the commutative blinding family (e.g. Boneh-Boyen IBE [2]). The existing work [1, 6] have shown that IBE schemes from the exponent inversion family and commutative blinding family can be tailored to be leakage-resilient ones. It is natural to ask if we can strengthen the IBE schemes from the full domain hash family to be leakage-resilient.

We give an affirmative answer to the above question by presenting an IBE scheme with the full domain hash structure based on a variant of Boneh-Franklin IBE [7]. Its leakage-resilient chosen plaintext security can be tightly reduced to the DBDH assumption in the relative leakage model and the random oracle model.

2 Preliminaries

Notations. $x \xleftarrow{R} S$ denotes that x is picked uniformly at random from the set S. We write PPT for probabilistic polynomial time. By $\mathsf{negl}(n)$ we denote a negligible function of n. We denote the bit-wise XOR operation by $\oplus$. We denote by $\mathcal{I}$ the identity space and by $\mathcal{SK}$ the private key space.

2.1 Bilinear Diffie-Hellman Assumption

The decisional BDH (DBDH) assumption [2,3] is defined via the following game: the challenger runs the bilinear group generator $\mathsf{GroupGen}(1^\kappa)$ to generate $(p, \mathbb{G}, \mathbb{G}_T, e)$, picks four random exponents x, y, z, w from $\mathbb{Z}_p$, then computes g^x, g^y, g^z, $T_0 = e(g, g)^{xyz}$ and $T_1 = e(g, g)^{xyw}$. We denote by D the tuple $(p, \mathbb{G}, \mathbb{G}_T, e, g, g^x, g^y, g^z)$. The challenger picks a random bit c and gives to the adversary $\mathcal{B}$ the challenge instance (D, T_c). We say $\mathcal{B}$ succeeds in solving the DBDH problem if it outputs the right guess c' for c at the end of the game, whose advantage is defined as:

$$|\Pr[c = c'] - 1/2| = |\Pr[\mathcal{B}(D, e(g, g)^{xyz}) = 0] - \Pr[\mathcal{B}(D, e(g, g)^{xyw}) = 0]|$$

Definition 2.1 *The (t, ϵ)-DBDH assumption holds if no t-time adversary has at least ϵ in solving the DBDH problem in $\mathbb{G}$.*

2.2 Randomness Extractors

The following notions and primitives will be used in our construction. We refer the readers to [1, 15] for a complement knowledge.

For a random variable X, we define $\mathbf{H}_\infty(X) = -\log(\max_x \Pr[X = x])$ as its min-entropy. We use the notion of *average min-entropy* [8] which captures the remaining unpredictability of a random variable X conditioned on another random variable Y, formally defined as

$$\tilde{\mathbf{H}}_\infty(X|Y) = -\log\left(E_{y\leftarrow Y}\left[\max_x \Pr[X=x|Y=y]\right]\right)$$

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

The average min-entropy measures exactly the optimal probability of guessing X given knowledge of Y. The following lemma was proved in [9] regarding average min-entropy:

Lemma 1. *For any random variables X, Y, Z, if Y has 2^ℓ possible values, then $\tilde{\mathbf{H}}_\infty(X|(Y,Z)) \geq \tilde{\mathbf{H}}_\infty(X|Z) - \ell$.*

The statistical distance between two random variables X, Y over a finite domain Ω is defined as

$$\mathbf{SD}(X,Y) = \frac{1}{2}\sum_{\omega\in\Omega} |\Pr[X=\omega] - \Pr[Y=\omega]|$$

Same as [1, 6, 15], a main tool used in our construction is the strong randomness extractor, which is formally defined as follows to the setting of the average min-entropy.

Definition 2.2 *A polynomial-time function $\mathsf{ext} : \mathbb{G}\times\{0,1\}^\mu \to \{0,1\}^m$ is an average case (k,ϵ)-strong extractor if for all pairs of random variables (X,Y) such that $X \in \mathbb{G}$ and $\tilde{\mathbf{H}}_\infty(X|Y) \geq k$, we have that*

$$\mathbf{SD}((\mathsf{ext}(X,U_\mu),U_\mu,Y)),(U_m,U_\mu,Y)) \leq \epsilon$$

where $\mathbb{G}$ is a non-empty set, and U_μ, U_m are two uniformly distributed random variables over $\{0,1\}^\mu$, $\{0,1\}^m$ respectively.

Dodis et al. [8] proved that any strong extractor is in fact an average-case strong extractor, for a proper setting of the parameters:

Lemma 2. *For any $\delta > 0$, if ext is a worst case $(m - \log(1/\delta),\epsilon)$-strong extractor, then ext is also an average-case $(m,\epsilon+\delta)$-strong extractor.*

As a specific example, they proved the following lemma which essentially gives an explicit construction of an average-case strong extractor:

Lemma 3. *Let X, Y be two random variables such that $X \in \mathbb{G}$ and $\tilde{\mathbf{H}}_\infty(X|Y) \geq k$. Let $\mathcal{H} = \{H : \mathbb{G} \to \{0,1\}^m\}$ be a family of universal hash functions. If $m \leq k - 2\log(1/\epsilon)$ then we have*

$$\mathbf{SD}((H(X),U_s,Y)),(U_m,U_s,Y)) \leq \epsilon$$

2.3 Leakage Model for IBE Setting

In this paper we use the relative leakage model suitable for the IBE setting. The leakage-resilient chosen plaintext security is defined by the following $\mathsf{LeakCPA}$ game, which is refined from the $\mathsf{CpaLeak}$ game introduced in [6].

Setup. The challenger generates the public parameters mpk and the master secret key msk. It gives mpk to the adversary and keeps msk to itself.

Phase 1. The adversary can make one of the following two types of queries to the challenger:

1. $\mathsf{Leak}(I, h_i)$ query, where $h_i : \mathcal{SK} \to \{0,1\}^{\ell_i}$. The challenger checks if the overall amount leakage will exceed ℓ. If not, it responds with $h_i(sk)$. Otherwise it responds with a reject symbol $\perp$.
2. $\mathsf{Reveal}(I)$ query, where I is the identity. The challenger responds with the associated private key sk.

Challenge. The adversary submits two messages M_0, M_1 of equal size and a challenge identity I^*, with the restriction that I^* has not been revealed. The challenger picks a random bit β and encrypts M_β under I^*. It sends the resulting ciphertext C^* to the adversary.

Phase 2. The same as Phase 1 with the restriction that no leakage queries or reveal queries related to I^* are allowed.

Guess. The adversary outputs a bit β'. We say it succeeds if $\beta = \beta'$.

The advantage of an adversary $\mathcal{A}$ on breaking an IBE scheme $\mathcal{E}$ with security parameter κ and leakage bound ℓ is defined as $\mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathsf{CPALeak}}(\kappa, \ell) = |\Pr[\beta = \beta'] - \frac{1}{2}|$.

Definition 2.3 *An IBE scheme $\mathcal{E}$ is ℓ-leakage fully secure if for all PPT adversaries $\mathcal{A}$ it holds that $\mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathsf{CPALeak}}(\kappa, \ell) \leq \mathsf{negl}(\kappa)$.*

3 Our Scheme

Our scheme consists of the following four algorithms:

Setup. Run $\mathsf{GroupGen}(1^\kappa) \to (p, \mathbb{G}, \mathbb{G}_T, e)$, pick $x \xleftarrow{R} \mathbb{Z}_p$, $g_2 \xleftarrow{R} \mathbb{G}^*$, and a cryptographic hash function $H : \{0,1\}^* \to \mathbb{G}$. Let $g_1 = g^x$, $\ell = \ell(\kappa)$ be an upper bound on the amount of leakage. Then set an average-case $(\log|\mathbb{G}_T| - \ell, \epsilon_{\mathsf{ext}})$-strong extractor function $\mathsf{ext} : \mathbb{G}_T \times \{0,1\}^\mu \to \{0,1\}^n$. The message space is $\mathcal{M} \in \{0,1\}^n$, while $mpk = (g, g_1, g_2)$ and $msk = x$.

KeyGen. For a given identity I, pick $t \xleftarrow{R} \mathbb{Z}_p$, compute $u = H(I)$, and then generate the private key for I as $sk = (d_1, d_2) = (t, (ug_2^{-t})^x)$.

Encrypt. To encrypt a message M under identity I, pick an exponent $r \xleftarrow{R} \mathbb{Z}_p$ and a seed $s \xleftarrow{R} \{0,1\}^\mu$ for the extractor function, generate the ciphertext as $C = (c_1, c_2, c_3, c_4) = (g^r, s, e(g_1, g_2)^r, M \oplus \mathsf{ext}(e(u, g_1)^r, s))$.

Decrypt. To decrypt a ciphertext $C = (c_1, c_2, c_3, c_4)$ encrypted under I using the associated private key $sk = (d_1, d_2)$ to compute $M = c_4 \oplus \mathsf{ext}(e(c_1, d_2)c_3^{d_1}, c_2)$. It is easy to verify that if the private key matches, we get the right decryption.

3.1 Security Analysis

Theorem 3.1 *If the DBDH assumption holds and the extractor's second parameter ϵ_{ext} is negligible in κ, then the proposed scheme is ℓ-leakage secure, where $\ell = \log|\mathbb{G}_T| - k$ and k is the extractor's first parameter.*

To prove the theorem, we organize the proof as a sequence of games, which are defined as follows:
$\mathbf{Game_{Real}}$: The real $\mathsf{CPALeak}$ game.

$\mathbf{Game_{Final}}$: The real $\mathsf{CPALeak}$ game except in the challenge phase the challenger generates the ciphertext as follows:

$$\begin{array}{ll} z, w \xleftarrow{R} \mathbb{Z}_p, \beta \xleftarrow{R} \{0,1\} & W = e(u^*, g_1)^z e(g_1, g_2)^{t^*(w-z)} \\ c_1^* = g^z & c_2^* \xleftarrow{R} \{0,1\}^\mu \\ c_3^* = e(g_1, g_2)^w & c_4^* = M_\beta \oplus \mathsf{ext}(W, c_2^*) \end{array}$$

where t^* is the tag of private key sk^* of the challenge identity I^*, z and w are randomly picked from $\mathbb{Z}_p$. The challenge ciphertext is $C^* = (c_1^*, c_2^*, c_3^*, c_4^*)$. Note that if $w \neq z$, then C^* is not a valid ciphertext since it is only decrypted correctly when using the private key with tag t^*.

Lemma 3.2 *If there exists a PPT algorithm $\mathcal{A}$ such that $\mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Real}}} - \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Final}}} = \epsilon$, then we can build a PPT algorithm $\mathcal{B}$ with advantage ϵ in breaking the DBDH problem.*

Proof. Suppose $\mathcal{B}$ is given a DBDH challenge $(p, \mathbb{G}, \mathbb{G}_T, e, g, g^x, g^y, g^z, T)$. We now describe how it interacts with $\mathcal{A}$ in the following game:
Setup. $\mathcal{B}$ sets $g_1 = g^x$ (implicitly sets $msk = x$), $g_2 = g^y$, picks a suitable extractor function ext, then gives $\mathcal{A}$ the public parameters $mpk = (p, \mathbb{G}, \mathbb{G}_T, e, g, g_1, g_2, \mathsf{ext})$.
Hash queries. For a fresh hash query on I, $\mathcal{B}$ picks $a, t \xleftarrow{R} \mathbb{Z}_p$ and responds with $u = g^a g_2^t$.

KeyGen queries. For an arbitrary identity I, $\mathcal{B}$ computes a private key for it as follows: (1) compute $u = H(I)$; (2) set $d_1 = t$, $d_2 = g_1^a = (ug_2^{-t})^x = (g^a g_2^t g_2^{-t})^x$; (3) return $sk = (d_1, d_2)$.
We note that the keygen queries are always implicitly called by $\mathcal{B}$ when it answers the associated leak queries and reveal queries.
Phase 1. To answer the leak queries and reveal queries issued by $\mathcal{A}$, $\mathcal{B}$ creates two lists L and K, which are initially empty. L is a list of triples of identities, private keys, and a leakage counter, while K is a list of tuples of identities, private keys.
– $\mathsf{Leak}(I, h_i)$ query: $\mathcal{B}$ checks if there is a tuple $\langle I, sk\rangle$ in the existing K list. If it is not $\mathcal{B}$ runs $sk \leftarrow \mathsf{KeyGen}(msk, I)$, inserts the tuple (I, sk) to the K list and the triple $\langle I, sk, 0\rangle$ to the L list. After this step there must exists a triple $\langle I, sk, num\rangle$ in the L list, $\mathcal{B}$ checks if $num + \ell_i \leq \ell$. If this is true, it responds with $h_i(sk)$ and sets $num \leftarrow num + \ell_i$ in $\langle I, sk, num\rangle$. Otherwise $\mathcal{B}$ responds with a reject symbol $\perp$.

– $\mathsf{Reveal}(I)$ query: $\mathcal{B}$ checks if there is a tuple $\langle I, sk\rangle$ in the K list. If it is $\mathcal{B}$ responds with sk. If it is not $\mathcal{B}$ runs $sk \leftarrow \mathsf{KeyGen}(msk, I)$, inserts the tuple $\langle I, sk\rangle$ to the K list and the triple $\langle I, sk, 0\rangle$ to the L list, and responds the leak query with sk.
Notice that $\mathcal{B}$ can calculate a valid private key for any identity. Therefore, $\mathcal{B}$ is able to answer all the leakage queries $\mathsf{Leak}(I, h_i)$ and reveal queries $\mathsf{Reveal}(I)$, with the corresponding private key $sk = (d_1, d_2)$.
Challenge. $\mathcal{A}$ submits two messages M_0, M_1 and an identity I^* on which it want to be challenged to $\mathcal{B}$. $\mathcal{B}$ computes $sk^* = (d_1^*, d_2^*) = (t^*, g_1^{a^*})$, then generates the challenge ciphertext as follows:

$$\begin{array}{ll} \beta \xleftarrow{R} \{0,1\} & c_1^* = g^z \\ c_2^* \xleftarrow{R} \{0,1\}^\mu & c_3^* = T \\ W = e(c_1^*, d_2^*)(c_3^*)^{d_1^*} = e(g^z, g_1^{a^*})T^{t^*} & c_4^* = M_\beta \oplus \mathsf{ext}(W, c_2^*) \end{array}$$

Phase 2. The same as Phase 1.
Guess. $\mathcal{A}$ outputs a guess β'. $\mathcal{B}$ returns 0 if $\beta = \beta'$ or 1 if $\beta \neq \beta'$.

We will prove that the advantage of $\mathcal{B}$ in breaking the DBDH problem is ϵ. To see this, notice that if $T = e(g,g)^{xyz}$ the challenge ciphertext is a correct ciphertext according to the original encryption algorithm and thus $\mathcal{A}$ plays the $\mathbf{Game_{Real}}$. This is because $W = e(g^z, g_1^{a^*})T^{t^*} = e(g^{a^*}, g_1^z)e(g_2^{t^*}, g_1^z) = e(g^{a^*}g_2^{t^*}, g_1^z) = e(u^*, g_1)^z$ as one can easily verify. Thus the probability that $\mathcal{A}$ succeeds in the game is exactly $\frac{1}{2} + \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Real}}}$. Since $\mathcal{B}$ outputs 0 when $\mathcal{A}$ succeeds we get that

$$\Pr[\mathcal{B}(D, e(g,g)^{xyz}) = 0] = \frac{1}{2} + \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Real}}}$$

On the other hand if $T = e(g,g)^{xyw} = c_3^*$ then $\mathcal{A}$ essentially plays the $\mathbf{Game_{Final}}$, because $W = e(g^z, g_1^{a^*})T^{t^*} = e(g^{a^*}, g_1^z)e(g_2^{t^*}, g_1^{(w-z)+z}) = e(u^*, g_1)^z e(g_1, g_2)^{t^*(w-z)}$ as one can easily verify. Therefore we have that

$$\Pr[\mathcal{B}(D, e(g,g)^{xyw}) = 0] = \frac{1}{2} + \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Final}}}$$

Combining the above equations we get that the advantage of $\mathcal{B}$ in DBDH is $\left|\Pr[\mathcal{B}(D, e(g,g)^{xyz}) = 0] - \Pr[\mathcal{B}(D, e(g,g)^{xyw}) = 0]\right| = \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Real}}} - \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Final}}} = \epsilon$. Therefore we prove the lemma. $\square$

Lemma 3.3 *For any PPT adversary $\mathcal{A}$ we have* $\mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Final}}} \leq 2\epsilon_{\mathsf{ext}}$.

Proof. In the $\mathbf{Game_{Final}}$, it is true that $W = e(u^*, g_1)^z e(g_1, g_2)^{t^*(w-z)}$. where t^* is the tag of the private key for I^*. If we assume that the exact private key with tag t^* is perfect hidden from the adversary, then W distributes uniformly at random in $\mathbb{G}_T$, and therefore the challenge ciphertext C^* is totally independent of M_β in an PPT adversary $\mathcal{A}$'s view. This is because $w = z \bmod p$ with negligible probability in κ and t^* is chosen randomly for I^*.

Suppose we denote by R the set of all terms (public parameters, private keys, challenge ciphertext) given to the adversary $\mathcal{A}$ except the leakage, the random seed c_2^*, and the part of the challenge ciphertext c_4^*, then according to the above argument $\tilde{\mathbf{H}}_\infty(C|R) = \log|\mathbb{G}_T|$. But the attacker has access to at most ℓ bits of leakage from the private key, i.e. to a random variable Y with 2^ℓ values, thus by lemma 1 we know that

$$\tilde{\mathbf{H}}_\infty(C|(Y,R)) \geq \tilde{\mathbf{H}}_\infty(C|R) - \ell = \log|\mathbb{G}_T| - \ell$$

According to the definition of $(\log|\mathbb{G}_T| - \ell, \epsilon_{\mathsf{ext}})$-strong extractor we have that $\mathbf{SD}((\mathsf{ext}(W,S), S, Y, R), (U_m, S, Y, R)) \leq \epsilon_{\mathsf{ext}}$, where S is the random variable for the seed $c_2^* \in \{0,1\}^\mu$ distributed uniformly at random, Y, R are the values of all the random variables known to the adversary: leakage and the rest, respectively. Thus the statistical distance of $c_4^* = M_\beta \oplus \mathsf{ext}(W, c_2^*)$ from the uniform distribution is at most ϵ_{ext} for each β. The statistical distance between the two possible ciphertexts is at most $2\epsilon_{\mathsf{ext}}$ and no adversary (even an unbounded one) can distinguish them with advantage more than this. $\square$

Suppose ϵ_{DBDH} is the maximum advantage of all PPT adversaries in the DBDH game. Then according to the above lemma, for any PPT adversary $\mathcal{A}$ we have $\mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Real}}} - \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Final}}} \leq \epsilon_{DBDH}$. Therefore

$$\mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Real}}} \leq \mathrm{Adv}_{\mathcal{A},\mathcal{E}}^{\mathbf{Game_{Final}}} + \epsilon_{DBDH}(\kappa) \leq 2\epsilon_{\mathsf{ext}}(\kappa) + \epsilon_{DBDH}(\kappa)$$

The proposed scheme is leakage-resilient CPA secure if both $\epsilon_{DBDH}(\kappa)$ and $\epsilon_{\mathsf{ext}}(\kappa)$ are negligible functions of κ. This proves the theorem. $\square$

References

1. Alwen, J., Dodis, Y., Naor, M., Segev, G., Walfish, S., Wichs, D.: Public-key encryption in the bounded-retrieval model. In: Advances in Cryptology - EUROCRYPT 2010. LNCS, vol. 6110, pp. 113–134. Springer (2010)
2. Boneh, D., Boyen, X.: Efficient Selective-ID Secure Identity Based Encryption without Random Oracles. In: Advances in Cryptology - EUROCRYPT 2004. LNCS, vol. 3027, pp. 223–238 (2004)
3. Boneh, D., Franklin, M.K.: Identity-Based Encryption from the Weil Pairing. SIAM Journal on Computation 32, 586–615 (2003)
4. Boneh, D., Gentry, C., Hamburg, M.: Space-efficient identity based encryption without pairings. In: 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007). pp. 647–657. IEEE Computer Society (2007)
5. Boyen, X.: General ad hoc encryption from exponent inversion ibe. In: Advances in Cryptology - EUROCRYPT 2007. LNCS, vol. 4515, pp. 394–411 (2007)
6. Chow, S.S.M., Dodis, Y., Rouselakis, Y., Waters, B.: Practical leakage-resilient identity-based encryption from simple assumptions. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS 2010. pp. 152–161. ACM (2010)
7. Coron, J.S.: A variant of Boneh-Franklin IBE with a tight reduction in the random oracle model. Des. Codes Cryptography 50(1), 115–133 (2009)
8. Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. SIAM J. Comput. 38(1), 97–139 (2008)
9. Dziembowski, S.: Intrusion-resilience via the bounded-storage model. In: Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006. LNCS, vol. 3876, pp. 207–224. Springer (2006)
10. Gentry, C.: Practical Identity-Based Encryption Without Random Oracles. In: Advances in Cryptology - EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464 (2006)
11. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, STOC. pp. 197–206. ACM (2008)
12. Halderman, J.A., Schoen, S.D., Heninger, N., Clarkson, W., Paul, W., Calandrino, J.A., Feldman, A.J., Appelbaum, J., Felten, E.W.: Lest we remember: Cold boot attacks on encryption keys. In: Proceedings of the 17th USENIX Security Symposium. pp. 45–60 (2008)
13. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In: Advances in Cryptology - CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer (1999)
14. Lewko, A.B., Waters, B.: New techniques for dual system encryption and fully secure hibe with short ciphertexts. In: Theory of Cryptography, 7th Theory of Cryptography Conference, TCC 2010. LNCS, vol. 5978, pp. 455–479. Springer (2010)
15. Naor, M., Segev, G.: Public-key cryptosystems resilient to key leakage. In: Advances in Cryptology - CRYPTO 2009. LNCS, vol. 5677, pp. 18–35. Springer (2009)
16. Waters, B.: Efficient Identity-Based Encryption Without Random Oracles. In: Advances in Cryptology - EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127 (2005)